Outlier detection in non-stationary time series applied to sewer network monitoring

Ali Shakil, Mohammad Ali Khalighi, Pierre Pudlo, Cyril Leclerc, Dominique Laplace, François Hamon, Alexandre Boudonne

Outlier Detection in Non-stationary Time Series Applied to Sewer Network Monitoring

Ali Shakil[1,2,4], Mohammad Ali Khalighi[1], Pierre Pudlo[3], Cyril Leclerc[4], Dominique Laplace[2], François Hamon[5], Alexandre Boudonne[5]

[1]Aix Marseille University, CNRS, Centrale Marseille, Institut Fresnel, Marseille, France

[2]Service d'Assainissement Marseille Métropole (SERAMM), Marseille, France

[3]Institut Mathématique de Marseille (I2M), Marseille, France

[4]LYRE Research Center, SUEZ Co., Bordeaux, France [5]GreenCityZen Co., Marseille, France

***Abstract*—We consider the case of data processing for a sewer infrastructure where water drains are equipped with waste-level sensors, which frequently send the related data to a data processing unit. In order to understand the dynamics of waste accumulation within the whole drain network, the collected data should first be pre-processed by removing the unreliable (or, in other words, noisy) measurements. As we show, the evolution of the waste inside a drain can be modeled by a non-stationary discontinuous time series model. Due to the chaotic aspect of the waste and the hostile conditions under which the sensor should operate, the observed time series can include *outliers* in the form of peaks, which should be removed from the raw data prior to any data processing. This paper proposes an efficient *data cleaning* algorithm that makes a good compromise between computational complexity and performance. This latter is evaluated in terms of the probabilities of peak detection (i.e., detecting actual outliers) and false detection (i.e., incorrectly denoting measurements as outliers). A trade-off between these two criteria should be made by setting appropriately the detection threshold (which, in the proposed method, does not depend on the mean or variance of the data). For instance, for a threshold of 2.5, the algorithm provides a correct outlier detection probability of 0.85 and a false detection probability of 2.5×10^{-2}. The efficiency of the proposed algorithm is demonstrated by applying it to real measurement data.**

***Index Terms*—Sewer network monitoring; Smart city; Data cleaning; Outlier detection; Internet-of-Things.**

I. Introduction

In a more than ever connected world, the Internet-of-Things (IoT) paradigm offers promising capabilities for efficient resource management and environment protection at large scale. In the smart-city context, IoT allows an efficient use of energy resources such as electricity and water through the deployment of dedicated sensors and connectivity solutions. Combining IoT and artificial intelligence (AI) opens doors for the deployment of efficient smart systems, capable of adapting to the actual environmental/field conditions, ultimately enabling optimal management of the resources [1], [2]. Typical examples are smart grids, smart water resource management and quality monitoring, smart sewerage network, etc. [3], [4], [5]. In particular, smart management of potable water distribution and sewerage systems can optimize resource consumption and reduce the impact on the environment [6], [7]. Indeed, the ecological impact of an inefficient sewerage system can be devastating, appealing for emergency deployment of adequate measures by the suppliers.

This work focuses on the case of a sewerage system for a large city and considers the deployment of an IoT-enabled smart network for managing the water drains as part of the sewer infrastructure. These drains collect rain and water but are also subject to urban wastes (they are not systematically equipped with bars to prevent waste accumulation inside them). The waste can enter the sewerage system, potentially causing important damages, e.g., obstructing the drain evacuation, resulting in local flooding, or entering the pipes of the sewerage system, damaging then the downstream equipment. Also, too much waste accumulation in the drains results in overflowing in the street. In particular, in some locations in Marseille where the drains of the stormwater network are directly connected to the sea, the rain can directly flush the waste into to the sea throughout the pipes, causing hence disastrous damage to the environment. As such, sewer networks are critical and complex systems that need significant maintenance and surveillance, thus the rapidly increasing deployment of IoT-based solutions over the past few years to ensure quality of life and protection of the environment. Examples include automatic or semi-automatic condition assessment to identify damages in pipes [8], water quality monitoring [9], detection and prediction of sewer overflow [10], or measurement of hazardous gases in sewer networks [11].

In this paper, we focus on the waste level measurement for the storm-water network, and more specifically, on the processing of the associated collected data for the whole sewerage system. In fact, analyzing these data allows to understand the dynamics of waste accumulation inside the drains, and the appropriate actions that should be taken to ensure proper operation of the whole network. Although this application seems to be rather simple at a first glance, the related data processing can be challenging in practice. Indeed, due to the chaotic aspect of the waste and the hostile conditions in which the sensor is installed, the observed time series can present erroneous peaks. The detection and removal of outliers from the raw data collected from a wireless sensor network is an important preliminary step prior to data analysis [12], [13], e.g., in view of developing a reliable mathematical model or training an AI-based algorithm [14].

For this purpose, we propose in this paper an efficient outlier detection method with a low computational complexity. The proposed method allows quick cleaning of the data without any need to labeling them, as well as without any modification of the other data points. This latter is critically important in order not to bias the data, which will otherwise result in information loss, impacting hence the mathematical modelling and the event prediction of the network. The originality of this work is related to the nature of collected data in the form of time series, with the properties of being non-stationary and discontinuous, to which most of the previously-proposed solutions in the literature cannot be applied, as explained in detail later in Section 3.

The remainder of the paper is organized as follows. Section II explains in more detail the considered sewerage system, the sensors, and the nature of the collected data. Next, Section III describes the related work on outlier detection in time series. Then, Sections IV and V present our proposed peak detection algorithm and its performance evaluation, respectively. Concluding remarks and discussions are provided in Section VI.

II. Considered Sewerage System

A. Context

Towards the deployment of IoT solutions for smart-city applications [15], [16], one big project of the French company SUEZ has been the digitalization of the water cycle in the city of Marseille, the second largest city of France with over 1.8 million inhabitants in the metropolis. This includes the instrumentation of the city's stormwater network and, more precisely, the corresponding water drains by means of a wireless sensor network (WSN) to monitor the volume of accumulated waste. The objective is to get a deeper understanding of the dynamics of the network and to intervene in a smart manner (i.e., when and where necessary) for waste removal before it causes environmental, health, olfactory, or visual pollution. In other words, the collected data from the sensors are used to establish mathematical models to predict the waste accumulation throughout the Marseille metropolitan area.

B. The Stormwater Network

Given the size of the metropolitan area and its particularities, the associated water and sewer networks are massive and quite complex. The sewerage system can consist of three types of sub-networks: the wastewater and the rainwater networks, and the combined wastewater/rainwater sewer network (the old infrastructure), see Fig. 1. Here, we focus on the above-mentioned second and third networks, which form together the so-called stormwater network.

