Temporal Pattern-Based Denoising and Calibration for Low-Cost Sensors in IoT Monitoring Platforms

Xhensilda Allka, Pau Ferrer-Cid, Jose M. Barcelo-Ordinas, and Jorge Garcia-Vidal

***Abstract*—The introduction of low-cost sensors (LCSs) in air quality Internet of Things (IoT) monitoring platforms presents the challenge of improving the quality of the data that these sensors provide. In this paper, we propose two algorithms to perform denoising and calibration for LCSs used in IoT monitoring platforms. Sensors are first calibrated in-situ using linear or nonlinear machine learning models that only take into account instantaneous measurements. The best calibration model is used to estimate the values measured by the sensor during the sensor deployment. To improve the values of the estimates produced by the in-situ calibration model, we propose to take into account the temporal patterns present in signals such as temperature or tropospheric ozone that have regular patterns, e.g. daily. The first method, which we call temporal pattern-based denoising (TPB-D), performs signal denoising by projecting the daily signals of the in-situ calibrated LCS onto a subspace generated by the daily signals stored in a database taken by reference instruments. The second method, which we call temporal pattern-based calibration (TPB-C), considers that if we also have a reference instrument co-located to the LCSs over a period of time, we can correct with a linear mapping with regularization the daily LCS signals projected in the subspace produced by the reference database to be as similar as possible to the projected signals of the co-located reference instrument. The results show that TPB-D improves the estimates made by in-situ calibration by up to 10-20% while the TPB-C improves the estimates made by in-situ calibration by up to 20-40%.**

***Index Terms*—Low-cost sensors, air quality, monitoring networks, sensor calibration, machine learning.**

I. Introduction

Lately, there has been a growing interest in the development of air quality IoT monitoring platforms using LCSs. These networks are composed of nodes that include inexpensive sensors that measure pollutants such as NO_2, NO, O_3, PM_x, and environmental variables, such as temperature and relative humidity [1]–[3]. There is some consensus that the calibration of LCSs in controlled laboratory chambers performs better than when the sensors are tested against a naturally varying atmospheric composition in the field [4]. A common practice to solve this problem is to carry out a calibration in uncontrolled environments [2], [3], [5], also called in-situ calibration, which consists of placing the node with the sensors near the area where the node will be deployed, and measuring over a period of time, contrasting the measurements taken with those made by a reference instrument such as a reference station managed by government agencies. In-situ calibration is carried out using supervised machine learning methods, either linear or nonlinear, which are trained to map the measurements obtained by the sensor to air quality concentrations, obtaining the optimal coefficients or hyperparameters of the calibration model. Using these hyperparameters obtained in the calibration phase, the node is deployed and the estimation of the pollutant concentrations measured from the measurements taken by the sensor is made. Examples of machine learning models used lately in the calibration of low-cost air quality sensors in uncontrolled environments are multiple linear regression (MLR), support vector regression (SVR), random forest (RF), Gaussian processes (GP), K-nearest neighbor (KNN), or neural networks (NN), [6]–[13].

In-situ calibration models take as input the raw sensor measurement value obtained over a period of time to produce the estimates. These models do not take into account patterns, neither spatial nor temporal, in the signal to be calibrated. In this paper, we propose to use techniques based on the extraction of regular temporal patterns in the signal to enhance the estimation of the in-situ calibration. This technique has been used in other fields such as face image reconstruction, where relevant information is extracted from a face, encoded, and then compared to a database of similarly encoded models (faces) [14]. Similarly, we will extract the relevant features from daily signals, encode them, and then compare them to a set of daily signals from reference instrumentation data stored in a database in what we call a temporal pattern-based denoising mechanism (TPB-D). In addition, to improve the denoised signal, we will recalibrate the signal using a linear mapping of the projected daily calibrated LCS signals to the projected daily reference signals in what we call a temporal pattern-based calibration (TPB-C) mechanism. Specifically, in this article we:

1) propose the TPB-D mechanism that takes daily signals from in-situ calibrated LCSs and projects them into a subspace created with a database of daily signals from reference instrumentation. The database is built with signals from reference instruments deployed near or in the same location as the LCSs deployment,
2) propose the TPB-C mechanism which, in the presence of a reference instrument co-located to the LCSs over a period of time, adds to the denoising stage a recalibration

Xhensilda Allka (xhensilda.allka@upc.edu), Pau Ferrer-Cid (pau.ferrer.cid@upc.edu), Jose M. Barcelo-Ordinas (jose.maria.barcelo@upc.edu) and Jorge Garcia-Vidal (jorge.garcia@upc.edu) are with the Department of Computer Architecture, Universitat Politecnica de Catalunya (UPC), Barcelona, Spain.

DOI 10.1109/TIM.2023.3239626

process using a linear mapping with regularization that maps the projected daily data from the in-situ calibrated LCSs onto the projected daily data from the co-located reference instrumentation,

3) analyze the parameters influencing the performance of the TPB-D and TPB-C mechanisms on four data sets taken from real IoT deployments in Spain and Italy with LCSs measuring tropospheric ozone (O_3) and temperature.

The outline of this paper is as follows: section II shows the related work. Section III describes the proposed temporal pattern-based denoising and calibration methods. Next, section IV presents the data sets used in this work, and section V evaluates the performance of the mechanisms applied to four data sets in real IoT deployments. Finally, section VI provides the conclusions of the paper.

II. Related Work

Air quality monitoring networks are composed of reference nodes with very accurate sensors that are continuously recalibrated. Recently, it has been proposed to join LCS nodes to the reference station network [15] forming an heterogeneous air quality monitoring network. Such LCS nodes incorporate sensors that are more inaccurate and whose calibration is performed at the time of deployment. The way to calibrate such LCSs in an uncontrolled environment (i.e., without specific calibration chambers) is to first place the sensors near accurate instruments such as reference stations in the deployment area and use supervised machine learning algorithms [2], [3], [5]. The data from the reference stations act as reference values of the supervised regression model and allow to obtain the hyperparameters of these models. This way of calibrating sensors with a machine learning algorithm with reference values taken with instruments that take accurate values is called in-situ calibration. The choice of calibration mechanism depends on the response of the sensor and in general can be linear or nonlinear. Most commonly, the calibration of an air quality sensor requires an array of sensors since the target pollutant to be estimated may have dependencies on several gases and environmental variables [1], [16], [17].

There is a large literature on how to calibrate LCSs with machine learning algorithms, both linear and nonlinear. Among the algorithms used to calibrate sensors for NO_2, NO, O_3, PM_x, CO, SH_2, CO_2, etc., we find the use of multiple linear regression (MLR) [6]–[8], [13], support vector regression (SVR) [6], [8], [11], random forest (RF) [6], [8], [11]–[13], Gaussian processes (GP) [11], K-nearest neighbor (KNN) [6], or neural networks (NN), [9], [10], [13]. More recently, the use of neural networks that take into account the temporal trend of the data has been proposed for air quality LCSs calibration [18], [19]. Nevertheless, the data quantity requirements of these models can be limiting in LCS environments and their applicability is in itself a field of research that needs its own in-depth study.

There are several techniques to recalibrate the sensors without removing and relocating them to a reference station. Saukh *et al.* [17] recalibrate LCSs located on top of buses when they encounter fixedly located reference stations. Cui *et al.* [20] propose that, after in-situ calibration, sensors should be recalibrated using mobile instruments that are brought close to the sensor location for a period of time. Tancev *et al.* [21] install reference instruments in arbitrary vehicles and recalibrate the LCSs when the vehicles are in proximity to the sensors.

Denoising signals by truncating the singular value decomposition (SVD) matrices up to a few largest singular value components, and then reconstructing a denoised data matrix by using these singular vectors is a well-known technique in signal processing with examples in electronics [22], computer vision [14], and other multimedia application areas [23]. Moreover, the SVD has been used in singular spectral analysis for forecast prediction in air quality monitoring networks [24]. In addition, the SVD has been used as a method of empirical orthogonal function (EOF) to identify spatial air quality index (API) patterns in China [25]. Ding *et al.* [26] made use of the SVD to identify the spatial correlation between aerosol concentrations and meteorology and urbanization indicators. Overall, the SVD has proven the be useful in the identification of spatio-temporal patterns as well as for signal and trend denoising.

The references presented in this related work show that the current state-of-the-art techniques for in-situ calibration provide point-to-point estimates using machine learning methods such as MLR, RF, SVR or ANN. To go beyond the limits of these methods and improve the estimates, we propose to combine these methods with other information obtained from the LCS and reference instrumentation data, such as temporal patterns. Summarizing, in this paper we propose the TPB-D and the TPB-C mechanisms for LCSs, which make use of the temporal patterns learned by means of the SVD of reference instrumentation data stored in a database. More precisely, these mechanisms pose three advances with respect the in-situ calibration state-of-the-art methods:

1) are models that provide an improvement of the estimate obtained by the already calibrated sensors,
2) the real air quality patterns are learned and trained using the singular value decomposition of reference instrumentation data, which is commonly available, instead of learning the patterns directly from the LCS measures,
3) the convexity of the models makes them easy to train, being of particular importance in IoT environments with possible calibration data constraints.

III. Temporal pattern-based denoising (TPB-D) and calibration (TPB-C) mechanisms

In this section, we present two temporal pattern-based mechanisms for LCS calibration based on two stages. In the first stage, the sensor is placed next to a reference instrumentation, e.g., a governmental reference station, for a period of time, and an in-situ sensor calibration is performed with a linear or nonlinear supervised machine learning algorithm. The temporal correlations present in data are not taken into account at this stage. For this purpose, the N samples taken are considered to be chosen with equal probability in the training data set or in the test data set.

