Agric Res (September 2024) 13(3):574–585
https://doi.org/10.1007/s40003-024-00705-x

FULL-LENGTH RESEARCH ARTICLE

Enabling Smart Agriculture: An IoT-Based Framework for Real-Time Monitoring and Analysis of Agricultural Data

Faruk Enes Oguz[1,2] · **Mahmut Nedim Ekersular**[2] · **Kubilay Muhammed Sunnetci**[2,3] · **Ahmet Alkan**[2]

Received: 7 November 2023 / Accepted: 22 January 2024 / Published online: 27 February 2024

Abstract With the progress in sensor and cloud technologies in contemporary times, a range of intelligent agriculture applications has gained considerable prominence. It is predicted that these developments can continue to pique the interest of researchers in the future. On the other hand, it is seen that IoT (Internet of Things)-based models are used in various fields. Herein, the primary objectives of this study are to enable farmers to remotely monitor and manage field conditions through sensor technology and IoT integration. In addition, these technological advancements make it possible to take the required measurements. Farmers can optimize their agricultural practices based on the analysis of the data obtained for this application. Thus, the aim is to manage the agricultural process more effectively and efficiently. In this study, an IoT-based framework is proposed for agricultural data monitoring. Light, temperature–pressure, smoke, humidity, and soil dryness values can be measured from GY-30, BME280, MQ-2, DHT11, and YL-69, respectively. An ESP-32S development board is used to collect data from sensors, and this board is coded using Arduino IDE. Subsequently, using ESP-32S, it is sent to the ThingSpeak cloud service provided by MATLAB via a Wi-Fi connection. Thus, these data can be easily transferred to MATLAB. We create a user-friendly Graphical User Interface application so that the data can be monitored and analyzed in MATLAB as well as ThingSpeak. This application allows users to monitor the data flow in real time and can easily provide the requested values such as maximum, minimum, mean, standard deviation, and current with the help of a button. In addition, the proposed system sends an e-mail to the user when soil dryness and smoke values exceed a certain threshold value. The results obtained in the study indicate that the proposed model can save time and labor in addition to providing reliable and fast data flow.

Keywords ESP-32S · IoT · MATLAB · Smart agriculture · Remote monitoring · ThingSpeak

Introduction

As time goes by and technology continues to advance, artificial intelligence and IoT-based models have become increasingly usable in virtually every sphere of life [2, 8, 13, 17, 23]. While this is the case, it can be predicted that these technologies, which can be utilized in various fields from healthcare to education, can be widely used in the future. Unfortunately, it is seen that IoT-based technologies have not been used effectively in agriculture, which is one of these areas, and the existing techniques that are widely used do not include IoT-based technologies that can be monitored remotely [5]. Monitoring values such as

✉ Faruk Enes Oguz
farukenes.oguz@mku.edu.tr

Mahmut Nedim Ekersular
mahmutnedimekersular@ksu.edu.tr

Kubilay Muhammed Sunnetci
kubilaysunnetci@osmaniye.edu.tr

Ahmet Alkan
aalkan@ksu.edu.tr

[1] Department of Biomedical Device Technology, Hassa Vocational School, Hatay Mustafa Kemal University, Hatay, Turkey

[2] Department of Electrical and Electronics Engineering, Kahramanmaras Sütçü Imam University, Kahramanmaras, Turkey

[3] Department of Electrical and Electronics Engineering, Osmaniye Korkut Ata University, Osmaniye, Turkey

temperature, relative humidity, soil dryness, pressure, light, and smoke is crucial for agricultural practices.

In addition, monitoring these values continuously and instantaneously by users with user-friendly software can help them to save a significant amount of time or labor. It is also significant to consider the fact that these data can contribute to the optimum sustainability of agricultural production [12].

The use of IoT in agriculture allows farmers to make more informed decisions by measuring soil quality, plant health, and environmental factors through sensors. Thus, providing farmers with remote access enables the enhancement of sustainability and efficiency in agriculture within the scope of this study. In this way, it is anticipated that the agricultural sector will become more resilient and adaptable to climate changes.

The current climate change, alterations in temperature and precipitation patterns, and their impact on plant growth cycles pose challenges for crop growers. Furthermore, the gradual decrease in water resources necessitates a more conscientious approach to water management. The rising temperatures and changing precipitation patterns in the region affect the soil moisture balance in agriculture. This situation diminishes the effectiveness of traditional irrigation methods and puts pressure on the sustainability of water resources. It is significant to collect and analyze agricultural data through sensors to develop efficient irrigation strategies that take climate change into account.

In this study, measurements of light, temperature and pressure, smoke, humidity, and soil moisture are taken using GY-30, BME280, MQ-2, DHT11, and YL-69 sensors. These data are collected using the ESP-32S development board, which is also programmed using the Arduino IDE. Data are transmitted to the ThingSpeak cloud via Wi-Fi and then transferred to MATLAB. Subsequently, the obtained real-time data are integrated into a user-friendly GUI application through graphical representations. Additionally, the software is capable of providing users with the maximum, minimum, mean, standard deviation, and current values of these data upon request. Furthermore, a system that sends an email to the user when the threshold value for the designated sensor output is exceeded has been proposed in this study.

In light of all this information, the main contributions of this study are presented as follows:

Real-time transfer and visualization of agricultural data to the cloud system

Design of a user-friendly GUI application that can display six channels of agricultural data

Informing the user by sending an e-mail in risky situations

The proposed method's ability to reliably save time and labor.