The role of a stormwater network is to absorb the rainwater and to avoid flooding. To do so, the network includes over 18,000 drains that need to be maintained meticulously, which includes cleaning the waste that accumulates therein over time. The possibility of large quantities of generated waste, the elaborate network, and the hilly topography of the city are

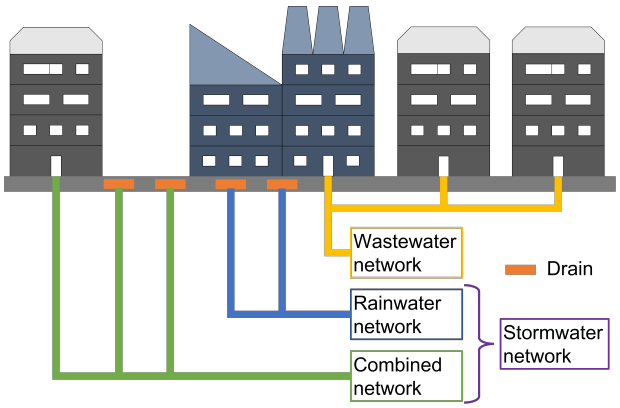


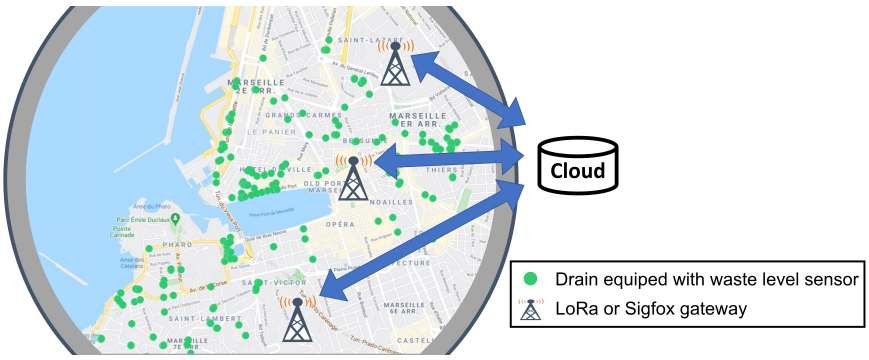
Fig. 1: Illustrating the three existing types of sewer networks in the city of Marseille.

factors among others that make this task particularly complex. Today, the maintenance system is inefficient and non-optimized; the whole process necessitates human intervention and supervision, which includes visual assessment of the waste level inside the drains. The maintenance is currently ensured by the *cleaning staff* who visit each drain regularly and collect the waste if needed. The installation of the above-mentioned WSN allows improving the efficiency of network maintenance and avoiding unnecessary human intervention. So far, more than 3,500 sensors have been installed and the goal is to reach around 5,000 connected sensors by the end of 2022. The current network covers about half of the city, i.e., an area of around 128 km^2.

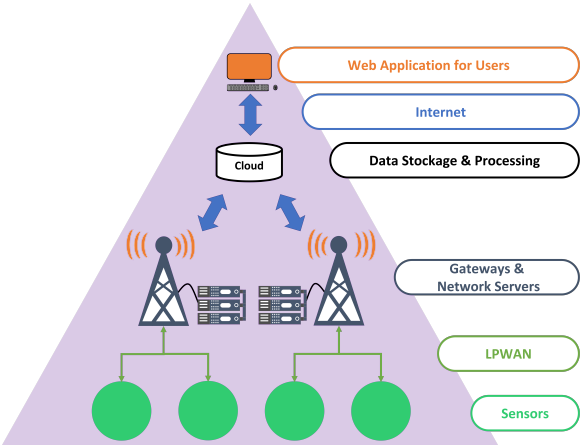
C. Sensor Network

In the considered network, the waste level inside each drain is monitored using ultrasonic sensors connected through a low-power wide-area network (LPWAN), more specifically a LoRa network [17], [18]. A few measurements are done per 24h, and the data is sent to a server through LoRa gateways, where the collected data from the ensemble of the sensors are processed. To provide a more practical view of a typical network, Figures 2(a) and 2(b) show an illustration of the actual network deployed around Marseille downtown (the *Vieux Port* district), and a schematic of the network architecture, respectively. Details on the employed sensors and the other parts of the network are provided in the appendix.

Figure 3 shows a simple illustration of a drain in the form of a box-shape container with some forms of waste inside. An ultrasonic sensor is installed on the top of the drain, pointing toward the bottom. By measuring the delay between an emitted pulse and its echo, the distance to the reflective surface is calculated; this latter can correspond to waste, water or the bottom of the drain (if it is empty). This measured distance, denoted by D, is in fact the data that is then transmitted through the WSN, and which is considered hereafter. Note that a small distance signifies a high waste level W_ℓ, where $W_\ell = H_d - D$ with H_d denoting the drain depth.



(a)



(b)

Fig. 2: Illustration of the LoRa Network: (a) Part of the sewer monitoring network deployed in the *Vieux Port* district at downtown of Marseille with green circles showing the location of the connected drains and towers representing LoRa gateways through which the data is sent onto a cloud. (b) Schematic of the network architecture.

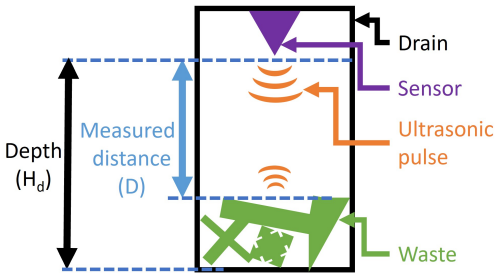


Fig. 3: Schematic of an ultrasonic sensor in a typical drain, with some waste (in green). The waste level is measured based on the delay between the emitted pulse and the received echo.

D. Characteristics of the Collected Data

Each sensor measures the waste level within a drain at regular (adjustable) time intervals. The resulting univariate measurement data describe the evolution of the waste level corresponding to the specific drain, which can be modeled as a non-stationary, discontinuous time series. Note, the non-stationarity of the collected data is due to the complex context and the role of numerous exogenous and environmental factors such as rain, wind, topography, etc., which are specific to each drain (and hence, to the corresponding time-series), and in addition, change over time depending on the unknown contextual factors. A few typical examples of such behaviors are:

- Drains with small variations of the waste level, i.e., accumulating gradually with waste, illustrated in the example of Figure 4(a);
- Drains with sudden variations of the measured distance, e.g., when exposed to large-size wastes, or when it is cleaned, illustrated in the example of Figure 4(b);
- Drains subject to rain where the waste level can either decrease, e.g., flush type behavior where the waste is evacuated in the network or in the streets, or increase, i.e., the rain brings more waste to the drain (these cases are not illustrated for the sake of brevity).