TABLE I
LIST OF SYMBOLS.

Symbol	Meaning
$\mathbf{x_s} \mid \mathbf{x_R}$	Sensor signal | Reference instrument signal
$\mathbf{x}_c$	Estimated signal calibrated *in-situ*
N	# of samples for in-situ calibration
D	# of samples taken in a day
P	# of features for sensor calibration
M	# of days taken for denoising in TPB-D
L	# of days taken for training the linear mapping in TPB-C
$f(\cdot)$	Machine learning-based sensor calibration model
$\mathbf{s}_{\mathrm{O_3}} \mid \mathbf{s}_{\mathrm{NO_2}}$	O_3 | NO_2 sensors
$\mathbf{s}_{\mathrm{T}} \mid \mathbf{s}_{\mathrm{RH}}$	Temperature | Relative humidity sensors
U|V	Left | Right singular vector matrix
$\Sigma \mid \lVert\cdot\rVert_F$	Singular value matrix | Frobenius norm of a matrix
$\kappa \mid \lambda$	SVD subspace range | Regularization hyperparameter

In the second stage, an improvement of the in-situ calibration is performed by taking into account the daily temporal correlations of the data. A first enhancement mechanism is to use data obtained from a reference instrumentation, or a set of reference instruments, stored in a database and grouped into daily blocks, and where the temporal patterns present in the reference data are learned by singular value decomposition. Then, the daily calibrated LCS data are projected onto the left singular vectors obtained by SVD in a denoising step, TPB-D mechanism. A second mechanism consists of placing a reference instrument near the LCSs, so that, after the denoising step, the projected daily in situ calibrated LCS data are linearly mapped to the projected daily instrumentation reference data to recalibrate the sensor, TPB-C mechanism. This linear mapping uses regularization to avoid over-fitting the data. In this way, the denoising and linear mapping steps are decoupled from the in-situ calibration and a sensor already calibrated in-situ can be corrected. Table I summarizes the different symbols used throughout the paper. Bold lowercase/uppercase symbols denote vectors/matrices and lowercase symbols denote scalars.

A. In-situ calibration

In this first stage, the sensor is co-located in-situ at a reference station/instrumentation close to the deployment site, where the sensor is calibrated. This technique is commonly known as in-situ calibration in uncontrolled environments [2], [3], and consists of estimating air quality concentrations through supervised machine learning models (linear or nonlinear) using arrays of sensors [6], [8], [9], [12]. These models compare the raw sensor data with the data from the reference instrumentation and measure the error using metrics such as root mean square error (RMSE). Hence, during the calibration period, a set of tuples $\{(y_i, \mathbf{x}_i)\}_{i=1}^{\mathrm{N}}$ are obtained, where $\mathbf{x}_i \in \mathbb{R}^{\mathrm{P}}$ are the array of sensor measurements, where P is the size of the sensor array to be introduced as covariates in the calibration model, and $y_i \in \mathbb{R}$ are the reference values. Then, a machine learning model is assumed to learn the function $f{:}\mathbb{R}^{\mathrm{P}} \rightarrow \mathbb{R}$, mapping the low-cost sensors' values to calibrated sensor values, $y_i \sim f(\mathbf{x_i})$, Figure 1.

We use the root mean square error (RMSE) and the coefficient of determination (R^2) to evaluate the goodness-of-fit of the calibration models. In general, the difference in the choice of model lies in how linear the sensor behavior is. If the linearity is high, nonlinear methods do not improve a multiple linear regression. In case the sensors have nonlinear behavior, nonlinear models such as random forest, support vector regression, or k-nearest neighbor have shown similar performances in terms of RMSE and R^2 for the same data set if enough data is available [6]. In this paper, we use three of the most commonly used machine learning models for in-situ calibration in the literature; multiple linear regression (MLR), random forest (RF), and support vector regression (SVR). Thus, we explore the performance of one linear model and two nonlinear models. Further information about the in-situ calibration using supervised machine learning model can be found in the literature [6], [9], [12], [27].

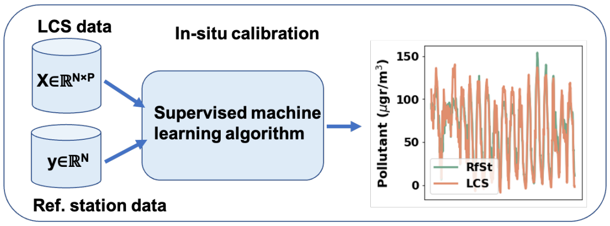


Fig. 1. In-situ calibration: raw data from an array of sensors is calibrated in-situ using supervised machine learning algorithms to produce the target sensor estimations.

B. Temporal pattern-based mechanisms

To improve the estimation, i.e., to decrease the RMSE value, we propose in this second stage to use the temporal daily correlations of reference instrumentation. To do this, calibrated sensor data are taken at a temporal granularity, in this case daily, to take advantage of the temporal relationships of the data. The temporal pattern-based (TPB) approach, Figure 2, consists of a denoising step based on a singular value decomposition generated subspace, in which the in-situ calibrated LCS data are projected onto the left-singular vector space obtained from a database of data collected by reference instruments on a daily basis. Then, the projected sensor signal can be recalibrated by a linear mapping onto the left singular vector space of the reference data projection. Finally, regularization is necessary to avoid over-fitting.

We observe three main factors for the TPB mechanisms to work. First, the signal to be calibrated with these methods present regular daily patterns, such as O_3 or temperature phenomena. Signals that have a lot of variability, such as $PM_{2.5}$ or NO, and that are very irregular, are more difficult to calibrate with this method, since the signal measured in one day may not be well represented in the database. The second factor is the quality of the estimation made by the in-situ calibration. Sensors whose calibration performance is already good in in-situ calibration have little room for improvement, while sensors with a not so good in-situ calibration performance can potentially be improved. The third factor is that depending on the reference data availability it is possible to perform only a denoising step (TPB-D) or a denoising step in conjunction with a linear mapping that recalibrates the LCSs (TPB-C). The denoising step requires a set of daily patterns in the database of one or more reference instruments at the LCSs location area with which to construct the subspace using the SVD. If in addition, we want to recalibrate the LCSs, we need to train the linear mapping with the projected in-situ calibrated signals taken at the same instants from the LCSs and from a co-located reference instrument. This linear mapping is similar to that followed in in-situ calibration, but on daily signals instead of individual measurements. If such co-located reference instrumentation is not available, only the TPB-D mechanism can be performed. In the following, we first describe the TPB-D, and then describe the TPB-C method that includes denoising and linear mapping of the projected signals.

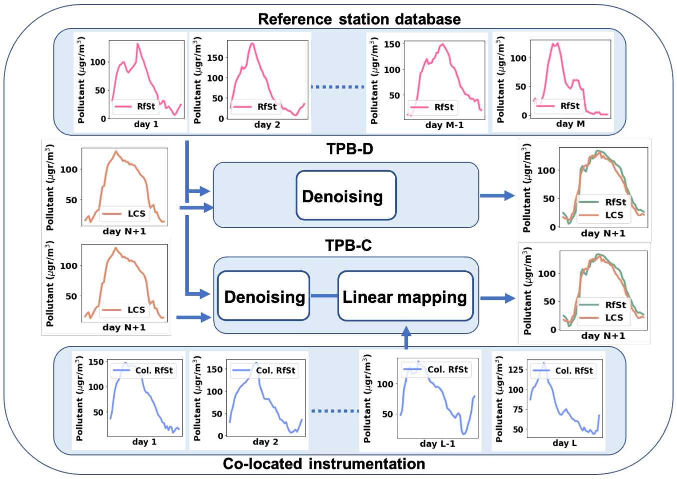


Fig. 2. Temporal pattern-based denoising (TPB-D) and calibration (TPB-C) mechanisms.

1) Temporal pattern-based denoising (TPB-D): consider, as an example of regular pattern signal, an air quality signal that follows daily patterns, such as tropospheric ozone (O_3). Let us now consider that we have a database or set of daily ozone signals, taken every interval T, produced by reference instruments such as reference governmental stations. Each signal is then represented as a vector $\mathbf{x}_{R_m}$ of dimension D, and we consider a set of M signals in the database[1]. The database signals are arranged in matrix form $\mathbf{X}_R{=}[\mathbf{x}_{R_1},\ldots,\mathbf{x}_{R_M}]{\in}\mathbb{R}^{\mathrm{D}\times\mathrm{M}}$. The objective is to compare a set of daily measurements taken by a LCS, and calibrated by an in-situ calibration model, with the set of daily measurements encoded in the database formed by the reference stations. Recall that this database of reference instrumentation may consist of a co-located reference instrument or a set of reference instruments deployed in the sensor area. Thus, it is not necessary to have an instrument in the same place as the sensor but the temporal trends can be approximated from measurements of instruments around the sensor. For this purpose, we compute the singular value decomposition of the centered data matrix, thus obtaining the eigenvectors [14] of the covariance matrix of the set of daily values of the reference station.