Related Work

In this section, we present the studies in the literature on this subject as follows. One of these studies aims to control the temperature and humidity values in the agricultural field, using an Android-based application for flexibility and functionality [10]. In the study, monitoring systems using an embedded web server with an IP connection are examined. The study aims to bring together electronic devices, software, and system prototype and provide web organization. Using an IoT-based system in agriculture to meet user demands, this study is optimized in terms of power consumption, simple installation, and maintenance. STM32F103 is used in the study and cloud technologies are used for storage analysis. As a result of the study, it is stated that this system can be used successfully [7]. In this study conducted in Sao Paulo, Brazil, a real-time and IoT-based system is employed to monitor the condition of soil and the environment. The system utilizes temperature, humidity, soil electrical conductivity, soil reflectance, a global positioning system, and ZigBee radio for communication [14]. This study, which proposes a GSM-based system to overcome the problems in standard environmental monitoring models, includes wireless environment monitoring equipment and mobile phones. If it is determined in the study that the environmental parameters exceed the appropriate value, the proposed model sends a warning message to the mobile phone [26]. In another review, considering the rapid growth of the world population and the consequent substantial increase in food demand, it is stated that traditional agricultural methods inefficiently utilize resources. It is pointed out that IoT-based automation is gaining popularity as a response to the growing demand for agricultural practices. In the study, a comprehensive literature review of next-generation agricultural technologies is provided, comparing sensors, communication protocols, and IoT-enabled devices. The aim of the study is to assist in the selection of IoT technologies for specific agricultural applications [18]. It has been indicated that an IoT-supported approach, combining mobile, cloud, sensor, and data-sharing features, for detecting insects, monitoring plant health, and maintaining the quality of harvested crops in greenhouse farming operations, is expected to enhance agricultural practices in farming sector by 90–95% [25]. In another IoT study designed for use in greenhouses, it is aimed to measure temperature using wireless sensors and an SMS system. In addition, the system also has the capability to warn farmers against temperature changes in the greenhouse and to take precautions against it [3]. ZigBee and GPRS networks are used together to collect agricultural information in the system designed for remote monitoring. Data collection has been carried out with the CC2430 radio frequency chip. The base station used in the study is designed based on an ARM9 processor and embedded Linux operating system and communicates with the remote monitoring center via GPRS. Based on the conducted tests, it has been concluded that the designed system is a successful model for remote monitoring of agricultural information [27]. It has been stated that the development of this conceptual system, which uses wired sensors to remotely monitor humidity and temperature values within a greenhouse, and notifies about deviations in these values through SMS alerts while employing the hypothetical-deductive method, would be beneficial for farmers [4]. In this IoT-based system developed for achieving a more sustainable agricultural activity, various sensors have been integrated to acquire data such as temperature, humidity, and pH in order to facilitate the hydroponic cultivation of plants. These data, along with external input variables such as district name, crop name, area, and type of hydroponic system, are combined and transmitted to a microcontroller for processing. The data are also sent to the cloud, providing notifications to farmers. In the proposed web application of the study, sensor and user inputs are fed into a random forest algorithm to provide an estimation of crop yield [15]. In another study, the integration of IoT technology with various networks has been examined, proposing a remote monitoring system where internet and wireless communication collaborate, working together [28].

Materials and Methods

Materials

The study was conducted in Gaziantep, Turkey, specifically in a hobby garden focusing on the Thorny Crown plant. The data obtained were collected under the conditions of this plant in the hobby garden. However, the results of the study are not limited to this specific hobby garden; they can also provide general information that may be applicable in different greenhouse or agricultural land conditions. The sampling interval from the sensor nodes has been set to 15 s. The data are recorded in 6 channels for a total of 4483 data for approximately 19 h. We can also present data channels as private or public in ThingSpeak. While publicly presented channel data can be observed by everyone, private channels can only be observed by users who are allowed with mail authorization. In addition, the location where the data are collected can be shown by marking, except for the data channels. The sensor used in the study and the data received are also presented in Table 1. In light of this information, the data obtained from which sensors and how is given below.

Temperature and Pressure

Changes in temperature and pressure affect agriculture [11]. Therefore, following these parameters is significant for productivity increase in agriculture. BME280 sensor is used for temperature and pressure measurement (https://electronics.semaf.at/BME280-I2C-or-SPI-Temperature-Humidity-Pressure-Sensor). This sensor is used for weather measurements capable of measuring temperature, pressure, and altitude. The sensor is presented to the customers as calibrated by the manufacturer. It is used only for temperature and pressure measurement in this study. It can measure the temperature between − 40 and 85 °C with an accuracy of ± 1.0 °C and the pressure between 300 and 1100 hPa with an accuracy of ± 1 hPa [6]. The BME280 sensor is used with a 3.3 V supply voltage, and it can transmit data using both SPI and I2C protocols. Having two different I2C data addresses (0 × 76, 0 × 77) allows two sensors to be used at the same time, and the low current consumption feature is significant. In this study, it is preferred to use the I2C communication protocol.

Humidity

Humidity is measured using the DHT11 sensor (https://www.pngkit.com/bigpic/u2t4r5a9q8o0t4o0/). The DHT11 sensor can measure both humidity and temperature [9, 20]. It is used only for relative humidity measurement in this study. DHT11 is a negative temperature coefficient thermistor-based sensor, it can measure humidity between 20 and 80% with 5% sensitivity [20]. It has a sampling rate of 1 Hz, can operate with 3.3 V or 5 V, and uses a communication protocol similar to the one-wire protocol [9, 20].

Smoke

Smoke measurement is important for agriculture in several ways. Basically, this measurement is evaluated in order to detect a possible fire that may occur in the agricultural field. But apart from studies, measurement of air quality and exposure to smoke is of great importance in protecting both worker and crop health [22]. Within the scope of the study, the MQ-2 sensor is used for smoke measurement (https://www.indiamart.com/proddetail/mq-2-gas-sensor-module-19274547712.html). The MQ-2 sensor can detect flammable gas and smoke at concentrations between 300 and 10,000 ppm. This sensor, which operates with 5 V

Table 1 Sensors used in the study and their intended use

Sensor	Measurement	Purpose of usage
GY-30	Light intensity (lux)	Measuring the amount of sunlight
BME280	Temperature (C°) and pressure (hPa)	Measuring temperature and pressure
MQ-2	Smoke (ppm)	Measuring flammable gas and smoke
DHT11	Relative humidity (%)	Measuring humidity in the air
YL-69	Soil moisture (–)	Measuring soil dryness

supply voltage, can produce digital TTL level or analog output [16]. In addition, sensitivity can be adjusted with the potentiometer on it.

Light

The amount of light and its variation are important for agricultural productivity, as they affect the plant's growth factor [1]. In the study, the GY-30 sensor is used for light measurement (https://www.roboshop.com.tr/gy-30-isik-yogunlugu-sensoru-aydinlik-modulu-gy30-bh1750?limit=100). The GY-30 sensor can measure light intensity in lux and from 0 to 65535lxx. In addition, it is a sensor that directly gives digital output and uses the I2C protocol [21].

