IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 20, 2019, accepted September 8, 2019, date of publication September 16, 2019, date of current version October 1, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2941521

A Mobile Greenhouse Environment Monitoring System Based on the Internet of Things

XIA GENG[1], QINGLEI ZHANG[2], QINGGONG WEI[1], TONG ZHANG[1], YU CAI[1], YONG LIANG[1], AND XIAOYONG SUN[1]
[1]School of Information Science and Engineering, Shandong Agricultural University, Tai'an 271018, China
[2]Shenxian Meteorological Bureau, Liaocheng 266590, China

Corresponding authors: Yong Liang (yongl@sdau.edu.cn) and Xiaoyong Sun (johnsunx1@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 31571306.

ABSTRACT A knowledge of the environmental information of different spaces of large greenhouse is a prerequisite for effective control, and multipoint monitoring is therefore needed. In view of the problems of current greenhouse environmental monitoring, a mobile greenhouse environment monitoring system was designed based on the Internet of Things. A four-layer system architecture with outstanding motion control functions was constructed that uses mobile acquisition rather than multiple sensing nodes to realize the automatic collection of greenhouse environmental information and capture pictures of the crops with low cost. In this study, a Raspberry Pi and an Arduino chip were combined for the first time in agriculture greenhouse environmental monitoring, with the former serving as the data server and the latter as the master chip for the mobile system. Firstly, the application layer server was deployed on the Raspberry Pi, secondly, due to its compact size and stable performance, Raspberry Pi and sensors etc. were all integrated into the mobile system, shortening the physical distance between the data acquisition end and the data processing end, and serial communication was used. In addition, a dedicated communication protocol with Cyclic Redundancy Check (CRC) checking was designed to reduce data loss at the transmission layer. The data was denoised using a limiting filtering algorithm and a weighted average filtering algorithm to improve quality of the data. The experimental results show that the system can effectively realize multi-point environmental monitoring of the greenhouse.

INDEX TERMS Environmental monitoring, greenhouse, Internet of Things, mobile system.

I. INTRODUCTION

According to Ren *et al.* [1], with the continuous development of intelligent agriculture and precision agriculture, facility agriculture in the form of greenhouses has begun to occupy an important position in China's agricultural sector. As shown by Zheng [2], a greenhouse environment is an uncertain, nonlinear and time-varying distributed parameter system. Therefore, it is necessary to monitor the greenhouse environment in real time to make the greenhouse environment (temperature, humidity and illumination, etc.) most suitable for the growth of crops. Also, this helps achieve high quality and high yield of crops.

As the most fundamental and important way of achieving the automation and efficient management of greenhouses, greenhouse environmental monitoring has been studied by researchers worldwide. Montoya *et al.* [3] developed and integrated a multi-platform application based on Android technology and using a wireless mesh network to monitor and control intensive agriculture. Pahuja *et al.* [4], Lee *et al.* [5], Nagesh Kumar [6], He *et al.* [7] and Du *et al.* [8] studied remote measurement and control technology for greenhouse environments. Ren *et al.* [1] combined the Internet of Things (IoT) and WeChat to design and develop a greenhouse environmental monitoring and temperature prediction system that effectively realized the lightweight design of the data acquisition end. In view of the problems faced by previous agricultural wireless sensor networks, such as high energy consumption, high cost and poor transmission performance, Chen *et al.* [9] designed a wireless sensor network that works within the China-specific 780 MHz frequency band and fully complies with the IEEE802.15.4c standard, and applied this to greenhouse environmental monitoring. To efficiently monitor the greenhouse environment,

---

The associate editor coordinating the review of this manuscript and approving it for publication was Chin-Feng Lai.

Li *et al.* [10] developed a common platform for an IoT measurement and management system based on an Android intelligent gateway and Google Web Toolkit remote web server, thus effectively avoiding the secondary development of web server and intelligent gateway applications caused by changes in the sensor and data acquisition unit node, and providing a common platform for the monitoring and managing of agricultural information. Guo and Ma [11] and Shang *et al.* [12] used advanced temperature and humidity sensors to collect temperature and humidity signals, and applied a single chip microcomputer (SCM) as a hardware core to monitor and control the greenhouse environment in real time. Zheng [13] used wireless sensors to collect various environmental factors in the greenhouse, and then adopted ZigBee technology to realize a self-organizing wireless sensor network, giving intelligent control of the greenhouse environment. Yang *et al.* [14] introduced WiFi wireless communication technology to design a greenhouse environment monitoring system based on agricultural IoT, which is suitable for large-area agricultural parks including multiple greenhouses. Lu *et al.* [15] designed a distributed temperature monitoring system for a greenhouse, consisting of a gateway module, a bus communication module, SCM temperature acquisition nodes and a bus temperature acquisition module. Kong [16] designed a greenhouse environment monitoring system based on the Internet of Things, this system uses Time Division (TD) terminals for data collection and Time Division-Synchronous Code Division Multiple Access (TD-SCDMA) wireless communication networks for data transmission.

Regardless of whether the system is designed for a single large greenhouse or for multiple greenhouses, a knowledge of the environmental information of different spaces is a prerequisite for effective control, and multipoint monitoring is therefore needed. However, the majority of currently available greenhouse environment monitoring systems, whether based on wired or wireless transmission, need to deploy a large number of sensors, control modules or data communication modules at the perceptual layer of the IoT. These systems have several drawbacks, such as high cost, unreliability, difficulty in maintenance etc.

Thanks to the development of agricultural robots in recent years, mobile environment detection system has been proposed [17], [18]. Robots play an important role in greenhouse technology and equipment [19]. At this stage, the main research and application direction of greenhouse robots is to improve the level of agricultural intelligent automatic production [20], [21]. The robots are mainly used in greenhouse operations such as picking, spraying, patrolling, navigation and handling [22]–[25]. For example, Acaccia *et al.* [26] proposed a virtual prototype for plant health monitoring and service robots that distribute local chemicals, drugs and fertilizers to greenhouse plants. This model has not been implemented in the crop filed yet. Martinet and Simon [27] developed a micro-climate multi-criteria decision-making expert system model based on fuzzy rules. In the control system, nodes communicated with each other through the WSN or with the greenhouse server. Real-time environmental data were transmitted to remote users through the non-cellular network. This system separates the mobile device from the database and therefore does not provide timely feedback.

Due to the need for intelligence, greenhouse robots generally have complex control systems, which have the disadvantage of high manufacturing costs. Unlike robots, which focus more on intelligent control, environmental monitoring systems are more concerned with the effective and timely monitoring of environmental information. However, there are few studies of mobile greenhouse environment monitoring systems, especially when it comes to mobile greenhouse environmental monitoring based on agricultural IoT.

As a result, we develop a system, which combines the Raspberry Pi and the Arduino chip together for the first time in agriculture greenhouse environmental monitoring. The whole system is integrated into the mobile system, shorting the physical distance between the data acquisition end and the data processing end. Thus, the cost of data collecting and data processing is reduced, and the reliability of data transmission are improved.

II. OVERALL DESIGN OF THE SYSTEM

The mobile greenhouse environment monitoring system proposed in this study is based on an IoT architecture. Following the development of IoT technology, its architecture is becoming increasingly standardized. Several authors [23], [28]–[32] have studied IoT architectures. Compared with other environmental monitoring systems based on IoT architectures, the system proposed in this study is characterized by its mobility, and the motion of the equipment itself in the greenhouse needs to be well controlled. Therefore, it is necessary to combine the motion control of the system and IoT technology. Compared with the automatic control function of the perceptual layer in the three-layer architecture of IoT, the function of motion control of the system is more abundant and independent. So, the motion control function as the control layer is divided separately in this system architecture, which aims to emphasize the control of the equipment itself and the environment. In fact, the adjustment in architecture is the refinement and improvement to the function of three-layer IoT architecture. The system therefore adopts an architecture containing four layers, from the bottom to the top, these are the perceptual layer, the control layer, the transmission layer and the application layer (Fig. 1).

