Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4518599, 12 pages
https://doi.org/10.1155/2022/4518599

Research Article

A Data-Secured Intelligent IoT System for Agricultural Environment Monitoring

Qing Zhou,[1] Minghua Xiao,[1] Lei Lu,[1] Jun Zeng,[1] Wenting He,[1] Chao Li,[2] and Yulun Shi[3]

[1]*School of Information Engineering, Jiangxi College of Applied Technology, Ganzhou, China*
[2]*Zhijiang College, Zhejiang University of Technology, Hangzhou, China*
[3]*Sunyard System Engineering Co., Ltd., Hangzhou, China*

Correspondence should be addressed to Chao Li; alexlee779@outlook.com

Received 29 January 2022; Accepted 14 June 2022; Published 28 June 2022

Academic Editor: Liran Ma

Collecting environmental information of crop growth and dynamically adjusting agricultural production has been proved an effective way to improve the total agricultural yield. Agricultural IoT technology, which integrates the information sensing equipment, communication network, and information processing systems, can support such an intelligent manner in the agricultural environment. Traditional agricultural IoT could meet the service demand of small-scale agricultural production scenarios to a certain extent. However, the emerging application scenario of the agricultural environment is becoming more and more complicated, and the data nodes of the underlying access to IoT backend system are increasing in large number, while the upper-layer applications are requiring high quality of data service. Hence, the traditional architecture-based (i.e., centralised cloud computing) IoT systems suffer from the problems such as small network coverage, data security issue, and limited power supply time while attempting to provide high-quality services at the edge of the network. Emerging edge computing offers the opportunity to solve these issues. This paper builds an intelligent IoT system for agricultural environment monitoring by integrating edge computing and artificial intelligence. We conducted an experiment to validate the proposed system considering the reliability and usability. The experimental results prove the system's reliability (e.g., data packet loss rate is less than 0.1%). The proposed system achieves the concurrency of 500TPS and the average response time of 300 ms, which meet the practical requirements in agricultural environment monitoring.

1. Introduction

Recently, the Internet of Things (IoT) is being applied to several fields, such as agriculture, logistics, and transportation [1–3]. Using various types of integrated microsmall sensors, wireless sensor networks (WSNs) can achieve real-time detection, acquisition, and sense of various objects [4]. The transmission of various information in physical space (such as temperature and humidity, moisture, and pressure) can make users intuitively understand the information. In China, the establishment of agricultural IT infrastructures is still in its infancy. Therefore, it is impossible to obtain timely information on all farms and different locations of crop growth environment parameters. In order to detect the temperature and suffocation environment, the environmental instruments equipped in the farms are manually operated on-site, which is time-consuming and inefficient. This situation makes the farmers monitor and control the climatic conditions are adversely affected, which in turn affects the improvement of crop yield and quality [5]. Therefore, the method to improve the accuracy and effectiveness of the crop growth environment in the acquisition of various environmental parameters is expected.

On the other hand, the operation of various types of environmental control equipment to achieve intelligent operation and remote control has become a growing concern. The use of the IoT, cloud computing, big data, and other information technology promotes the transformation and upgrading of the entire agricultural industry chain and vigorously drives the development of intelligent agriculture. IoT can effectively

reduce human consumption and accurately capture crop environment and other information and timely carry out actions. But traditional IoT platforms usually adopt a centralised architecture, in which the network bandwidth and processing capacity of the central node could be the bottlenecks for the horizontal expansion of the system [6]. The IoT network edges of large-scale heterogeneous generate massive amounts of heterogeneous data. The long network links for data access operations reduce the performance and efficiency of centralised data storage architectures. The resource constraints of IoT nodes make them dependent on the IoT platform to provide rich services to the external, and the long network links for edge nodes to access IoT services under the centralised IoT architecture make the network latency high, which makes it difficult for edge IoT nodes to get real-time services [7, 8]. By incorporating the computing model of edge computing (EC), an edge processing layer is employed at the near-device end, effectively reducing the workload of network and computation. Edge clouds rely on shorter network links with service invokers, making it possible to provide low-latency network services and low-cost resource access compared to the traditional cloud computing model. However, the edge cloud itself is limited in resources and can easily become a bottleneck for the platform's services on the edge side of the network.

