






# Evaluating Correlations in IoT Sensors for Smart Buildings

Davide Andrea Guastella<sup>1,3</sup><sup>a</sup>, Nicolas Verstaevl<sup>2</sup><sup>b</sup>, Cesare Valenti<sup>3</sup><sup>c</sup>,  
Bilal Arshad<sup>4</sup><sup>d</sup> and Johan Barthélemy<sup>4</sup><sup>e</sup>

<sup>1</sup>*Institut de Recherche en Informatique de Toulouse, Université Toulouse III Paul Sabatier, France*

<sup>2</sup>*Institut de Recherche en Informatique de Toulouse, Université Toulouse I Capitole, France*

<sup>3</sup>*Dipartimento di Matematica e Informatica, Università degli Studi di Palermo, Italy*

<sup>4</sup>*SMART Infrastructure Facility, University of Wollongong, Wollongong 2522, NSW, Australia*

**Keywords:** Smart Building, Smart Cities, IoT Sensors, Evolutionary Approach, Sensors Correlation.

**Abstract:** In this paper we introduce a dataset of environmental information obtained via indoor and outdoor sensors deployed in the SMART Infrastructure Facility of the University of Wollongong (Australia). The acquired dataset is also made open-sourced along with this paper. We also propose a novel approach based on an evolutionary algorithm to determine pairs of correlated sensors. We compare our approach with three other standard techniques on the same dataset: on average, the accuracy of the evolutionary method is about 62,92%. We also evaluate the computational time, assessing the suitability of the proposed pipeline for real-time applications.

## 1 INTRODUCTION AND MOTIVATIONS

Smart buildings, like smart cities, are truly complex systems (Nigon et al., 2017). Indeed, if we set aside the social and organizational aspects of smart buildings and focus only on technology, smart buildings are equipped with numerous heterogeneous sensors and networks. Smart buildings are open, meaning that new sensors and data can appear or disappear at any time, and they have to face non-linear and unpredictable dynamics. Putting humans in the loop to design adaptive people-centric control systems will add another layer of complexity. This involves designing systems equally complex, as expressed by Ashby's law of requisite variety claiming that *if a system is to be stable, the number of states of its control mechanism must be greater than or equal to the number of states in the system being controlled* (Ashby, 1991).


This level of complexity, combining a dynamic and unpredictable nature of malfunction events, the noisiness of the data, and human activities make those


systems difficult to design through traditional methods, as not all states of the system are known *a priori*.


In this paper, we propose and evaluate a novel methodology to dynamically exploit the correlations between sensors to provide experts with automatic information about the building state. Studying how sensors are correlated or uncorrelated allows detecting variations of behavior in the building that might be indicative of an anomaly such as a malfunctioning air conditioning system, or a window that stays open during the night (Houssin et al., 2020). Because those anomalies can occur unexpectedly, the correlations must be processed in real-time (i.e. within fixed time constraints).


The contribution of this article is twofold:


- we introduce an **open dataset of one year of heterogeneous environmental information** acquired through internal and external sensors deployed in the SMART Infrastructure Facility at the University of Wollongong (Australia). Making this dataset open sourced will allow the scientific and industrial communities to have a consistent set of environmental information to be used for simulations, testing applications and comparing methodologies in a smart building context;
- we **propose a novel solution based on an evolutionary algorithm to detect in real-time highly correlated pairs of heterogeneous sensors.**

<sup>a</sup> <https://orcid.org/0000-0002-6865-1833>

<sup>b</sup> <https://orcid.org/0000-0002-7879-6681>

<sup>c</sup> <https://orcid.org/0000-0002-4961-2054>

<sup>d</sup> <https://orcid.org/0000-0002-6078-0235>

<sup>e</sup> <https://orcid.org/0000-0002-7800-5309>

Our proposal addresses the properties of **openness**, **heterogeneity** and **unpredictability** (Guastella and Valenti, 2018). The novelty of our contribution is:

- proposed an approach to determine pairs of highly correlated sensors whose information dynamics are similar in time;
- it does not make any assumption on the topography of the environment where the sensors are deployed. This property makes the solution generic so that it can be applied regardless of the building configuration.
- proposed a methodology that uses heterogeneous information to measure the correlation between pairs of sensors. Moreover, our approach exploits sensors whose type of perceived information is not known *a priori*.