The main function of the perceptual layer is to obtain the temperature, humidity, illumination, carbon dioxide concentration and pictures of the crop. By realizing control of the lower computer, the control layer is responsible for the automatic tracking in the greenhouse. It also accurately determines the location of each data acquisition point. It also controls the data acquisition mode and format, and converts and encapsulates some data at the perceptual layer. The transmission layer completes the reliable transmission of data between

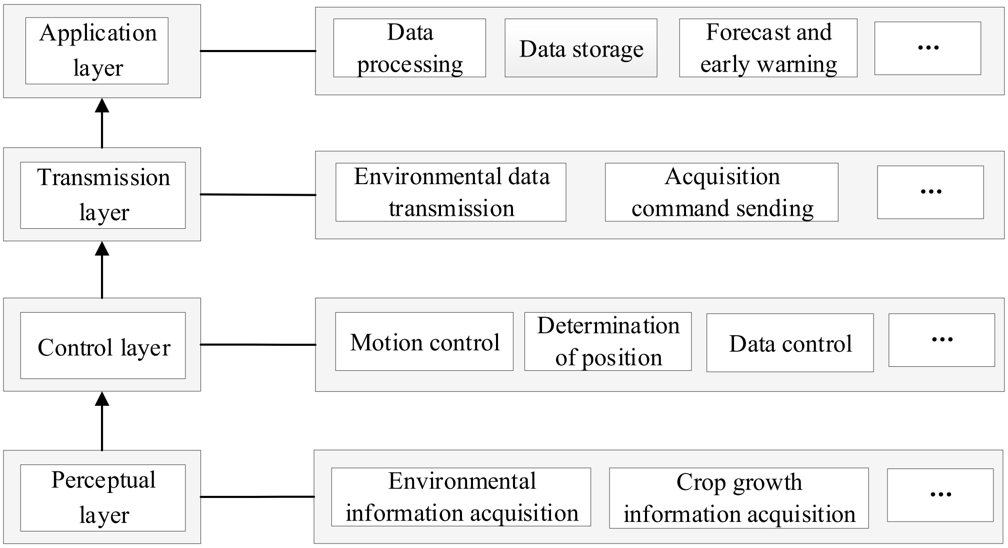


FIGURE 1. The system architecture includes four layers: perceptual layer, control layer, transmission layer and application layer.

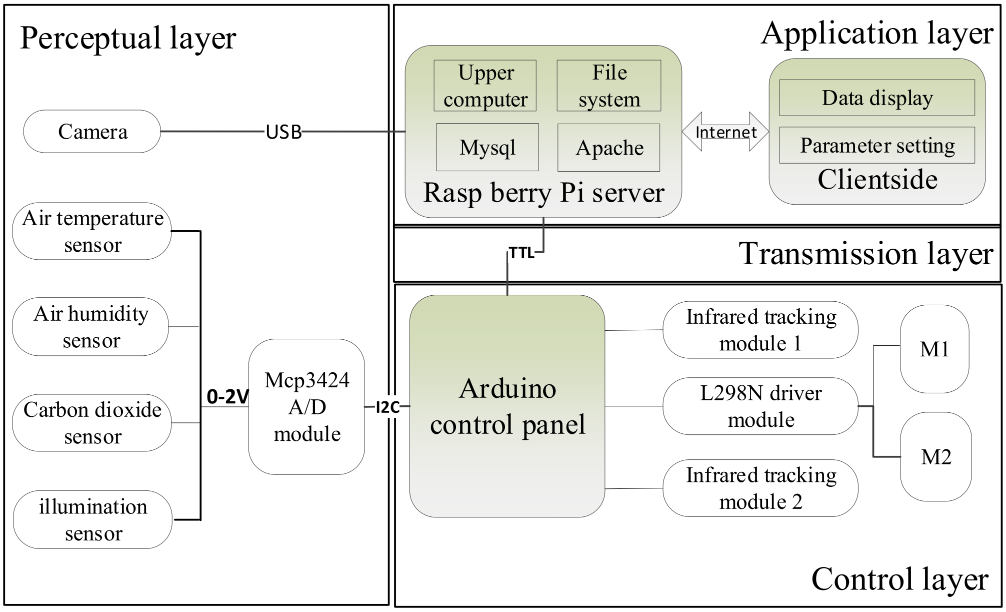


FIGURE 2. The hardware components of four-layer system architecture. The parameter setting page of client side can fix automatic fixed point mode and automatic tracking mode, as well as data collection frequency, etc.

the perceptual layer and the server, and develops relevant protocols to improve data quality. The main function of the application layer is to implement all user-oriented applications such as data processing, data storage, data management, forecasts and early warning, service decision making etc.

The system hardware block diagram, showing the four-layer architecture of the system, is given in Fig. 2.

The perceptual layer includes a camera, an air temperature sensor, an air humidity sensor, a carbon dioxide sensor, an illumination sensor and an AD conversion module. The control layer mainly consists of the Arduino main control panel, the LP298N motor drive module, a motor and an infrared tracking sensor. Using a Raspberry Pi as the data server, serial communication is adopted for data transmission at the transmission layer due to its relative security and reliability. The server side of the application layer of the system is built on the Raspberry Pi, including an upper computer to receive the data, a database, a file system and a web server. Of these, the upper computer is responsible for receiving data and storing it in the MySQL database; the database is used to store the collected data; the file system is used to store the greenhouse pictures captured by the Raspberry Pi; and the web server manages and views the data.

III. SYSTEM HARDWARE DESIGN

A. MOBILE CONTROL MODULE

1) DESIGN OF THE AUTOMATIC TRACKING SCHEME

There are currently two cost-effective forms of automatic tracking schemes: ultrasonic positioning and infrared tracking. Of these, the anti-interference of ultrasonic tracking is poor and the transmitter requires a power supply, making it very inconvenient to use [33], [34]. Infrared tracking is therefore used in the proposed system.

Infrared tracking works as follows. The infrared emitting diode of the infrared sensor probe continuously emits infrared rays, and when these fall onto the surface of a dark or black object, the reflected light is weak or even absent. Thus, the phototransistor cannot receive the reflected light, and the sensor output will be high; conversely, when infrared light falls onto the surface of a light or white object, the reflected light is strong enough for the phototransistor to receive the reflected light, and the sensor output will be at a low level.

Black and white tapes are used for infrared tracking in this system. Since tracking is difficult at night [35], [36], brightness diodes are added at both ends of the infrared sensor to illuminate the road at night and improve nighttime tracking.

The system uses two five-way tracking sensors manufactured by YWRobot, which are mounted on the front and rear of the mobile car, respectively, and which work separately when the mobile car is moving forward or backward. The five-way tracking sensor has five output pins, corresponding to five infrared probes numbered A, B, C, D and E, respectively. During the tracking process, the tracking sensor has six possible states (Fig. 3).

Fig. 3(a) shows the normal state of system operation.

When the states shown in Fig. 3(b) and (c) occur, the output of infrared probe A is one or zero, whereas the outputs of the infrared probes B, C, D and E are all zero, and the control system moves straight ahead.

When the state shown in Fig. 3(d) occurs, the outputs of infrared probes A, C and E are zero, the output of infrared probe B is one, the output of infrared probe D is zero or one, and the control system turns left until it reaches the state shown in Fig. 3(a).

When the state shown in Fig. 3(e) occurs, the outputs of infrared probes A, B and D are zero, the output of infrared probe C is one, and the output of infrared probe E is zero or one, and the control system turns right until it reaches the state shown in Fig. 3(a).

When the state in Fig. 3(f) occurs, the outputs of infrared probes A, B, C, D and E are one, and the system reaches the end of its trajectory, undergoing reverse motion or stopping.

2) MODES OF OPERATION

The system has three modes of operation: manual mode, automatic fixed point mode and automatic tracking mode, and these can be set via the web front end of the system.

In manual mode, the upper computer in the Raspberry Pi terminal does not send any commands but simply receives the data sent by the Arduino. The Arduino, when is started, will enter the manual mode within five seconds after receiving the remote control command. In manual mode, the infrared remote control technology can be used to make the system move forward or backward, turn left or right and to acquire information. The manual mode is further divided into a manual tracking mode and a manual fixed mode. A track is not needed in the manual mode for tracking.

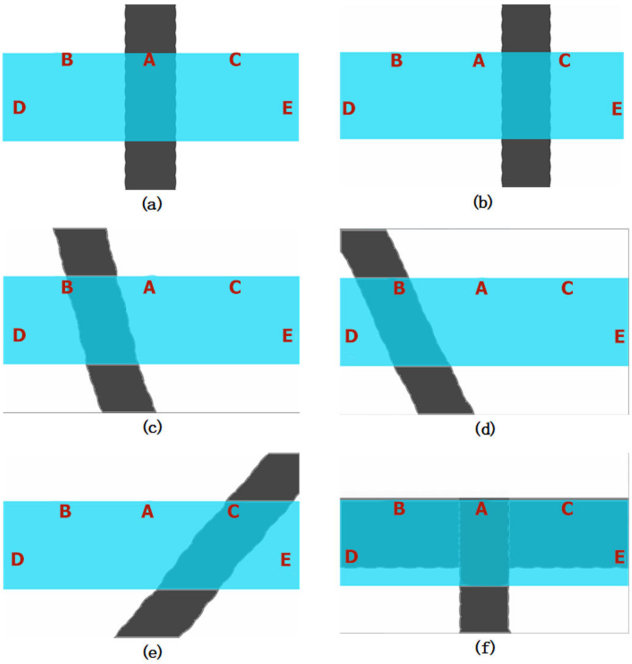


FIGURE 3. Six states of tracking process: (a) shows the normal state of system operation, (b) and (c) control system to move straight ahead, (d) and (e) control system to turn left or right respectively, and (f) shows the system reaching the end of its trajectory.