For agricultural IoT, the introduction of EC means that many tasks that used to need to be processed in the cloud can be done locally with artificial intelligence algorithms and data fusion algorithms and can greatly accelerate the response speed of agricultural information and improve monitoring accuracy and more targeted development of agricultural environment management strategies in the monitoring coverage area. This paper combines the Internet of Things and its artificial intelligence technology to build a wide coverage, low power consumption IoT monitoring system suitable for monitoring agricultural environment. It achieves unified management of IoT resources and cooperative computing for environmental monitoring tasks. In this design, the contextual specificity of the environmental monitoring, coupled with the high requirement of real-time data processing, a new agricultural, environmental monitoring IoT architecture model is highlighted. The main contribution in this paper are summarised as follows:

(1) Techniques related to building an edge computing platform for environment monitoring were investigated. The functional architecture and data communication architecture of the edge computing gateway were studied

(2) A collaborative IoT cloud-edge architecture was proposed to realise the unified heterogeneous resource management. It enables the compatibility of the resource identification and mapping mechanism to various kinds of IoT identification standards and realises the unification of the platform resource description methods to reduce the complexity of resource description

(3) The application of LSTM-based environmental indicator prediction algorithm in environmental monitoring was explored. Besides, a visualization dashboard for agricultural environmental monitoring data is built. The data collected by all monitoring nodes are displayed in a dynamic visualization, and the corresponding decision-making support was enabled by the predicting module

The rest of this paper is organised as follows: Chapter 2 introduces the related work, Chapter 3 presents the system architecture and functional details, Chapter 4 presents the experiment and discussion of the proposed system, and Chapter 5 concludes the work.

2. Related Work

2.1. Internet of Things. The Internet of Things (IoT) was first proposed by Prof. Kevin Ashton of the MIT Auto-ID Lab in 1999. IoT was initially designed to solve the problems in supply chain management [9]. The traditional IoT platform architecture usually consists of a data sensing layer, network layer, and business logic layer [10]. Such a centralised architecture has the advantages of easy construction and efficient resource management when the scale of the system is small-size or medium-size. However, with the expansion of the IoT system, service demand is increasing (e.g., response time, intelligent analysis, data storage, and privacy protection). With the popularity of 5G technology, this demand further increases [11, 12], a trusted computer or cluster of computers deployed at the edge of a network with rich service resources to provide computing and data storage services to nearby mobile devices. In 2011, Bonomi et al. proposed the concept of fog computing [13], which introduces a fog computing layer between the device and the cloud. It uses the local fog devices (e.g., routers, IP video cameras, and switches) to process some task requests in close proximity, thereby reducing the number of tasks transmitted to remote cloud computing centres. In 2013, Ryan proposed the early concept of edge computing [14] to address the problem of the rapidly growing number of mobile edge devices. In recent years, distributed IoT architecture based on edge computing has attracted the attention from many researchers [15–17]. Existing studies or edge computing platforms only consider a single edge cloud's vertical application in an IoT scenario, without considering multiple edge clouds in heterogeneous scenarios. Guo et al. built the first virtual fencing system based on a wireless sensor network and implemented a research test for automatic grazing of arable cattle [18]. Vijayakumar and Ramya designed a low-cost and real-time water quality monitoring system by a wireless sensor network, which requires no wiring and has the advantages of flexible deployment and low cost [19]. [20] explored the ZigBee technology in significant field conditions to ensure stable operation of wireless transmission in irrigation area environment. LoRaWAN is designed for long-range communication and networking devices using LoRa (Long Range Radio) technology and can be independently networked from wireless operators [21]. Sendra

et al. develop a monitoring system for large-scale farming, which can measure temperature, relative humidity, wind speed, and carbon dioxide in the farming environment [22]. Valente et al. developed a LoRaWAN-based terminal node equipped with a cluster of asynchronous serial protocol sensors that can measure environmental parameters (including atmospheric pressure, lightning strike count, and soil conductivity) [23].