The rest of this paper is organized as follows: Section 2 presents the proposed approach, describing how it is capable of determining pairs of correlated sensors based on their perceptions. Section 3 presents the results obtained from our method using the data acquired from the SMART Facility. In Section 4, we conclude and point out some future perspectives.

## 2 METHODOLOGY

In this section, we present a novel approach that reminds an evolutionary algorithm for determining correlations among sensors in smart buildings.

Along with this method, we present three other original pipelines to compare the results obtained from the evolutionary approach and also to illustrate techniques that can be used to exploit the introduced dataset. Fig. 1 shows the main steps of our methodology and the three other pipelines. The described techniques do not require any *a priori* information on the topology of the environment. All the input parameters we used have been obtained on an experimental basis; our results refer to their best configuration.

The following section describes the proposed pipeline. Sections 2.2, 2.3 and 2.4 present the techniques used to compare the results obtained through the evolutionary method.

### 2.1 Evolutionary Technique

The proposed approach (diagram ① in Fig. 1) consists of three steps: (i) acquiring information from the sensors; (ii) smoothing the data; (iii) applying the evolutionary approach using the information resulting from the previous operations.

Our technique takes inspiration from genetic algorithms, even though for this research we considered only one generation round. Indeed, although in our particular example the number of sensors is limited, we propose a general technique (i.e. with any number of sensors, eventually heterogeneous). To make sure that computation is performed in a predefined amount of time, we applied both crossover and mutation operators on the population every time until at least one sensor acquires new information.

#### 2.1.1 Coding

The population has 30 individuals (possible solutions). Each individual consists of a vector of 5 *genes*, where each gene represents the information acquired by a couple of sensors during the last two hours. In our experiments, we have made sure that all sensors acquire data every 2 minute, thus a gene is composed of two vectors of length 60. These values can be tailored to a specific problem.

#### 2.1.2 Fitness Function

To evaluate the goodness of a solution, it is necessary to define a numerical function that returns a score for each individual. Let  $S_k = \{P_1, P_2, \dots, P_n\}$  be the  $k$ -th individual with  $n = 5$  genes. A gene  $P_i = (T_i^{(1)}, T_i^{(2)})$  contains the times series  $T_i^{(1)}$  and  $T_i^{(2)}$  obtained by a given pair of sensors.

For performance reasons, we pre-calculate the smoothed version  $\bar{T}_i^{(1)}$  and  $\bar{T}_i^{(2)}$  of  $T_i^{(1)}$  and  $T_i^{(2)}$  as described in (Guastella et al., 2019); this technique was modified to provide an estimate for each available information.

The correlation  $\rho_i$  between  $T_i^{(1)}$  and  $T_i^{(2)}$  is:

$$\rho_i^k = |d(T_i^{(1)}, \bar{T}_i^{(1)}) - d(T_i^{(2)}, \bar{T}_i^{(2)})|, \quad (1)$$

with the distance between two time series  $T_a$  and  $T_b$ :

$$d(T_a, T_b) = \sum_{j \in [1, \gamma]} |t_j^{(a)} - t_j^{(b)}|, \quad (2)$$

where  $t_j^{(a)}$  and  $t_j^{(b)}$  are the  $j$ -th information of the data series  $T_a$  and  $T_b$  respectively. The smaller the difference  $d$  between two time series is, the more similar these two time series are.

The fitness of the individual  $S_k$  is calculated as the average correlation among the sensors in  $P_i \in S_k, \forall i$ :

