Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ARTICLE TYPE

# A high performance IoT solution to reduce frost damages in stone fruits

Miguel A. Guillén | Raquel Martínez-España | Belén López | José M. Cecilia*

[1]Dept. of Computer Engineering, Universidad Católica de Murcia, Murcia, Spain

**Correspondence**
*Corresponding author José M. Cecilia
Email: jmcecilia@ucam.edu

**Present Address**
Campus de los Jerónimos, Guadalupe 30107, (Murcia) Spain

**Summary**

Agriculture is one of the key sectors where technology is opening new opportunities to break up the market. The Internet of Things (IoT) could reduce the production costs and increase the product quality by providing intelligence services via IoT analytics. However, the hard weather conditions and the lack of connectivity in this field limit the successful deployment of such services as they require both; fully connected infrastructures and highly computational resources. Edge computing has emerged as a solution to bring computing power in close proximity to the sensors, providing energy savings, highly responsive web services and the ability to mask transient cloud outages. In this paper, we propose an IoT monitoring system to activate anti-frost techniques to avoid crop loss, by defining two intelligent services to detect outliers caused by the sensor errors. The former is a nearest neighbor technique and the latter is the k-means algorithm, which provides better quality results but it increases the computational cost. Cloud vs. edge computing approaches are analyzed by targeting two different low-power GPUs. Our experimental results show that cloud-based approaches provides highest performance in general but edge computing is a compelling alternative to mask transient cloud outages and provide highly responsive data analytic services in technologically hostile environments.

**KEYWORDS:**
Frost crops, Precision Agriculture, IoT system, Edge Computing, GPUs

## 1 | INTRODUCTION

Mainly motivated by the high competition of the market, the high demands on the quality of products and the lack of scarce resources such as water [1], agriculture is constantly evolving towards the adoption of new technologies. Increasingly more farmers are applying new technologies to obtain products with higher quality and less cost. A new research area has emerged as a result of these concepts called Precision Agriculture (PA) [2]. PA is an innovative, integrated and internationally standardized approach to increase the efficiency of resource use and to reduce the uncertainty of decisions required to manage variability on farms [3]. One of the greatest dangers in agriculture is arising from inclement weather. Floods or constant temperature changes are only some examples that may jeopardize the harvest, having dramatic consequences [4]. Particularly damaging is the case of the Region of Murcia (Southeastern Spain), where crops suffer temperature variations of up to 20 degrees Celsius ($^{o}$C) on the same day. For example, temperatures at midday could reach values equal to or even higher than 20$^{o}$C in winter, and the temperature at night could even become negative. These temperature variations make the flowering of trees predictable, particularly in stone fruit trees. When the tree is fully bloomed, the negative temperatures may cause the flower to freeze, which could be translated in losing the harvest, and thus, having significant economic loses. Some techniques are available to protect crops from low temperatures such as protective covers, artificial fog, fans, application of chemical treatments or sprinkler irrigation[5],[6].

The anti-freeze techniques are often too expensive and must be applied several hours in advance to really prevent frost in the crop. The main problem for farmers is the lack of frost prediction, as these techniques take several hours to protect the crop. Therefore, this challenging scenario requires several ingredients: (1) real-time weather information for the crop in question, (2) analytical models that, from the information gathered, are capable of predicting a frost and (3) enough time to be able to take appropriate action. Indeed, Farmers can take advantage of technological innovations in the areas of computer science, telecommunications and agriculture, to answer these factors. One of the leading drivers of the digital revolution is the Internet of Things (IoT), where devices and humans are fully connected to the global network. The IoT is being widely applied in agriculture, as they are evolving towards continuous monitoring and automatized systems[7]. The IoT revolution is underpinned by two key factors (1) data, which carries hidden patterns, correlations and other valuable insights, and (2) real-time data analytics, since knowledge is often time-sensitive, and useful only within a specific time-frame[8]. In this paper, we propose an Internet of Things (IoT) infrastructure to prevent frost to avoid the loss of the harvest. This infrastructure is composed of several modules, in which the main contributions of this paper are developed. These contributions are listed below:

1. A *data acquisition* module is developed based on LoRa to provide good ratios of coverage, end-node's power consumption, and scalability[9].
2. An IoT *intelligent* module to make predictions about when the actuator should be switched on, is included in the infrastructure. This module is composed of several algorithms to identify outliers based on the nearest neighbor technique and the k-means clustering algorithm. The early detection of outliers is an important issue, since they can cause an incorrect operation of the actuators. For instance, sprinklers could be activated resulting in a loss of water and money or not activated, or activated too late, resulting in a loss of the crop.
3. The time and high availability requirements found in this problem demand a high-performance architecture for these algorithms to be valid for frost prediction. Therefore, we propose a *performance evaluation* on low-power GPU-based computing architectures at the edge compared to their cloud-based counterpart versions.
4. The IoT system proposed is able to activate the anti-frost technique of sprinkler irrigation due to its effectiveness and low installation cost compared to other techniques.

The rest of the paper is organized as follows: Section 2 contextualizes this research and identifies the main novelties of this work with respect to previous works. Section 3 describes the components of the proposed IoT intelligent monitoring system and its practical implementation. Section 4 provides evaluation results that demonstrate the accurate operation of our platform. Finally, Section 5 provides some conclusions and directions for future work.

## 2 | RELATED WORK

The capacities that IoT offers can be applied in agricultural and farming applications. In[10], some applications that use sensors and IoT technologies are applied to agriculture. Some of these applications are irrigation management system to optimize the water use in farming; control pest, disease and fertilizers to increase the crop quality; farming systems monitoring to improve management in large agricultural fields; cattle movement monitoring using radio frequency identifier; remote control and diagnosis of farming machinery; monitoring the ground water quality and the greenhouse gases, just to name a few. In[11], a monitoring system to control the quality of peppers in greenhouse is presented. The topology shown is composed of a wireless sensor network (WSN) to control crops in greenhouses. The main drawback of this work is that their solution is not extrapolated to other environments. The authors of[12] also present a study where they assess wireless sensor networks operation, reliability and accuracy in greenhouses. The authors perform an analysis of the collected data to research possible problematic situations for the growing plants caused by climatic heterogeneity inside the greenhouse.

In[13], a WSN is proposed to monitor water level in the farm areas for precision agriculture. In[14], a similar network is implemented, but in this case, it is used to control the waste of water. It noteworthy to highlight that in 136 days, they obtained a 90% reduction of water consumption compared with traditional irrigation systems. In[15], the development and use of a wireless radio frequency technology for the performance analysis of an irrigation scheme is presented. The authors study the use of water with different crops by using telemetry, concluding farmers behavior management is extraordinarily complex due to the multiple factor implicated in the process. In[16], the authors evalaute whether FIWARE platform is suitable for the development of agricultural applications. They propose an IoT scenario in the field of precision agriculture/viticulture for managing a large number of resources associated with sensors and actuators. The sensors detect data related to the environment and the condition of the vineyards (e.g. water and nutrients). The actuators are installed in crop irrigation systems.

Another work where the authors propose an improvement for irrigation in agriculture is presented in[17]. These authors implement an intelligent system for the control of bicarbonates in irrigation for precision hydroponic crops in order to improve water quality in hydroponic agriculture. They use different wireless sensors, including a pH sensor to measure nutrients. In[18], the authors present a wireless monitoring system for honey bee hives. The IoT system architecture enables the easy deployment and scalability in the field. They used commercial nodes adapted to the measurements of the hives. The nodes allow the use of different technologies according to the scope range, including the following: 3G, GPRS, LoRa, Sigfox, ZigBee or Wi-Fi. The LoRa technology has been used in agriculture[19] to propose an embedded system called MoniSen that is capable of monitoring environmental parameters within a 10 km radius. The proposed hardware architecture is designed for continuous monitoring of five environmental parameters: air temperature and humidity, soil temperature, soil moisture and light intensity. The authors of[20] present an intelligent IoT communication system manager as a low-cost irrigation controller. The proposal is a irrigation tool that periodically samples real-time data such as the variable irrigation rate, the vegetation index and irrigation events(flow rate, pressure level or wind speed.