In the automatic fixed point mode, the upper computer of the Raspberry Pi terminal sends a fixed point acquisition command to the Arduino every hour. After receiving the data acquisition instruction, the Arduino will conduct a data acquisition to obtain a piece of data.

In the automatic tracking mode, the upper computer at the Raspberry Pi terminal sends an automatic tracking command to the Arduino. After receiving the command, the Arduino will then perform data acquisition of a tracking cycle.

B. TRANSMISSION CONTROL MODULE

1) DATA TRANSMISSION

This system integrates the Raspberry Pi server into the mobile system, which markedly reduces the physical distance between the upper and lower computers. Since the system is characterized by a small amount of communication data and high requirements for the reliability of communication, a Transistor-Transistor Logic (TTL) serial port is used to conduct the communication between the upper computer and the lower computer.

To ensure the reliability and accuracy of information transmission, the data transmission protocol is customized.

TABLE 1. Protocol stack format.

Byte	Length
Beginning character	1
Information length	2
Command type	2
Extended area	2
Area number	1
Reason for transmission	1
Transmission destination	1
Information body	Variable
CRC check	2
End character	1

TABLE 2. Command type.

The 1st byte	The 2nd byte	Meaning
0x0A	0x0A	Automatic control
0x0A	0x0C	Automatic transmission
0x0B	0x0A	Automatic control
0x0B	0x0C	Automatic transmission

TABLE 3. Reason for Transmission.

Byte	Meaning
0x03	Request from the upper computer
0x04	Response of the lower computer
0x05	Response of the upper computer
0x06	Burst information

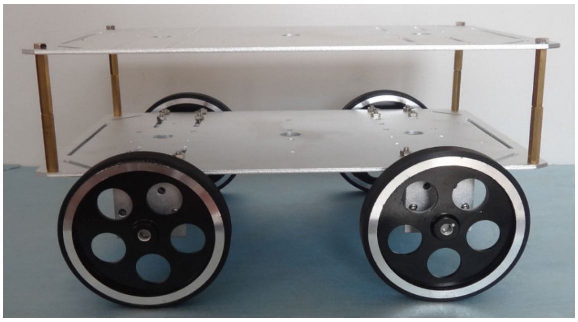


FIGURE 4. The overall size of the system is 30∗20∗20 cm, and the diameter of the wheel is 9.5 cm.

As shown in Table 1, the length unit is 1 byte. The protocol stack format is designed as follows: the initial character is 0xBB; the final character is 0xEE; the information length is 2 bytes (indicating the length of the data); the command type is 2 bytes (indicating the type of command); there is an extended area of 2 bytes (reserved for future function expansion, with a default of zero); the area number is 1 byte (corresponding to the greenhouse number); the reason for transmission is 1 byte (indicating the reason for transmitting this information); the transmission destination is 1 byte (indicating the receiving end of the information); 0x0A represents the upper computer; 0x0B denotes the lower computer; and the CRC check code is 2 bytes.

The specific command types and reasons for transmission are shown in Tables 2 and 3, respectively.

As can be seen from Table 2, if the second byte of the command type is 0x0A, the information is a control command. In this case, if the body of the information is 0x1A, this means that the lower computer's tracking acquisition instruction is turned on. If the body of the information is 0x1B, the lower computer's fixed-point acquisition instruction is turned on. If the second byte of the command type is 0x0C, this indicates a data transmission instruction and the length of the information body is variable according to the specific transmitted data.

2) PROCESSING OF ACQUIRED DATA

In the process of data collection, due to the interference of several random factors, a small number of outliers can be found among the acquired data, which affect the quality of data acquisition. The system integrates a limiting filtering algorithm with a weighted average filtering algorithm to reduce these singular values in the acquired data.

Two successive samplings are subtracted using the limiting filtering algorithm, and the absolute value of the increment is compared with the maximum difference A allowed by the two samplings. The size of A depends on the specific conditions of the sensor under test. If this is less than or equal to the maximum allowable difference A, the current sampling is valid; otherwise, the last sampled value will be taken as the sample for this data [37].

The data values are collected 5 times in a row, and analyzed by weighted average filtering algorithm. Then the results are multiplied by different weighting coefficients respectively and accumulated. The weighting coefficients are generally ranked from small to large, in order to highlight the effects of several successive samplings, thus improving the system's understanding of the trend in the parameter changes. The weighting coefficients are decimals whose sum is equal to 1. The accumulated sum after the weighting operation (that is, D in (1)) is the effective sampling value.

$$D = \sum_{i=0}^{N-i} C_i X_{N-1}, \tag{1}$$

where D is the weighted average of N sampling values; X_{N-i} is the N-th sampling value; N is the number of samplings; and C_i is the weighting coefficient, which reflects the proportion of the various sampling values in the average [38].

C. BEARING STRUCTURE OF THE SYSTEM

In order to ensure that the system can move in a stable manner at a constant velocity in the greenhouse along the specified route, the chassis shown in Fig. 4 is designed as a carrier for the system. The overall size of the moving car is 30∗20∗20 cm, and there is enough space for hardware

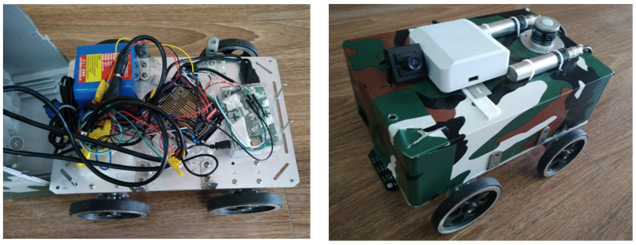


FIGURE 5. Four geared motors are used as power. The maximum running velocity of the system is 30 cm/s, and the maximum load is 10 kg.

equipment. The diameter of the wheel is 9.5 cm, guaranteeing good trafficability. , and four geared motors are used as power. The maximum running velocity of the moving car is 30 cm/s, with a maximum load of 10 kg.

An assembly drawing of the mobile greenhouse environment monitoring system, including the design of the above hardware system, is shown in Fig. 5.

IV. SYSTEM SOFTWARE DESIGN

The design of the system software includes development of the Arduino control program, the upper computer software and the data management system, and the design of the corresponding database. (The software is available on the request)

A. IMPLEMENTATION OF THE ARDUINO CONTROL PROGRAM

The development environment of the Arduino control program is the official Integrated Development Environment (IDE) provided by Arduino, which uses the Arduino language i.e. the various application programming interfaces provided by the Arduino core library file.

When the Arduino system is powered on, system initialization is performed, including initialization of the infrared receiving module and the serial communication module, with a baud rate of 9600 for serial communication. The timer is then turned on. If the infrared receiving module receives an infrared remote control command within five seconds, it enters either the manual or automatic mode, and the mode can only be modified by rebooting the system. During operation of the system, the value of the global variable flag is continuously monitored. When the flag is zero, the system moves; when the flag is one, the system stops.

If the system enters the manual mode, it first determines whether the command is an acquisition or a motion command by analyzing the infrared control code. If it is a motion command, the motion of the car (e.g. going forward, backward, left or right) is controlled according to the content of the code. If it is an acquisition command, the system will acquire the environmental data and then upload these data through the serial port. After executing this command, the system enters a waiting state for the next remote command.

If the system enters the automatic mode, then a CRC check will be performed upon receiving the serial port command. If the check is passed, the command will be parsed according to the above mentioned data communication protocol, thereby distinguishing the command type. If it is an automatic tracking acquisition command, a second thread will be started, and this new thread will turn on the timer to ensure that the control system acquires environmental data every 30 seconds. If it is an automatic fixed point acquisition command, data will be acquired once and the system will then wait for the next serial command.

The main thread determines whether or not the end point of the track has been reached by the tracking sensor. If not, the car moves forward until it reaches the end point. Upon reaching the end point, the system determines whether the car has reached the starting point. If not, it moves backwards until it reaches the starting point, stops the motion, ends the second thread, and then waits for the next serial port command.