$$f(S_k) = \frac{\sum_i \rho_i^k}{n}. \quad (3)$$

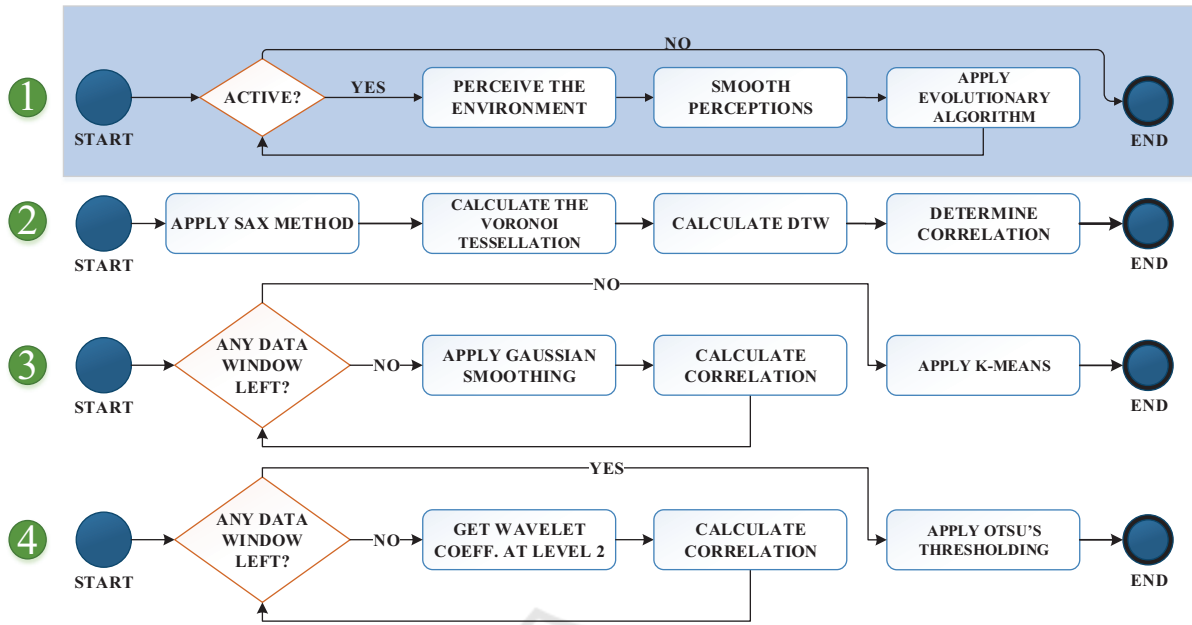


Figure 1: Main steps of the proposed evolutionary method for determining highly correlated pairs of sensors in smart buildings (diagram ①) and the pipelines used for comparing the obtained results (diagrams ②–④).

### 2.1.3 Crossover Operator

We have implemented the crossover operator as a single cut in a random position. This operator returns two child individuals and to maintain constant the size of the population over the entire process, both parents and children are compared together according to their fitness and only the best two of them are chosen to remain in the population. This means that the possibility that only parents remain alive is allowed, compared to other techniques that replace parents with offsprings, anyhow and independently of their quality. The two individuals to be recombined are chosen randomly: this does not exactly follow the evolutionary strategy but allows a greater gene variability.

### 2.1.4 Mutation Operator

To ensure the effective exploration of the solution space, the mutation operator replaces the sensors coded by a gene with a couple of sensors taken randomly. This operator is carried out on just one gene per individual.

### 2.1.5 Output

This evolutionary approach is carried out on a single generation and identifies quickly couples of possible correlated sensors. There is no guarantee of obtaining a correct solution, but this can be used as pre-processing for the criteria described in 3.2.

## 2.2 SAX-based Technique

The SAX-based pipeline (diagram ② in Fig. 1) uses moving average filtering, the SAX method (Wang et al., 2019), *Voronoi tessellation* (Guastella and Valenti, 2018) and *Dynamic Time Warping* (DTW) (Kenji Iwana and Uchida, 2020). Fig. 2 shows the individual steps of the technique. The first step involves acquiring raw data from sensors, the noise is then removed by applying a moving average filter of size 3 to each time series. The SAX method is applied to transform the denoised time series into strings, where the number of alphabets is limited to 20 symbols; this transformation emphasizes the differences between the values perceived by the sensors. Each time series, now converted to a string, is then compared with the others to determine the one that minimizes the correlation according to the DTW distance. The Voronoi tessellation is then applied to determine the rough relevance area sensors.

Let  $TS = (ts_1, ts_2, \dots, ts_n)$  be the set of time series obtained each one from the sensors  $S = (s_1, s_2, \dots, s_n)$ . Each time series has been denoised and transformed to string. A sensor  $s_i$  is correlated with a sensor  $s_k \in S$  if their Voronoi regions are adjacent and the DTW distance between their time series is minimized:

$$\operatorname{argmin}_k (\operatorname{DTW}(ts_i, ts_k) \wedge \operatorname{adj}(R_{s_i}, R_{s_k})) \quad (4)$$

where  $\operatorname{DTW}(ts_i, ts_k)$  calculates the DTW between the time series  $ts_i$  and  $ts_k$  respectively,  $\operatorname{adj}$  is a boolean