In practice, the measured distances by the sensors are corrupted by *noise* and *undesired outliers*. In our case, due to working in a hostile environment subject to dust, humidity, etc., the sensors' measurements have a precision of about ± 2 cm at most, which is considered as the measurement *noise*. Moreover, the measured data may include some *peaks* (i.e., the outliers), which are either because of the chaotic nature of the waste inside the drain (see the example in Figure 5(a)) or the form of the drain itself, resulting in multiple echos of the

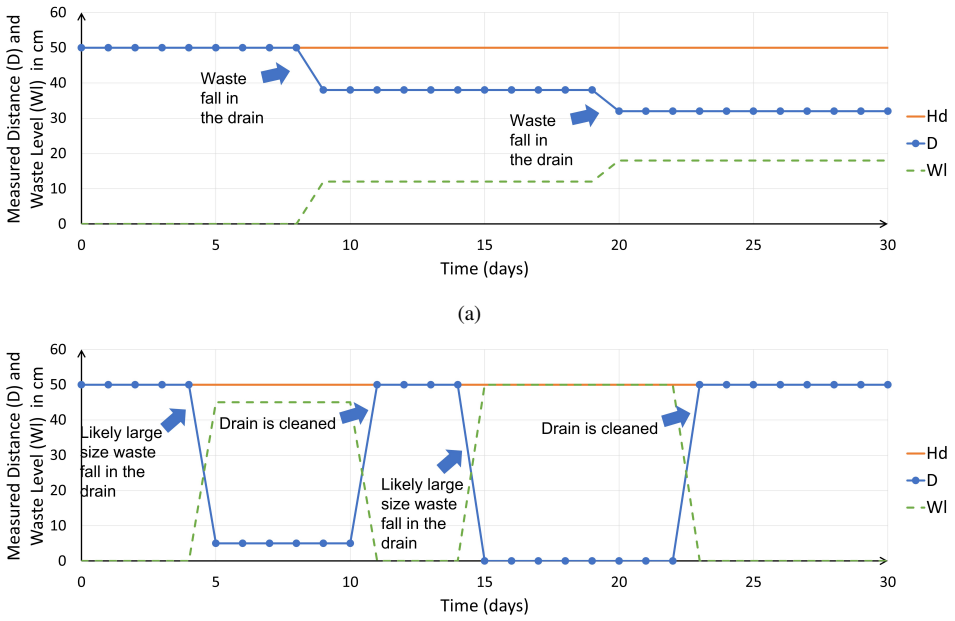


Fig. 4: Simple illustration of the evolution of the waste level W_ℓ and the measured distance D by a sensor; $H_d = 50\,\text{cm}$; (a) slow and steady variations, (b) sudden variations of W_ℓ.

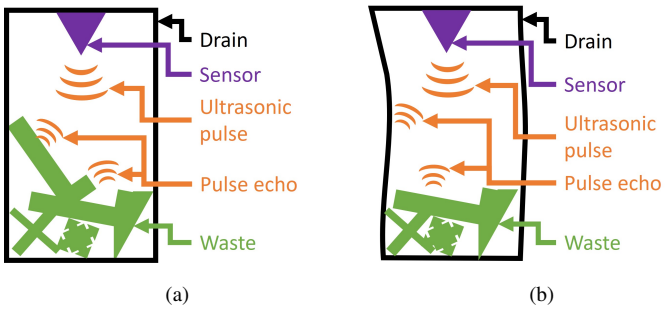


Fig. 5: Examples of typical cases resulting in the appearance of peaks in the collected data: (a) chaotic wastes inside drain causing random ultrasonic echos;
(b) non-uniform drain shape causing ultrasonic signal bouncing on its walls.

sensor's ultrasonic signal (see the example in Figure 5(b)).

We have shown in Figure 6 the measured data of a typical drain sensor during an interval of about one month with two measurements per 24h, for a drain of depth $H_d \approx 55\,\text{cm}$. As we can observe, the measured data contains a few high peaks. Note that the occurrence of successive peaks can be due to the form and volume of the waste inside the drain. This specific case study has been selected among the collected data in the following since it represents a relatively difficult case, i.e., containing two waste falls (around days 9 and 22, as indicated in Figure 6), as well as several peaks (including two successive peaks), in a relatively short measurement interval (i.e., less than 15 days from the time the drain is subject to waste accumulation).

Obviously, to study the dynamics of waste accumulation within the network, the collected data should first be pre-processed by removing the above-described outliers, which is the aim of this work. Note that, this can represent a real challenge when considering the substantial amount of collected data in the whole network composed of thousands of sensors.

III. Related Work

As explained above, in this work, we particularly focus on the problem of peak detection and their removal from the raw data. As mentioned in the Introduction, there are numerous previously-proposed works on similar topics, which have considered outlier detection in time series [19], [20]. However, most of these solutions are not suitable for use in our specific context, where we are concerned with a large quantity of non-stationary and unlabeled univariate data with a complex evolution over time, depending on a number of different (and even unknown) contextual parameters, e.g., rain, wind, slope of the street, proximity to markets, etc. To elucidate the requirement of developing a new solution for the considered application, we present in the following a brief description of

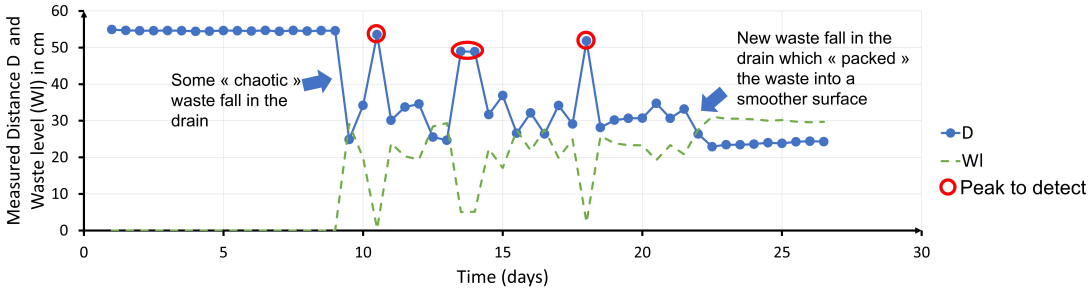


Fig. 6: Example of the collected data from a drain sensor with the measured waste level.

some classical but also some recently-proposed methods and explain their inappropriateness for use in our case.

A simple way of removing peaks from data would be to do some sort of low pass filtering to remove the "high frequency" noise. However, this cannot be used in our case due to the nature of the measurements as signal discontinuities, inherent in the data, will be discarded. Even filters that preserve signal discontinuities such as a median filter [21] will result in a loss of useful information due to data smoothing since it cannot distinguish between temporarily local events in the data (useful information here) and the noisy peaks. Moreover, data smoothing such as exponential smoothing [22] will bias any post-classification or prediction of the data, which is, of course, undesirable. For solutions such as adaptive piecewise constant approximation of a time series [23], where outliers are removed within each data segment independently, the performance highly depends on the number of segments initially set, which is difficult to optimize in our case.