Rather than supporting multithreading, Arduino uses ProtoThreads to simulate multithreading with macros. ProtoThreads can carry out the operating system's functions and also save memory to a certain extent.

Fig. 6 shows a flow chart for the Arduino program.

B. IMPLEMENTATION OF THE UPPER COMPUTER SOFTWARE

This system uses Raspberry Pi 3B+ as the server platform. The Raspberry Pi can support a variety of mainstream programming languages. Since Python has the advantages of fast operation and low memory usage, is free and open source, requires no compilation, and is a cross-platform language, it is used as the development language for the server side of the upper computer. In this system, the upper computer does not need to carry out human-computer interaction functions such as data viewing; thus, the upper computer software only runs in the background after being powered on, in the form of a system service.

Once the Raspberry Pi is powered on, it will automatically start the upper computer software in the background, load the required software package and read the set of operational parameters from the database, which mainly concern the operating modes of the equipment, including the automatic tracking, automatic fixed point and manual modes. The peripheral equipment will then be initialized, including the camera and serial port.

The operating mode of the equipment then needs to be determined. If this is the automatic tracking or automatic fixed point mode, a new thread called a timing thread is initiated. The preset time is then read, and based on this, the data acquisition instruction of the corresponding mode is sent through the serial port, using the above mentioned communication protocol, at a fixed time.

The main thread waits for data, which are received via the serial port. All data undergo a CRC check. If the data pass the check, data parsing will be performed; otherwise, these data will be discarded rather than processed, and the main thread will continue waiting for data received via the serial port.

Data analysis is used to determine whether a command is a data command. If so, the capturing thread will be started to capture pictures and save them to the file system. The main

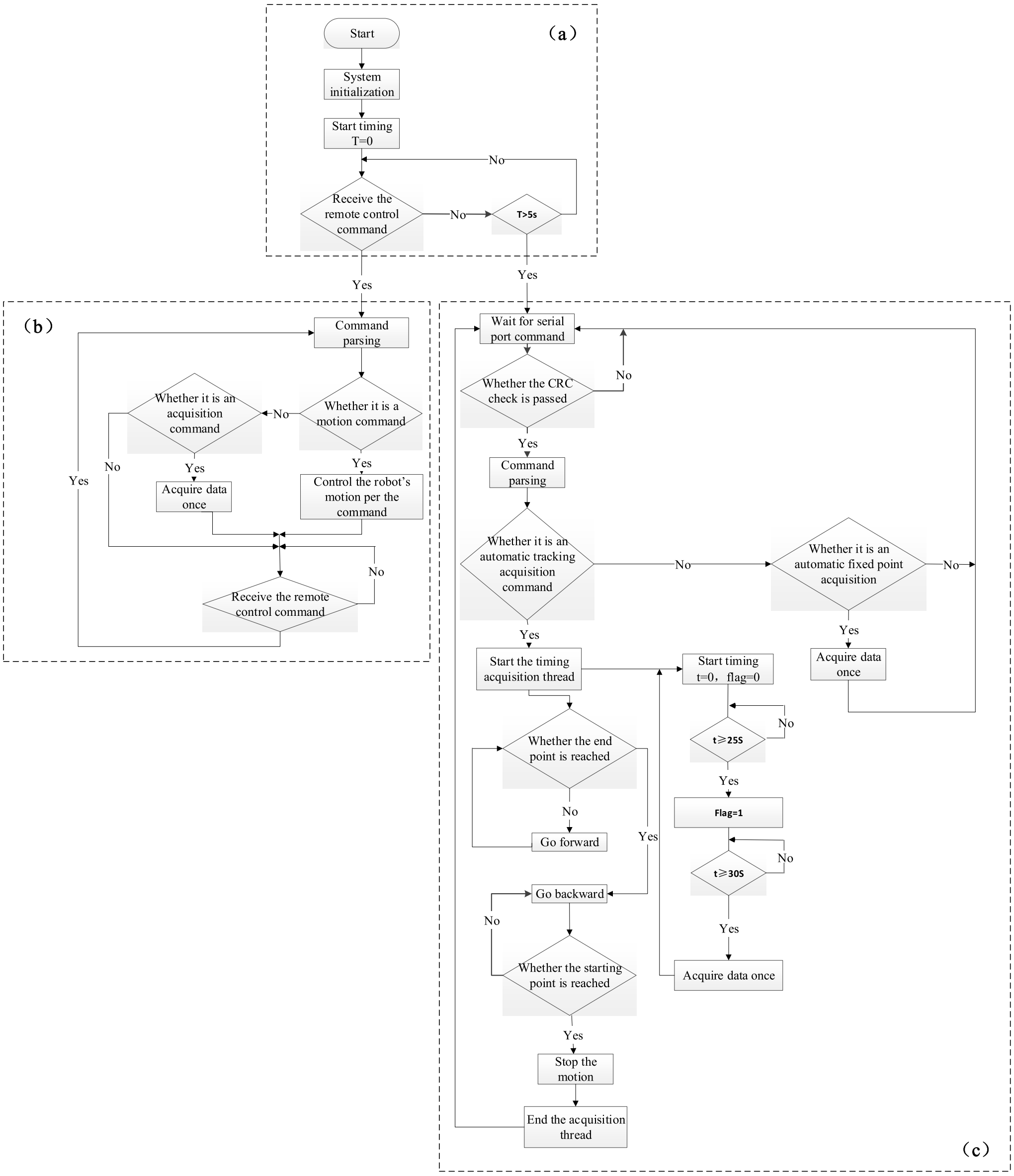


FIGURE 6. **The Arduino running flowchart. (a) System initialization module, (b) manual mode module, and (c) automatic mode module.**

thread then saves the data to the database and continues to wait for data received by the serial port.

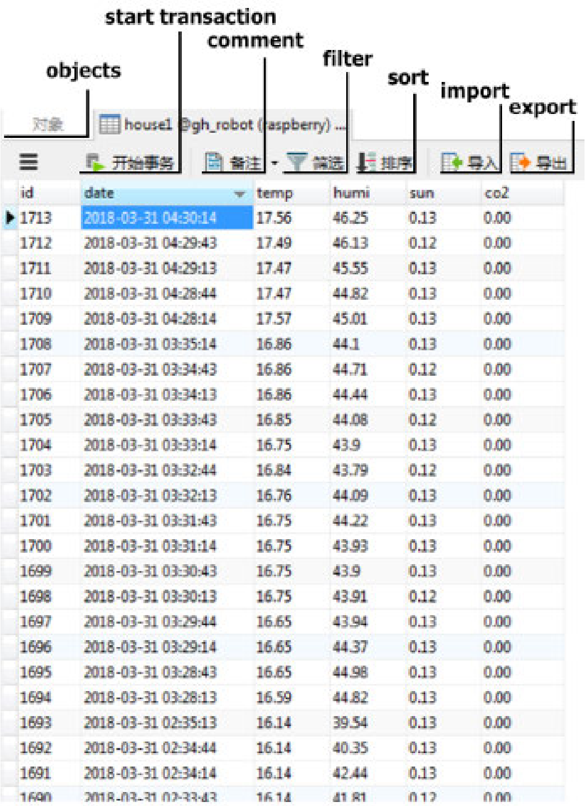
Fig. 7 shows the data stored on the upper computer. Since the carbon dioxide sensor does not have a power supply in the motion mode, data about carbon dioxide are not collected here. Fig. 8 shows the pictures captured by the upper computer.

C. IMPLEMENTATION OF THE DATA MANAGEMENT SYSTEM

The data management system is a human-computer interaction system with functions such as the viewing and managing of real-time and historical data, downloading and saving of acquired data, setting operating parameters and modes. In order to allow it to run on a Raspberry Pi, the data management system is simple and lightweight. The integrated development environment is Zend Studio, which is developed using the open source Model View Controller (MVC) framework ThinkPHP. The Bootstrap framework is used at the front end, and the ECharts plugin is used to display the charts.

One journey of the system back and forth in the greenhouse is referred to as a data acquisition period, and takes between seven and eight minutes. Within a given data acquisition period, the same acquisition point generates two pieces of data in succession. To reduce the data error for the same acquisition point, the acquired data are processed as follows.