Soil Moisture

YL-69 sensor is employed for soil moisture measurement (https://www.hobidevre.com/toprak-nem-sensoru). The YL-69 sensor is used for measuring soil moisture, and its probes are inserted into the soil to conduct measurements [19]. Due to the difference in resistance caused by the soil or the liquid being immersed, there occurs a voltage difference between the probe terminals. Depending on the magnitude of this voltage difference, the moisture level can be measured. The higher the moisture level in the soil, the higher the conductivity. The sensor consists of two parts: the pad probe that detects the water content and the electronic board with an analog–digital converter on it. In addition, it can give analog or digital output [19, 24].

Methods

In this section, we aim to explain the methodologies used in the study. As mentioned before, in this article, we aim to obtain and analyze data by using IoT technologies for agricultural monitoring so that it can be controlled remotely. Thus, we aim to create a platform where we can provide services to people who are interested in this subject by making use of user-friendly software applications and a notification system. For this purpose, we use five different sensors to monitor six measurements in this procedure, which is aimed to be placed in an agricultural area first. The sensors used in the study are GY-30, BME280, MQ-2, DHT11, and YL-69.

Within the specified time interval, these sensors can measure light, temperature–pressure, smoke, humidity, and soil dryness values, respectively. At this juncture, we provide the relevant connections to the ESP-32S development board (https://joy-it.net/en/products/SBC-NodeMCU-ESP32) by using all of these sensors together in this study. We use I2C, AOUT, and one-wire for the sensors used in the study, and we present it in the following figure. Afterward, we obtain data from these sensors in the designated agricultural area. While these data are being obtained, data are transferred to the IoT service called ThingSpeak using Wi-Fi technology. In this platform, after the live data stream is aggregated in the cloud and visualized in ThingSpeak (https://www.mathworks.com/products/thingspeak.html), we transfer it to MATLAB.

Following, we offer a user-friendly GUI application designed for the user to better understand and analyze these data in MATLAB. This application can easily present the statistical information requested for the data obtained in the relevant time period with the help of simple buttons to the users. In this application, we prefer the maximum, minimum, mean, standard deviation, and current values depending on the sensors. Moreover, we propose a system that continuously checks the available data in the sensors for soil dryness and smoke values and sends an e-mail to the user according to the values determined by the user. In this study, we set these values to 90. In light of all this information, we present the block diagram of the proposed approach in the study in Fig. 1. In addition, we render the experimental setup of our study and the serial port output in Fig. 2. Herein, it is seen that five different sensors are used and with the help of the ESP-32S development board, we can follow the updated data via the serial port while continuing the data flow from the sensors using Wi-Fi. In this study, we also code the ESP-32S development board using

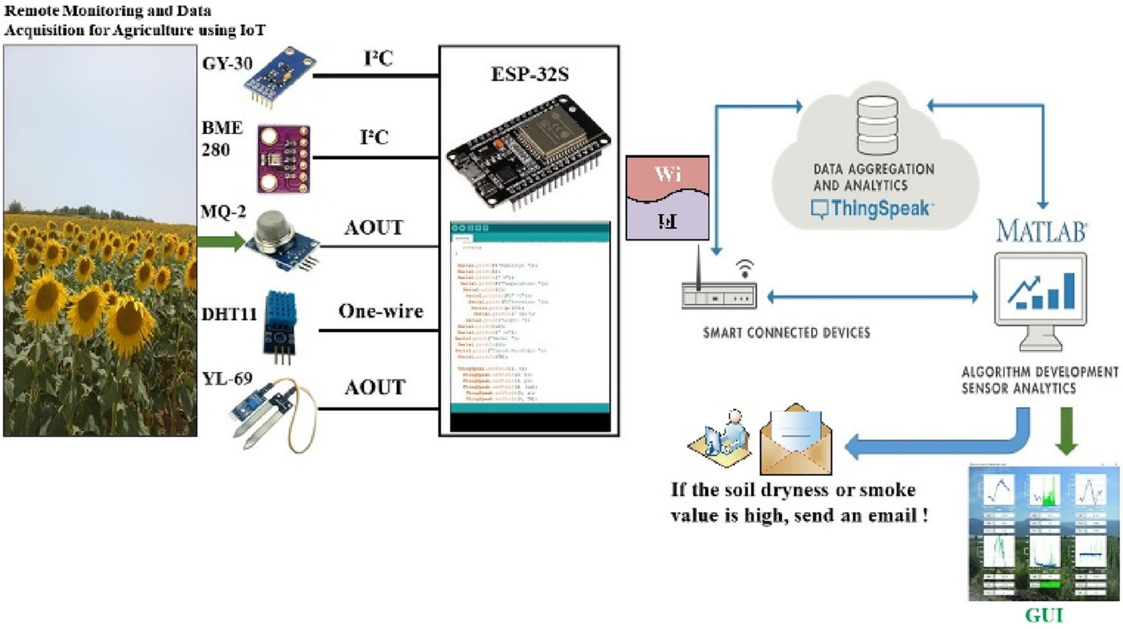
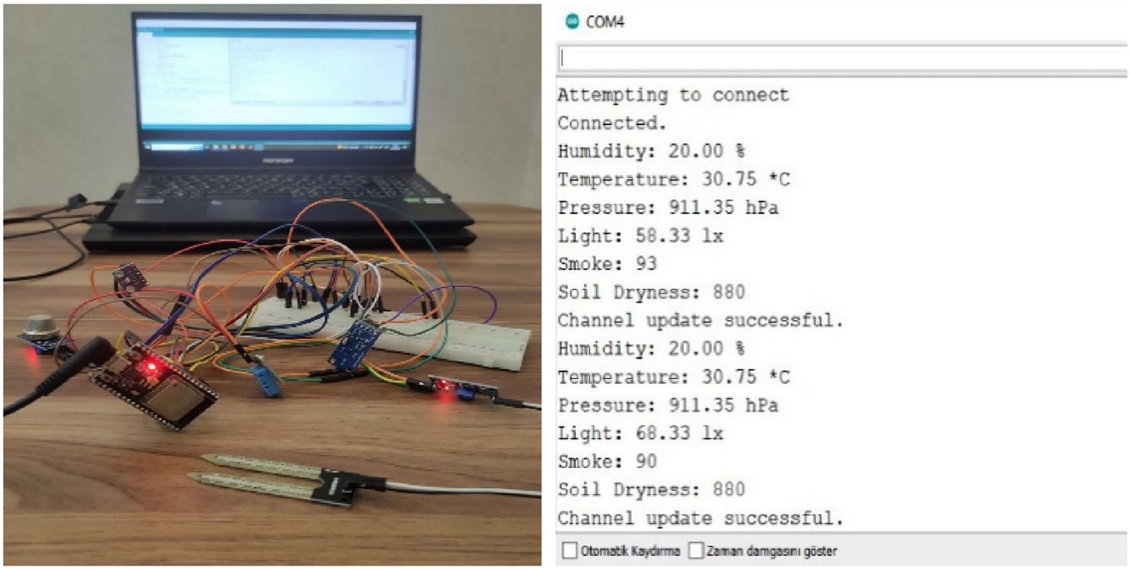