Concerning statistical anomaly detection methods proposed in the literature, which are mostly prediction based, the classical approaches based on auto-regressive (AR), moving average (MA), or ARMA models require second order stationarity (i.e., in terms of mean and variance), which is not valid in our case. It is the same for AR integrated MA (ARIMA) method, which requires stationarity in the sense of signal variance [24]. Also, the iterative outlier removing approach based on extreme studentized deviate test requires the number of outliers to be known [25].

Some other investigated approaches for outlier detection are based on machine learning. There, most of the proposed algorithms such as one-class support vector machines (OC-SVM) [26] are either supervised or semi-supervised, which require labeled outliers or at least a set of *clean* data (without outliers), and hence, cannot be applied to the raw data in our case. A few unsupervised outlier detection algorithms have also been proposed for time series, such as peer-group analysis [27], which consist in characterizing an expected pattern of behavior between similar objects. However, in our case, almost every drain has its own characteristics, and could be considered as independent from the others. Another classical unsupervised technique is that of sub-sequence time series clustering (STSC) [28], which consist in applying the K-means algorithm to time series using a sliding window. Again, each drain having its specific characteristics, the parameter K (the number of clusters in the K-means algorithm) needs to be set in a personalized way, thus resulting in prohibitive data processing complexity in our case. Another solution can be to use auto-encoder based methods, which is a special artificial neural network used to learn an "encoding" (i.e., an optimal representation of the signal) in an unsupervised manner, e.g., using long-short term memory (LSTM) based auto-encoder [29]. However, such methods usually necessitate a significant volume of data to optimize the parameters of the underlying neural network, e.g., number of layers, number of cells in each layer, window size, smoothing window size, etc.

IV. Outlier Cleaning from Collected Data

In general sense, outlier detection can be considered as a classification problem, where a common approach consists in transforming the time series into a scatter point representation that facilitates separation between normal and abnormal points. Our proposed solution is based on this idea, which is in fact used in some of the methods described in the previous section, where this transformation has been done using sliding windows. Our proposed method, however, realizes this in a much simpler way.

In fact, in order to transform the time series data into a scattered point representation, the idea here is to give a score to each point in order to characterize its "abnormality." Then, different metrics can be used to detect the noisy peaks, as described in the following.

A. Z-Score Method

Z-score is one of the most classical metrics used for anomaly detection [30]. Given a data-set of size N, $\boldsymbol{X} = [X_1, \cdots, X_N]$ with mean μ and standard deviation σ, the associated Z-score to each measurement X_i, denoted here by $Z(X_i)$, is given by $Z(X_i) = (X_i - \mu)/\sigma$. Note, it implicitly assumes a Gaussian distribution for the data. The higher the Z-score, the higher is the probability that a given measurement corresponds to an outlier. Typically, outlier detection is done using a threshold: if a measurement's Z-score exceeds the threshold, it is considered as an outlier. This will obviously eliminate a certain percentage of the data, for example, a threshold of 2 and 3 results in discarding $\sim 4.6\%$ and $\sim 0.3\%$ of the data, respectively. In our case, considering a rough estimate of noisy peaks of 1% of the collected data, the corresponding threshold is about 2.5, based on the assumption of Gaussian distribution for $\boldsymbol{X}$.

Since in our case we are concerned with non-stationary data (i.e., changing μ and σ over time), the Z-score method should be applied on a sliding window of appropriate length, depending on the dynamics of waste accumulation, in order to adjust the outlier detection threshold. As an example, we have illustrated in Figure 7 the measured data of Figure 6, where the Z-score method is applied either to the whole data set (see Figure 7(a)) or to a sliding window of length 10 corresponding to 5-day measurements (see Figure 7(b)), considering a threshold of 2.5. Obviously, decreasing the threshold will improve peak detection but it will also result in false peak detection, and vice versa. It is worth mentioning that, as the statistics of waste accumulation are different from one drain to another, the considered threshold should be set in a "personalized" way for every drain and further adapted to environmental and seasonal conditions, for instance, due to the non-stationarity of the waste accumulation process. However, performing such a personalized and adaptive thresholding is challenging and would require, for instance, to go through change point detection in order to delimit the changes in the distribution, resulting in considerable data processing complexity.

B. Opposite Variation Detection

Looking for a better outlier detection, we firstly developed a simple intuitive method, which consist in identifying outliers as corresponding to an "unusual" change in the trend of the measured distance, i.e., a sudden large (negative or positive) variation of D. We will refer to this method as opposite variation detection (OVD). Let us consider the time derivation of X at instance i as follows:

$$\delta_i = \frac{X_{i+1} - X_i}{\Delta_t}, \tag{1}$$

with Δ_t representing the time interval between two successive measurements. To identify a change in the trend of X, we look for a change in the sign of δ_i by defining κ_i as follows:

$$\kappa_i = \frac{\text{sign}(\delta_{i+1}) - \text{sign}(\delta_i)}{2}, \tag{2}$$

where sign(.) is the sign function. Here, κ_i takes the values ± 1 if the sign of δ_i changes, and zero otherwise; X_i is considered as a potential peak if $\kappa_i \neq 0$. Based on this, we define the outlier detection metric θ_i as follows:

$$\theta_i = \kappa_i \ \min(|\delta_{i+1}|, |\delta_i|), \tag{3}$$

where $|.|$ denotes the absolute value. The idea behind (3) is to associate with each point an abnormality score, which is obtained from the absolute value of the local slopes of the D plot (with respect to the previous and the next data points). This latter is the minimum of the two calculated slopes in order to account for signal discontinuities due to normal variations of the waste level in the drain or the measurement noise. This way, "potential peaks" would correspond to relatively large values of θ_i.

To explain this more clearly, two examples are shown in Figure 8, where in Figure 8(a) P_1 is a "normal" point whereas in Figure 8(b) P_2 is a peak to detect. Here, both points can be considered as potential peaks since from (2) the corresponding κ are non-zero. The metric θ is in fact calculated so as to distinguish between these two cases. Let us denote the corresponding metric by θ_{P_1} and θ_{P_2}. The slopes of the D plot at P_1, i.e., δ_{P_1} are denoted by $SP_{1,1}$ and $SP_{1,2}$; likewise the slopes at P_2 are denoted by $SP_{2,1}$ and $SP_{2,2}$. Here, for point P_1, the smallest slope is $SP_{1,2}$, and hence, $\theta_{P_1} = |SP_{1,2}|$. Also, for point P_2, the smallest slope is $SP_{2,1}$, and hence, $\theta_{P_2} = |SP_{2,1}|$. We notice that $\theta_{P_1} < \theta_{P_2}$, in other words, P_2 is more likely to be a peak than P_1, which is indeed the case here.

We have further shown in Figure 9 an illustration of outlier detection by calculating the metric θ and applying a threshold to it. Comparing this figure with Figure 6, one can see that for a threshold of 2.5, the OVD method has detected two peaks, but has missed the two others. In fact, the OVD metric is close to zero for the case of successive peaks, which is, of course, unsuitable.