However, to the best of our knowledge, we have not found any work that applies an IoT infrastructure to the field of frost prevention or cares about the quality of the data collected on the sensors and their validity. In addition to being in a new field of application, the IoT proposal made in this study applies techniques to correct and evaluate the quality of sensor data to detect outliers. Outlier detection is an important are within data mining and, particularly relevant in precision agriculture[21]. In WSN, outliers are defined as "measures that deviate significantly from the normal pattern of detected sensed data"[22]. This definition is based on the fact that sensor nodes are assigned in WSN to monitor the physical world and, therefore, there may be a pattern that represents the normal behavior of the detected data. It is important to consider that outlier detection is a task to be highlighted within the data processing. There are many outlier detections techniques in the literature as applied to WSN. In[23] a review of these techniques is carried out, showing a categorization of them into statistical-based approaches, *nearest neighbor-based* approaches, classification-based approaches, *clustering-based* approaches and spectral decomposition-based approaches. In addition the authors of[21] add to this classification of outliers detection techniques, the artificial intelligence techniques with fuzzy logic, such as artificial neural networks. Both classifications of the outlier techniques agree that KNN and clustering techniques are basic and effective techniques and they are also interpretable so they become appropriate techniques to be used by farmers. In the literature, there are different types of algorithms designed to detect outlier using the KNN algorithm and designing different distance measurements, we refer the reader to[24],[25],[26],[27],[28],[29],[30],[31] for insights. Some similar happens with clustering algorithms for outlier detection, some examples include[32],[33],[34],[35],[36].

# 3 | THE IOT INFRASTRUCTURE TO AVOID THE FROST DAMAGE.

This section introduces the main components of the proposed IoT solution to minimize frost damage. Figure 1 shows the conceptual architecture of the proposed IoT monitoring system to reduce the frost crops. The system consists of three main components: (1) climate sensors and anti-frost actuators, (2) an intelligent data processing system and (3) a monitoring component. The intelligent component is halfway between the cloud and the actuator, and in this paper, we analyze where it is most feasible, effective and reliable for that component to work. These three components are fully connected through a Wireless Sensor Network (WSN) to create an out-of-the-box IoT precision agriculture tool for minimizing damages caused by frost. In summary, the system monitors the climate data of an agricultural plot and, depending on the data, the system acts by activating the anti-frost system and alerting the farmer to make the appropriate decisions if this would be necessary. In what follows, we explain the main modules of the IoT infrastructure.

## 3.1 | Wireless Sensor Network (WSN)

One of the most important decisions relies on the selected wireless communication technology. Currently, there exists a myriad of IoT communication protocols that can be used for the interconnection of IoT nodes[7]. There are many technologies available

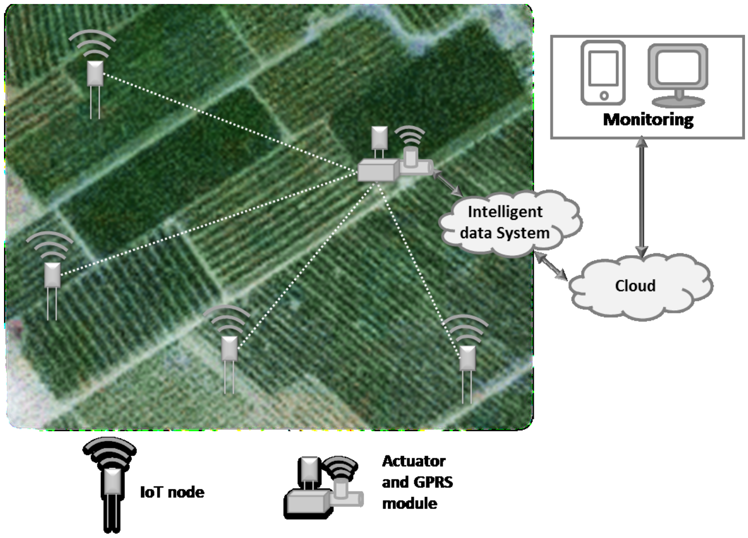



**FIGURE 1** The IoT System Architecture

in the market but we have focused on those that achieve long transmission distances, low power consumption, minimal data transfer and are feasible to implement in outdoor facilities. There are several data transmission standards and protocols, including SigFox†, LoRa‡ and ZigBee§, which meet these characteristics.

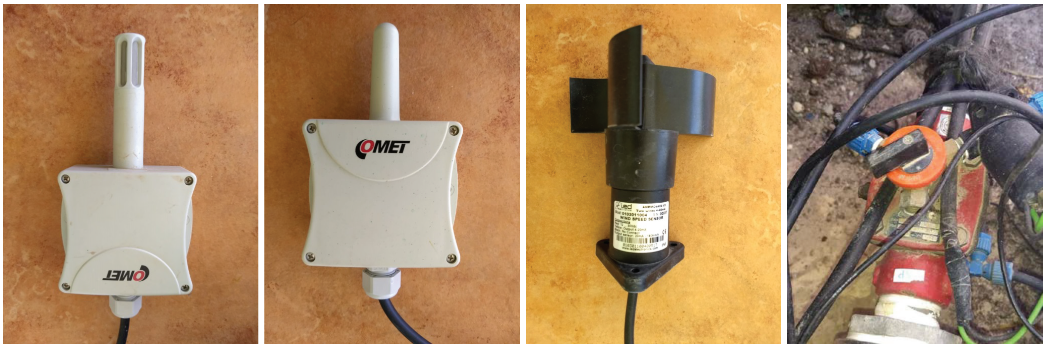
In a first scenario, we designed an infrastructure with nodes using ZigBee. This choice was made because ZigBee wireless protocol is considered one of the best candidate technologies for the agriculture and farming domains [17,19]. Although according to its specification it reaches 100 meters, in real use environments, ZigBee does not allow transmission distances greater than 60 meters outdoors [37]. This is a big issue for deploying IoT infrastructures in outdoor crops, and thus we decide to use the LoRa communication protocol. Among the advantages of technology, we may highlight the low power consumption and the great coverage it is able to reach. The choice to use LoRa instead of Sigfox is based on the possibility of deploying your own network, as Sigfox imposes certain restrictions on its use.

## 3.2 | IoT Sensor Hardware Description

**FIGURE 2** IoT node

†https://www.sigfox.com/
‡https://lora-alliance.org/
§http://www.zigbee.org/

The workflow starts at the sensor level where the climate data information is collected. The hardware architecture of the sensor network is the 4H remote control system of the company Hidroconta[¶]. Figure 2 shows the IoT node developed by this company. The IoT node incorporates temperature, humidity and wind speed sensors. These sensors have been calibrated by the company, so we have not had to perform any calibration phase. The sensors are connected to the analog inputs available on the IoT node, which can be expanded with as many sensors as required, while the actuator is connected to one of the four outputs for 12V latch solenoid valves. The sensors and the actuator used in the proposed IoT solution are the following:



(a) Humidity sensor (b) Temperature sensor (c) Wind speed sensor (d) Actuator(valve)

**FIGURE 3** Sensors and actuators of the IoT node.

- Humidity sensor: The humidity sensor also measures temperature, the implanted model is the P3110E from COMET [#] (see Figure 3a). With a temperature range of -30°C to 80°C with an error of $\pm$ 0.6°C and a humidity range of 0 to 100% with an error of $\pm$3%).
- Temperature sensor: The temperature sensor is the model 60P8610 from COMET (see Figure 3b). The temperature range is from -20°C to 60°C with a resolution of 0.1°C.
- Wind speed sensor: This sensor is the model PCE-WS A from the company PCE Instruments [‖] (see Figure 3c). It has a measurement range between 3 and 180 Km/h with an error of $\pm$ 1Km/h.