Remote Monitoring and Data Acquisition for Agriculture using IoT
GY-30
I²C
BME 280
I²C
MQ-2
AOUT
DHT11
One-wire
YL-69
AOUT
ESP-32S
Wi Fi
DATA AGGREGATION AND ANALYTICS
ThingSpeak
MATLAB
SMART CONNECTED DEVICES
ALGORITHM DEVELOPMENT SENSOR ANALYTICS
If the soil dryness or smoke value is high, send an email !
GUI

Fig. 1 Block diagram of the proposed approach based on IoT



```
Attempting to connect
Connected.
Humidity: 20.00 %
Temperature: 30.75 *C
Pressure: 911.35 hPa
Light: 58.33 lx
Smoke: 93
Soil Dryness: 880
Channel update successful.
Humidity: 20.00 %
Temperature: 30.75 *C
Pressure: 911.35 hPa
Light: 68.33 lx
Smoke: 90
Soil Dryness: 880
Channel update successful.
```

☐ Otomatik Kaydırma ☐ Zaman damgasını göster

Fig. 2 IoT-based experimental setup and serial port output proposed in the study

the Arduino IDE (the codes used in the study are shared on the https://github.com/fenesoguz/IoT-Agriculture). When critical values determined by the user according to the geographical region are observed, the features of initiating and stopping irrigation can be integrated into the proposed system by performing irrigation control.

Data Sending to the ThingSpeak

This section discusses sending data to ThingSpeak via Wi-Fi using ESP-32S. At this juncture, as illustrated in the pseudocode in Table 2, the initial step involves adding libraries pertaining to sensors and communication protocols. In order for ESP-32S to establish Wi-Fi connectivity through the modem, the Wi-Fi SSID and password are defined. For data retrieval from ThingSpeak, the Read API Key is utilized, and for data writing, the Write API Key comes into play. Hence, in order to accomplish data transmission to ThingSpeak, both the Channel ID and Write API Key need to be defined.

Subsequently, distinct configurations are set for five sensors. To visualize the transmitted data over Wi-Fi and

Table 2 Pseudocode of ThingSpeak data sending used in the study

```
Initialize
Include libraries.
Define Wi-Fi SSID and password.
Describe ThingSpeak channel ID and Write API Key.
    for i=1:number of sensors
    Configure i-th sensor.
    end
Begin serial port.
    While (infinite loop)
        if Wi-Fi connection time<15
        Begin Wi-Fi.
        Read sensor data.
            if isnan(sensors data)
            Return read sensors data.
            else
            Write read values to serial port.
            Send data to specified channel numbers of ThingSpeak.
            end
        if channel updated successful=yes

        Return read sensor data.

        else
        Print an error report.
        end
    end
```

the update status of the channel, a serial port is initiated. Then, the Wi-Fi connection is established, with a defined delay of 15 s. At this point, data are read from the sensors, and the read data are scrutinized; if any empty data are detected, the process loops back to the sensor data reading function. Successfully read data from the sensors are written to the serial port. Following this, the acquired sensor data are written sequentially to the designated channels in ThingSpeak. If the system successfully records the data in ThingSpeak, a new sensor reading process commences. However, if the process encounters an error, an error report is generated.

Analysis of Receiving Data

In this part, we explain how we analyze the data obtained from the sensors after transferring them to MATLAB. In this study, we also provide the pseudocode of the user-friendly GUI application that we have designed to provide these analyzes and present them to the user. As mentioned before, we express that we use different calculations according to the sensor type and the data collected. Herein we provide graphs for each sensor output where users can easily observe the relevant area. In the study, we determine the maximum, minimum, mean, standard deviation, and current value instantly and continuously for the data taken in a certain time interval.

On the other hand, we aim to send an e-mail to the user via ThingSpeak in case of high soil dryness and smoke values. In addition, we need to express that different values can be determined according to the demands of the users. Within the scope of the analysis of the received data, we give the pseudocode of the designed GUI application in Table 3. Thus, it can be seen how the data are analyzed in this application, which is used in the study or designed for the user.

Visualization and GUI App

This section describes how the user-friendly GUI application proposed in the study has been designed and how the data have been visualized. First, we create the figure section of the GUI application. While adding the graphics where users can see the current data to the mentioned section, we reflect the temperature, relative humidity, soil dryness, pressure, light, and smoke values obtained from each sensor to the related graphics with the help of ThingSpeak ChannelID and field ID. In the designed user-friendly GUI application, incoming data are transferred to the relevant figures every 3 s.

Table 3 Pseudocode of the GUI application presented in the study

```
1.   Initialize
2.   Select GUI background image.
         While (infinite loop)
3.       Obtain data from sensors utilizing readChannelID and fieldID.
4.       Plot the acquired data in 2-D.
5.       Specify the colors for the X-Y axis and the Grid.
6.       Assign a pause of 3 seconds for each graph.
             for i=1:(number of buttons used)
7.           Set the i-th pushbutton and i-th edit text.
8.           Calculate max, min, mean, standard deviation, and current values with pushbutton and print them to the screen.
                 if smoke>90
9.               Print a red warning on the screen.
                 else
10.              Print a green warning to the screen.
                 end
             end
11.      Stop the application when requested by the user.
         end
```

These values can also be determined according to the data collection times of the sensors. At this stage, data are being transmitted in the order shown in the graphic, and to allow users to access desired values within the specified time interval, we are adding buttons to this figure file that provide mathematical information such as maximum, minimum, mean, standard deviation, and current values. In parallel with the created figure file, we connect the relevant codes to design the GUI application in which the extensions of the graphics and buttons are included. Afterward, current values for smoke or soil dryness values can be determined depending on the user's request. In this designed user-friendly GUI application, we set this value as 90 for smoke. If this value is greater than 90, a red warning is given, and if it is smaller, it gives a green warning. Thus, if the smoke value is greater than 90, the user can understand that there is a situation that needs attention in the specified area.