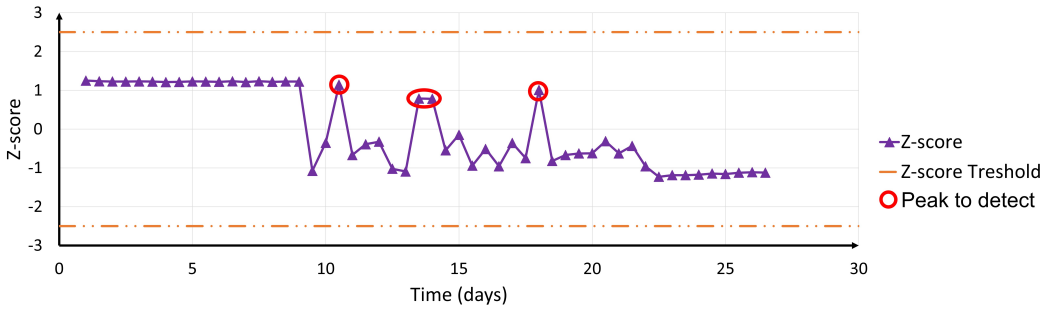
C. Proposed Solution

As discussed in the two previous subsections, both sliding window Z-score-based and OVD methods show limited performance in practice in detecting outliers in the collected data. In fact, these methods are based on metrics which are centered at 0 when applied to "clean" data. Here, we propose a more efficient feature-based solution, that we will call *peak-pattern-based Z-score* (PPZ), by combining the two ideas and applying them to a two-dimensional (2D) representation of data, as described in the following. The idea behind this 2D feature-based approach is to complete the information provided by Z-score method with that obtained from OVD.

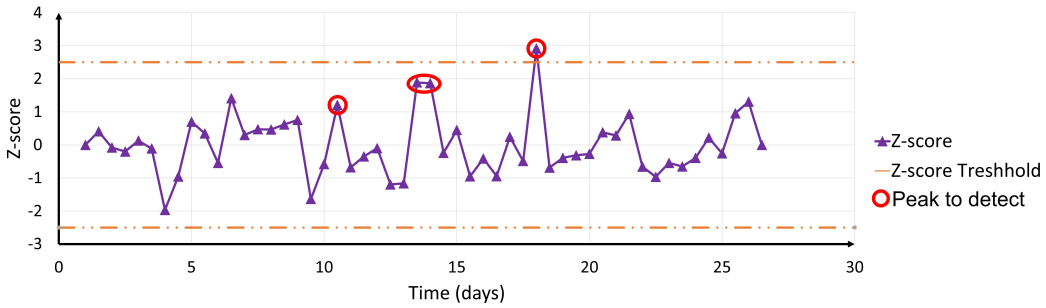
We propose to use the two previously-presented metrics to obtain a scatter plot of data, centered at $(0, 0)$. The points corresponding to outliers will be placed distant from the center, which can be distinguished from the clean data by applying a 2D threshold, i.e., in the form of an ellipse in the 2D representation. In other words, for each point X_i of the time series, Z-score and θ metrics are calculated, resulting in two vectors $\boldsymbol{Z}$ and $\boldsymbol{\theta}$, respectively. Threshold setting for outlier detection is based on eigen-decomposition of the (2×2) covariance matrix $\boldsymbol{K}_{\boldsymbol{Z},\boldsymbol{\theta}}$, called standard deviational ellipse (SDE) [31]; the resulting eigenvectors determine the angle of the threshold ellipse, while the axes length of this latter is set by multiplying the square root of the corresponding eigenvalues by a constant ξ.

In fact the proposed approach relies on the simplifying assumption that the time series $\boldsymbol{X}$, modeled as samples of a random variable, follows a Gaussian distribution whose mean and standard variation are almost constant within the time window of data collection. For instance, the considered period in the provided examples (see Figure 6) is 120 hours, i.e., five days, which corresponds to 10 points with sensing rate of 2 measurements per day. Under this assumption, the calculated Z-score will also follow a Gaussian distribution. As for θ metrics, it will follow a pseudo-Gaussian distribution, i.e., with a higher occurrences of zeros due to κ.

Figure 10 illustrates outlier detection using the proposed PPZ method, applied to the set of collected data from Figure 6.

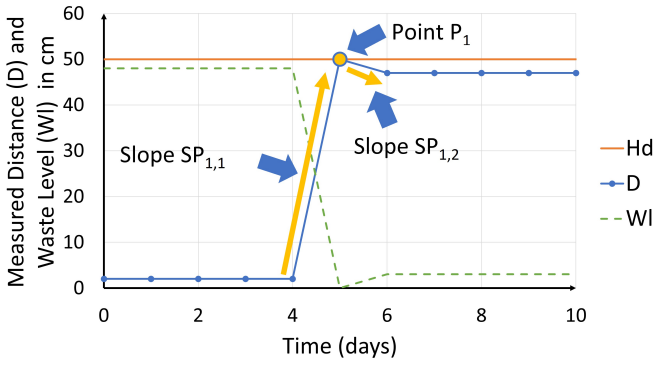


(a)

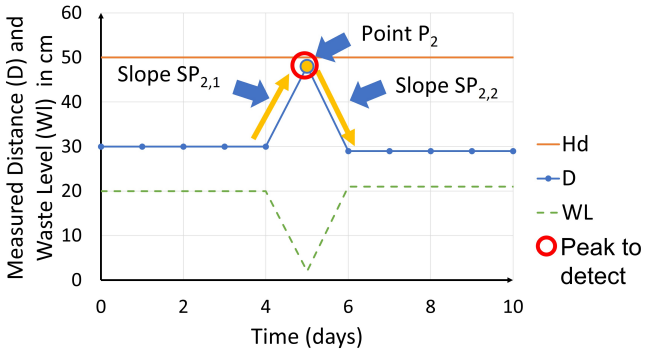


(b)

Fig. 7: Example of outlier detection using Z-score method on the data showed in Figure 6. (a) Z-score of the data calculated on the whole window (no peaks are detected);
(b) Z-score calculated on a sliding window of length 10 points (5 days), in this case only one peak is correctly detected. The outlier detection threshold is considered at ± 2.5.



(a)



(b)

Fig. 8: Illustration of the idea behind the θ metric: (a) Normal point, P_1, (b) a peak to detect, P_2. We have $SP_{1,2} < SP_{1,1}$ and $SP_{2,1} < SP_{2,2}$. Also, $|\theta(P_1)| < |\theta(P_2)|$.