Let $\bar{\mathbf{x}}_R{=}\frac{1}{M}\sum_{m=1}^{\mathrm{M}}\mathbf{x}_{R_m}$ be the average daily reference instrument measurements, and $\hat{\mathbf{x}}_{R_m}{=}\mathbf{x}_{R_m}{-}\bar{\mathbf{x}}_R$ the centered version of each daily element of the database signals. Denoting by $\hat{\mathbf{X}}_R{=}[\hat{\mathbf{x}}_{R_1},\ldots,\hat{\mathbf{x}}_{R_M}]{\in}\mathbb{R}^{\mathrm{D}\times\mathrm{M}}$ the centered matrix, we obtain the SVD of $\hat{\mathbf{X}}_R$ as:

$$\hat{\mathbf{X}}_R = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^T \tag{1}$$

Where $\mathbf{U}{\in}\mathbb{R}^{\mathrm{D}\times\mathrm{D}}$ and $\mathbf{V}{\in}\mathbb{R}^{\mathrm{M}\times\mathrm{M}}$ are the left and right singular vector matrices, and $\boldsymbol{\Sigma}{\in}\mathbb{R}^{\mathrm{D}\times\mathrm{M}}$ is the diagonal singular value matrix, with singular values $\sigma_1{\geq}\sigma_2{\geq}\ldots{\geq}\sigma_D{\geq}0$. Then, applying the Eckart–Young theorem, the best κ-rank approximation of matrix $\hat{\mathbf{X}}_R$ can be obtained taking the singular vectors related to the κ-largest singular values, $\hat{\mathbf{X}}_{R_\kappa}{=}\mathbf{U}_\kappa\boldsymbol{\Sigma}_\kappa\mathbf{V}_\kappa^T$, with $\mathbf{U}_\kappa{\in}\mathbb{R}^{\mathrm{D}\times\kappa}$, $\mathbf{V}_\kappa{\in}\mathbb{R}^{\mathrm{M}\times\kappa}$ and $\boldsymbol{\Sigma}_\kappa{\in}\mathbb{R}^{\kappa\times\kappa}$ expressed in their truncated form. Afterwards, the LCS samples can be projected onto the subspace generated by the left-singular vectors $\mathbf{U}_\kappa$. The idea behind this operation lies in projecting the in-situ calibrated data into a subspace generated by the most important latent patterns of the reference instruments encoded in the database.

Suppose now that we have a new day of LCS data estimated with an in-situ calibration scheme as explained in section III-A. The aim is to improve the accuracy of the in-situ calibrated LCS data, encoded in the vector $\mathbf{x}_c{\in}\mathbb{R}^{\mathrm{D}}$, by projecting it onto the subspace generated by $\mathbf{U}_\kappa$, and then perform a signal reconstruction. First, we find $\hat{\mathbf{x}}_c{=}\mathbf{x}_c{-}\bar{\mathbf{x}}_R$, the difference between the daily in-situ calibrated LCS data with the daily average measurements of the reference stations in the database. The new estimated vector will be given by:

$$\tilde{\mathbf{x}}_c = \bar{\mathbf{x}}_R + \mathbf{U}_\kappa\mathbf{U}_\kappa^T\hat{\mathbf{x}}_c. \tag{2}$$

Note that in the first stage, in-situ calibration, we estimate the hourly data taking into account the interactions of other input variables such as temperature, relative humidity, or other gases, while in the second stage, we perform an adjustment of the in-situ calibrated data by projecting the in-situ calibrated LCS daily data in the subspace created by daily signals provided by the reference stations database. The algorithm 1 shows the in-situ calibration process and the denoising step; lines 1-2 perform the in-situ calibration using any supervised regression model. Lines 3-5 obtain a low-rank subspace from the daily reference instrumentation records in the database and lines 6-9 denoise the newly collected daily data.

Algorithm 1 Temporal pattern-based denoising (TPB-D).

```
Input: ${\mathbf{X}, \mathbf{y}, f(\cdot), \mathbf{X}_{R_M}, \kappa}$
1: $f \leftarrow$ Train_Model$(f, \mathbf{X}, \mathbf{y})$        ◁ In-situ calibration
2: $\mathbf{X}_c \leftarrow f(\mathbf{X})$
3: $\bar{\mathbf{x}}_R{=}1/M\sum_{m=1}^{M}\mathbf{x}_{R_m}$        ◁ Denoising stage
4: $\hat{\mathbf{x}}_{R_m} \leftarrow \mathbf{x}_{R_m}{-}\bar{\mathbf{x}}_R$; $\hat{\mathbf{X}}_{R_M} = [\hat{\mathbf{x}}_{R_1},\ldots,\hat{\mathbf{x}}_{R_M}]$
5: $\mathbf{U}, \boldsymbol{\Sigma}, \mathbf{V}^T \leftarrow$ SVD$(\hat{\mathbf{X}}_R)$
6: while $\mathbf{x}_{new}$ do        ◁ New daily data
7:     $\mathbf{x}_c \leftarrow f(\mathbf{x}_{new})$
8:     $\tilde{\mathbf{x}}_c \leftarrow \bar{\mathbf{x}}_R + \mathbf{U}_\kappa\mathbf{U}_\kappa^T(\mathbf{x}_c - \bar{\mathbf{x}}_R)$
9: end while
```

[1] For simplicity, in this paper, we assume that T=1 hour, and then D=24.

2) Temporal pattern-based calibration (TPB-C): in addition to the denoising step, to improve the sensor in-situ calibration, we need reference instrumentation co-located next to the LCSs during a time interval. In this case, we can perform a denoising operation together with a linear mapping of the daily in-situ calibrated LCSs projected onto the subspace generated by the left-singular vectors of the signals obtained in the database with the projection of signals produced by the reference instrument that is co-located next to the LCSs. Thus, once we have the sensor data on a daily basis $\mathbf{x}_c$ and the subspace generated by the left-singular vectors $\mathbf{U}_\kappa$ obtained by SVD on a set of daily signals taken from the database, we proceed to recalibrate the daily low-cost calibrated vectors using the daily data from a co-located reference instrument $\mathbf{x}_{R_l}\in\mathbb{R}^{\mathrm{D}}$, taking l=1, ..., L days. It is worth noting that the linear mapping requires L signals obtained by a reference instrument co-located at the LCS site during the same time period as the LCS signals. On the contrary, the denoising step requires M signals to learn the temporal patterns by means of SVD, which can come from one or several instruments in the LCS deployment area, and can be obtained during different time periods. Then, we project both the vector of daily in-situ calibrated sensor measurements and the vector of daily reference instrumentation measurements onto the subspace generated with the set of signals from the reference instrumentation database:

$$\boldsymbol{\alpha}_{c_l} = \mathbf{U}_\kappa^T(\mathbf{x}_{c_l} - \bar{\mathbf{x}}_R) = \mathbf{U}_\kappa^T \hat{\mathbf{x}}_{c_l} \quad (3)$$

$$\boldsymbol{\alpha}_{R_l} = \mathbf{U}_\kappa^T(\mathbf{x}_{R_l} - \bar{\mathbf{x}}_R) = \mathbf{U}_\kappa^T \hat{\mathbf{x}}_{R_l} \quad (4)$$

Now that both vectors of measurements $\{\boldsymbol{\alpha}_{c_l}, \boldsymbol{\alpha}_{R_l}\}$, with $\boldsymbol{\alpha}_{c_l}, \boldsymbol{\alpha}_{R_l}\in\mathbb{R}^\kappa$, lie in the same subspace, a correction of the LCS vector can be performed to approximate the low-cost in-situ calibrated sensor projected data to the reference instrumentation projected data. The estimation of the projections is a linear regression problem with coefficient matrix $\mathbf{\Phi}\in\mathbb{R}^{\kappa\times\kappa}$:

$$\mathbf{U}_\kappa^T \hat{\mathbf{x}}_{R_l} \approx \mathbf{\Phi}\mathbf{U}_\kappa^T \hat{\mathbf{x}}_{c_l} \quad (5)$$

Matrix $\mathbf{\Phi}$ is obtained by putting the problem in a regularized least squares form, where it is intended to obtain the matrix $\tilde{\mathbf{\Phi}}$ that minimizes the errors in the multivariate linear regression:

$$\min_{\mathbf{\Phi}} \|\mathbf{A}_{R_L} - \mathbf{\Phi}\mathbf{A}_{c_L}\|_F^2 + \lambda \|\mathbf{\Phi}\|_F^2 \quad (6)$$

Where $\mathbf{A}_{c_L} = [\boldsymbol{\alpha}_{c1}, \ldots, \boldsymbol{\alpha}_{cL}] = \mathbf{U}_\kappa^T \hat{\mathbf{X}}_{c_L}$ and $\mathbf{A}_{R_L} = [\boldsymbol{\alpha}_{R_1}, \ldots, \boldsymbol{\alpha}_{R_L}] = \mathbf{U}_\kappa^T \hat{\mathbf{X}}_{R_L}$ are the matrices formed appending the projected vectors measured during L days, and $\lambda\in\mathbb{R}$ is the hyperparameter that controls the regularization term penalizing the Frobenius norm of $\mathbf{\Phi}$. The regularization parameter λ controls the magnitude of the $\mathbf{\Phi}$ matrix, since due to the size of the matrix, there may be over-fitting. A closed-form expression for the recalibration matrix $\tilde{\mathbf{\Phi}}$ is obtained by solving the unconstrained optimization problem (6), where we have made the gradient of the objective function with respect to matrix $\mathbf{\Phi}$ equal to zero:

$$\tilde{\mathbf{\Phi}} = \mathbf{A}_{R_L}\mathbf{A}_{c_L}^T(\mathbf{A}_{c_L}\mathbf{A}_{c_L}^T + \lambda\mathbf{I}_\kappa)^{-1} \quad (7)$$

With $\mathbf{I}_\kappa\in\mathbb{R}^{\kappa\times\kappa}$ the identity matrix. The recalibration estimate of a new daily vector of LCS in-situ calibrated measurement $\mathbf{x}_c$ can be obtained as:

$$\tilde{\mathbf{x}}_c = \bar{\mathbf{x}}_R + \mathbf{U}_\kappa \tilde{\mathbf{\Phi}} \mathbf{U}_\kappa^T(\mathbf{x}_c - \bar{\mathbf{x}}_R) \quad (8)$$

The algorithm 2 shows the whole calibration process: lines 1-2 perform the in-situ calibration using any supervised regression model. Lines 3-5 obtain a low-rank subspace from the daily reference instrumentation records in the database and lines 6-9 project the daily LCS measurements and daily reference instrumentation data into the subspace. Line 10 trains the linear mapping model to find the optimal correction matrix $\tilde{\mathbf{\Phi}}$. Finally, lines 11-14 correct the newly collected daily data.