Hereabouts, we present the flowchart of the user-friendly GUI application, which consists of start, plot and calculation blocks, as shown in Fig. 3. In addition, we provide a screenshot of the application called Remote_Agriculture_Monitoring_with_IoT, which we designed using this block diagram, as shown in Fig. 4. As can be seen from this figure, this graphical user interface application presents the temperature, relative humidity, pressure, light, smoke, and soil dryness values to the user as °C, %, hPa, lux, constant, and, constant, respectively. While the previous and current values can be easily read from the graph, the relevant date and time range can also be seen in this application. As can be seen in the figure, maximum, minimum, mean, standard deviation, and current values are provided when the user requests or push the button. After pushing the button, the requested information is presented to the user in approximately 3–4 s. In addition to continuously collecting up-to-date data from the sensors as long as the application is running, the user can easily close this application when requested, that is, the while loop ends when the user requests it.

Results and Discussion

In the study, firstly, light, temperature–pressure, smoke, humidity, and soil dryness values are measured in an agricultural field determined using sensors named GY-30, BME280, MQ-2, DHT11, and YL-69. Data are obtained from these sensors using the ESP-32S development board and coding is completed using the Arduino IDE compiler. Afterward, the data are transmitted to the ThingSpeak cloud service via a Wi-Fi connection, and these data are transferred to MATLAB. In addition, a user-friendly GUI