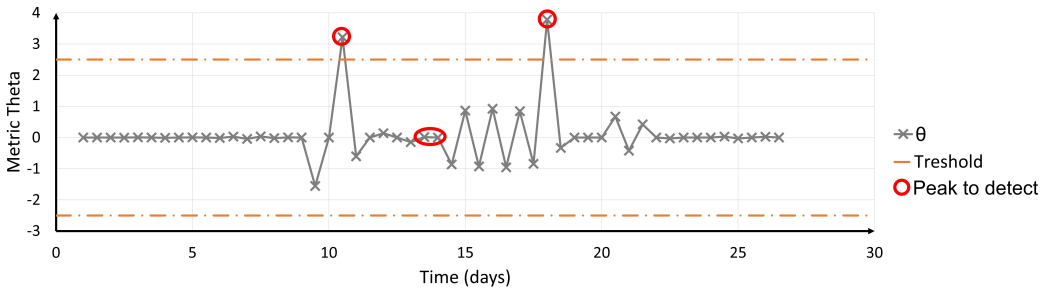


Fig. 9: OVD algorithm applied to the data in Figure 6, considering a threshold of 2.5 (in absolute value) on the calculated metric θ.

Here, the constant ξ is fixed to 2.5, as previously considered for Z-score and OVD methods. As can be seen, each point in the 2D plot corresponds to the calculated Z-score and θ metrics for every measurement (as already shown in Figures 7(b) and 9). The method allows the detection of all peaks, albeit incorrectly identifying the point at Day 4 as a peak.

V. Performance Evaluation of the Proposed Algorithm

The common approach for comparing the performances of different outlier detection methods is to contrast the ROC (Receiver Operating Characteristic) curves. These are obtained upon calculation of the confusion matrix, with elements consisting of the number of "true positives" N_{TP} (i.e., correctly detected outliers), "true negatives" N_{TN} (i.e., correctly not-detected normal points), "false positives" N_{FP} (i.e., normal points identified incorrectly as outliers) and "false negatives" N_{FN} (i.e., undetected outliers). ROC curves are obtained by plotting the so-called "true positive rate" TPR, defined as $N_{\text{TP}}/(N_{\text{TP}}+N_{\text{FN}})$, versus the "false positive rate" FPR, defined as $N_{\text{FP}}/(N_{\text{FP}}+N_{\text{TN}})$.[1]

Performance evaluation can be done based on either simulated or real labeled data. Obviously, the former approach is pertinent only when the simulated data are representative of the real data. This appears to be challenging in our case since each drain has a distinct behavior, related to a number of environmental parameters. Nevertheless, as a preliminary performance study of the proposed algorithm, we applied it to a set of simulated data, generated based on a piecewise constant or a piecewise linear time series (based on some arbitrary parameters and probabilistic laws) to which a Gaussian noise was added to represent the sensor measurement noise as well as a few peak-type points as outliers (e.g., using an exponential distribution with an offset).[2]

Applying the proposed algorithm to such generated data was quite promising (results are not shown for the sake of brevity). However, this cannot be considered as representative of the performance on the real data, given the complexity of the collected data in practice. Therefore, we have decided to evaluate the algorithm performance by applying it to real (measured) data. This, however, necessitates manual labeling of a data set in order to identify with certitude the peaks present in the data (and to see whether or not they are actually detected through outlier detection).

For this purpose, a data set of two months from around 300 sensors was labeled, representing a total of about $33,600$ points, among which over 460 points (i.e., 1.3% of the data) were labeled as peaks. To label the data manually, for each sensor, the time series was visualized as a scatter plot, where each point was labeled or not as a peak (this was a tedious task, of course). We have contrasted the corresponding ROC curves of the proposed PPZ method with those of Z-score and OVD in Figure 11, where we can notice the superiority of the former. Note, for the PPZ algorithm, the threshold parameter ξ can be set so as to ensure a required minimum TPR or a maximum FPR. A good compromise can be made by setting $\xi = 2.5$, which results in a TPR of 0.85 and an FPR of 2.5×10^2, as shown in the figure. Note, the performances of the algorithms can further be quantified by calculating the air under each ROC curve. In this case, the airs are 0.92, 0.95 and 0.98 for the Z-score, OVD, and PPZ methods, respectively.

Lastly, concerning the computational complexity of our proposed method, for a sensor with N measurements, the calculation of each metric has a complexity of $\mathcal{O}(N)$. Then, the algorithm calculates the threshold ellipse, which is based on eigen-decomposition of the covariance matrix of dimension (2×2), as described in Subsection IV-C, with relatively low complexity [32]. Meanwhile, the estimation of each of the

[1]Note, other metrics could also be used to evaluate the performance of the algorithms including the so-called "precision" Pr defined as $N_{\text{TP}}/(N_{\text{TP}}+N_{\text{FP}})$, or the $F1$-score, defined as the harmonic mean between TPR and precision, i.e., $2\,Pr\,\text{TPR}/(Pr+\text{TPR})$.

[2]More specifically, two approaches were used for data generation in these simulations. The first approach consisted of generating data according to a piecewise constant signal level of length N divided into P equal segments. The value of each segment was set to an integer, generated randomly according to a uniform distribution $\mathcal{U}(0, 100)$. By the second approach, data were generated according to a piecewise linear signal level variation, where the slope of each segment was generated randomly following a uniform distribution $\mathcal{U}(-1, 1)$. In both cases, a normalized Gaussian noise was added to each point to represent the measurement noise; then, P points, selected randomly, were modified by adding an exponential noise with an offset to simulate the outlier peaks.