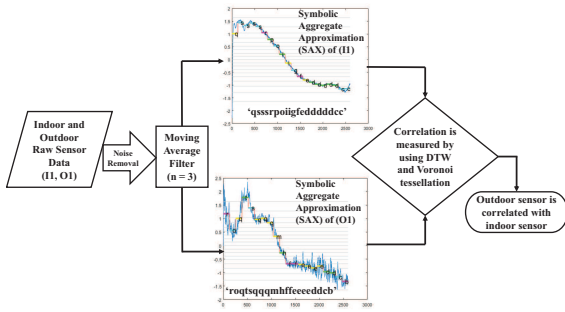


Figure 2: Block Diagram of SAX pipeline.

operator that returns 1 if the Voronoi regions  $R_{s_i}$  and  $R_{s_k}$ , associated to  $s_i$  and  $s_k$  respectively, are adjacent.

### 2.3 Gaussian Smoothing based Technique

This technique (diagram ④ in Fig. 1) is based on the Gaussian smoothing technique (Pociask et al., 2018) and  $k$ -means clustering (Shyr-Shen et al., 2018): this technique calculates the difference between time series and smoothed versions, then measures the correlation and apply a  $k$ -means clustering algorithm to separate the sensors into two groups (i.e. indoor, outdoor).

This pipeline iterates all the possible pairs of sensors. Furthermore, for each pair the algorithm iterates on blocks of 60 information.

Let  $D_i$  and  $D_k$  be the windows of 60 information perceived from the sensors  $s_i$  and  $s_k$  respectively, relative to the time interval  $T = [t - 60, t]$  with  $|T| = 60$ . The method calculates two sets  $DS_i$  and  $DS_k$  where  $ds_i^\ell \in DS_i$  and  $ds_k^\ell \in DS_k$  are computed by smoothing  $\ell$  times the data windows  $D_i$  and  $D_k$  through the Gaussian algorithm. The pipeline evaluates a set  $C$  of correlations, where  $c^\ell \in C$  is the average of the differences between  $D_i$  and  $D_k$  and the respective data windows obtained by applying  $\ell$  times the Gaussian smoothing:

$$c^\ell = \frac{\sum(|ds_i^\ell - D_i| - |ds_k^\ell - D_k|)}{w} \quad (5)$$

where  $ds_i^\ell \in DS_i$  and  $ds_k^\ell \in DS_k$  are the smoothed time series,  $D_i$  and  $D_k$  the input time series,  $w = |D_i| = |D_k| = 60$  is the number of samples for each time series. The standard deviation of the values in  $C$  is the output of the method for a pair of sensors  $s_i$  and  $s_k$ .

Compared to the previous pipeline, this approach process data from 60 samples, consecutive in time, at each iteration. We examine all the possible pairs of sensors for each sample window: this requires a significant amount of computational time. Although the algorithm has determined 89% of correctly correlated

pairs, the computational time ( $\sim 82$  seconds) makes the application unsuitable in real-time contexts.

### 2.4 Pearson based Technique

The third technique (diagram ④ in Fig. 1) uses the Pearson correlation coefficient (Zhou et al., 2016) and wavelet decomposition (Sciortino et al., 2017). This pipeline iterates over windows of 60 samples, consecutive in time, for all the pairs of sensors in the dataset. The first step of the pipeline consists of calculating the wavelet decomposition of both the time series of the current sensors pair. The wavelet decompositions are calculated for the entire time series using the Daubechies wavelet of order 4 (Vonesch et al., 2007). Then, the technique iterates all the information in blocks of 60 samples for all pairs of sensors. At each iteration, the Pearson coefficient is calculated from the wavelet coefficient of both current sensors, for the current time instant. This process results in a set  $P = \{\rho_1, \rho_2, \dots, \rho_n\}$  of Pearson coefficients. The correlation between the two sensors considered in the current iteration is calculated as the average of the Pearson coefficients in  $P$ . The last step consists of determining which pairs of sensors have been correctly matched. For this purpose, we applied Otsu's thresholding technique (Otsu, 1979) to calculate a threshold value used to separate the set of correlation coefficients into two separate and disjointed classes. Once the two classes have been obtained, we identified the correctly matched pairs of sensors.

The computation time of this technique is lower as compared to the previous ones (it takes just about 1 second to process the entire dataset), however, the accuracy is considerably lower, returning only 22% of correctly matched pairs.