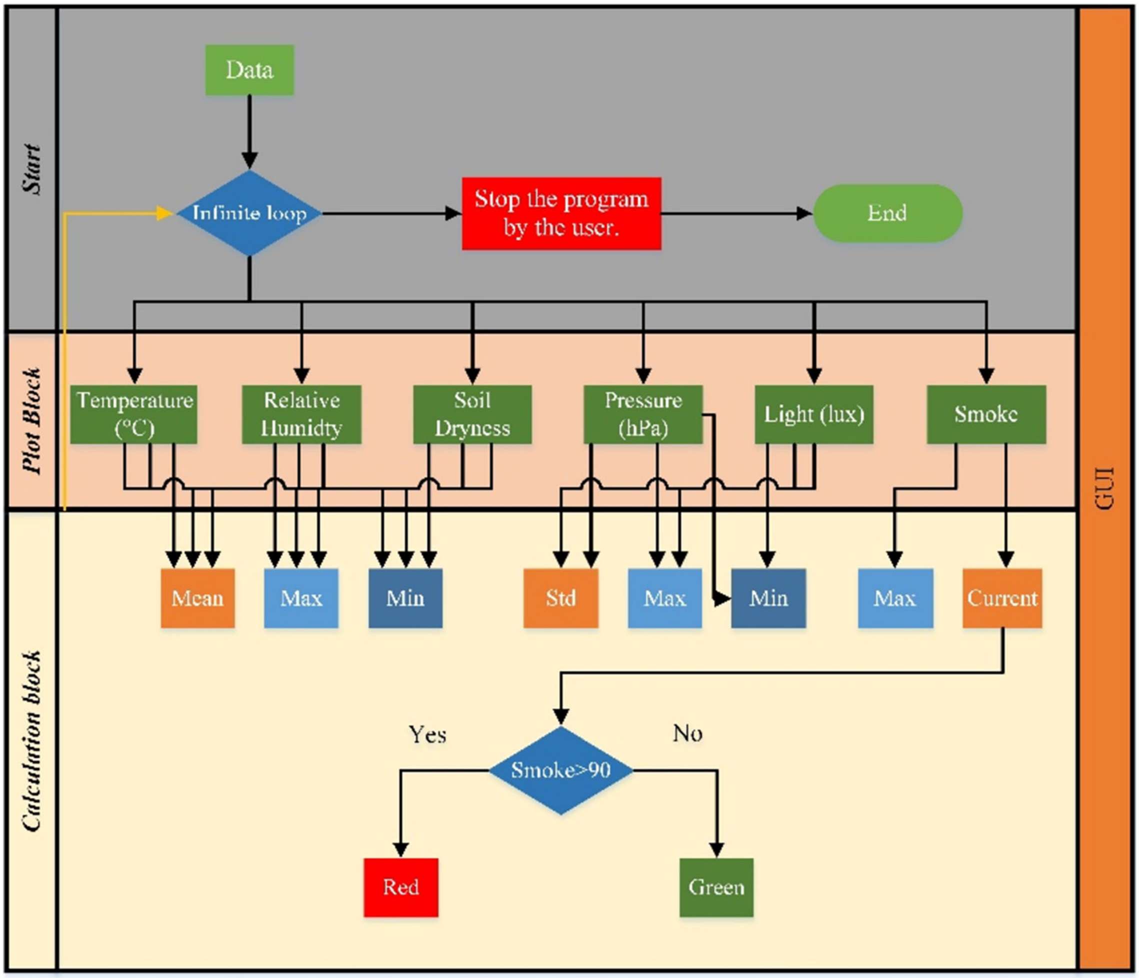


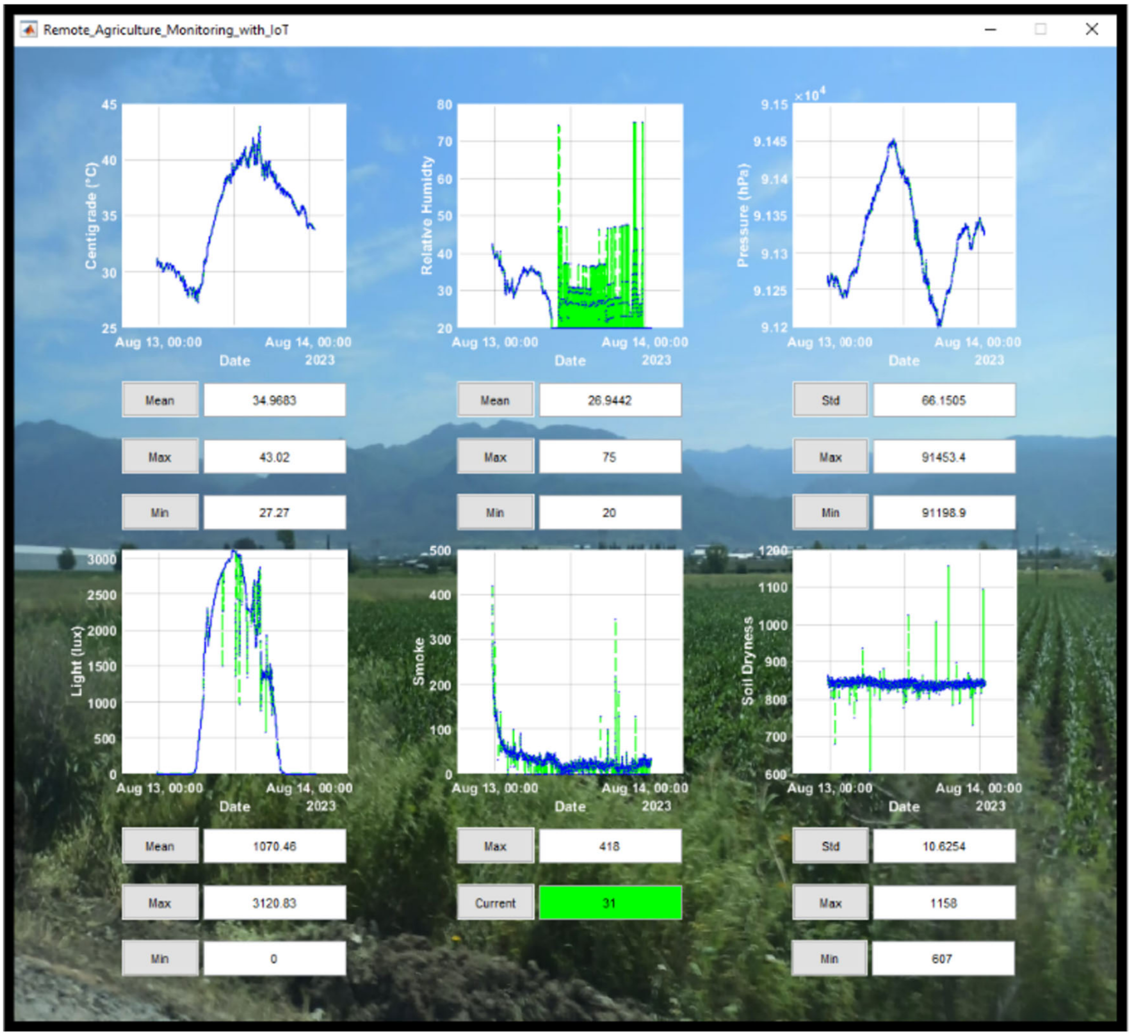
Fig. 3 Flowchart of the GUI application presented in the study

application that displays real-time data flow is designed within the scope of this study. In this software, temperature, relative humidity, soil dryness, pressure, light, and smoke values are instantly and continuously displayed on the graphs. When the user pushes the relevant button in the GUI, it presents statistical values such as maximum, minimum, mean, standard deviation, and current to users in about 3–4 s. In this GUI application, if the smoke value is higher than the specified value, it gives a red-colored warning to the user, if not, it gives a green-colored warning and informs the user about the relevant situation. In addition to all these, if the smoke and soil dryness values are higher than the threshold value, it provides services to the user by sending an e-mail.

The sensor data obtained in the study are given in Fig. 5. Herein, different graphics can also be added to this section. As shown in Fig. 5, five different channels, namely temperature, humidity, pressure, light, and soil dryness, are graphically visualized. In addition, a counter called “Field 5 Gauge” has been added for the smoke, and when this counter value is greater than 90, the “Fire Alarm” section gives a red warning to the user. The threshold value of the counter can also be changed with the help of this interface. In the study, it is also shown in the “Channel Location” that the experimental setup is carried out in Gaziantep/Turkey.

While the code is running in the proposed system, a trigger e-mail is sent to the user when it is detected that the threshold values for smoke and soil dryness are exceeded. As an example, the screenshot of an e-mail sent to the user when the threshold value for soil dryness is exceeded is presented in Fig. 6. Here, when it is determined that the soil moisture is low, it can be seen that an e-mail with the content “I need water” is sent with the time and date information. The user takes necessary actions such as turning the irrigation system on or off, as they can access relevant location and similar information.

As a result, smart agriculture applications are becoming increasingly important in today’s world, with various examples that enhance productivity, save human labor, and



Mean 34.9683
Max 43.02
Min 27.27

Mean 26.9442
Max 75
Min 20

Std 66.1505
Max 91453.4
Min 91198.9

Mean 1070.46
Max 3120.83
Min 0

Max 418
Current 31

Std 10.6254
Max 1158
Min 607

Fig. 4 Screenshot of the GUI application designed in the study

reduce time gaining popularity day by day. Studies in the literature show the applicability and widespread use of this study. Utilizing sensor and IoT technologies, this study involving remote access in the field of agriculture encompasses controls and interventions, when necessary, and implements a local application.

Thus, a useful smart agriculture application is presented to the user by processing and interpreting the data collected from the agricultural field. An affordable, practical, and easily usable smart agriculture application has been developed in the study. Considering these aspects, the proposed system seems to be competitive according to the literature [3, 4, 7, 10, 14, 15, 18, 25–28]. When examining the studies in the literature, one of the competitive aspects of the proposed method is its user-friendly GUI application that can graphically display real-time data flow. On the other hand, the use of the ThingSpeak cloud system in this study and its ability to enable real-time data analysis in various ways on the MATLAB platform also highlight the study. In the proposed method, the integration of real-time data into the GUI not only provides time and labor savings for users but also offers statistical analysis options.