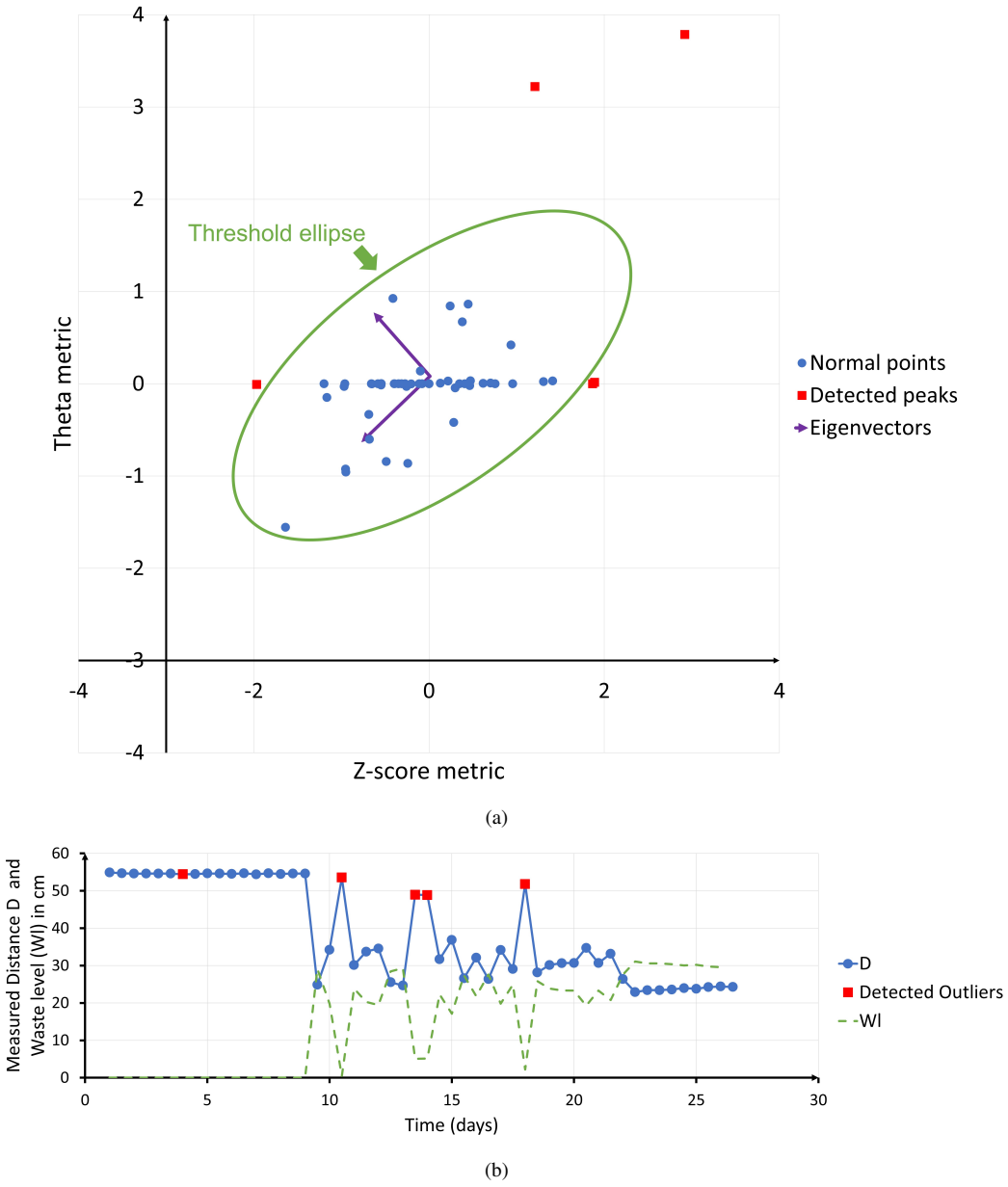


Fig. 10: Illustration of outlier detection using the proposed outlier detection algorithm applied to the data of Figure 6 considering $\xi = 2.5$: (a) scatter plot of Z-score and θ metrics; (b) the resulting detected peaks in the original time series.

four entries of the covariance matrix entails a complexity on the order of $\mathcal{O}(N)$. So, overall, the proposed method has a computational complexity on the order of $\mathcal{O}(N)$.

VI. Conclusions and Discussions

We proposed in this paper a new algorithm for peak-type outlier detection applied to sewer network collected data. The proposed PPZ method provides superior performance compared to the classical Z-score method with a relatively low additional compositional complexity. The main idea behind this algorithm has been to enhance the performance of the Z-score metric by augmenting it with a pattern-detection metric designed here to detect peak-type outliers. In other words, the Z-score metric provides a large spectrum outlier detection but necessitates the signal to follow a Gaussian distribution as well as the use of an optimal sliding window length. However, the arbitrary distribution of the signal and its non-stationarity make theses two requirements difficult to achieve in general. On the other hand, by the OVD method, the θ metric was specifically designed to detect peak-type outliers, independently from the window size. As such, combining the complementary metrics of Z-score and OVD through 2D thresholding for outlier detection results in considerable performance improvement, while entailing a relatively low computational complexity.

Despite the advantages of the proposed method, highlighted in the paper, it could have limited performance in specific circumstances in practice. The first consideration concerns the calculation of the metric θ for the time series, which

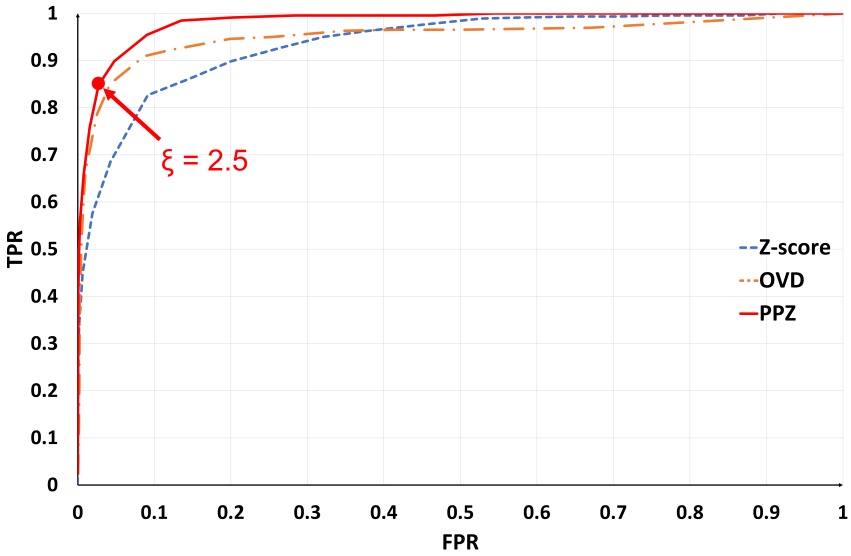


Fig. 11: Contrasting ROC curves of the Z-score, OVD, and the proposed PPZ method applied to labeled data.

requires that the measurement rate of the sensor should be the same for the considered time interval (remember that the sensing rate of each individual sensor can be modified, if needed, to have more or less precise evolution of the waste accumulation within the corresponding drain). Similarly, the proposed approach assumes that the parameters of the collected data are almost unchanged during the considered interval. Any abrupt change in the properties of the data (e.g., flash function after waste accumulation due to heavy rain, or drain cleaning) will affect the performance of the algorithm. Moreover, the selection of the time window width for the Z-score metric (taken as 120 hours in the presented results) should be adapted for each specific sensor and the dynamics of waste accumulation in the corresponding drain.

To improve the outlier detection performance, future work will consider the use of more elaborate clustering methods (in contrast to the considered 2D SDE), such as one-class-support-vector-machine (OCSVM) [33], when a sufficient amount of clean (outlier free) data is available, required for algorithm training. This has additionally the advantage of developing a boundary function adapted to each sensor. Also, in the case of recurrent successive peaks, a customized metric can be developed, adapted to such specific patterns.

Appendix

The network uses an LPWAN network for data transmission from the sensors, that can be according to LoRa (used here) or Sigfox technologies. The LoRa gateways are collocated with network servers. The distance between a sensor and the corresponding BS in our networks varies between a few meters up to a few hundreds of meters. The collected data are transferred to a cloud via the network server for storage in a database and processing. The users can access the data via a web application connected to the cloud. More details on the network and sensors' specifications are provided in the following.

- Sensor: Ultrasonic level measurement sensor ("Hummbox Level" third generation, made by GreenCityZen Co., based on Microchip's SAM microcontroller with its development tools and programming environment);
- Cloud: The Things Network (TTN);
- Wireless connectivity: LoRa;
  - Carrier frequency: 868 MHz,
  - Transmission power: 14 dBm,
  - Effective Isotropic Radiated Power (EIRP): 16 dBm,
  - Receiver sensitivity: −148 dBm,
  - Number of channels: 3 to 8.