The most important variable to address the frost prediction is the air temperature, but other variables such as humidity or wind speed are also relevant in this domain. Including more variables offer the possibility to create more accurate models and thus, being able to predict the frost earlier. Moreover, sometimes it is also necessary to measure the same variable several times in different places. This is the case with temperature, where height can change the capture value. In our case, the humidity sensor measures both the humidity and air temperature. Having two temperature sensors helps us to detect outliers and/or temperature errors and, on the other hand, it increase the possibility of detecting the lowest temperature to activate the anti-frost technique as soon as possible.

One of the IoT communication nodes also has an actuator connected to it that activates and deactivates the anti-frost sprinkler technique. Specifically, a lach type solenoid (see Figure 3d). Among its characteristics, we may highlight the existence of 3 direct action ways, resistance of up to 80°C and 12 bar of pressure. Beyond sensors and actuators, the IoT node also has hardware dedicated to connectivity where the sensors are connected. An important aspect is that one of the IoT nodes acts as a gateway to send data to the cloud via GPRS connection. While the others connect to it using LoRa technology. All of them are autonomous since they have a battery 6 volts (V) and 12 amp (A) per hour and a solar panel of 12V and 5 watts (W). More than sufficient capacities as the consumption in "low" mode is 126$\mu$A and 19mA with GPRS connection.

Finally, each IoT node has a micro controller with 256 KB of firmware storage and 96 KB of volatile memory for data. The latter can be expanded with a non-volatile external memory of 244 KB, and thus store up to 20.000 records. Therefore, each

[¶] http://www.hidroconta.com/
[#] https://www.cometsystem.com/
[‖] https://www.pce-instruments.com/

node, in addition to sending the information to the gateway, stores the information in its memory so that if there is a failure in the communication it can be forwarded once it is restored.

## 3.3 | IoT Intelligent Component

There are many applications where *real-time data mining* of sensor data to make intelligent decisions is essential[38]. One of this application is the approach proposed in this study for monitoring low temperatures in crops. Data measured and collected by WSNs is often unreliable and inaccurate[23]. The quality of data set might be affected by noise and error such as missing values, duplicated data, or inconsistent data. The quality of this data is due to the fact that the low cost and low quality sensor nodes have stringent resource constraints such as energy (battery power), memory, computational capacity, and communication bandwidth. Besides, operations of sensor nodes are frequently susceptible to environmental effects. Thus, it is inevitable that in such environments some sensor nodes do not work properly, which may result in noisy, faulty, missing and redundant data, being this type of data called outliers.

The prediction of frost temperatures needs reliable and error-free data, therefore, it is necessary to apply a processing technique to these data to avoid creating confusion and errors in the IoT system proposed and implicitly to the farmer. This work propose two different algorithms for outlier detection within this context; they are: (1) k-nearest neighbors and (2) k-means clustering information. With these techniques, we not only detect outliers but also we eliminate/transform these outlier in useful information. These two techniques are within the categorization presented in the section 2. Considering the profile of the problem and the rejection that exists in some farmers to the implementation of the new technology and after analyzing the different outliers detection techniques, these two techniques have been selected based on their interpretability and their easy explanation of its operation without going into technical details.

**Neighborhood-based outlier detection**

The neighborhood-based outlier detection is a lightweight technique which combines the same variables from different IoT nodes. As mentioned above, each IoT node is equipped with two temperature sensors, (the humidity sensor also measures the temperature). This provides temperatures at different heights and allows a real-time outlier checking. Moreover, each IoT node stores its closest neighbors to compare its measurements with those taken by its neighbors. Each time a new IoT node is introduced into the network, it notifies all IoT nodes within a radius to be included as a neighbor. This avoids performing the search every time a value has to be checked. The process of verification of the distance can be done offline without penalizing the proposed system. It is noteworthy to highlight that the distance here also takes into account the altitude at which the sensor is located as this is a relevant parameter in temperature. Therefore, the distance between IoT nodes is always displayed taking into account the latitude, longitude and altitude in the different part of plots and experience of the farmer.

**Algorithm 1** Outlier detection technique based on k-nearest neighbors.

**procedure** KNN
**forall** *T from the whole of IoT sensors*
    $outlier1 = Diff(T, T_{1-KNN})$
    $outlier2 = Diff(T, T_{2-KNN})$
    **if** $|outlier1| \leq 1.0$ **then**
        $Out = T$
        **else if** $(|outlier1| \geq 1.0)\&(|outlier2| \leq 2.5)$
        $Out = Average(T_{1-KNN}, T_{2-KNN})$
        **else** $Out = T_{1-KNN}$
    **end if**
**end for**
**end procedure**

Algorithm 1 shows the general scheme of the outlier detection technique of the nearest neighbor. This technique uses the two nearest neighbors, being $1-KNN$ the air temperature sensor of the IoT node and $2-KNN$ the nearest neighbor in distance that is not deployed in the IoT node. This decision to use only 2 nearest neighbors is supported by previous experiments, where it was tested that when using information from distant neighbors, the outliers value was distorted, introducing even more noise in the data. Thus, for each temperature, the difference between the temperature of the node $T$ (node under evaluation) with the temperature of the node $T_{1-KNN}$ (outlier1) and with the temperature of the node $T_{2-KNN}$ (outlier2) is calculated. When the temperature $T$ is less than 1.0$^o$C from the nearest neighbor, there is no outlier. If the temperature $T$ is greater than 1.0$^o$C from its nearest neighbor $(1 - KNN)$ and less than 2.5$^o$C from its second nearest neighbor $(2 - KNN)$, there is an outlier and the temperature shown by the system is the average of the two neighbors. If these two conditions are not given, then it is an outlier but its second nearest neighbor $(2 - KNN)$ can contain another outlier, therefore the temperature of the nearest neighbor is assigned as the output of the system. The thresholds of 1.0 and 2.5$^o$C have been established on the basis of experimental evidence and farmers' opinions and assessments.

**K-Means clustering**

The k-means[39] is a well-known clustering technique which is heavyweight computationally speaking. It is an iterative procedure widely used in pattern recognition and data mining to search for statistical structures in data. It is an unsupervised data mining algorithm to make cluster from high-dimensional data. Given a data matrix composed of observations and variables, the objective is to cluster the observations into groups that are internally homogeneous and heterogeneous from group to group. The $k$ of the k-means clustering method indicates the number of groups which is actually an input parameter in the algorithm. The k-means algorithm establishes prototypes or $centroids$, which are points that represents each cluster. To decide which point belongs to each cluster, the k-means uses the Euclidean distance as a measure of the similarity between observations and clusters. The traditional algorithm uses an iterative refinement.

Given an initial set of k clusters $(c_1, c_2, ..., c_k)$ where k is an input parameter the algorithm alternates two main steps:

1. **Assignment step** where each point is assigned to the "closest" cluster; i.e. the cluster with the the least squared Euclidean distance between the cluster prototype and the point .

2. **Update step** calculates the new means to be the prototypes of the new clusters.

The k-means algorithm has been used in the literature for improving the outlier removal[40,41]. Actually, the term outlier is categorized into two new concepts (Internal and External outlier) when clustering techniques are applied to identify them. *External outlier* is when the outlier is an element of the group which has few data and is located far from other groups. *Internal outlier* is a point that belongs to a particular group but is located far from the centroid. In this work, we focus on the internal outliers, since in this context, the external outliers do not make sense because there is a large amount of data increasingly generated each hour to be analyzed. Once the corresponding clusters have been obtained by means of K-means, the outlier value $x$ is detected as follows:

- The cluster $c$ corresponding to the value $x$ is assigned.
- The maximum distance between each centroid and the furthest value $x^f$ belonging to that cluster is calculated, taking into account the signs. An outlier value can be detected by low or high temperature. This distance is called $Max$.
- If the distance of $x$ to the centroid of $c$ is greater than $Max_c$ and in addition the distance between $x$ and $x_c^f$ is greater than $1°C$, then the value is considered outlier.
- The outlier is corrected taking as value the centroid value of the cluster.