## 3 EXPERIMENTAL RESULTS

This section introduces the real dataset acquired by the sensors installed at SMART Facility building and the results obtained through the techniques described in the previous section. We also carry out a description of the evaluation methods used to compute the results obtained by the presented pipelines along with their computational times.

### 3.1 Experimental Context

The SMART Infrastructure Facility at the University of Wollongong (NSW, Australia) was created in 2011 and provides specialist laboratories for 150 staff and 200 postgraduate research students. In 2018, the

Table 1: Temperature and Humidity comparison for both Pycom and Droplet modules.

	Pycom	Droplet
Temperature variation	-40 to +85 $\pm 0.5^\circ$	0 to +65 $\pm 1^\circ$
Humidity variation	0 to 100%, $\pm 0.4\%$ RH	0 to 100%, $\pm 0.008\%$ RH

SMART building was equipped with a Droplet environmental sensors from Nube iO. A fleet of 140 sensors monitor the temperature, humidity, atmospheric pressure and movements inside every room inside the Facility. These sensors are battery-powered and rely on a long-range, low-power LoRaWAN network for transmitting data every 30 second. The LoRaWAN gateway then transmits the incoming data to a local database. The physical location of the sensors and the open database containing all the records and time series data for 2019 are publicly available and detailed in (Barthélemy et al., 2020b).

In addition to the existing Droplet sensors, mobile sensors based on a Pycom platform have been developed to capture the temperature and humidity at a different location inside and outside the building, every 2 minutes, thus enabling the rapid deployment and testing of different scenarios. Fig. 3 show the two types of sensors used for the experimentation.



Figure 3: Pycom (left) and Nube iO (right) modules.

The comparison between the Droplet and Pycom modules for both temperature and humidity is shown in Table 1, as these are the two data parameters used to perform correlation in this study.

The scenario investigated in this work is illustrated in Fig. 4 and the data used in the remainder of this Section is also open and available (Barthélemy et al., 2020a).

### 3.2 Evaluation Method

The techniques proposed in the previous section have been validated in four versions of the presented dataset, containing different sensors deployed in the SMART Facility building:

- *Test #1*: all sensors, inside and outside the build-

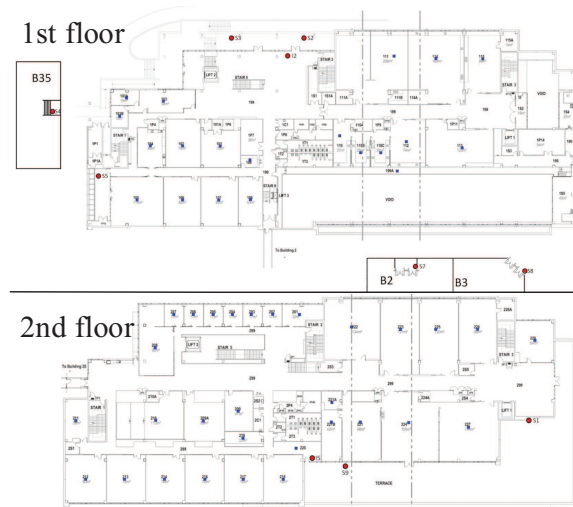


Figure 4: Location of the Droplet sensors (blue squares) and Pycom sensors (red dots). B35, B2 and B3 are the buildings next to the SMART Facility Building.

ing;

- *Test #2*: only the sensors inside the building;
- *Test #3*: external sensors and internal sensors on the 2nd floor;
- *Test #4*: external sensors and internal sensors on 1st floor.

The techniques have been evaluated on the same set of sensors but using different types of information: (i) temperature, (ii) humidity and (iii) temperature and humidity combined. This last combination was tested only on the evolutionary technique because it is the only one capable of integrating heterogeneous information.

We used two criteria to assess the quality of the solution obtained by the proposed techniques:

1. a pair of sensors are considered to be correctly matched if their information is correlated and both its sensors are internal or external to the building;
2. a pair of sensors are considered to be correctly matched if their information is correlated and the relevant areas of sensors are close one to each other.

The second criterion introduces a spatial proximity constraint: rather than evaluating if two sensors are both inside or outside the building, we calculate the Voronoi tessellation by using the position of sensors, which results in a set of adjacent regions that represent the rough relevance area of sensors. Two sensors are paired if their Voronoi regions overlap. In the case of the anomalous behavior of a sensor, this information can be useful for the domain expert to choose the sensor characterized by both spatial proximity and

Table 2: Percentage of sensor pairs correctly matched by the evolutionary technique.