References

[1] S. Yao, Y. Zhao, A. Zhang, S. Hu, H. Shao, C. Zhang, L. Su, and T. Abdelzaher, "Deep learning for the Internet of Things," *Computer*, vol. 51, no. 5, pp. 32–41, 2018.

[2] S. Helal, F. C. Delicato, C. B. Margi, S. Misra, and M. Endler, "Challenges and opportunities for data science and machine learning in IoT systems a timely debate: Part 2," *IEEE Internet of Things Magazine*, vol. 4, no. 2, pp. 46–50, 2021.

[3] E. Hossain, I. Khan, F. Un-Noor, S. S. Sikander, and M. S. H. Sunny, "Application of big data and machine learning in smart grid, and associated security concerns: A review," *IEEE Access*, vol. 7, pp. 13 960–13 988, 2019.

[4] A. C. D. S. Junior, R. Munoz, M. D. L. A. Quezada, A. V. L. Neto, M. M. Hassan, and V. H. C. D. Albuquerque, "Internet of Water Things: A remote raw water monitoring and control system," *IEEE Access*, vol. 9, pp. 35 790–35 800, 2021.

[5] Q. F. Hassan, *Internet of Things A to Z: Technologies and Applications*. Wiley IEEE Press, 2018.

[6] A. A. Nasser, M. Z. Rashad, and S. E. Hussein, "A two-layer water demand prediction system in urban areas based on micro-services and LSTM neural networks," *IEEE Access*, vol. 8, pp. 147 647–147 661, 2020.

[7] H. M. Mustafa, A. Mustapha, G. Hayder, and A. Salisu, "Applications of IoT and artificial intelligence in water quality monitoring and prediction: A review," in *International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India.*, 2021, pp. 968–975.

[8] R. Rayhana, Y. Jiao, A. Zaji, and Z. Liu, "Automated vision systems for condition assessment of sewer and water pipelines," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 4, pp. 1861–1878, 2021.

[9] S. O. Olatinwo and T.-H. Joubert, "Enabling communication networks for water quality monitoring applications: A survey," *IEEE Access*, vol. 7, pp. 100 332–100 362, 2019.

[10] D. Zhang, "Sewer system control using artificial intelligence, hydraulic model, and Internet of Things," Ph.D. dissertation, Norwegian University of Life Sciences, 2019.

[11] N. N. Kasat, P. D. Gawande, and A. D. Gawande, "Smart city solutions on drainage, unused well and garbage alerting system for human safety," in *International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-19), Nagpur, India*, 2019, pp. 1–6.

[12] A. K. M. Al-Qurabat and S. A. Abdulzahra, "An overview of periodic wireless sensor networks to the internet of things," *IOP Conference Series: Materials Science and Engineering*, vol. 928, no. 3, p. 32055, Nov 2020.

[13] A. Finjan, H. M. Salman, and A. Qurabat, "Important extrema points extraction-based data aggregation approach for elongating the WSN lifetime," *International Journal of Computer Applications in Technology*, vol. 68, p. 357, 01 2022.

[14] A. A. Cook, G. Misirli, and Z. Fan, "Anomaly detection for IoT time-series data: A survey," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6481–6494, 2020.

[15] S. P. Mohanty, U. Choppali, and E. Kougianos, "Everything you wanted to know about smart cities: The Internet of Things is the backbone," *IEEE Consumer Electronics Magazine*, vol. 5, no. 3, pp. 60–70, 2016.

[16] M. A. Pradhan, S. Patankar, A. Shinde, V. Shivarkar, and P. Phadatare, "IoT for smart city: Improvising smart environment," in *International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai,India*, 2017, pp. 2003–2006.

[17] U. Raza, P. Kulkarni, and M. Sooriyabandara, "Low power wide area networks: An overview," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 855–873, 2017.

[18] M. Centenaro, L. Vangelista, A. Zanella, and M. Zorzi, "Long-range communications in unlicensed bands: the rising stars in the IoT and smart city scenarios," *IEEE Wireless Communications*, vol. 23, no. 5, pp. 60–67, 2016.

[19] M. Braei and S. Wagner, "Anomaly detection in univariate time-series: A survey on the state-of-the-art," *ArXiv*, 2020.

[20] A. Blázquez-García, A. Conde, U. Mori, and J. A. Lozano, "A review on outlier/anomaly detection in time series data," *ACM Comput. Surv.*, vol. 54, no. 3, apr 2021.

[21] B. Justusson, "Median filtering: Statistical properties," in *Two-Dimensional Digital Signal Processing II*. Springer, 1981, pp. 161–196.

[22] R. Hyndman, A. Koehler, K. Ord, and R. Snyder, *Forecasting with exponential smoothing. The state space approach*. Springer, 2008.

[23] M. C. Dani, F.-X. Jollois, F. Cassiano, and M. Nadif, "Adaptive Threshold for Anomaly Detection Using Time Series Segmentation," *ICONIP*, Nov. 2015.

[24] W. Wei, *Time Series Analysis: Univariate and Multivariate Methods, 2nd edition, 2006*, 2006.

[25] J. Hochenbaum, O. S. Vallis, and A. Kejariwal, "Automatic anomaly detection in the cloud via statistical learning," *CoRR*, vol. abs/1704.07706, 2017.

[26] J. Ma and S. Perkins, "Time-series novelty detection using one-class support vector machines," in *Proceedings of the International Joint Conference on Neural Networks, Portland, Oregon, USA*, vol. 3, 2003, pp. 1741–1745 vol.3.

[27] Z. Ferdousi and A. Maeda, "Unsupervised outlier detection in time series data," in *22nd International Conference on Data Engineering Workshops (ICDEW), Atlenta, GA, USA*, 2006, p. 121.

[28] K. Peker, "Subsequence time series (sts) clustering techniques for meaningful pattern discovery," in *International Conference on Integration of Knowledge Intensive Multi-Agent Systems, Waltham, MA, USA*, 2005, pp. 360–365.

[29] O. I. Provotar, Y. M. Linder, and M. M. Veres, "Unsupervised anomaly detection in time series using lstm-based autoencoders," in *2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT), Kyiv, Ukraine*, 2019, pp. 513–517.

[30] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2001.

[31] B. Wang, W. Shi, and Z. Miao, "Confidence analysis of standard deviational ellipse and its extension into higher dimensional euclidean space," *PLoS ONE*, vol. 10, no. 3, pp. 60–67, 2015.

[32] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*. Cambridge University Press, 2006.

[33] J. Ma and S. Perkins, "Time-series novelty detection using one-class support vector machines," in *International Joint Conference on Neural Networks*, vol. 3, 2003, p. 17411745.