This K-means technique is highly parallel as the Euclidean distance calculation between points and clusters can be fully performed in parallel. Some works have proposed parallel implementations of the k-means algorithm in different platforms, including mutlicore CPUs, GPUs and FPGAs[42,43,44]. Actually, the RAPIDS library recently announced by Nvidia[45] includes k-means as an accelerated method for clustering observations. In this work, we use three different implementations of k-means for outlier removal, they are sequential implementation in ANSI C to run on single-thread CPUs, multicore implementation using OpenMP[46] and GPU implementation using CUDA[47].

When a sensor repeatedly obtains outliers during an hour of monitoring means that the IoT node is inactive and their values are not used to make decisions. In the test phase, we have implemented the KNN outliers detection technique in some sensors and the K-means technique in others to analyse the efficiency, speed and quality of the results. Once the data is pre-processed to ensure data reliability, then the IoT actuators are activated if the temperature is less than or equal to 3.0°C. This decision rule is made by the experience and knowledge of experts in the field (agronomists). The rule is based on the need to moisten the entire crop area before the temperature drops from zero in order to make the anti-frost technique more effective. When the actuator is activated, the irrigation valve is opened, which allows the application of the anti-frost sprinkler irrigation technique. The IoT system deactivates the actuators when the following two scenarios occur:

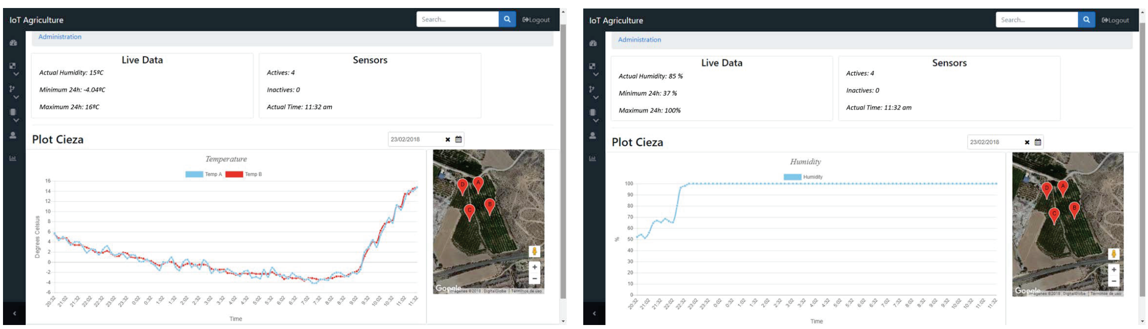
- When negative temperatures have occurred and after these negative temperatures occur two hours with positive temperatures above 7°C.
- When temperatures do not become negative, but are below 2.5°C, and spend two hours with temperatures above or equal to 4°C. The actuators are activated again if the 2.5°C rule is met.

These temperature thresholds have been established on the basis of the experiments carried out by the farmers and it should be noted that in the tests carried out with the IoT system proposed in this manuscript the results have been positive using these thresholds.

## 3.4 | Monitoring Interface

The monitoring interface provides interpretability and simplicity to reach the end user of the system. It must be appreciated that many farmers may not be familiar with the technology, hence the development of the monitoring interface must be friendly. For the development of this monitoring interface there was the possibility to choose between open source IoT platforms as Kaa[**], Fiware[††], ThingSpeak[‡‡], between others. However, due to the particularity of the problem presented, the application of outlier detection, the configuration of specific alerts that farmers demand and the activation of the irrigation valves, an ad-hoc monitoring interface has been developed. This allows us to add the specific preferences for each farmer and to configure the alarms or conditions that are necessary to avoid damage to crops due to frost.

Figure 4a and Figure 4b show a part of the administration panel of the IoT monitoring system. It shows the data related to a plot containing four IoT nodes active. The location data of the IoT nodes has been altered to avoid the exact location of the nodes for privacy reasons.



(a) Temperature view (b) Humidity view

**FIGURE 4** Monitoring interface for the intelligent system

In the calendar icon, the user can pick a day and the interface will show the sensor selected in the map (Figure 4a air temperature sensors and Figure 4b humidity sensor). Each user has his/her own personalized administration panel. These figures show the farmer has available the number of active nodes, the minimum and maximum value of the sensor and the current value. This monitoring interface has been developed using NodeJS and AngularJS technologies. As a database manager, we work with MySQL 5.7 which allows us to store data of JSON type. This allows us to simplify the process of data manipulation. Also, we have implemented a MongoDB database to increase system scalability.

[**] https://www.kaaproject.org/
[††] https://www.fiware.org/
[‡‡] https://thingspeak.com/

### 3.5 | Deployment of the IoT monitoring system

The IoT system has been deployed in two plots of the Region of Murcia (Spain). The IoT nodes are actively working to date. This deployment was carried out according to farmers' requirements due to the temperature difference between the canopies and the trunk of the trees. The IoT nodes are implanted in a 2-meter high stainless steel support. The wind speed and temperature sensors are located in the upper part (see Figure 5a). The humidity sensor is integrated in the part of the IoT node (see Figure 5b) that is protected from rainfall with its corresponding insulators to prevent it from getting wet and sending erroneous data. In addition, the display has one of the IoT nodes connected to the valve (the IoT actuator) which opens and closes to activate or deactivate the anti-frost technique (see Figure 5c). Each plot has a actuator deployed, and three IoT nodes each with with two temperature sensors, one for humidity and wind speed. The two temperature sensors of the same IoT node are at different heights in the support, and all the IoT nodes have the same configuration, although it is possible to define different heights for each sensor.