In terms of the computational complexity of the proposed models, the most expensive operations are SVD (2) and matrix multiplication and inversion (7). Since the matrices involved in this application are not large, there are no limitations, and no need to use advanced algorithms for matrix decomposition and multiplication, current linear algebra software can easily handle these operations. The computational cost of TPB-D is O(M^3) and that of TPB-C is O(M^3+κ^3).

Algorithm 2 Temporal pattern-based calibration (TPB-C).

```
Input: ${\mathbf{X}, \mathbf{y}, f(\cdot), \mathbf{X}_{R_M}, \mathbf{X}_{R_L}, \kappa, \lambda}$
 1: $f \leftarrow$ Train_Model($f, \mathbf{X}, \mathbf{y}$)                          ◁ In-situ calibration
 2: $\mathbf{X}_c \leftarrow f(\mathbf{X})$
 3: $\bar{\mathbf{x}}_R = 1/M \sum_{m=1}^{M} \mathbf{x}_{R_m}$                          ◁ Denoising stage
 4: $\hat{\mathbf{x}}_{R_m} \leftarrow \mathbf{x}_{R_m} - \bar{\mathbf{x}}_R$; $\hat{\mathbf{X}}_{R_M} = [\hat{\mathbf{x}}_{R_1}, \ldots, \hat{\mathbf{x}}_{R_M}]$
 5: $\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}^T \leftarrow$ SVD($\hat{\mathbf{X}}_R$)
 6: $\hat{\mathbf{x}}_{c_l} \leftarrow \mathbf{x}_{c_l} - \bar{\mathbf{x}}_R$; $\hat{\mathbf{X}}_{c_L} = [\hat{\mathbf{x}}_{c_1}, \ldots, \hat{\mathbf{x}}_{c_L}]$     ◁ Data projection
 7: $\hat{\mathbf{x}}_{R_l} \leftarrow \mathbf{x}_{R_l} - \bar{\mathbf{x}}_R$; $\hat{\mathbf{X}}_{R_L} = [\hat{\mathbf{x}}_{R_1}, \ldots, \hat{\mathbf{x}}_{R_L}]$
 8: $\mathbf{A}_{c_L} \leftarrow \mathbf{U}_\kappa^T \hat{\mathbf{X}}_{c_L}$
 9: $\mathbf{A}_{R_L} \leftarrow \mathbf{U}_\kappa^T \hat{\mathbf{X}}_{R_L}$
10: $\tilde{\mathbf{\Phi}} \leftarrow$ Train_$\mathbf{\Phi}$($\mathbf{A}_{c_L}, \mathbf{A}_{R_L}, \lambda$)          ◁ Linear mapping stage
11: while $\mathbf{x}_{new}$ do                                           ◁ New daily data
12:     $\mathbf{x}_c \leftarrow f(\mathbf{x}_{new})$
13:     $\tilde{\mathbf{x}}_c \leftarrow \bar{\mathbf{x}}_R + \mathbf{U}_\kappa \tilde{\mathbf{\Phi}} \mathbf{U}_\kappa^T(\mathbf{x}_c - \bar{\mathbf{x}}_R)$
14: end while
```

IV. Data sets

To demonstrate the performance of the TPB methods, we will use four data sets taken with LCSs in real IoT platforms. The first two data sets, D.1 and D.2, are O_3 data taken with metal-oxide (MOX) and electro-chemical (EQ) low-cost sensors during 2017 and 2018 in the H2020 CAPTOR project, where three testbeds were deployed in Austria, Italy and Spain [28]. For this study, we will use four nodes called Captor nodes deployed in the Spanish testbed, data set D.1, and two nodes called Raptor nodes, data set D.2, in the Italian testbed. The nodes were placed next to reference stations managed by governmental agencies that report data using high-precision instrumentation [29].

Each Captor node was developed by the Universitat Politecnica de Catalunya (UPC, Spain) and consists of a box with four SGX Sensortech MICS 2614 metal-oxide ozone (O_3) sensors, one air temperature (T), and one air relative humidity (RH) sensor (DHT1 Grove). For this paper, we have selected four nodes with sixteen O_3 sensors as target sensors, Table II, data

TABLE II

DATA SET 1 PRESENTS NODES WITH FOUR METAL-OXIDE (MOX) O_3 SENSORS AND A TEMPERATURE/RELATIVE HUMIDITY SENSOR PER NODE. DATA SET 2 PRESENTS NODES WITH ONE ELECTROCHEMICAL (EQ) O_3 SENSOR PER NODE. DATA SETS 3 AND 4 PRESENT ONE TEMPERATURE SENSOR PER NODE.

Data set	Node name	Sensor target	Sensor array labels	Sensor type	Calibration place	Period (dd/mm/yyyy)	# of days
D.1	C-17009	O_3	s_1,s_2,s_3,s_4/s_T/s_{RH}	MICS 2614/DHT1-Grove	Palau Reial (Spain)	19/06/2017-05/10/2017	80
D.1	C-17013	O_3	s_1,s_2,s_3,s_4/s_T/s_{RH}	MICS 2614/DHT1-Grove	Manlleu (Spain)	10/05/2017-4/10/2017	126
D.1	C-17016	O_3	s_1,s_2,s_3,s_4/s_T/s_{RH}	MICS 2614/DHT1-Grove	Vic (Spain)	27/05/2017-04/10/2017	114
D.1	C-17017	O_3	s_1,s_2,s_3,s_4/s_T/s_{RH}	MICS 2614/DHT1-Grove	Tona (Spain)	10/05/2017-29/09/2017	121
D.2	R-N69	O_3	s_1/s_{NO_2}/s_T/s_{RH}	OX-B431/NO2-B43F/DHT1-Grove	Monte Cucco (Italy)	21/06/2018-25/09/2018	84
D.2	R-N212	O_3	s_1/s_{NO_2}/s_T/s_{RH}	OX-B431/NO2-B43F/DHT1-Grove	Osio Sotto (Italy)	27/06/2018-24/09/2018	87
D.3	T-17009	T	s_T	DHT1-Grove	Palau Reial (Spain)	07/04/2017-04/10/2017	102
D.3	T-17016	T	s_T	DHT1-Grove	Vic (Spain)	26/05/2017-05/10/2017	116
D.4	T-17027	T	s_T	DHT1-Grove	Palau Reial (Spain)	05/02/2019-12/07/2019	125
D.4	T-17032	T	s_T	DHT1-Grove	Palau Reial (Spain)	06/02/2019-11/05/2019	93

set D.1: the node labeled C-17009 located in an urban area of the city of Barcelona called Palau Reial (Spain), the nodes labeled C-17013, C-17016, and C-17017 located in the cities of Manlleu, Vic and Tona (Spain), which are about 80 km away from Barcelona, in semi-urban areas.

The Raptor node was built by Universite Clermont Auvergne (UCA) in France. Each Raptor node includes one Alphasense OX-B431 electro-chemical O_3 sensor, one Alphasense NO2-B43F electro-chemical NO_2 sensor, a temperature sensor and a relative humidity sensor (DHT1 Grove). The Raptor platform is composed by two boxes: an outdoor box is powered by a 9V 4000mAh battery for a lifetime of three months, and connected using a IEEE802.15.4 (ZigBee) wireless access medium to an indoor box that acts as local server, powered by an external power supply and connected to Internet using Wifi or 3G. For this paper, we have selected two nodes with two electrochemical O_3 sensors as target sensors, Table II, data set D.2: node R-N69 located in Monte Cucco which is an urban area in the province of Piacenza (Italy) and node R-N212 located in Osio Sotto in Bergamo (Italy).

Finally, we define two other data sets called D.3 and D.4 with temperature as target sensors. The data set D.3 consists of two nodes in the cities of Barcelona (Palau Reial area, Spain) and Vic (Spain) during the year 2017. Dataset D.4 consists of two nodes in the city of Barcelona (Palau Reial area, Spain) during 2019. All these nodes included DHT1 Grove temperature sensors.

In all data sets, the samples are hourly (T=1 h) and are taken with the same aggregation strategy as the reference stations; minute samples are taken and aggregated into hourly samples [30]. In order to have daily 24-hour samples, in case of data loss, e.g. due to communication subsystem failures or node maintenance, any imputation method can be applied [31]. In our case, we perform an interpolation (average between previous and next value) if only one sample is missing and discard the whole day if there is more than one loss in one day.