Suppose that n sets of data are obtained within one data acquisition period, and these are denoted as $D_1, D_2, \ldots, D_n$.



id	date	temp	humi	sun	co2
1713	2018-03-31 04:30:14	17.56	46.25	0.13	0.00
1712	2018-03-31 04:29:43	17.49	46.13	0.12	0.00
1711	2018-03-31 04:29:13	17.47	45.55	0.13	0.00
1710	2018-03-31 04:28:44	17.47	44.82	0.13	0.00
1709	2018-03-31 04:28:14	17.57	45.01	0.13	0.00
1708	2018-03-31 03:35:14	16.86	44.1	0.13	0.00
1707	2018-03-31 03:34:43	16.86	44.71	0.12	0.00
1706	2018-03-31 03:34:13	16.86	44.44	0.13	0.00
1705	2018-03-31 03:33:43	16.85	44.08	0.12	0.00
1704	2018-03-31 03:33:14	16.75	43.9	0.13	0.00
1703	2018-03-31 03:32:44	16.84	43.79	0.12	0.00
1702	2018-03-31 03:32:13	16.76	44.09	0.13	0.00
1701	2018-03-31 03:31:43	16.75	44.22	0.13	0.00
1700	2018-03-31 03:31:14	16.75	43.93	0.13	0.00
1699	2018-03-31 03:30:43	16.75	43.9	0.13	0.00
1698	2018-03-31 03:30:13	16.75	43.91	0.12	0.00
1697	2018-03-31 03:29:44	16.65	43.94	0.13	0.00
1696	2018-03-31 03:29:14	16.65	44.37	0.13	0.00
1695	2018-03-31 03:28:43	16.65	44.98	0.13	0.00
1694	2018-03-31 03:28:13	16.59	44.82	0.13	0.00
1693	2018-03-31 02:35:13	16.14	39.54	0.13	0.00
1692	2018-03-31 02:34:44	16.14	40.35	0.13	0.00
1691	2018-03-31 02:34:14	16.14	42.44	0.13	0.00
1690	2018-03-31 02:33:43	16.14	41.81	0.12	0.00

FIGURE 7. The data screenshot obtained in automatic tracking mode using Navicat, in which "temp" represents "temperature", "humi" represents "humidity", "sun" represents "illumination", and "co2" represents "carbon dioxide". The data are stored in the MySQL database of Raspberry Pi.

FIGURE 8. The images captured by the upper computer.

The data acquired at the ith acquisition point will be processed as follows:

$$L_i = \frac{D_i + D_{n+1-i}}{2} (i = 1, 2, \ldots, n/2) \tag{2}$$

V. RESULTS AND DISCUSSION

An experiment was carried out in Greenhouse No. 10 of the College of Horticulture Science and Engineering at Shandong Agricultural University. Greenhouse No. 10 is planted with chrysanthemums at various stages of production, and is divided into several planting areas from east to west. Fig. 9 shows the path plan of the greenhouse.

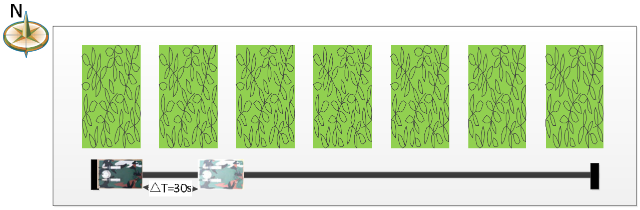


FIGURE 9. The system moves at a constant velocity from east to west along the path of the greenhouse, and the data are collected every 30 seconds.

A. ANALYSIS OF DATA ACQUISITION INTERVAL

In the automatic tracking acquisition mode, the system moves at a constant velocity, and the position of the acquisition point can be calculated based on the acquisition time interval between the two sets of data. Hence, the accuracy of the data acquisition interval directly determines the accuracy of the calculated location of the acquisition point.

Based on the actual layout of the greenhouse, the system is set to acquire data every 30 seconds by adjusting the velocity of movement of the system. Monitoring data acquired over five full days are used to calculate the time interval between the two successive sets of data from each acquisition period.

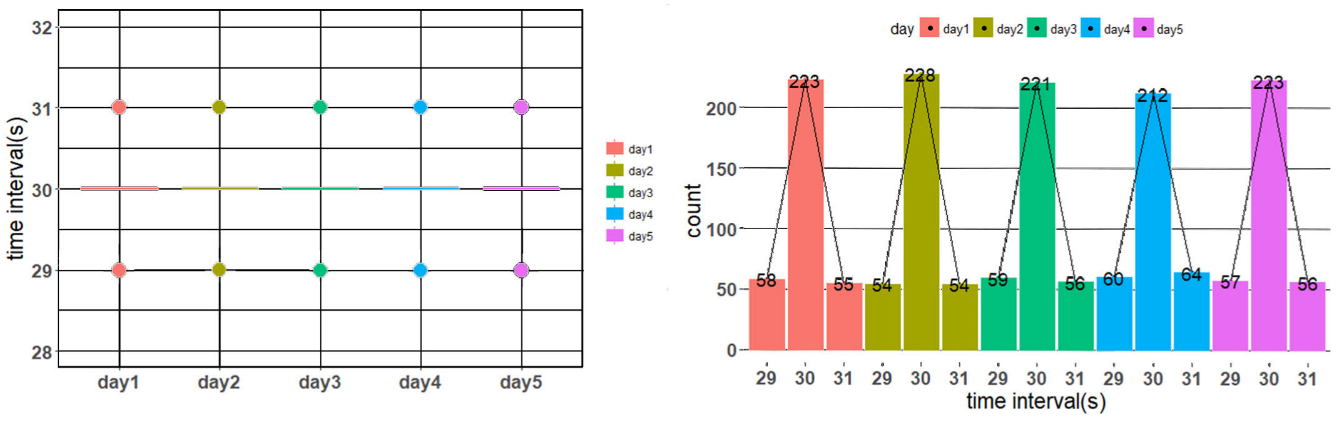
A box diagram and histogram are drawn for the sampling time interval, as shown in Fig. 10. It can be intuitively seen that the time interval for most of the data acquisition is 30 s; the average time interval can be calculated as 29.99 s, thereby indicating that the time interval for data acquisition for the system is accurate.

B. ANALYSIS OF RESULTS

Figs. 11–13 show line charts for the variations in temperature, humidity and illumination in the greenhouse over five consecutive days, and these indicate that the three greenhouse environmental parameters monitored by the system basically conform to the daily variation trends of temperature, humidity and illumination. As shown in Fig. 11, due to poor ventilation, the temperature in the greenhouse sometimes exceeds 40°C. Since it was cloudy on the second day, the temperature did not vary as strongly as on the other four days. In order to avoid extremely strong sunlight, the management personnel applied a sunshade net on three of the days, meaning that the temperature in the greenhouse began to decrease at 10 a.m. The variation in the illumination shown in Fig. 13 was also caused by the same factor. The abnormality of the data on the first day was caused by an incorrectly set measurement range for the sensor, and the data were restored to normal after this error was corrected. The experimental results show that this mobile system can effectively monitor the environmental information of the greenhouse.

C. COMPARISON OF DATA ACQUISITION BEFORE AND AFTER FILTERING

Data acquisition without filtering algorithm was carried out from 8:25 am to 13:00 pm. Data acquisition with filtering



(a)

(b)

FIGURE 10. Sampling plots (a) time interval versus days, and (b) sampling counts versus time intervals.

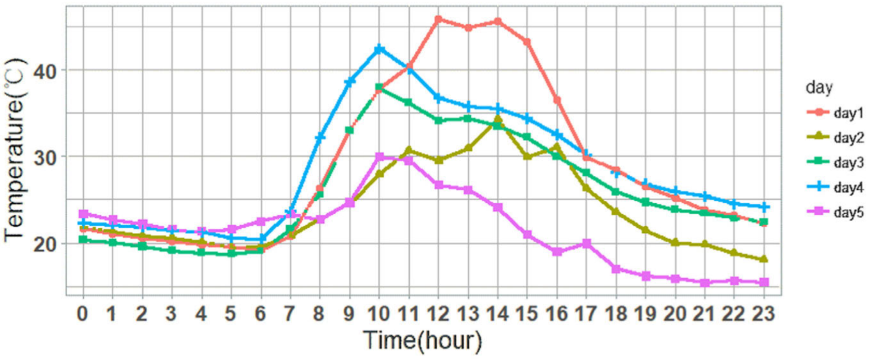


FIGURE 11. Scatter plot of temperature versus time. The peaks are between 10:00 am and 14:00 pm.