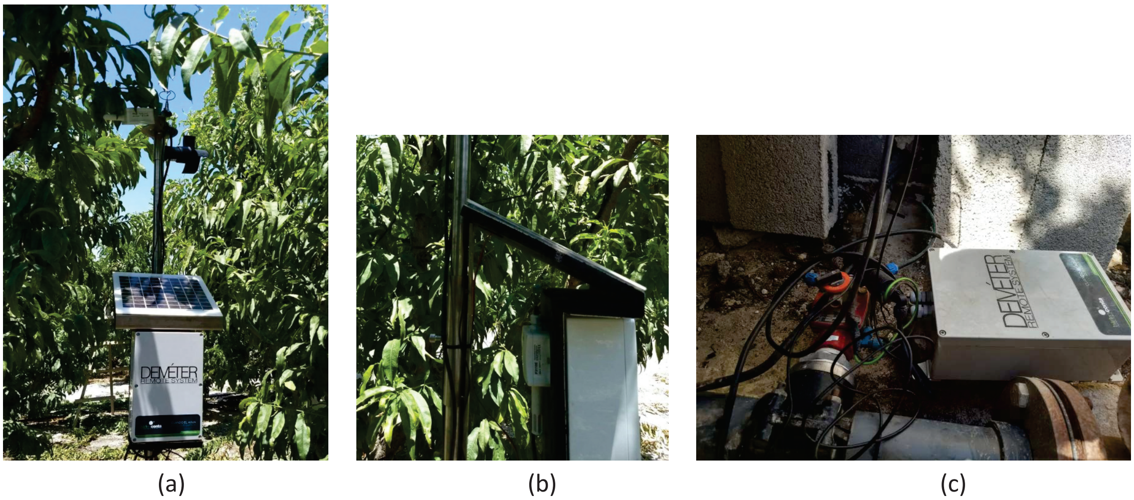


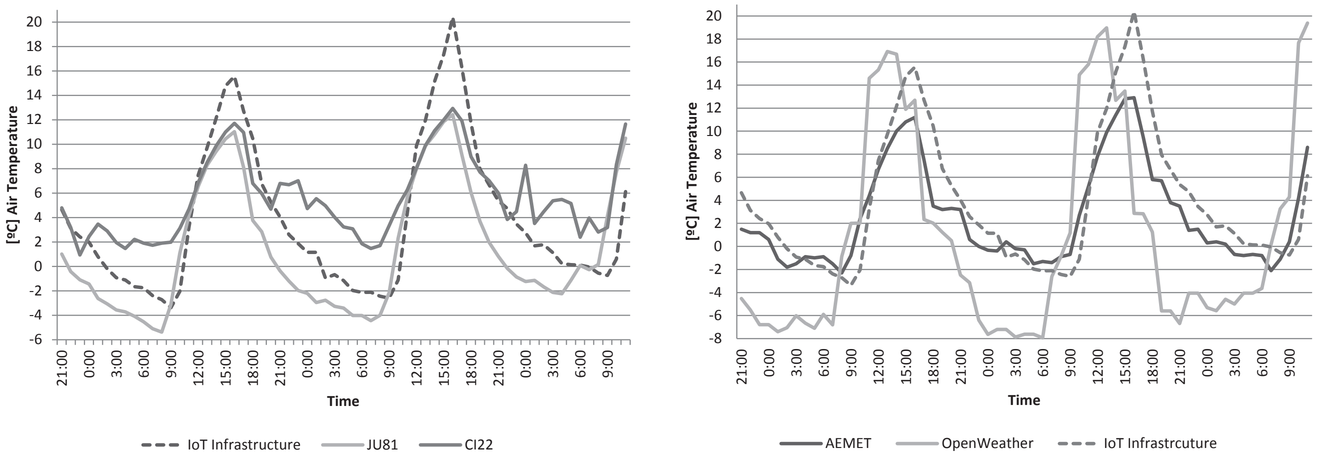

**FIGURE 5** (a) Deployment of an IoT node with the three sensors in a real environment. (b) Detail of the displayed humidity sensor. (c) Deployment of the actuating IoT node in a real environment.

The proposed IoT system was put into operation at the beginning of October of 2017 and the data used in these assessment experiments are related to the month of November, December 2017 and the January, February, March, November and December 2018 and January, February 2019. We have to take into account for the frost domain that we are interesting to analyzed and tested low temperatures. The IoT nodes pick up temperatures 24 hours a day with a frequency between 5 and 10 minutes. However, to assess the behavior, we rely on the temperatures collected between 5 pm a day and 12 am hours a day following. Thus analyzing the temperatures in this range, we avoid problems with high temperatures at midday since these temperatures can be influenced by the position of the IoT nodes, i.e. whether these nodes are in the sun or not. It worth mentioning that this IoT infrastructure is designed to be scalable. The user can include as IoT nodes as desired, and it can include more sensors per IoT node using expansion slots. This offers the possibility to apply this infrastructure in different agricutlural context.

## 4 | RESULTS AND DISCUSSION

This section shows the results of the proposed IoT solution to address the problem of reducing frost damage to crops. It is structured as follows. First of all, we motivate the need of deploying an IoT system to deal with the frost problem instead of relying on coarse-grained forecast systems, where the predictions expand to a wide geographical area. The IoT infrastructure is then evaluated in several ways: (1) the most appropriate frequency and position (height) for collecting temperatures are analyzed to determine the best sensor configuration and (2) the intelligent component of IoT is evaluated from the quality's (outlier's removal) and performance's (edge vs. cloud approach) point of views.

## 4.1 | Global vs Local Temperature




(a) Global Temperature Prediction vs. Local Real Temperature

(b) Global Real Temperature vs. Local Real Temperature

**FIGURE 6** Comparative of Global and Local Temperatures

General weather forecasting has several limitations. Among them, we may highlight the following: (1) the measurements are generally provided hourly and (2) it is a coarse-grained prediction based on weather station at a given location.The prevention of frost requires real-time information for the particular area where the crops are located. Figure 6a shows a comparison between the temperature measurements made by the proposed IoT infrastructure and the predictions made from the Spanish Meteorological Agency (AEMET) and the OpenWeather website [§§] between 21.00 hours on 10/01/2019 and 11.00 hours on 13/01/2019. Although the trend is quite similar, there are significant differences between them. Openweather's prediction is less accurate than AEMET's, AEMET's prediction still differs greatly from in situ IoT infrastructure. Particularly detrimental to us are the differences in low temperatures, as such a deviation could mean the loss of the crop. Moreover, our IoT temperature information is also compared to weather stations that provide real-time information. These stations belong to the Murcian institute for agricultural and food research and development (IMIDA)[¶¶]. In particular, the two IMIDA stations closest to the crop are selected, i.e. CI22 and JU81. These stations are located at a maximum distance of 10 km from the IoT infrastructure. Figure 6b shows that there are still big differences with the actual temperature collected by our IoT infrastructure.

## 4.2 | Difference of temperature according to height and frequency capture

Figure 7 shows temperature variation when it is measured at different frequencies, i.e. 10 and 5 minutes. This is commonly used frequencies for collecting temperatures in different IoT scenarios such as IMIDA. As the frequency of data collection increases, more information becomes available and, therefore, additional patterns could be detected. Whenever the temperature begins to drop, the 5-minute temperature appears to indicate the trend to the next temperature, although this does not have to happen in all cases. For example, the temperature at 3:04 am was 0.79$^{o}$C, 5 minutes later the temperature was 0.94$^{o}$C and at 3:14 am was 1.08$^{o}$C, so the intermediate reading at 5 minutes indicates that the temperature was rising. Therefore the frequency has been set every 5 minutes to provide greater information, an accuracy in displaying temperatures and warning farmers of possible frost early and to activate the anti-frost technique.

Figure 8 shows the air temperatures for sensors A and B where sensor A is located higher than sensor. As explained above, the proposed IoT infrastructure is composed of up to two IoT nodes at different heights (see figure 5). Temperature differences of these sensors never exceed one degree Celsius. For instance, from 2:42 am to 3:52 am, the sensor A had a higher temperature than the sensor B. This is not a single event, but it occurs several times. Moreover, it is worth highlighting that the biggest

[§§] www.openweather.com
[¶¶] http://www.imida.es/

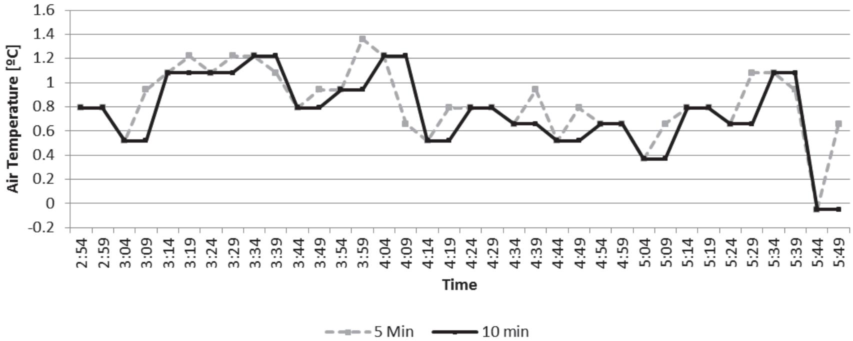



**FIGURE 7** Comparative of IoT nodes collect temperature frequencies every 5 to 10 minutes

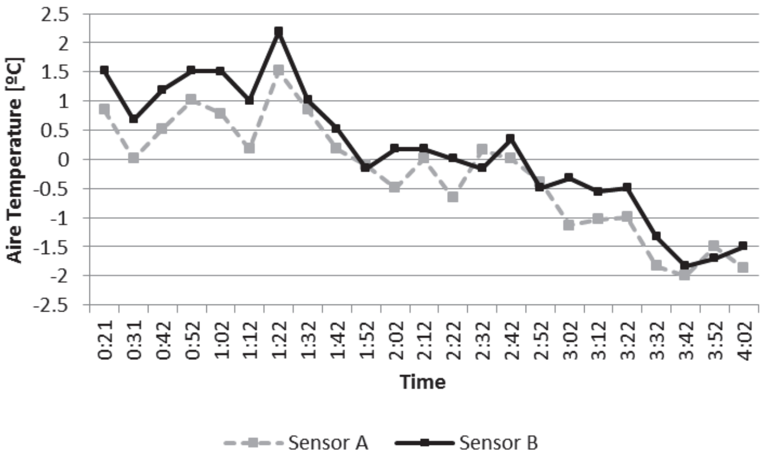



**FIGURE 8** Temperature comparison of sensors at different heights.

temperature differences occurs at 1:12 am and 3:02 am. In addition, it should be noted that the greatest temperature differences occur at 1:12 am and 3:02 am. Having information at several heights enriches the information available for decision-making. A degree of temperature difference can mean the loss of the crop, and thus it is important to activate the antifreeze techniques in the worst-case scenario.