V. RESULTS

In this section, we present different experiments we have tested using the proposed algorithms and their results.

First, we perform the in-situ calibration using linear and nonlinear machine learning models, while subsequently, by introducing the daily patterns, we show how TPB-D and TPB-C improve the accuracy of the in-situ calibration. The methodology explained in section III is implemented as follows:

1) to train the in-situ calibration model, 21 days are randomly selected and a data set of size 21 days times 24 hours (504 hourly samples) is generated. This data set is then divided into a training data set (80%) and a test data set (20%). Linear (multiple linear regression) and nonlinear (support vector regression and random forest) machine learning models are applied to in-situ calibrate the target sensors,
2) by selecting the best performing in-situ calibration model, the remaining days are estimated, and these calibrated data will be used in the TPB-D and TPB-C models. For the same time interval, a daily reference database is constructed for each of the data sets. These days are used to build the denoising step. Finally, 80% of the days are taken for training the linear mapping and 20% of the days for testing. The hyperparameters for the denoising and for the linear mapping method are chosen by performing a 10-fold cross-validation (CV) procedure on the training data.

The algorithms are implemented in Python 3.9.9, while the libraries that are used to obtain the results are pandas, NumPy, and sklearn. The computer used for the experiments has an Intel(R) Core(TM) i5-10600 CPU @ 3.30GHz 3.31 GHz processor with 16GB RAM[2].

A. In-situ calibration

For the in-situ calibration, linear and nonlinear models are used to calibrate the low-cost sensors. As low-cost gas sensors have cross-sensitivities with other gases and their performance also depends on temperature and relative humidity, in the calibration process for O_3 those will be taken into account [6], [7], [9], [11], [12]. For each metal-oxide O_3 low-cost sensor in data set D.1, a data matrix $\mathbf{X}=[\mathbf{s}_i, \mathbf{s}_T, \mathbf{s}_{RH}]\in\mathbb{R}^{\mathrm{N}\times 3}$, is generated with three feature columns, corresponding to a O_3 sensor, and the corresponding temperature and relative humidity sensors. In data set D.2, we are using electrochemical

[2] The code and the data sets used in this paper are available at authors web page http://sans.ac.upc.edu/?q=node/256

O_3 sensors and for this kind of technology the calibration performance also depends on the measurements of NO_2. Thus, for data set D.2 the model will be trained using the data matrix $\mathbf{X}=[\mathbf{s}_i, \mathbf{s}_{No_2}, \mathbf{s}_T, \mathbf{s}_{RH}]\in\mathbb{R}^{\mathrm{N}\times 4}$, while for data set D.3 and D.4, as the sensors are measuring temperature, and it doesn't have any cross sensitivity with any pollutant or phenomena, the data matrix for these data sets is $\mathbf{X}=[\mathbf{s}_T]\in\mathbb{R}^{\mathrm{N}}$. Once the data matrix for each one of the LCSs is generated, the calibration model will be trained using linear and nonlinear models. Multiple linear regression (MLR), support vector regression (SVR) and random forest (RF) are compared to calibrate the sensors.

Table III presents the results in terms of testing RMSE and R^2 for each of the sensors in each of the data sets considered. The results indicate that the nonlinear methods outperform the linear methods. In the case of temperature sensors, SVR gives the best results, while in the case of O_3 (metal-oxide and electrochemical) there is not a big difference between RF and SVR, but since SVR performs better than RF for most sensors, we select this method as the calibration method, and we will estimate all sensors using this method. We also observe a large variability in the results, even with sensors from the same manufacturer. For instance, among the MOX sensors placed in Vic, node C-17016, sensor s_1 has an RMSE of 17.29 µg/m^3 and R^2 of 0.83, and sensor s_4 has an RMSE of 11.49 µg/m^3 and R^2 of 0.93 with the same SVR method. The two electrochemical sensors also present variable results, one with RMSE of 13.65 µg/m^3 and R^2 of 0.88, and the other with RMSE of 11.60 µg/m^3 and R^2 of 0.94 with SVR. Finally, the temperature sensors present RMSE between $0.92°C$ and $1.5°C$ with R^2 of 0.85 and 0.75 respectively. In conclusion, this variability present in the in-situ calibration results shows room for improvement for the worst performing sensors.

TABLE III
IN-SITU CALIBRATION USING MULTIPLE LINEAR REGRESSION (MLR), SUPPORT VECTOR REGRESSION (SVR) AND RANDOM FOREST (RF). RESULTS SHOW TESTING RMSE AND R^2. THE BEST ESTIMATION METHOD FOR EACH SENSOR IS SHOWN IN BOLD.

Node Name	MLR		SVR		RF	
	RMSE (µg/m^3)	**R^2**	**RMSE** (µg/m^3)	**R^2**	**RMSE** (µg/m^3)	**R^2**
C-17009 s_1	14.64	0.7	**10.8**	**0.83**	11.7	0.8
C-17009 s_2	13.51	0.74	**10.75**	**0.83**	11.4	0.82
C-17009 s_3	10.47	0.81	9.1	0.86	**9.05**	**0.86**
C-17009 s_4	14.08	0.72	**10.1**	**0.85**	10.8	0.84
C-17013 s_1	11.41	0.93	**9.67**	**0.95**	10.32	0.94
C-17013 s_2	12.85	0.91	**9.72**	**0.95**	10.37	0.94
C-17013 s_3	17.12	0.85	**12.54**	**0.92**	12.62	0.92
C-17013 s_4	12.26	0.92	**10.32**	**0.94**	10.38	0.94
C-17016 s_1	19.34	0.79	17.29	0.83	**15.08**	**0.87**
C-17016 s_2	16.3	0.85	13.8	0.89	**13.4**	**0.9**
C-17016 s_3	15.2	0.87	12.78	0.9	**11.88**	**0.92**
C-17016 s_4	13.7	0.89	11.49	0.93	**10.39**	**0.94**
C-17017 s_1	12.45	0.89	**9.6**	**0.93**	10.2	0.92
C-17017 s_2	13.49	0.87	**10.51**	0.92	10.96	0.91
C-17017 s_3	14.89	0.84	13.15	0.87	**12.6**	**0.88**
C-17017 s_4	12.02	0.89	**9.37**	**0.94**	9.7	0.93
R-N212 s_1	13.72	0.87	13.65	0.88	**13.18**	**0.89**
R-N69 s_1	11.78	0.94	**11.6**	**0.94**	12.8	0.93
	RMSE (ºC)	**R^2**	**RMSE** (ºC)	**R^2**	**RMSE** (ºC)	**R^2**
T-17009 s_T	1.96	0.81	**1.5**	**0.75**	1.5	0.72
T-17016 s_T	1.02	0.97	**0.95**	**0.94**	1.01	0.93
T-17027 s_T	1.18	0.93	**1.1**	**0.95**	1.18	0.94
T-17032 s_T	1.27	0.82	**0.92**	**0.85**	0.93	0.83

B. Temporal pattern-based methods

We choose SVR as the calibration method, and with the hyperparameters obtained in section V-A, we estimate O_3 concentration values or temperature values for subsets of D.1-D.4 with 60-105 days depending on the data set. The new RMSE and R^2 values of this data set are shown in the first column of table IV. For clarity, we have chosen a single sensor from each node. Now, in order to improve the in-situ calibration of this new data by taking into account the daily pattern data, we will first learn the daily pattern data from the reference stations where the sensors were located and which are stored in the database to study the performance obtained with the TPB-D model. Next, we will investigate the performance of the TPB-C model, selecting both the daily patterns involved in the denoising step and those involved in the linear mapping training data set. In both cases, the validation of the TPB-D and TPB-C mechanisms is performed on a test data set. The results of this section consider that to construct the subspace in both TPB-D and TPB-C, we take those days from the database containing data from a reference station co-located with the LCSs.

1) TPB-D performance: Given the formerly in-situ calibrated data, we propose to improve the estimation by projecting into a low-rank subspace spanned by the κ most important left singular vectors. The only hyperparameter we need to estimate for this method is the range of the subspace over which we are projecting and then reconstructing the signal. In (2), when κ=24, the denoising step is exactly the in-situ calibration value since $\mathbf{U}\mathbf{U}^T=\mathbf{I}$, so there is no improvement for this value of the hyperparameter. On the other hand, when κ<24, $\mathbf{U}_\kappa\mathbf{U}_\kappa^T\neq\mathbf{I}$, and the signal is denoised. Figure 3 shows the average CV RMSE with respect to κ, the rank of the subspace. The value of κ changes for each data set and for each sensor in the same data set, but there is not high variability between them. Thus, for O_3 (metal-oxide and electrochemical) and for temperature sensors, the value of κ ranges between 3 and 9. In short, the cross-validation results show that in all cases denoising with low-rank subspaces improves the estimation since in none of the cases the minimum RMSE is reached for κ=24.