Conclusions

An IoT-based system that can obtain data for a specified agricultural area and be monitored remotely is proposed in this manuscript. In this system, sensors named GY-30, BME280, MQ-2, DHT11, and YL-69 can measure light, temperature–pressure, smoke, humidity, and soil dryness, respectively. For this purpose, the necessary procedures for the ESP-32S development board are followed using I2C, AOUT, and one-wire. The coding part of the ESP-32S development board is completed using the Arduino IDE. Here, the sensors are constantly collecting data and

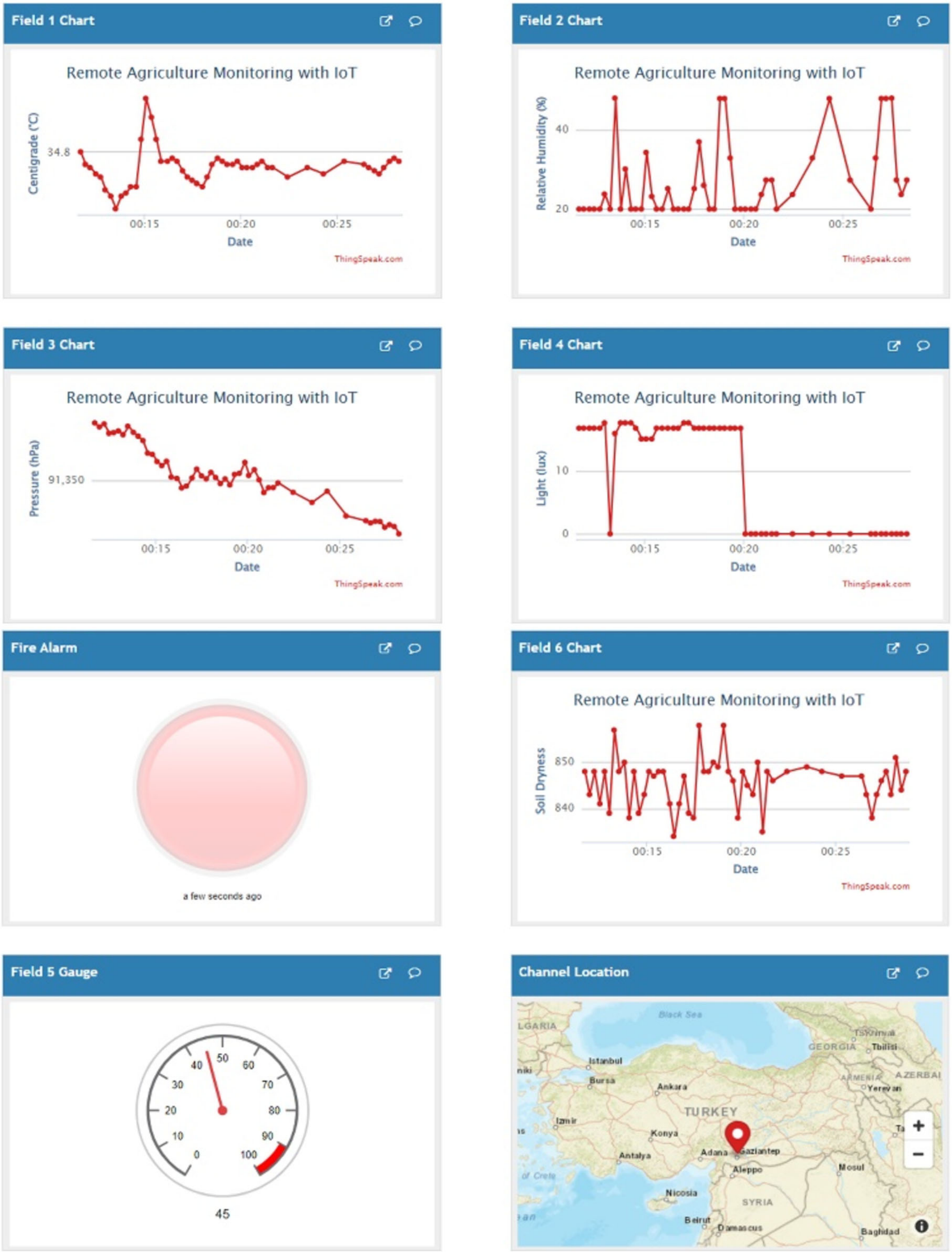
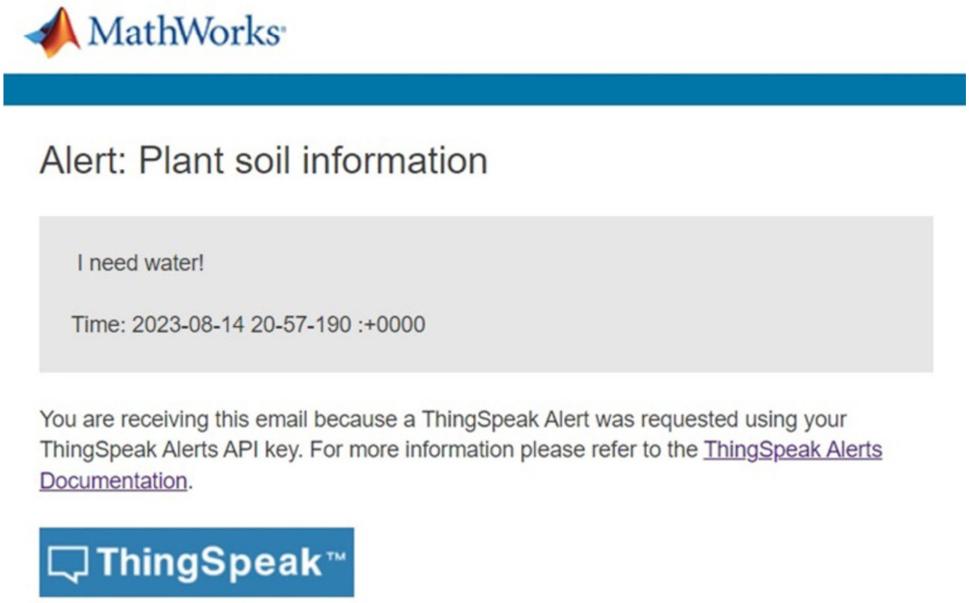


Fig. 5 ThingSpeak channel monitoring screenshot

transmitting these data to the ThingSpeak environment with the help of Wi-Fi. The current data stream is then collected in the cloud and visualized in ThingSpeak. Herein, these obtained data are transferred to MATLAB. So, the GUI application of this system can be designed.



Alert: Plant soil information

I need water!

Time: 2023-08-14 20-57-190 :+0000

You are receiving this email because a ThingSpeak Alert was requested using your ThingSpeak Alerts API key. For more information please refer to the ThingSpeak Alerts Documentation.