## 4.3 | Assessing the IoT intelligent component

As described in Section 3.3, the IoT intelligent component is composed of two outlier detection techniques based on the KNN and k-means algorithms. The detection of outliers in this system is a fundamental task due to the possible errors that sensors may have. Actually, outliers are a common problem in agriculture due to the harsh weather conditions in which IoT infrastructures are deployed. This section evaluates the intelligent component of the IoT infrastructure proposed. To perform this evaluation in parallel, two sensors have implemented the outlier detection technique based on K-means and the rest of sensors have used the KNN technique because the latter technique requires some neighbors to check its feasibility.

Figure 9(a) and Figure 9(b) show ten temperatures relative to January 6, 2018 between 22:20 and 23:55 hours. The dots in bold represent outliers. Figure 9(a) shows the raw temperatures that includes the outlier values. Figure 9(b) shows the outlier values corrected by the KNN technique (see Algorithm 1). In the Figure 9(a), the temperatures marked as outliers are 5.36°C and 2.36°C. The KNN replaces these values by 4.02°C and 4.65°C respectively. The former is replaced by 1-KNN and the latter is calculated by the average of the temperatures between 1-KNN and 2-KNN. If these outliers are not corrected, the actuators (valves) would be activated incorrectly, resulting in a loss of resources.

Figure 9(c) and Figure 9(d) show the 5 clusters made for one sensor by the k-means algorithm during January 7, 2018 between 10:00 p.m. and 8:00 a.m. of the following day. Clusters are built using data from 1 month prior to that date and are updated if

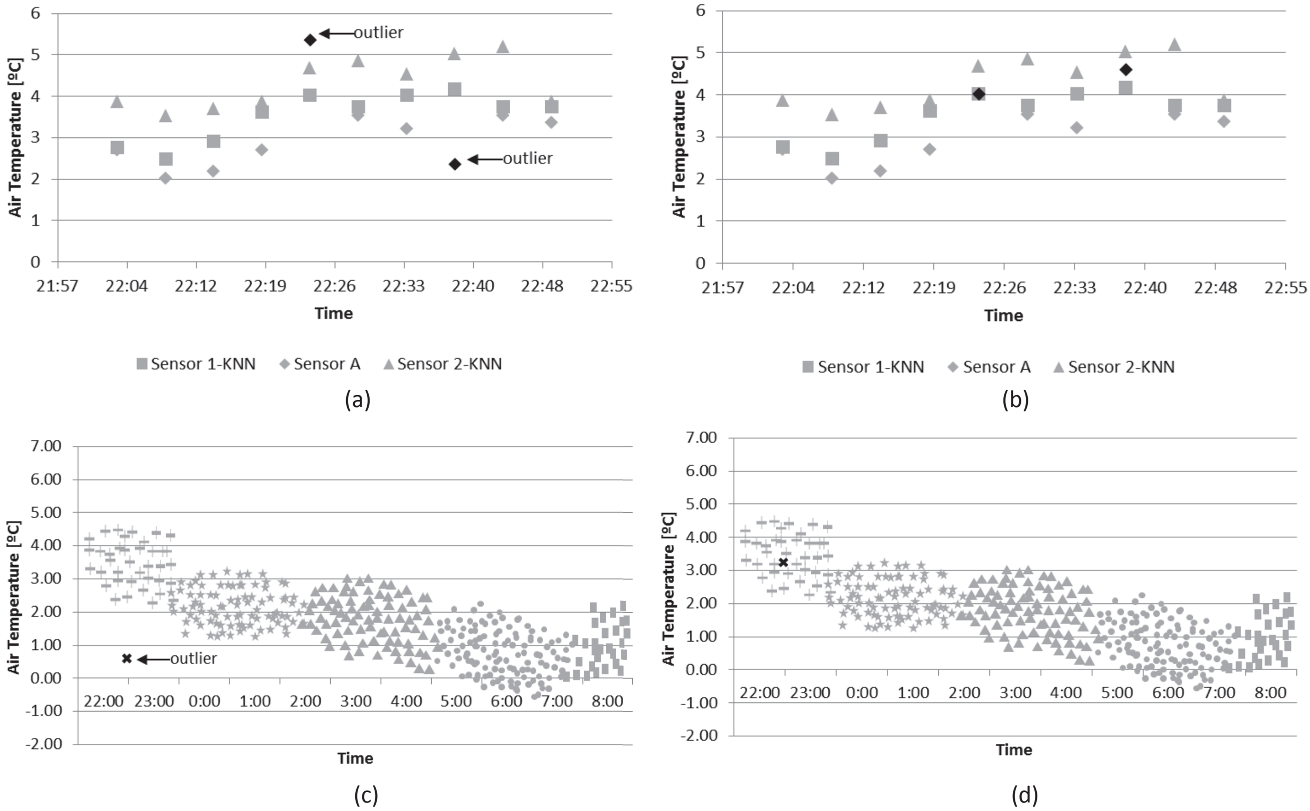



**FIGURE 9** (a) Air temperature detecting outlier with KNN technique. (b) Air temperature with the outlier corrected with KNN technique. (c) Air temperature detecting outlier with K-means technique. (d) Air temperature with the outlier corrected with K-means technique.

necessary each time a new value is collected. The outlier detected between 22.00 and 23.00 hours corresponds to a temperature of 0.6°$C$. The outlier is detected since the temperature obtained by the sensor is higher than 1.0°$C$ with respect to the maximum distance between the temperature furthest from the centroid. The value of the corrected outliers is the centroid value of the cluster to which this value belongs to. The K-means algorithm requires much more information than KNN for the detection of outliers, but it provides a higher accuracy.

## 4.4 | Cloud Vs. Edge computing

The k-means algorithm is a more accurate tool for outlier detection but it is computationally expensive. In the agriculture application domain, the lack of connectivity, the low-bandwidth connection is big trouble when IoT analytics services are deployed. Therefore, this section analyzes the k-means algorithm from the computational point of view. Computational experiments have been carried out considering the IoT infrastructure under study where cloud and edge computing approaches are evaluated. For the cloud-based approach, the IoT infrastructure is connected to a computational back-end called *Heterolistic*, which includes 2 hexa-core Intel Xeon E5-2650 at 2.20 GHz, 128 GB of RAM, private L1 and L2 caches of 32 KB and 256 KB per node, and a L3 cache of 32 MB shared by all the cores of a socket. Moreover, it includes 2 GPUs: Nvidia GTX 1080 Ti (Pascal), with 12 GB and 3584 cores (28 SM and 128 SP per SM), and a Nvidia TITAN X (Maxwell) with 3072 (24 SM and 128 SP per SM) cores and 12 GB of Global Memory. For the edge computing evaluation, the nodes are System on Chip (SoC) based on Nvidia Jetson family. Particularly, Jetson TK1 and TX1 are under evaluation. The NVIDIA Tegra K1 SoC uses a NVIDIA Kepler GPU core (192 CUDA cores) with NVIDIA 4-Quad-Core ARM Cortex-A15 CPU, 2 GB x16 Memory with 64-bit Width and 16 GB 4.51 eMMC Memory. The NVIDIA Tegra X1 SoC uses a NVIDIA Maxwell GPU core (256 CUDA cores) and 4-Quad-Core ARM A57/2 MB L2, 4 GB 64 bit LPDDR4 (25.6 GB/s) and 16 GB eMMC, SDIO, SATA.

A set of hydrometeorological data with up to 239,108 instances is used for the evaluation, each with a maximum of 13 variables. The data set contains hourly information from January 2012 to April 2018 for the following variables: mean humidity, maximum humidity, minimum humidity e, mean radiation, maximum radiation, mean wind speed, maximum wind speed, mean wind direction, precipitation, mean pressure vapour deficit, dew point, difference with the mean temperature of one hour from the previous hour and minimum temperature.