	Temperature	Humidity	Temperature & Humidity
Test #1	56,67%	63,33%	70,00%
Test #3	46,67%	50,00%	36,67%
Test #4	46,67%	40,00%	56,67%

high informative correlation. For each technique, the following steps are carried out to determine whether two sensors are correlated according to this criterion:

1. compute the Voronoi regions separately for the sensors of each floor of the building;
2. determine the pairs of sensors using one of the described techniques;
3. stack the Voronoi tessellations calculated for each floor;
4. check if the two regions overlap: if so, sensors are correlated.

We applied this criterion to the techniques that use the evolutionary approach, the Pearson correlation and Gaussian smoothing. The SAX-based method already uses the Voronoi tessellation to determine whether sensors are correlated or not.

### 3.3 Experimental Results

Table 2 shows the percentage of correct pairs obtained by the proposed evolutionary method using the first criterion. The average of the percentages of correct pairs for each information is as follows: 62,50% for temperature, 63,33% for humidity and 65,83% for combined temperature and humidity. The results obtained by combining temperature and humidity are on average better (about 3%) than the tests conducted using only information of the same type. Moreover, we report a better accuracy in test #1 using heterogeneous information: the 70% of correlated pairs have been correctly matched by the evolutionary method.

The evolutionary approach carries out a single evolution by using the data acquired from the last 2 hours, as described in Section 2.1. This behavior is contradictory to the normal functioning of the genetic algorithms, where usually a variable number of evolutions are carried out on a single set of information.

By using the SAX-based technique, 85% of pairs of sensors are considered as correct. However, this pipeline requires about 15 seconds to compute the result. Concerning other methods, the SAX-based technique determines pairs of sensors that have a high correlation and spatial proximity thanks to the Voronoi tessellation. The technique based on Pearson correlation provided an average of 48% of correlated sensor pairs using temperature and humidity separately. The

Table 3: Percentage of sensor pairs correctly matched by the described techniques.

		Temperature	Humidity
Test #1	SAX	87,80%	78,05%
	Pearson	17,32%	6,95%
	Gaussian F.	28,42%	23,66%
Test #3	SAX	91,67%	83,33%
	Pearson	21,74%	9,42%
	Gaussian F.	42,75%	48,19%
Test #4	SAX	82,35%	82,35%
	Pearson	12,50%	2,94%
	Gaussian F.	24,27%	21,32%

Table 4: Computational times, in seconds, required by the described techniques. The evolutionary is omitted, as it requires about 50ms on average to compute a solution.

		Temperature	Humidity
Test #1	SAX	24,25	26,05
	Pearson	2,1803	2,1775
	Gaussian F.	146,3742	156,1693
Test #2	SAX	20,24	20,22
	Pearson	1,3	1,4
	Gaussian F.	107,0053	100,1627
Test #3	SAX	10,87	11,45
	Pearson	0,7	0,8
	Gaussian F.	50,7633	51,0154
Test #4	SAX	7,1	5,33
	Pearson	0,3	0,3
	Gaussian F.	25,3639	23,8448

technique based on the Gaussian smoothing provided an average of 37,89% of correctly correlated sensor pairs using temperature, 54,09% using humidity.

Table 3 shows the percentage of correct pairs obtained by these techniques. These results refer to the pairs obtained by applying the first criterion, based on the choice of sensors both indoor or outdoor. This means that the 85% of pairs of sensors returned by this pipeline contain either indoor sensors or outdoor sensors.

Table 4 reports the computational time required by the described techniques to determine the pairs of correlated sensors. The table does not list the computational time for the evolutionary method: this requires about 50ms to evaluate a solution for each test and each combination of data.

Among the techniques listed in Table 4, the one based on the Pearson correlation requires the least computational time. However, this depends on the number of sensors available in the environment: test #1 requires about 2 seconds whereas test #4, which considers a smaller set of sensors, requires 0,3 seconds. Therefore, these techniques are not suitable for real-time applications.

Table 5 shows the percentage of pairs obtained by applying the Voronoi proximity criterion on the solutions obtained by all the described techniques. For each pair of sensors, we verify whether the Voronoi

regions of the two sensors, which describe their relevance area, are in proximity; if so, the sensors are considered as correlated. The remaining pairs of sensors are omitted.