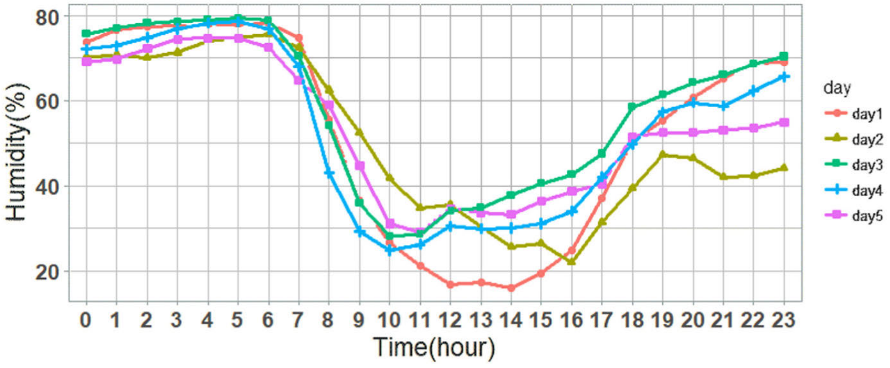


FIGURE 12. Scatter plot of humidity versus time. The minimum are between 10:00 am and 16:00 pm.

algorithm was conducted from 13:50 pm to 21:00 pm. Data were acquired every five minutes.

Fig. 14 shows a comparison of the data filtering curves before and after filtering, based on the acquisition of temperature in the greenhouse. It can be seen that when the limiting filtering algorithm and the weighted average filtering algorithm are integrated, the quality of the acquired data can be effectively improved and meaningless outliers can be reduced. The error rate of the unfiltered data acquisition reaches 3.57%, while that of the filtered data acquisition is zero.

D. COMPARISON OF SERIAL AND GPRS COMMUNICATION

Two sets of Arduino chips are used as the lower computer. Equipment A communicates with the server via the

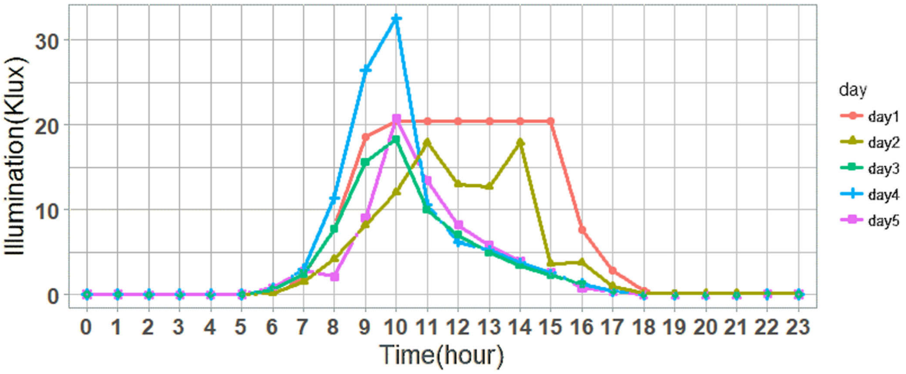
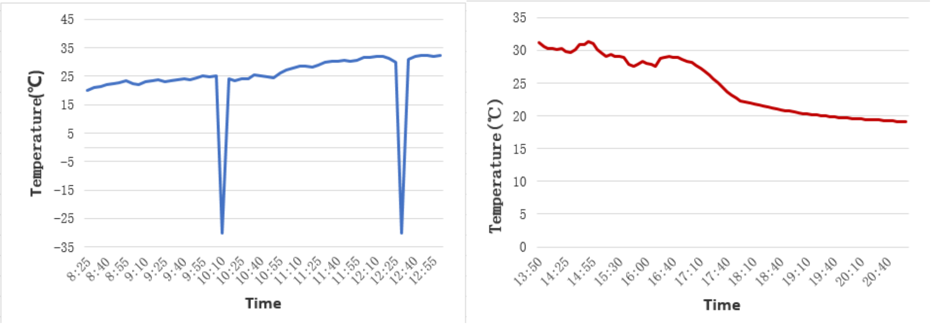


FIGURE 13. Scatter plot of illumination versus time. The peaks are between 10:00 am and 15:00 pm.



(a) (b)

FIGURE 14. Comparison of data filtering curves. (a) Air temperature before filtering, and (b) air temperature after filtering.

TABLE 4. Comparison of data acquisition success rate between two communication modes.

Equipment No.	Transmission mode	Test duration (h)	Times of data acquisition	Pieces of data	Success rate (%)
A	TTL	24	288	287	99.65
B	GPRS	24	288	268	93.06

TTL serial port; equipment B communicates with the server through the H7210 General Packet Radio Service (GPRS) communication module. Both A and B use the same batch of sensors and Analogue/Digital (A/D) conversion modules. Data acquisition is performed every five minutes for 24 hours.

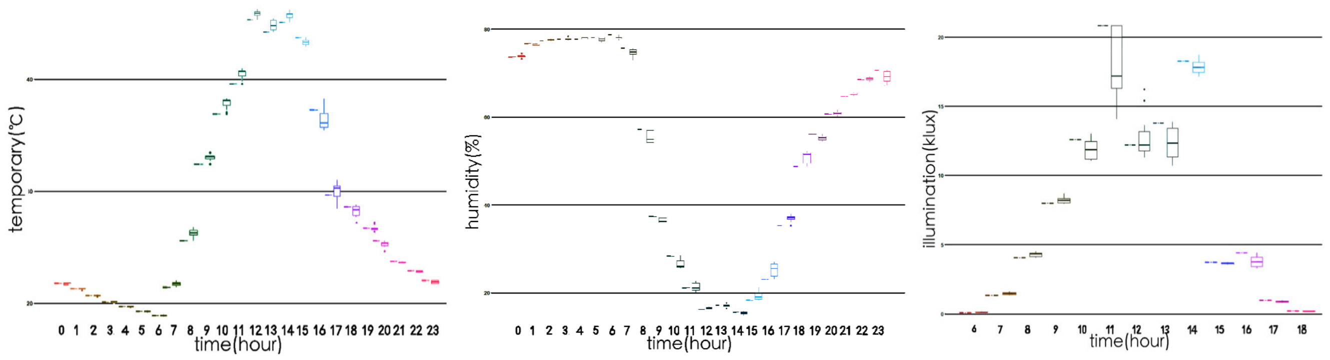
The results are shown in Table 4. With consistent external conditions, the system with TTL serial port transmission has a data acquisition success rate of 99.65%, and the success rate of the system with GPRS wireless transmission is 93.06%. The reliabiliy of TTL serial port communication is much higher than that of GPRS wireless communication. The use of TTL serial port for data transmission in short-range communication can effectively improve the reliability of the system.

E. COMPARISON OF AUTOMATIC TRACKING ACQUISITION MODE AND AUTOMATIC FIXED POINT ACQUISITION

Fig. 15 shows a comparison of the data acquired using different sampling modes in one day. Since the illumination is zero at night, only data acquired between 06:00 and 18:00 are compared.

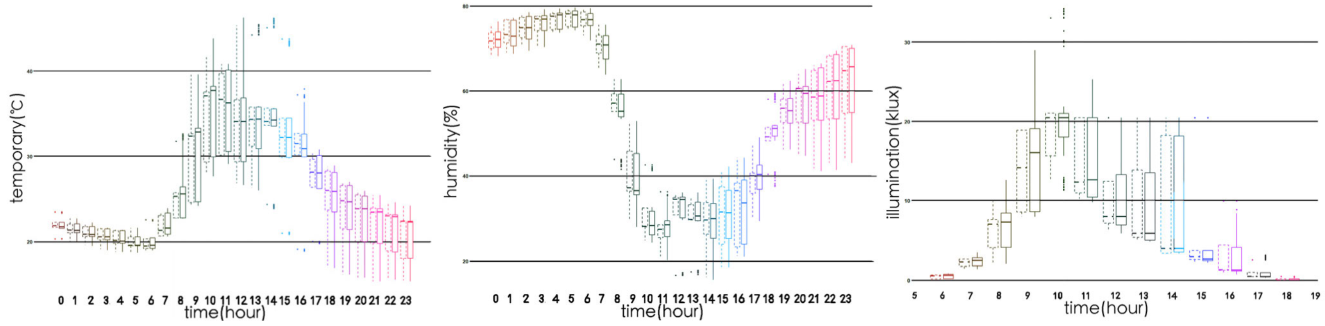
As shown in Fig. 15, each time point corresponds to two box lines, where the left side represents data acquired in the automatic fixed point acquisition mode and the right side represents data acquired in automatic tracking acquisition mode. It can be seen that at the same point in time, there are differences in the temperature, humidity and illumination at different locations in the greenhouse. The data acquired in automatic tracking acquisition mode can better reflect the various environmental parameters in the greenhouse, while the data acquired in automatic fixed point acquisition mode deviate from the data acquired in automatic tracking acquisition mode, and this deviation is not regular. Therefore, the data acquired at a single fixed point cannot accurately reflect information about the various parameters in the greenhouse.