Table IV shows the testing RMSE and the coefficient of determination R^2 for the in-situ calibration and TPB-D method for one sensor from each node of each data set described in table II. For data set D.1, as each captor node has four O_3 sensors at each node, we have selected sensor s_1, while for other data sets the only sensor present at each node is selected. Comparing the results of the in-situ calibration on the test set before and after denoising, it is observed that for MOX sensors, the error in estimating O_3 concentrations decreases between 9.77% for sensor C-17016 s_1 and 17.58% for sensor C-17009 s_1. We note that good sensors with R^2 greater than 0.80 improve less, for example, the C-17016 s_1 increases R^2 from 0.83 to 0.86, while the C-17009 s_1 sensor has a larger margin of improvement with R^2=0.57 increasing to R^2=0.71. Electrochemical O_3 sensors and temperature sensors have a similar behavior. The worse the R^2 of the in-situ estimation, the better the percentage improvement provided by the TPB-

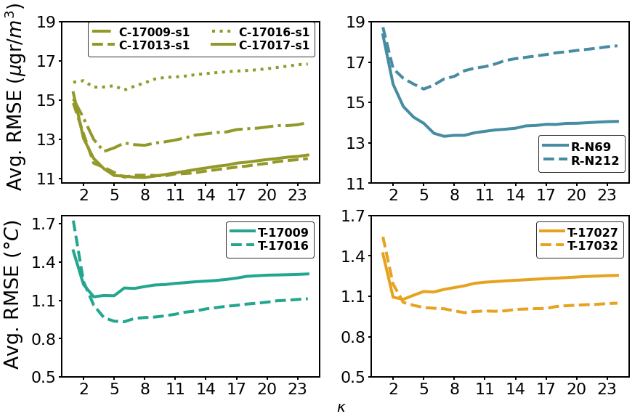


Fig. 3. Impact of the range of the κ subspace on the TPB-D model where lines represent the average CV RMSE. The upper left subplot represents the result for 4 MOX sensors in data set D.1; upper right: 2 EQ sensors in data set D.2; lower left: 2 temperature sensors in data set D.3; lower right: 2 temperature sensors in data set D.4.

D mechanism. However, we can observe that in all cases the percentage of improvement does not exceed 20%.

2) TPB-C performance: By introducing the correction matrix $\mathbf{\Phi}$, it is expected to have a better estimation of the sensor calibration with respect to only performing the denoising step. As mentioned in section III-B2, in contrast to the TPB-D model, where we only need a subspace created by the daily signals taken from the reference instrumentation, and where the signal obtained by the sensors is projected, now to train the model it is necessary to project both the daily signal captured by the reference instrumentation co-located to the LCS, and the daily signal captured by the LCS, calibrated in-situ, onto the subspace created by reference instrumentation signals stored in the database. Then, we perform a linear regression of the daily projected LCS signal on the daily projected signal of the reference co-located instrumentation. The coefficients to be estimated are the coefficients of the $\mathbf{\Phi}$ matrix, whose dimension is $\mathbb{R}^{\kappa \times \kappa}$.

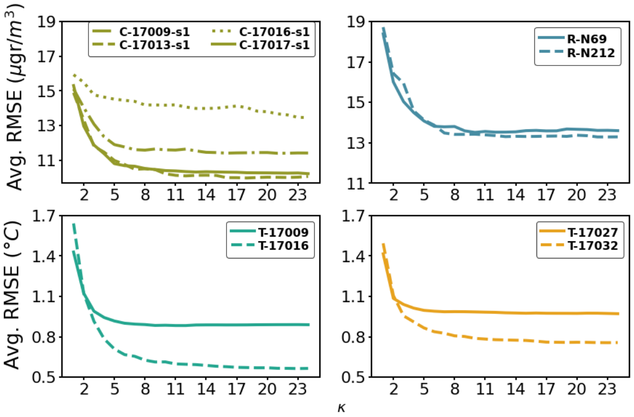


Fig. 4. Impact of subspace range κ in the TPB-C model: lines represent the average RMSE value in the CV data set. The upper left subplot represents the result for 4 MOX sensors in data set D.1; upper right: 2 EQ sensors in data set D.2; lower left: 2 temperature sensors in data set D.3; lower right: 2 temperature sensors in data set D.4.

In order to prevent over-fitting, besides estimating the subspace range κ, it is necessary to estimate the regularization parameter λ. Both hyperparameters, κ and λ, are estimated by performing a 10-fold CV procedure. Figure 4 plots the average CV RMSE curves when the best κ is set. The value of the subspace range varies from 9 to 24. Comparing with the denoising step, it is observed that here the value of the hyperparameter κ is higher. However, the regularization parameter λ in (6) overcomes the over-fitting problem when κ is large.

The TPB-C accuracy in terms of RMSE and R^2 are presented in table IV. The same sensors as in the denoising step are selected to show the improvement. For nodes with MOX O_3 technology sensors, the improvement ranges from 12.1% to 27.8%, while for nodes with EQ O_3 technology sensors, the improvement ranges from 9.36% to 26.71%. For the temperature sensors, the one with the least improvement is the T-17016 sensor that started with an in-situ calibration with an R^2 of 0.96, and improves by 15.65% (R^2=0.99), while the one with the most improvement is the T-17009 sensor that started with an in-situ calibration with an R^2 of 0.85, and improves by 38.8% (R^2=0.94). In conclusion, it is observed that TPB-C improves up to 20% with respect to TPB-D.

C. Impact of training data set size

In this section, the impact of the size of days taken to obtain the subspace where the data are projected in the TPB-D method is analyzed, and also the impact of the size of days taken to obtain the subspace where the data are projected and the linear mapping is trained in TPB-C is analyzed. To perform this analysis, the sensors C-17017s_1 as representative of O_3 and T-17016 as representative of temperature are selected. To proceed with the experiment, 20 days are randomly selected as the test data set. The performance of each method is analyzed if it is trained with a data set containing randomly selected blocks of 10, 11, 12, $\cdots$, 79 and 80 days for the O_3 and up to blocks of 75 days for temperature. For each block, 20 repetitions are performed by changing the seed in order to have a confidence interval calculated with a t-student with 95% confidence level. We observe, Figure 5, that the number of days required to train the model depends on the method and the sensor target. For sensors with very regular daily patterns such as temperature, the TPB-D denoising mechanism needs few data taken from the database, e.g., between 15 to 20 days, to converge. On the other hand, for sensors with significantly less regular daily patterns such as O_3, a larger data set, e.g., 30 to 40 days, is needed. In contrast, for the TPB-C method, more days are needed to train the model due to the linear mapping introduced since the optimal values of the coefficients of the Φ matrix have to be estimated. This implies that the higher the number of training data, the less over-fitting. It can be observed that for the O_3 sensor large training sets are needed, while for the temperature sensor, with more regular daily patterns, fewer days are needed.

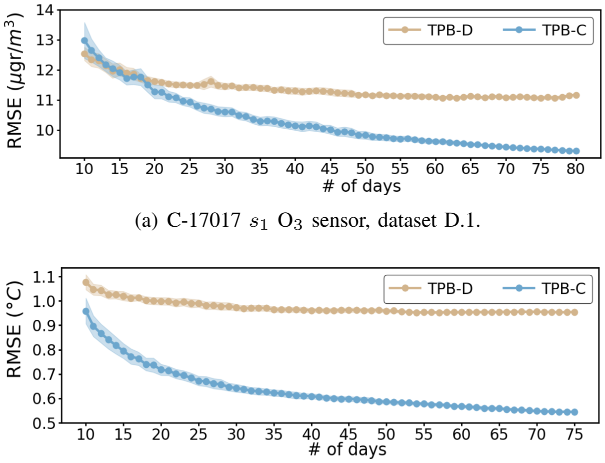
D. TPB-D performance using nearby reference stations

In the TPB-D method, it is not necessary to obtain the subspace in which the daily in-situ calibrated LCS data are

TABLE IV
PERFORMANCE OF THE IN-SITU CALIBRATION, TPB-D AND TPB-C ON TEST DATA SETS ALONG WITH THE BEST HYPERPARAMETERS FOUND BY CV.

Node Name	**In-situ (SVR)**		**TPB-D**			**TPB-C**			
	RMSE (μg/m^3)	**R^2**	**RMSE (μg/m^3)**	**R^2**	κ	**RMSE (μg/m^3)**	**R^2**	κ	λ
C-17009 s_1	15.24	0.57	12.56	0.71	4	11.38	0.76	21	3934
C-17013 s_1	11.66	0.92	10.01	0.94	6	9.34	0.95	16	3045
C-17016 s_1	15.55	0.83	14.03	0.86	6	13.67	0.89	24	4322
C-17017 s_1	12.88	0.87	11.17	0.90	5	9.30	0.93	24	4662
R-N69	14.07	0.92	12.61	0.93	7	13.12	0.93	10	4061
R-N212	20.51	0.76	18.34	0.81	5	15.03	0.87	23	4662
	RMSE (ºC)	**R^2**	**RMSE (ºC)**	**R^2**	κ	**RMSE (ºC)**	**R^2**	κ	λ
T-17009	1.8	0.85	1.56	0.89	3	1.10	0.94	9	17
T-17016	1.15	0.96	0.96	0.97	6	0.56	0.99	23	17
T-17027	1.5	0.89	1.3	0.91	3	1.22	0.93	20	58
T-17032	1.05	0.92	0.96	0.92	9	0.68	0.96	23	18

projected with a reference instrument co-located with the LCSs. The subspace can be created using reference stations that are in the vicinity of the LCSs and have a high correlation in their data with the calibrated LCSs. Although there are several techniques for discovering the relationships between network sensors, graph signal processing tools have been successfully applied in learning the graph that encodes the implicit relationships between sensors measuring natural phenomena. Thus, we have applied a smoothness-based graph learning technique [31], [32] in order to find the reference instruments most related to the node of interest and include these instruments in the reference station database. The results indicate that two nearby reference instruments can be included in the database for the TPB-D mechanism.