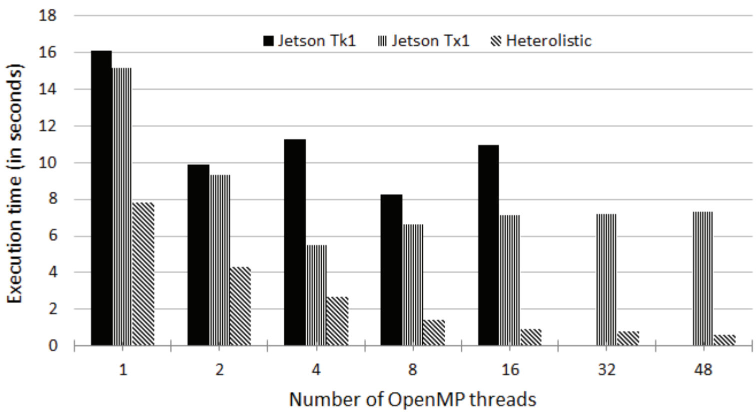



**FIGURE 10** Execution time in seconds of the k-means algorithm running 1000 iterations, 4 clusters and a hydrometeorological dataset with 13 variables and 239,108 instances

Figure 10 shows the execution time in seconds of the k-means algorithm for the target dataset. Three different multicore architectures (Jetson TK1, Jetson TX1 and Heterolistic) are analyzed by varying the number of threads. The highest performance is achieved with 8 threads in Jetson TK1, 4 threads in Jetson TX1 and 32 threads in Heterolistic. The speed-up factor between Jetson architectures is up to 1,5x and between Jetson TX1 and Heterolistic is almost 7x (Heterolistic classifies this dataset in 0,794 seconds and Jetson TX1 in 5,467 seconds). Therefore, the cloud-based approach offers much better performance than Jetson nodes and they are designed to be power-efficiency. From now on, we focus our performance evaluation on Jetson TX1 as it provides better performance than the previous generation.

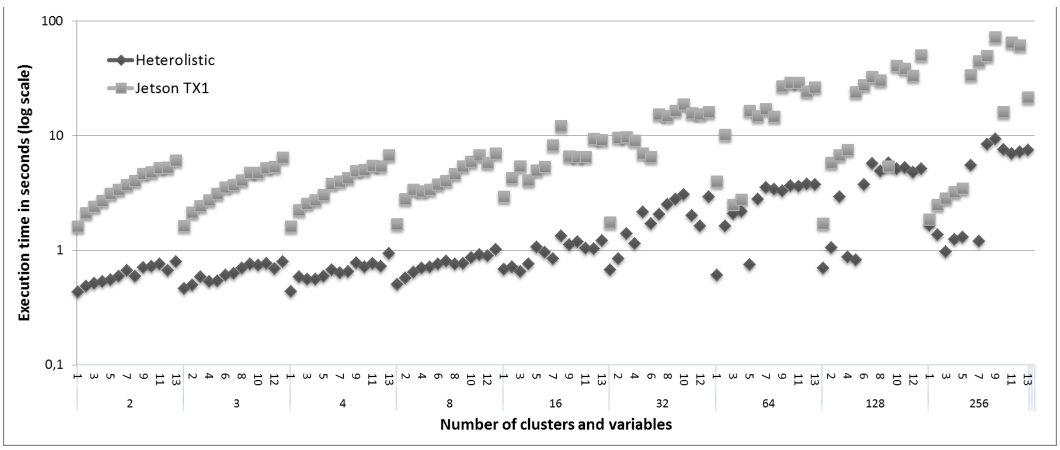



**FIGURE 11** Execution time in seconds of k-means algorithm running 1000 iterations for the hydrometeorological dataset with 239,108 instances. The number of variables (from 1 to 13) and clusters (2, 3, 4, 8, 16, 32, 64, 128, 256) are modified to analyze the behaviour of both Heterolistic and Jetson Tx1 under different workloads.

Figure 11 shows the execution time in seconds (logarithmic scale) of k-means algorithm running 1000 iterations for the hydrometeorological dataset with 239,108 instances. The number of variables goes from 1 to 13 and the number of clusters is in the range of (2, 3, 4, 8, 16, 32, 64, 128, 256). Notice that, the performance of this code is mainly affected by the number of clusters to be performed, the number variables and registers. The cloud-server (Hetorolistic) defeats by a wide margin the low-power consumption architecture Jetson Tx1, generally speaking. The speed-up factor first quartile is 2,3x, the median is 4,3x and the third quartile is 6,2x for all benchmarking configurations. However, the lack of connectivity and the low-bandwidth connections in agriculture scenarios may justify the possibility to compute on the edge for some cases. Particularly, there is up to 25% of the cases where the performance gap is only 2,3x (Q1) which can be hidden by the network overhead. The lowest differences in performance between both platforms are found when those variables are relatively small. Network latency tests have been carried out using the PingTools program[48], obtaining rates higher than 2 second. Although the device used for them is an Android phone configured to send through the GPRS network, the data obtained is considered better than what can be achieved by measuring directly on the device of the communications node. Therefore, it isn't contradicted the hypothesis that the computation in the device is more optimal for the scenario that is described in this article.

## 5 | CONCLUSIONS AND FUTURE WORK

Climate change is causing trees bloom earlier and the sudden changes in temperature can lead to severe economic damage to farmers, since freezing flowers and spoiling crops. In this article, we have proposed and deployed a high performance IoT system to reduce frost damages in stone fruit. We design the whole IoT infrastructure in a technological hostile area, which includes sensors/actuators, a wireless sensor network, an intelligent component to detect outliers in real time and a monitor interface. Regarding the intelligent component, two different algorithms to remove outliers are designed: the KNN and k-menans. The latter provides better quality results but it is a heavy workload that is well-suited for parallelization. Therefore, a comparison between cloud-edge approaches to perform the k-means algorithm is provided. Different low-power GPU-based computing architectures based on Nvidia Jetson architectures are analyzed at the edge on this IoT infrastructure. Indeed, the cloud-based architecture offers higher performance in general but edge computing is not too far away from it computationally speaking, and therefore, we really believe that it is a compelling alternative to mask transient cloud outages and provide highly responsive data analytic services in such technologically hostile environments. After the evaluation of the goodness of the IoT system, this is deployed and operating in a real environment giving positive and satisfactory results and covering the needs of farmers facing the problem of frost.

We acknowledge that the classification algorithms pointed out in this work are very basic and other classification algorithms, such as those within the umbrella of deep learning, could obtain better results. However, the computational requirements of these promising algorithms may be too high for this emerging landscape of computation and the power consumption may be a limiting factor. Increasing the computing requirements would require a broader evaluation of edge computing platforms, in terms of performance and power consumption. We definitely believe this is an interesting tradeoff to evaluate in future work.

## ACKNOWLEDGMENT

This work is supported by the Spanish Ministry of Science, Innovation and Universities under grants TIN2016-78799-P (AEI/FEDER, UE) and RTC-2017-6389-5. Finally, we thank the farmers for the availability of their resources to be able to asses and improve the IoT monitoring system proposed.

## References

1. María-Inmaculada MM, others . Water distribution management in South-East Spain: A guaranteed system in a context of scarce resources. *Science of the total environment* 2018.

2. Norton GW, Swinton SM. Precision agriculture: global prospects and environmental implications. In: Routledge. ; 2018: 269.

3. Liaghat S, Balasundram SK, others . A review: The role of remote sensing in precision agriculture. *American journal of agricultural and biological sciences* 2010; 5(1): 50–55.

4. Nelson GC, Mensbrugghe D, Ahammad H, et al. Agriculture and climate change in global scenarios: why don't the models agree. *Agricultural Economics* 2014; 45(1): 85–101.

5. Crookston RK. A top 10 list of developments and issues impacting crop management and ecology during the past 50 years. *Crop science* 2006; 46(5): 2253–2262.