This criterion filtered a small number of pairs in the case of test #4 for Pearson and Gaussian smoothing based techniques. This is because the number of pairs obtained by the two techniques is small for this test, and many of these resulting correlated pairs are also spatially correlated. The test #4 using the Pearson based method on humidity data returns 100% of pairs using the Voronoi criterion: only four pairs of sensors were determined by the technique, all characterized by spatial proximity.

On average, the application of the proximity criterion has filtered a greater number of pairs obtained by the evolutionary technique, about 14%, against 24% of the technique based on Gaussian smoothing and 30% of the technique using the Pearson correlation. This is because the evolutionary solution determines a limited number of sensor pairs, while the other techniques evaluate the correlation for each possible pair of sensors, resulting in large sets of pairs. Despite this, having a limited set of sensor pairs that have not only a high informative correlation but also physical proximity enables the domain expert to easily choose which sensor to use in case of unexpected anomalous situations, where the information of a malfunctioning sensor can be compensated by the one of the correlated and also near sensor.

For illustrative purposes, Fig. 5 shows the two floors of SMART Facility building and the Voronoi regions resulting from the spatial proximity criterion on the test #1 using the evolutionary technique. We observe that the method determines pairs of sensors situated outside and inside the building and that are in immediate proximity.

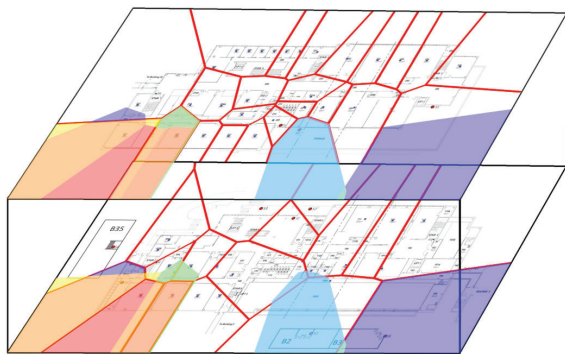


Figure 5: Voronoi tessellation resulting from the proximity criterion on test #1 using the evolutionary technique. Overlapping regions, depicted as the colored polygons, determine relevance area of correlated sensors.

Table 5: Pairs of sensors determined through the Voronoi criterion on the results obtained by the described techniques.

		Temperature	Humidity	Temperature & Humidity
Test #1	Pearson	21,83%	14,04%	X
	Gaussian F.	17,60%	19,59%	
	Evolutionary	16,67%	6,67%	13,33%
Test #2	Pearson	26,19%	12,77%	X
	Gaussian F.	19,01%	23,04%	
	Evolutionary	6,67%	20,00%	26,67%
Test #3	Pearson	23,33%	7,69%	X
	Gaussian F.	16,95%	19,70%	
	Evolutionary	6,67%	10,00%	3,33%
Test #4	Pearson	35,29%	100,00%	X
	Gaussian F.	42,42%	37,93%	
	Evolutionary	16,67%	26,67%	20,00%

The experiments were carried out on a machine equipped with i7-7820HQ, 32GB RAM and Windows 10. The computation takes less than 50ms to generate a new solution for a set of information perceived at a given time instant. The techniques were coded in MatLab language without particular optimizations; the evolutionary method was coded in Java language.

### 3.4 Discussion

The techniques described show promising results on the data acquired from the SMART Facility building. Among these, the evolutionary one allows addressing the need for on-line computation. Moreover, this evolutionary approach allows integrating heterogeneous, which is not possible by the other presented techniques. The criteria to determine the correlation between sensors allow the domain expert to extract useful knowledge from the perceived information: by using the criterion based on the correlation between internal or external sensors it is possible to determine the topography of the environment, discriminating between internal and external sensors. The criterion based on Voronoi tessellation allows determining pairs of sensors that have a high informative correlation and are also in the immediate vicinity. In both cases, the proposed techniques produced a significant amount of pairs of sensors.

The evolutionary approach determines pairs of sensors in real-time by using blocks of consecutive information in time. Nevertheless, the algorithm carries out only one evolution for each data window; this does not allow the achievement of an optimum. Using the evolutionary technique determines pairs of sensors whose correlation persists in time: at time instant  $t$ , the pairs of sensors determined by the technique are those that have survived the selection, therefore their correlation is high over time. The advantage of the evolutionary technique over the others also concerns

the computation time: this requires on average 50ms to compute a solution, while the SAX method requires 15 seconds.