To further compare the overall differences between data acquired in automatic fixed point acquisition mode and auto-



(a) (b) (c)

FIGURE 15. Comparison of the data acquired using different sampling modes in one day. (a) Comparison of temperature, (b) comparison of humidity, and (c) comparison of illumination.



(a) (b) (c)

FIGURE 16. Comparison of the data acquired using different sampling modes over five consecutive days. (a) Comparison of temperature, (b) comparison of humidity, and (c) comparison of illumination.

matic tracking acquisition mode, data acquired in these different modes over five consecutive days are compared (Fig. 16).

In Fig. 16, the dotted box line represents the data acquired in automatic fixed point acquisition mode, and the solid box line represents the data acquired in automatic tracking acquisition mode. On the whole, there is no significant difference between the data acquired in the two modes; however, there are more discrete points in the tracking acquisition mode, meaning that it has an advantage in terms of monitoring the maximum and minimum values at a given point in time.

To sum up, the automatic tracking acquisition mode outperforms the automatic fixed point acquisition mode in the monitoring of the environmental information of the greenhouse.

VI. CONCLUSION

(i) In this study, a mobile greenhouse environment monitoring system based on the IoT architecture is proposed. A Raspberry Pi and an Arduino chip are combined for the first time in agriculture greenhouse environmental monitoring, with the former serving as the data server and the latter as the master chip for the mobile system. A four-layer system architecture is constructed that provids a motion control function. And all four layers of the architecture are deployed on the mobile system, shortening the physical distance between the data acquisition end and the data processing end, to reduce the cost of data collecting and improve the real-time response to surrounding information and the reliability of data transmission. The system can operate in three modes to realize the automatic collection of multipoint greenhouse environmental information and the capture of pictures of crops with low cost.

(ii) The experimental results show that the system can accurately acquire data based on a set time interval, and that the position of the acquisition point can be accurately determined. With the filtering algorithm integrated, the error rate in data acquisition is zero, and with TTL serial communication, the success rate of data acquisition can reach 99.65%. The system can effectively realize the multi-point monitoring of the environmental information of the greenhouse, such as its temperature, humidity and illumination. The system therefore has high accuracy and reliability. Moreover, the use of a Raspberry Pi in the system can improve the quality of data transmission and the accuracy of monitoring of environmental information, indicating that the Raspberry Pi is of high value in this application.

(iii) An experimental comparison shows that the automatic tracking acquisition mode outperforms the automatic fixed point acquisition mode. The automatic tracking acquisition mode can not only reduce the number of equipment and sensors, but can also reduce the requirements for management and maintenance, and is of great assistance in achieving accurate planting.

This study can not only provide important basic data for intelligent monitoring of greenhouse crops and alarms for abnormalities found in the greenhouse environment, but also can offer a reference for the effective monitoring of other agricultural environments. In future work, deep learning technologies will be used to establish a chrysanthemum growing period assessment model in order to achieve intelligent monitoring of crops. Modeling analysis will be conducted on the changes in greenhouse temperature, humidity and illumination, and the predicted and absolute values of maximum deviation will be combined to achieve monitoring and alarms for abnormalities found in the greenhouse environment with a dynamic threshold.

REFERENCES

[1] Y. Ren, X. Chen, J. Jia, W. Gao, and J. Zhu, "Environment monitoring and temperature prediction in greenhouse based on wechat platform," *Nongye Jixie Xuebao.*, vol. 48, no. s1, pp. 302–307, Dec. 2017. doi: 10.6041/j.issn.1000-1298.2017.S0.046.

[2] X. Zheng, "Expert control systems for modern greenhouse climate," *Jidian Gongcheng.*, vol. 20, no. 3, pp. 42–45, 2003. doi: 10.3969/j.issn.1001-4551.2003.03.014.

[3] F. G. Montoya, J. Gómez, A. Cama-Pinto, A. Zapata-Sierra, F. Martínez, J. de la cruz, and F. Manzano-Agugliaro, "A monitoring system for intensive agriculture based on mesh networks and the Android system," *Comput. Electron. Agricult.*, vol. 99, pp. 14–20, Nov. 2013. doi: 10.1016/j.compag.2013.08.028.

[4] R. Pahuja, H. K. Verma, and M. Uddin, "A wireless sensor network for greenhouse climate control," *IEEE Pervasive Comput.*, vol. 12, no. 2, pp. 49–58, Apr. 2013. doi: 10.1109/MPRV.2013.26.

[5] W. S. Lee, V. Alchanatis, C. Yang, M. Hirafuji, D. Moshou, and C. Li, "Sensing technologies for precision specialty crop production," *Comput. Electron. Agricult.*, vol. 74, no. 1, pp. 2–33, Oct. 2010. doi: 10.1016/j.compag.2010.08.005.

[6] N. K. D. N, "ARM based remote monitoring and control system for environmental parameters in greenhouse," in *Proc. IEEE ICECCT*, Coimbatore, India, Mar. 2015, pp. 1–6. doi: 10.1109/icecct.2015.7226117.

[7] Y. He, M. Liang, L. Chen, D. Xu, and S. Du, "Greenhouse environment control system based on IOT," *Zhengzhou Daxue Xuebao Lixueban*, vol. 50, no. 1, pp. 90–94, Mar. 2018. doi: 10.13705/j.issn.1671-6841.2017263.

[8] S. Du, Y. He, M. Liang, L. Chen, J. Li, and D. Xu, "Greenhouse environment network control system," *Nongye Jixie Xuebao*, vol. 48, no. s1, pp. 296–301, Dec. 2017. doi: 10.6041/j.issn.1000-1298.2017.S0.045.

[9] X. Chen, P. Guo, and Y. Lan, "Design and experiment based on wireless sensor network with 780 MHz in greenhouse," *Trans. Chin. Soc. Agricult. Eng.*, vol. 30, no. 1, pp. 113–120, Jan. 2014. doi: 10.3969/j.issn.1002-6819.2014.01.015.

[10] P. Li, M. Chen, J. Wang, J. Zhou, and D. Xia, "Development of monitoring management system and data synchronization for greenhouse IOT," *Nongye Jixie Xuebao.*, vol. 46, no. 8, pp. 224–231, Aug. 2015. doi: 10.6041/j.issn.1000-1298.2015.08.031.

[11] P. Guo and J. Ma, "Design on intelligent environment monitoring system of agricultural greenhouse," *Zhongguo Nongjihua Xuebao*, vol. 37, no. 4, pp. 71–73, Apr. 2016. doi: 10.13733/j.jcam.issn.2095-5553.2016.04.017.

[12] M. Shang, Z. Xu, Z. Yin, and L. Qin, "Design of integrated control system of greenhouse based on Internet of Things," *Shandong Nongye Kexue*, vol. 50, no. 10, pp. 142–146, Oct. 2018. doi: 10.14083/j.issn.1001-4942.2018.10.030.

[13] H. Zheng, "Design and application for greenhouse environment monitoring system based on ZigBee," *J. Changchun Normal Univ.*, vol. 37, no. 8, pp. 42–45, Aug. 2018.

[14] F. Yang, T. Xie, Y. Wu, and W. Su, "Design of monitoring system for greenhouse environment of agricultural Internet of Things based on WIFI," *Jisuanji Celiang Yu Kongzhi*, vol. 25, no. 2, pp. 50–53, Feb. 2017. doi: 10.16526/j.cnki.11-4762/tp.2017.02.013.

[15] M. Lu, Y. Li, R. Li, and X. Cui, "Distributed temperature monitoring system in greenhouse," *Zhongguo Nongjihua Xuebao*, vol. 36, no. 4, pp. 59–64, Jul. 2015. doi: 10.13733/j.jcam.issn.2095-5553.2015.04.016.

[16] R. Kong, "Research of the greenhouse environment monitoring system based on Internet of Things," M.S. thesis, School Mech. Elect. Eng., Univ. Agricult. Hebei, Baoding, China, 2013.

[17] H. Ledford, "Plant biologists welcome their robot overlords," *Nature*, vol. 541, no. 7638, pp. 445–446, Jan. 2017. doi: 10.1038/541445a.

[18] A. Shafiekhani, S. Kadam, F. B. Fritschi, and G. DeSouza, "Vinobot and Vinoculer: Two robotic platforms for high-throughput field phenotyping," *Sensors.*, vol. 17, no. 1, p. 214, Jan. 2017. doi: 10.3390/s17010214.

[19] G. Hu, J. Rong, and X. Yue, "The design of a robot system in agricultural and forestry environmental monitoring based on FPGA," *J. Agric. Mech. Res.*, vol. 35, no. 7, pp. 124–126, Jul. 2013. doi: 10.3969/j.issn.1003-188X.2013.07.030.