(a) C-17017 s_1 O$_3$ sensor, dataset D.1.

(b) T-17016 temperature sensor, dataset D.3.

Fig. 5. Impact of training data set size: lines depict the average RMSE of the TPB-D and TPB-C models.

In this case, in the absence of a reference instrument in place, the CV procedure to estimate the value of the hyperparameter κ cannot be performed. Gavish *et al.* [33] propose the recovery of low-rank matrices by introducing singular value thresholds, so to estimate the value of κ we will use the method proposed in [33]. Assuming that we want to recover the centered matrix $\hat{\mathbf{X}}_R{=}[\hat{\mathbf{x}}_{R_1},\ldots,\hat{\mathbf{x}}_{R_M}]{\in}\mathbb{R}^{D\times M}$, the threshold singular value is given as:

$$\tilde{\sigma}{=}w(\beta)*\sigma_{med} \tag{9}$$

where σ_{med} is the median of the singular values, while $w(\beta)$ is obtained as:

$$w(\beta){\approx}0.56\beta^3-0.95\beta^2+1.82\beta+1.43 \tag{10}$$

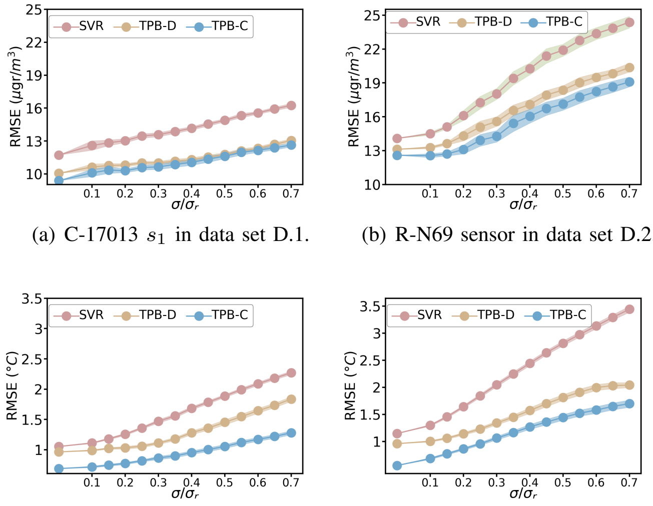
where $\beta{=}\mathrm{D/M}$, with D the dimension of daily data and M the number of days used by nearby reference stations. Finally, κ corresponds to the number of singular values that are greater than the threshold $\tilde{\sigma}$.

To show the impact of using nearby reference stations we have selected the C-17009 s_1 O$_3$ MOX sensor. The estimated κ value resulting from these methods is 5, which is close to the optimal values (κ=4) using the CV when using co-located reference instrumentation. The RMSE and R^2 values by applying the TPB-D are respectively 13.1 μgr/m^3 and 0.68. Comparing with the accuracy of the in-situ calibration, table IV, in which the values of RMSE and R^2 were 15.24 μgr/m^3 and 0.57, we observe that the use of the information from the neighboring reference station improves the quality of the calibrated data, although the efficiency gain is not as good as if the LCS is co-located next to the reference instrument were the RMSE and R^2 values were 12.56 μgr/m^3 and 0.71. The experiment shows how the TPB-D mechanism can be used when the LCS does not have a co-located reference instrument to project the daily values.

E. Noise sensitivity of the TPB methods

Generally, LCSs are prone to errors due to their data quality limitations. In this section, the sensitivity of the proposed method is studied, so our goal is to study how stable is the method's output $\tilde{\mathbf{x}}_c$ against different amounts of perturbations introduced into the sensor data. To simulate perturbations, we use a sampling procedure to introduce additive Gaussian noise ϵ into the input data (sensor raw data) with mean zero and variance σ^2, i.e., $\epsilon{\sim}N(0,\sigma^2)$. The sensors are calibrated in-situ with support vector regression. Increasing values of σ^2 emulate a worse sensor. In short, we evaluate how the in-situ calibration, the TPB-D model, and the TPB-C model behave with sensors perturbed with incremental values of error.

To show the sensitivity of the method, the sensor that performs best on each of the data sets is selected, respectively the C-17013 s_1, R-N69, T-17016, and T-17032 are selected. Figure 6 shows the testing set results when different amounts of noise are introduced, in terms of average RMSE and their confidence interval over 20 repetitions. Besides, the level of noise introduced is represented as the ratio between the standard deviation of the noise introduced σ and the sensor raw data standard deviation σ_r. The results show that the TPB-C method shows very good results in noise filtering and correction. Regarding the RMSE obtained by the in-situ calibration scheme, for O_3, it is observed that it is reduced between 9%-19% with TPB-D and 15%-22% with TPB-C, while for temperature it is reduced up to 40% with TPB-D and up to 50% with TPB-C. As a summary, in general, with moderate noise, the TPB-D reduces about 15% of the RMSE and TPB-C another 10% of the RMSE with respect to the in-situ calibration scheme. As for the value of the hyperparameters, in general for both TPB-D and TPB-C, we have observed that as the noise increases, the value of κ decreases, being between 3-4 for TPB-D and 7-8 for TPB-C. For higher values of noise, a higher value of the regularization hyperparameter λ is also required. Thus, the more perturbed the data are, the smaller the number of left singular vectors (i.e., patterns) we need to project and reconstruct the daily measurements and the higher the value of λ.



(a) C-17013 s_1 in data set D.1.

(b) R-N69 sensor in data set D.2.

(c) T-17016 sensor in data set D.3.

(d) T-17032 sensor in data set D.4.

Fig. 6. Noise sensitivity: lines represent the average RMSE in testing data set for SVR, TPB-D, and TPB-C models. The x-axis represents the ratio between the standard deviation of the perturbed data and the standard deviation of the raw data (σ/σ_r); confidence intervals (shaded area) are calculated as a t-student with 95% confidence level.

To sum up, the analysis for both methods has shown their capability to filter out and mitigate the effects of noisy data. For larger perturbations, the TPB-C model is less affected by noise than the TPB-D model, being able to filter out more noise. The value of the hyperparameters is also related to the value of the variance of the noise. Finally, the hyperparameter selection process shows how for large data perturbations the optimal number of left-singular vectors decreases.

VI. Conclusions

In this paper, we have proposed the temporal pattern-based denoising (TPB-D) and calibration (TPB-C) methods for the low-cost sensors. The proposed models are based on improving the performance of in-situ calibrated sensors by means of a machine learning model using temporal patterns extracted from a reference instrumentation database. In the first mechanism, TPB-D, the signal is denoised by a singular value decomposition using signals from reference stations in the vicinity of the sensor stored in a database. For the second mechanism, TPB-C, a reference instrument has to be placed for a period of time next to the LCSs, in the same way as the in-situ calibration, to perform a linear mapping with regularisation in addition to the denoising step.

The proposed algorithms have been tested using four data sets that comprise two data sets for tropospheric ozone LCSs (O_3 metal-oxide and O_3 electrochemical sensors) and two data sets for temperature LCSs. We have shown how these sensors can be calibrated in-situ with well-known machine learning mechanisms, such as multiple linear regression (MLR), random forest (RF) and support vector regression (SVR). Among these, SVR performs the best, and has been selected as the in-situ calibration method for the remainder of the paper. We have seen how the two mechanisms, TPB-D or TPB-C, improve the in-situ calibration results by 10-20% (TPB-D) and 20-40% (TPB-C) for the different sensors. In general, the improvement is greater when in-situ calibration performs worse, as both mechanisms have a greater margin of improvement than when in-situ calibration performs very well. Furthermore, we show that if there are nearby stations the TPB-D mechanism can use the data from these stations to improve the in-situ calibration, but for the second improvement, the TPB-C mechanism needs to be trained with a reference instrument co-located near the LCSs, in the same way as the in-situ calibration is performed. The behavior of the two mechanisms in the presence of data perturbations has also been analyzed, showing the ability of both methods to filter noise and improve accuracy with respect to in-situ calibration. As future work it would be interesting to study the use and adaptation of these techniques in the sensor relocation problem showing the possibility to create general or invariant temporal patterns and the possibility to use non-linear mappings for correcting the projected signals.

TPB-D and TPB-C algorithms work well when two circumstances are present. First, the estimates produced by the in-situ calibration method do not perform very well in terms of RMSE or R^2, and second, when the signal measured by the sensors has temporal patterns. On the other hand, when the in-situ calibration of the sensor performs very well or when the phenomenon has no temporal patterns, the method will only have the ability to denoise the signal. Moreover, temporal pattern-based (TPB) methods have two good qualities: first, in the presence of reference instrumentation, we can apply TPB-C that denoises and helps to recalibrate the sensor, and second, the method is robust in presence of noise, and in the worst case where we do not have reference instrumentation available to co-locate with the sensor, but there is reference instrumentation nearby, we can apply the TPB-D method and denoise the signal improving the quality of the estimates.

Acknowledgment

This work is supported by the National Spanish funding PID2019-107910RB-I00, and regional project 2017SGR-990.

References

[1] F. Concas, J. Mineraud, E. Lagerspetz, S. Varjonen, X. Liu, K. Puolamäki, P. Nurmi, and S. Tarkoma, "Low-cost outdoor air quality monitoring and sensor calibration: A survey and critical analysis," *ACM Trans. Sen. Netw.*, vol. 17, no. 2, may 2021.