Fig. 6 Screenshot of trigger e-mail sent to user

We design a user-friendly GUI application that can display six output values obtained from five sensors. In this application, we present values such as maximum, minimum, mean, standard deviation, and current to them as soon as the user requests them. In addition, if the smoke value is too high in the application, we print a red warning, otherwise a green warning. In addition to all these, we ensure that the user is informed by sending an e-mail if the soil dryness and smoke values, which can also be determined by the user, are higher than the determined value. On the other hand, the evaluation of the model proposed in the study in a short period of time is considered a limitation of it. However, this model can also be evaluated and tested over a long period of time. Considering that the proposed model can successfully provide data flow and convey the necessary information to the user, it can be stated that it can be used reliably and save labor and time. In future studies, it is aimed to further increase the number of sensors used, support them with artificial intelligence-based models and use them for different purposes.

Author contributions FEO, MNE, and KMS were responsible for methodology, software, investigation, writing—original draft, conceptualization, and resources. AA was involved in methodology, investigation, writing—original draft, conceptualization, and resources.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical Approval Not applicable.

References

1. Al Murad M, Razi K, Jeong BR, Samy PMA, Muneer S (2021) Light emitting diodes (LEDs) as agricultural lighting: Impact and its potential on improving physiology, flowering, and secondary metabolites of crops. Sustainability 13(4):1985
2. Alkan A, Abdullah MU, Abdullah HO, Assaf M, Zhou H (2021) A smart agricultural application: automated detection of diseases in vine leaves using hybrid deep learning. Turk J Agric For 45(6):717–729
3. Aziz IA, Hasan MH, Ismail MJ, Mehat M, Haron NS (2009) Remote monitoring in agricultural greenhouse using wireless sensor and short message service (SMS). Int J Eng Technol IJET 9(9):1–12
4. Aziz IA, Ismail MJ, Haron NS, Mehat M (2008) Remote monitoring using sensor in greenhouse agriculture. In: 2008 international symposium on information technology. vol 4, p 1–8. IEEE
5. Bhattacherjee SS, Shreeshan S, Priyanka G, Jadhav AR, Rajalakshmi P et al (2020) Cloud based low-power long-range iot network for soil moisture monitoring in agriculture In: 2020 IEEE sensors applications symposium (SAS). p 1–5. IEEE
6. Bosch (2018) BME280 Data sheet https://www.mouser.com/datasheet/2/783/BST-BME280-DS002-1509607.pdf
7. Changqing C, Hui L, Wenjun H (2018) Internet of agriculture-based low cost smart greenhouse remote monitor system. In: 2018 chinese automation congress (CAC). p 3940–3945. IEEE.
8. Ekersular MN, Alkan A (2024) Detection of COVID-19 disease with machine learning algorithms from CT images. Gazi Univ J Sci 37(1):169–181
9. Gay W (2018) Advanced raspberry pi: raspbian linux and GPIO integration. Apress, New york
10. Hashim NMZ, Mazlan SR, Aziz MA, Salleh A, Jaafar A et al (2015) Agriculture monitoring system: a study. Jurnal Teknologi 77(1):53–59
11. Hatfield JL, Boote KJ, Kimball BA, Ziska LH et al (2011) Climate impacts on agriculture: implications for crop production. Agron J 103(2):351–370
12. Kim WS, Lee WS, Kim YJ (2020) A review of the applications of the internet of Things (IoT) for agricultural automation. J Biosyst Eng 45:385–400

13. Madakam S, Lake V, Lake V, Lake V (2015) Internet of Things (IoT): a literature review. J Comput Commun 3(05):164
14. Maia RF, Netto I, Tran A LH (2017). Precision agriculture using remote monitoring systems in Brazil. In: 2017 IEEE global humanitarian technology conference (GHTC). p 1–6. IEEE.
15. Mamatha V, Kavitha JC (2023) Remotely monitored web based smart hydroponics system for crop yield prediction using IoT. In: 2023 IEEE 8th international conference for convergence in technology (I2CT). p 1–6. IEEE
16. MQ-2 (2023) Semiconductor sensor for combustible gas data sheet https://www.pololu.com/file/0J309/MQ2.pdf
17. Oğuz FE, Alkan A, Schöler T (2023) Emotion detection from ECG signals with different learning algorithms and automated feature engineering. Signal Image Video Process 17:1–9
18. Pathmudi VR, Khatri N, Kumar S, Abdul-Qawy ASH, Vyas AK (2023) A systematic review of IoT technologies and their constituents for smart and sustainable agriculture applications. Sci Afr 19:e01577
19. Randomnerdtutorials (2017) Guide for YL-69 https://kysungheo.com/wp-content/uploads/2022/07/instructions.261306374.pdf
20. Robocs D (2010) DHT 11 Data sheet https://www.digikey.com/htmldatasheets/production/2071184/0/0/1/dht11-humidity-temp-sensor.html
21. Rohm (2011) GY-30 Data sheet https://www.mouser.com/datasheet/2/348/bh1750fvi-e-186247.pdf
22. Schollaert C, Austin E, Seto E, Spector J, Waller S et al (2023) Wildfire smoke monitoring for agricultural safety and health in rural Washington. J Agromed 18:1–14
23. Sunnetci KM, Kaba E, Beyazal Çeliker F, Alkan A (2023) Comparative parotid gland segmentation by using ResNet-18 and MobileNetV2 based DeepLab v3+ architectures from magnetic resonance images. Concurr Comput: Pract Exp 35(1):e7405
24. Tan WY, Then YL, Lew YL, Tay FS (2019) Newly calibrated analytical models for soil moisture content and pH value by low-cost YL-69 hygrometer sensor. Measurement 134:166–178
25. Vimalnath S, Kumar KS, Naveen K, Kumar DS (2023) A remote monitoring greenhouse agricultural farming with edge computing system. In: 2023 9th international conference on advanced computing and communication systems (ICACCS). vol 1, p 1204–1207. IEEE
26. Wang Z, Zhao C, Zhang H, Fan H (2011) Real-time remote monitoring and warning system in general agriculture environment. In: 2011 international conference of information technology, computer engineering and management sciences. vol 3 , p 160–163. IEEE
27. Zhang Z, Wu P, Han W, Yu X (2017) Remote monitoring system for agricultural information based on wireless sensor network. J Chin Inst Eng 40(1):75–81
28. Zhao JC, Zhang JF, Feng Y, Guo JX (2010) The study and application of the IOT technology in agriculture. In: 2010 3rd international conference on computer science and information technology. vol 2, p 462–465. IEEE

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.