[20] N. Hu, "Development of farmland information collection robot," Ph.D. dissertation, Dept. Inf. Sci. Technol., Univ. Nanjing Forestry, Nanjingi, China, 2009.

[21] D. A. V. Slot and C. W. J. Hol, "Optimal manipulator design for a cucumber harvesting robot," *Comput. Electron. Agricult.*, vol. 65, no. 2, pp. 247–257, Mar. 2009. doi: 10.1016/j.compag.2008.11.004.

[22] Z. Zhao, X. Zhu, Y. He, H. He, H. Zheng, X. Wang, and D. An, "Application research of robot in greenhouse," *J. Agric. Mech. Res.*, vol. 37, no. 1, pp. 238–241, Jan. 2015. doi: 10.13427/j.cnki.njyi.2015.01.055.

[23] S. Jia, J. Li, Q. Qiu, and H. Tang, "New corridor edge detection and navigation for greenhouse mobile robots based on laser scanner," *Trans. Chin. Soc. Agricult. Eng.*, vol. 31, no. 13, pp. 39–45, Jul. 2015. doi: 10.11975/j.issn.1002-6819.2015.13.006.

[24] X. Niu, G. Gao, Z. Bao, and H. Zhao, "Path tracking of mobile robots for greenhouse spraying controlled by sliding mode variable structure," *Trans. Chin. Soc. Agricult. Eng.*, vol. 29, no. 2, pp. 9–16, Jan. 2013. doi: 10.3969/j.issn.1002-6819.2013.02.002.

[25] J. Wei, E. Lv, F. Wang, W. Lin, and Q. Ruan, "Greenhouse environmental monitoring robot path tracking algorithm and experiment," *J. Agric. Mech. Res.*, vol. 41, no. 8, pp. 13–20, Aug. 2019. doi: 10.13427/j.cnki.njyi.2019.08.002.

[26] G. M. Acaccia, R. C. Michelini, R. M. Molfino, and R. P. Razzoli, "Mobile robots in greenhouse cultivation: Inspection and treatment of plants," *Acta Horticulture*, vol. 453, pp. 52–60, Mar. 2003.

[27] G. Martinović and J. Simon, "Greenhouse microclimatic environment controlled by a mobile measuring station," *Wageningen J. Life Sci.*, vols. 70–71, pp. 61–70, Dec. 2014. doi: 10.1016/j.njas.2014.05.007.

[28] Y. Liu, "Research on architecture of IOT based on cloud computing," *Sci. Technol. Inf.*, vol. 3, no. 1, pp. 209–210, Jan. 2012. doi: 10.3969/j.issn.1001-9960.2012.01.138.

[29] P. Liu, S. Bi, X. Xue, Z. Deng, G. Zang, and Y. Gao, "Study on intelligent control system for agricultural production based on Internet of Things," *Jisuanji Celiang Yu Kongzhi*, vol. 19, no. 9, pp. 2154–2156, Sep. 2011. doi: 10.16526/j.cnki.11-4762/tp.2011.09.041.

[30] G. Wei, "Research on hierarchical standards on different architecture of IoT," *Comput. Technol. Develop.*, vol. 24, no. 12, pp. 221–229, Dec. 2014.

[31] G. Wang, "China's first national standard for the Internet of Things system has been officially released," *Internet Things Technol.*, vol. 7, no. 3, pp. 3–4, Mar. 2017.

[32] J. Zheng, H. Ruan, W. Feng, and S. Xu, "Agricultural IOT architecture and application model research," *Zhongguo Nong Ye Ke Xue*, vol. 50, no. 4, pp. 657–668, Feb. 2017. doi: 10.3864/j.issn.0578-1752.2017.04.006.

[33] S. Chen, "Design of ultrasonic sensor based on based on intelligent tracking and obstacle avoidance vehicle," *Ke Ji Feng*, vol. 2018, no. 1, p. 10, Jan. 2018. doi: 10.19392/j.cnki.1671-7341.201801010.

[34] S. Dai, "Design and implementation of ultrasonic indoor positioning system," M.S. thesis, Dept. Inf. Sci. Technol., Southwest Jiaotong Univ., Chengdu, China, 2017.

[35] C. Yang, J. Chu, K. Wang, J. Wang, and D. Chao, "Design of intelligent tracking and obstacle avoidance vehicle based on infrared sensor," *Keji Chuanbo*, vol. 9, no. 6, pp. 91–92, Mar. 2017. doi: 10.16607/j.cnki.1674-6708.2017.06.047.

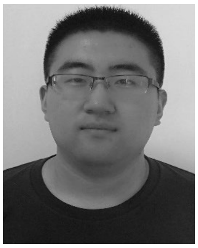
[36] Y. Zhang and R. Song, "Design of multi-route automatic tracking vehicle based on infrared sensor," *Light Ind. Sci. Technol.*, vol. 177, no. 8, pp. 49–52, Aug. 2013.

[37] D. Han, W. Yang, and W. Liu, "Improved clipping and filtering algorithm applied to reduce PAPR," *Beijing Youdian Daxue Xuebao*, vol. 37, no. 4, pp. 44–48, Aug. 2014. doi: 10.13190/j.jbupt.2014.04.010.

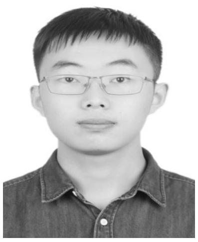
[38] Y. Gao, "Research on methods and algorithms of optimal and self-tuning multisensor measurement fusion filtering," Ph.D. dissertation, Dept. Electron. Eng., Heilongjiang Univ., Harbin, China, 2010.

XIA GENG received the master's degree in computer application technology from the Shandong University of Science and Technology, in 2007. She is currently pursuing the Ph.D. degree with the School of Mechanical and Electrical Engineering, Shandong Agricultural University, Tai'an, Shandong, China, where she is also an Associate Professor with the School of Information Science and Engineering. In the recent years, she has hosted or participated in many projects, including hosted one project of the Institute of Remote Sensing and Digital Earth of the Chinese Academy of Sciences and one open topic of the National Agricultural Information Engineering Technology Research Center. As a Key Personnel, she participated in three 863 sub-projects, two projects of National Surveying and Mapping Geographic Information Bureau, and one major agricultural application technology innovation project in Shandong. She has published over 30 journal or conference articles. Her research topics include the theory and application of Petri net, and the study of precision agriculture and information technology.

QINGLEI ZHANG received the master's degree in agricultural informatization from Shandong Agricultural University, in 2016. He is currently an Assistant Engineer with Shenxian Meteorological Bureau. He has been involved in the agricultural Internet of Things and agricultural meteorology research. He holds more than ten national patents and has published many articles in academic journals.



QINGGONG WEI is currently pursuing the bachelor's degree with Shandong Agricultural University. He was an Assistant with the Agricultural Big Data Research Center, Shandong Agricultural University, to do scientific research. His research interests include deep learning application and the agricultural Internet of Things.



TONG ZHANG is currently pursuing the bachelor's degree with Shandong Agricultural University. He majored in communications engineering and was an Assistant with the Agricultural Big Data Research Center, Shandong Agricultural University, to do scientific research. His research interests include deep learning application and communication engineering.

YU CAI received the bachelor's degree in engineering from Shandong Agricultural University, in 2017, where she is currently pursuing the master's degree in agricultural engineering and information technology. Her research interest includes agricultural big data.

YONG LIANG received the master's degree from Wuhan University and the Ph.D. degree from the Shandong University of Science and Technology. He is currently a Professor and a Doctoral Supervisor with the College of Information Science and Engineering, Shandong Agricultural University. In recent years, he has completed three national 863 projects. He has presided one national science and technology support plan project and two projects of the State Bureau of Surveying and Mapping. He has hosted five scientific research projects in Shandong and has published more than 60 articles in core journals. He received the Third Prize of the Shandong Science and Technology Award and the First Prize of the 6th Shandong Provincial Land and Resources Science and Technology Award. He has been rated as one of the top 100 academic and management leaders of China's informatization by the Chinese information industry. He was hired as a Review Expert by the National Natural Science Foundation, the National 863 Program, and the National Science and Technology Support Program. He was also a Review Expert of the International Cooperation Project, the National Science and Technology Award, and the Shandong Science and Technology Plan and Technology Award.

XIAOYONG SUN received the Ph.D. degree from Lowa State University, America. He is currently a Professor with the School of Information Science and Engineering, Shandong Agricultural University, Tai'an, Shandong, China. He has published 17 SCI articles. His research interests include data analysis, and data mining and bioinformatics.

• • •