## 4 CONCLUSIONS

The contribution of this article is twofold: firstly, we present a novel and open dataset of environmental information acquired from indoor and outdoor sensors deployed at SMART Infrastructure Facility at the University of Wollongong, Australia; secondly, we present a novel evolutionary approach to determine in real-time the correlation between pairs of different sensors.

This correlation is computed by using the information coming from the sensors and two criteria: the former based on the presence of both sensors inside or outside a building, the latter on the spatial distance among the sensors themselves. We experimentally verify that this allows determining accurately correlated pairs of sensors.

These techniques can be applied for different smart building applications such as environmental monitoring to optimize energy consumption or anomalies detection. In a real environment, sensors can be subject to unpredictable anomalies that cause missing information. A domain expert could be able to understand if a given sensor is malfunctioning or otherwise there is an emergency by using the proposed system. For example, a high temperature could indicate a malfunction, rather than the presence of fire. In this case, the system should assist the domain expert to determine whether a sensor is malfunctioning or there is an emergency.

We plan to extend our research by integrating more information (e.g. luminosity or noise) and by determining the state of devices (e.g. determining if a door is open or closed).

## REFERENCES

- Ashby, W. R. (1991). Requisite variety and its implications for the control of complex systems. In *Facets of Systems Science*, pages 405–417. Springer.
- Barthélemy, J., Arshad, B., Verstaevel, N., Guastella, D., and Perez, P. (2020a). Smart building additional data.
- Barthélemy, J., Arshad, B., Verstaevel, N., and Perez, P. (2020b). Smart infrastructure facility building data.
- Guastella, D. A., Camps, V., and Gleizes, M.-P. (2019). Estimating missing environmental information by contextual data cooperation. In Baldoni, M., Dastani, M., Liao, B., Sakurai, Y., and Zalila Wenkstern, R., editors, *PRIMA 2019: Principles and Practice of Multi-Agent Systems*, pages 523–531. Springer. doi: 10.1007/978-3-030-33792-6\_37.
- Guastella, D. A. and Valenti, C. (2018). Estimating missing information by cluster analysis and normalized convolution. In *2018 IEEE 4th International Forum on Research and Technology for Society and Industry (RTSI)*, pages 1–6. doi: 10.1109/RTSI.2018.8548454.
- Houssin, M., Combettes, S., Gleizes, M.-P., and Lartigue, B. (2020). SANDMAN: a Self-Adapted System for Anomaly Detection in Smart Buildings Data Streams. In *(to appear in) Proceedings of the 18th Adaptive Computing (and Agents) for Enhanced Collaboration (ACEC) at WETICE 2020*.
- Kenji Iwana, B. and Uchida, S. (2020). Time series classification using local distance-based features in multi-modal fusion networks. *Pattern Recognition*, 97:107024.
- Nigon, J., Verstaevel, N., Boes, J., Migeon, F., and Gleizes, M.-P. (2017). Smart is a matter of context. In *International and Interdisciplinary Conference on Modeling and Using Context*, pages 189–202. Springer.
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66.
- Pociask, E., Malinowski, K. P., Słęczak, M., Jaworek-Korjakowska, J., Wojakowski, W., and Roleder, T. (2018). Fully automated lumen segmentation method for intracoronary optical coherence tomography. *Journal of Healthcare Engineering*, 2018:1–13.
- Sciortino, G., Tegolo, D., and Valenti, C. (2017). Automatic detection and measurement of nuchal translucency. *Computers in Biology and Medicine*, 82:12–20.
- Shyr-Shen, Y., Shao-Wei, C., Chuin-Mu, W., Yung-Kuan, C., and Ting-Cheng, C. (2018). Two improved k-means algorithms. *Applied Soft Computing*, 68:747–755.
- Vonesch, C., Blu, T., and Unser, M. (2007). Generalized Daubechies wavelet families. *IEEE Transactions on Signal Processing*, 55(9):4415–4429.
- Wang, L., Lu, F., Cui, M., and Bao, Y. (2019). Survey of methods for time series symbolic aggregate approximation. In Cheng, X., Jing, W., Song, X., and Lu, Z., editors, *Data Science*, pages 645–657. Springer.
- Zhou, H., Deng, Z., Xia, Y., and Fu, M. (2016). A new sampling method in particle filter based on pearson correlation coefficient. *Neurocomputing*, 216:208–215.