[2] J. M. Barcelo-Ordinas, M. Doudou, J. Garcia-Vidal, and N. Badache, "Self-calibration methods for uncontrolled environments in sensor networks: A reference survey," *Ad Hoc Networks*, vol. 88, pp. 142–159, 2019.

[3] B. Maag, Z. Zhou, and L. Thiele, "A survey on sensor calibration in air pollution monitoring deployments," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4857–4870, Dec 2018.

[4] A. Lewis, W. R. Peltier, and E. von Schneidemesser, "Low-cost sensors for the measurement of atmospheric composition: overview of topic and future applications," *World Meteorological Organization*, 2018.

[5] F. Delaine, B. Lebental, and H. Rivano, "In situ calibration algorithms for environmental sensor networks: A review," *IEEE Sensors Journal*, vol. 19, no. 15, pp. 5968–5978, 2019.

[6] P. Ferrer-Cid, J. M. Barcelo-Ordinas, J. Garcia-Vidal, A. Ripoll, and M. Viana, "A comparative study of calibration methods for low-cost ozone sensors in iot platforms," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9563–9571, Dec 2019.

[7] ——, "Multi-sensor data fusion calibration in iot air pollution platforms," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 3124–3132, 2020.

[8] J. M. Cordero, R. Borge, and A. Narros, "Using statistical methods to carry out in field calibrations of low cost air quality sensors," *Sensors and Actuators B: Chemical*, vol. 267, pp. 245–254, 2018.

[9] L. Spinelle, M. Gerboles, M. G. Villani, M. Aleixandre, and F. Bonavitacola, "Field calibration of a cluster of low-cost commercially available sensors for air quality monitoring. part b: NO, CO and CO2," *Sensors and Actuators B: Chemical*, vol. 238, pp. 706–715, 2017.

[10] P. Han, H. Mei, D. Liu, N. Zeng, X. Tang, Y. Wang, and Y. Pan, "Calibrations of low-cost air pollution monitoring sensors for co, no2, o3, and so2," *Sensors*, vol. 21, no. 1, p. 256, 2021.

[11] P. Nowack, L. Konstantinovskiy, H. Gardiner, and J. Cant, "Machine learning calibration of low-cost no 2 and pm 10 sensors: non-linear algorithms and their impact on site transferability," *Atmospheric Measurement Techniques*, vol. 14, no. 8, pp. 5637–5655, 2021.

[12] A. Bigi, M. Mueller, S. K. Grange, G. Ghermandi, and C. Hueglin, "Performance of no, no_2 low cost sensors and three calibration approaches within a real world application," *Atmospheric Measurement Techniques*, vol. 11, no. 6, pp. 3717–3735, 2018.

[13] R. R. Kureshi, B. K. Mishra, D. Thakker, R. John, A. Walker, S. Simpson, N. Thakkar, and A. K. Wante, "Data-driven techniques for low-cost sensor selection and calibration for the use case of air quality monitoring," *Sensors*, vol. 22, no. 3, 2022.

[14] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[15] N. H. Motlagh, E. Lagerspetz, P. Nurmi, X. Li, S. Varjonen, J. Mineraud, M. Siekkinen, A. Rebeiro-Hargrave, T. Hussein, T. Petaja *et al.*, "Toward massive scale air quality monitoring," *IEEE Communications Magazine*, vol. 58, no. 2, pp. 54–59, 2020.

[16] S. De Vito, E. Esposito, N. Castell, P. Schneider, and A. Bartonova, "On the robustness of field calibration for smart air quality monitors," *Sensors and Actuators B: Chemical*, vol. 310, p. 127869, 2020.

[17] O. Saukh, D. Hasenfratz, and L. Thiele, "Reducing multi-hop calibration errors in large-scale mobile sensor networks," in *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*, 2015, pp. 274–285.

[18] H. Yu, Q. Li, R. Wang, Z. Chen, Y. Zhang, Y.-a. Geng, L. Zhang, H. Cui, and K. Zhang, "A deep calibration method for low-cost air monitoring sensors with multilevel sequence modeling," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 9, pp. 7167–7179, 2020.

[19] Y. Cheng, O. Saukh, and L. Thiele, "Sensorformer: Efficient many-to-many sensor calibration with learnable input sub-sampling," *IEEE Internet of Things Journal*, 2022.

[20] H. Cui, L. Zhang, W. Li, Z. Yuan, M. Wu, C. Wang, J. Ma, and Y. Li, "A new calibration system for low-cost sensor network in air pollution monitoring," *Atmospheric Pollution Research*, vol. 12, no. 5, p. 101049, 2021.

[21] G. Tancev and F. G. Toro, "Stochastic online calibration of low-cost gas sensor networks with mobile references," *IEEE Access*, vol. 10, pp. 13 901–13 910, 2022.

[22] S. K. Jha and R. Yadava, "Denoising by singular value decomposition and its application to electronic nose data processing," *IEEE Sensors Journal*, vol. 11, no. 1, pp. 35–44, 2010.

[23] M. Turk, "Over twenty years of eigenfaces," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 9, no. 1s, pp. 1–5, 2013.

[24] F. Espinosa, A. B. Bartolomé, P. V. Hernández, and M. Rodriguez-Sanchez, "Contribution of singular spectral analysis to forecasting and anomalies detection of indoors air quality," *Sensors*, vol. 22, no. 8, p. 3054, 2022.

[25] L. Zhang, Y. Liu, and F. Zhao, "Singular value decomposition analysis of spatial relationships between monthly weather and air pollution index in china," *Stochastic environmental research and risk assessment*, vol. 32, no. 3, pp. 733–748, 2018.

[26] S. Ding, J. He, D. Liu, R. Zhang, and S. Yu, "The spatially heterogeneous response of aerosol properties to anthropogenic activities and meteorology changes in china during 1980–2018 based on the singular value decomposition method," *Science of The Total Environment*, vol. 724, p. 138135, 2020.

[27] E. Esposito, S. De Vito, M. Salvato, G. Fattoruso, and G. Di Francia, "Computational intelligence for smart air quality monitors calibration," in *International Conference on Computational Science and Its Applications*. Springer, 2017, pp. 443–454.

[28] A. Ripoll, M. Viana, M. Padrosa, X. Querol, A. Minutolo, K. M. Hou, J. M. Barcelo-Ordinas, and J. García-Vidal, "Testing the performance of sensors for ozone pollution monitoring in a citizen science approach," *Science of the Total Environment*, vol. 651, pp. 1166–1179, 2019.

[29] J. M. Barcelo-Ordinas, P. Ferrer-Cid, J. Garcia-Vidal, M. Viana, and A. Ripoll, "H2020 project captor dataset: Raw data collected by low-cost mox ozone sensors in a real air pollution monitoring network," *Data in Brief*, vol. 36, p. 107127, 2021.

[30] P. Ferrer-Cid, J. Garcia-Calvete, A. Main-Nadal, Z. Ye, J. M. Barcelo-Ordinas, and J. Garcia-Vidal, "Sampling trade-offs in duty-cycled systems for air quality low-cost sensors," *Sensors*, vol. 22, no. 10, p. 3964, 2022.

[31] P. Ferrer-Cid, J. M. Barcelo-Ordinas, and J. Garcia-Vidal, "Data reconstruction applications for iot air pollution sensor networks using graph signal processing," *Journal of Network and Computer Applications*, p. 103434, 2022.

[32] P. Ferrer-Cid, J. M. Barcelo-Ordinas, and J. Garcia-Vidal, "Graph learning techniques using structured data for iot air pollution monitoring platforms," *IEEE Internet of Things Journal*, vol. 8, no. 17, pp. 13 652–13 663, 2021.

[33] M. Gavish and D. L. Donoho, "The optimal hard threshold for singular values is $4/\sqrt{3}$," *IEEE Transactions on Information Theory*, vol. 60, no. 8, pp. 5040–5053, 2014.

Xhensilda Allka is a Ph.D. student at the Statistical Analysis of Networks and Systems (SANS) research group, Universitat Politecnica de Catalunya (UPC). She holds a B.Sc and a M.Sc in Engineering of Mathematics and Informatics by the University of Tirana (UT). Her main research interests are the potential applications of data analytic methods to sensor data from IoT monitoring platforms.

Pau Ferrer-Cid is a PhD student at the Statistical Analysis of Networks and Systems (SANS) research group, Universitat Politecnica de Catalunya (UPC). He holds a B.Sc in Computer Science and a M.Sc in Data Science by the UPC. His main research interests are the applications of novel data analysis methods to sensor data coming from IoT platforms and the analysis of other kinds of data from fields like biology and computer vision.

Jose M. Barcelo-Ordinas is an Associate Professor at Universitat Politecnica de Catalunya (UPC) from 1999. He holds a PhD and B.Sc+M.Sc in Telecommunication Engineering and a B.Sc+M.Sc in Mathematics. He has participated in many European projects such as WIDENS, EuroNGI, EuroNFI, EuroNF NoE and H2020 CAPTOR. His currently research areas are wireless sensor networks, mobility patterns, and the statistical analysis of sensor data.

Jorge Garcia-Vidal holds a PhD in Telecommunication engineering (1992). He is a full professor at the Computer Architecture Department of UPC since 2003, and a senior researcher at Barcelona Supercomputing Center (BSC-CNS) since 2012. His main current research interest is in problems related with the capture, processing and statistical analysis of sensor data.