6. Guillén-Navarro MA, García nFP, na RME. IoT-based System to Forecast Crop Frost. In: ; 2017: 28-35.

7. Tzounis A, Katsoulas N, Bartzanas T, Kittas C. Internet of Things in agriculture, recent advances and future challenges. *Biosystems Engineering* 2017; 164: 31–48.

8. Gubbi J, Buyya R, Marusic S, Palaniswami M. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems* 2013; 29(7): 1645–1660.

9. Sanchez-Iborra R, Sanchez-Gomez J, Ballesta-Viñas J, Cano MD, Skarmeta AF. Performance Evaluation of LoRa Considering Scenario Conditions. *Sensors* 2018; 18(3): 772.

10. Ojha T, Misra S, Raghuwanshi NS. Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges. *Computers and Electronics in Agriculture* 2015; 118: 66–84.

11. Srbinovska M, Gavrovski C, Dimcev V, Krkoleva A, Borozan V. Environmental parameters monitoring in precision agriculture using wireless sensor networks. *Journal of Cleaner Production* 2015; 88: 297–307.

12. Ferentinos KP, Katsoulas N, Tzounis A, Bartzanas T, Kittas C. Wireless sensor networks for greenhouse climate and plant condition assessment. *Biosystems Engineering* 2017; 153: 70–81.

13. Lichtenberg E, Majsztrik J, Saavoss M. Grower demand for sensor-controlled irrigation. *Water Resources Research* 2015; 51(1): 341–358.

14. Gutiérrez J, Villa-Medina JF, Nieto-Garibay A, Porta-Gándara MÁ. Automated irrigation system using a wireless sensor network and GPRS module. *IEEE transactions on instrumentation and measurement* 2014; 63(1): 166–176.

15. Lorite IJ, Santos C, García-Vila M, Carmona M, Fereres E. Assessing irrigation scheme water use and farmers' performance using wireless telemetry systems. *Computers and electronics in agriculture* 2013; 98: 193–204.

16. Martínez R, Pastor JÁ, Álvarez B, Iborra A. A testbed to evaluate the fiware-based IoT platform in the domain of precision agriculture. *Sensors* 2016; 16(11): 1979.

17. Cambra C, Sendra S, Lloret J, Lacuesta R. Smart System for Bicarbonate Control in Irrigation for Hydroponic Precision Farming. *Sensors (Basel, Switzerland)* 2018; 18(5).

18. Gil-Lebrero S, Quiles-Latorre FJ, Ortiz-López M, Sánchez-Ruiz V, Gámiz-López V, Luna-Rodríguez JJ. Honey bee colonies remote monitoring system. *Sensors* 2016; 17(1): 55.

19. Ilie-Ablachim D, Pătru GC, Florea IM, Rosner D. Monitoring device for culture substrate growth parameters for precision agriculture: Acronym: MoniSen. In: IEEE. ; 2016: 1–7.

20. Cambra C, Sendra S, Lloret J, Garcia L. An IoT service-oriented system for agriculture monitoring. In: IEEE. ; 2017: 1–6.

21. Ayadi A, Ghorbel O, Obeid AM, Abid M. Outlier detection approaches for wireless sensor networks: A survey. *Computer Networks* 2017; 129: 319–333.

22. Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. *ACM computing surveys (CSUR)* 2009; 41(3): 15.

23. Zhang Y, Meratnia N, Havinga P. Outlier detection techniques for wireless sensor networks: A survey. *IEEE Communications Surveys & Tutorials* 2010; 12(2): 159–170.

24. Lijun C, Xiyin L, Tiejun Z, Zhongping Z, Aiyong L. A data stream outlier delection algorithm based on reverse k nearest neighbors. In: . 2. IEEE. ; 2010: 236–239.

25. Kontaki M, Gounaris A, Papadopoulos AN, Tsichlas K, Manolopoulos Y. Continuous monitoring of distance-based outliers over data streams. In: IEEE. ; 2011: 135–146.

26. Yang D, Rundensteiner EA, Ward MO. Neighbor-based pattern detection for windows over streaming data. In: ACM. ; 2009: 529–540.

27. Buthong N, Luangsodsai A, Sinapiromsaran K. Outlier detection score based on ordered distance difference. In: IEEE. ; 2013: 157–162.

28. Abid A, Kachouri A, Mahfoudhi A. Anomaly detection through outlier and neighborhood data in Wireless Sensor Networks. In: IEEE. ; 2016: 26–30.

29. Wang X, Wang XL, Ma Y, Wilkes DM. A fast MST-inspired kNN-based outlier detection method. *Information Systems* 2015; 48: 89–112.

30. Schubert E, Zimek A, Kriegel HP. Fast and scalable outlier detection with approximate nearest neighbor ensembles. In: Springer. ; 2015: 19–36.

31. Yosipof A, Senderowitz H. k-Nearest neighbors optimization-based outlier removal. *Journal of computational chemistry* 2015; 36(8): 493–506.

32. Yu D, Sheikholeslami G, Zhang A. Findout: finding outliers in very large datasets. *Knowledge and Information Systems* 2002; 4(4): 387–412.

33. Rahman MA, Ang KLM, Seng KP. Unique Neighborhood Set Parameter Independent Density-Based Clustering With Outlier Detection. *IEEE Access* 2018; 6: 44707–44717.

34. Zhang Z, Zhu M, Qiu J, Liu C, Zhang D, Qi J. Outlier Detection Based on Cluster Outlier Factor and Mutual Density. In: Springer. ; 2018: 319–329.

35. Campello RJ, Moulavi D, Zimek A, Sander J. Hierarchical density estimates for data clustering, visualization, and outlier detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 2015; 10(1): 5.

36. Timón I, Soto J, Pérez-Sánchez H, Cecilia JM. Parallel implementation of fuzzy minimals clustering algorithm. *Expert Systems with Applications* 2016; 48: 35–41.

37. Kuzminykh I, Snihurov A, Carlsson A. Testing of communication range in ZigBee technology. In: IEEE. ; 2017: 133–136.

38. Ma X, Yang D, Tang S, Luo Q, Zhang D, Li S. Online mining in sensor networks. In: Springer. ; 2004: 544–550.

39. Linde Y, Buzo A, Gray R. An algorithm for vector quantizer design. *IEEE Transactions on communications* 1980; 28(1): 84–95.

40. Yoon KA, Kwon OS, Bae DH. An approach to outlier detection of software measurement data using the k-means clustering method. In: IEEE. ; 2007: 443–445.

41. Chawla S, Gionis A. k-means–: A unified approach to clustering and outlier detection. In: SIAM. ; 2013: 189–197.

42. Li Y, Zhao K, Chu X, Liu J. Speeding up k-means algorithm by gpus. *Journal of Computer and System Sciences* 2013; 79(2): 216–229.

43. Hussain HM, Benkrid K, Erdogan AT, Seker H. Highly parameterized k-means clustering on fpgas: Comparative results with gpps and gpus. In: IEEE. ; 2011: 475–480.

44. Baydoun M, Ghaziri H, Al-Husseini M. CPU and GPU parallelized kernel K-means. *The Journal of Supercomputing* 2018: 1–24.

45. RAPIDS:Open GPU Data Science. urlhttps://rapids.ai/; .

46. OpenMP A. Openmp specification. *URL http://www. http://openmp. org/wp* 2017.

47. Nvidia C. Cuda c programming guide, version 9.1. *NVIDIA Corp* 2018.

48. PingTools: A useful set of network utilities. urlhttps://www.pingtools.org/; .

49. Guillén-Navarro M. MCGBARME. A Preliminary Study to Solve Crop Frost Prediction Using an Intelligent Data Analysis Process. *Ambient Intelligence and Smart Environments, Intelligent Environments 2018* 2018; 23: 97–106.

50. Challinor AJ, Watson J, Lobell D, Howden S, Smith D, Chhetri N. A meta-analysis of crop yield under climate change and adaptation. *Nature Climate Change* 2014; 4(4): 287.

51. Domínguez DR, Redondo RPD, Vilas AF, Khalifa MB. Sensing the city with Instagram: Clustering geolocated data for outlier detection. *Expert systems with applications* 2017; 78: 319–333.

□