2.2. Agricultural IoT and Intelligence Computing. Agricultural IoT is the domain application of IoT technology in the whole industry chain of production, management, operation, and service in agriculture [24, 25]. The most common agricultural IoT is used for production environment monitoring [19]. IoT technology is used to collect and obtain information of different elements in the agricultural production environment, including temperature and humidity, light, carbon dioxide, soil water content, and soil fertility. Early IoT applications focused on agricultural information sensing. For example, Wang et al. built a mobile observation system for bovine animals using pulse oximeters, respiratory sensors, body temperature sensors, environmental sensors, and GPS modules [26], which provided a monitoring tool to prevent the spread of diseases in the herd. González et al. developed a method to perform unsupervised behavioural classification by installing GPS sensors and movement collars on cattle to observe and record foraging [27]. Gill et al. propose a cloud-based information system that provides agriculture-as-a-service using cloud and big data technologies [28]. It collects information from different users through preconfigured devices and IoT sensors and processes it in the cloud using big data analytics. Zhu et al. design a dedicated IoT platform in precision agriculture and ecological monitoring [29]. The massive amount of data generated by agricultural, environmental monitoring exhibits complex and dynamic characteristics, and it usually involves multiple sectors, regions, and domains.

With the advancement of artificial intelligence, advanced scientific data processing algorithms are applied in multiple aspects such as air quality prediction and pollution source location. Environmental data are often a series of observations obtained from various physical quantities observed in temporal order, reflecting the characteristics of entity attributes over time, i.e., a multidimensional time series [30]. Time series usually carry a specific law of variation, which is determined by the intrinsic physical properties of the monitored indicators. Time series prediction refers to the process of mining the intrinsic law of change through a large amount of series data and predicting the next point in time based on this law. Popular time series algorithms, including ARIMA [31], etc. [32], used wavelet decomposition and reconstruction to smooth the time series and demonstrated its feasibility for atmospheric pollutant concentration analysis. The change of environmental information involves multiple factors and has nonlinear characteristics, which significantly impacts the accuracy of environmental information prediction. Neural network-based methods have the advantages of self-learning and self-evolving neurons, which are good at dealing with nonlinear models. Artificial neural networks have shown better performance in environmental prediction, and the first one used by related scholars was BP neural network [33], which is with simple structure. Still, it cannot record the features of the previous moment or multiple moments to be used as learning, which leads to the poor prediction and poor generalisation. RNN [34] and many other neural networks have been applied to environmental information prediction, and certain improvements have been achieved. The various environmental monitoring parameters are affected by many factors such as climatic conditions and geographic conditions and do not show linear characteristics.

3. Data-Secured Intelligent Edge Cloud Architecture for IoT

As shown in Figure 1, the workflow is clearly described. This system integrates various kinds of sensors, RFID, video, and other sensing and monitoring devices to collect specific information of farm. This system introduces a cloud-edge collaboration mechanism to process, analyse, and store data to improve the efficiency of network bandwidth utilisation and guarantee the high quality of the platform's external services. The system integrates wireless sensor networks and achieves stable and reliable data transmission through 5G networks. The system fuses and processes the obtained massive agricultural data and realises fully automated monitoring and intelligent analysis of agrarian environment in combination with intelligent terminals. The system's main objectives are as follows: (1) to realise unified management of heterogeneous IoT resources. The proposed architecture uses mapping technology to achieve compatibility with various IoT identity standards and adopts customised identity within the system. A unified resource descriptor is realised by abstracting the behaviour and attributes of resources. (2) To realise a cloud-edge computing services. Data resources are exchanged to each edge cloud through the cloud computing centre. The services of each edge cloud are integrated to provide intelligent decision support for agricultural environment monitoring through intelligent algorithms for hierarchical processing and computing massive data.

3.1. Cloud-Edge Collaborative Architecture. The platform provides comprehensive IoT perceptive ability. The terminal is equipped with ULG series collection and transmission integrated equipment and a set of crop growth related sensors (e.g., soil moisture sensor, soil pH sensor, air temperature and humidity sensor, soil temperature and humidity sensor, light intensity sensor, and carbon dioxide concentration sensor). The platform's architecture consists of an intelligent sensing layer, cloud-edge collaboration layer, heterogeneous network layer, business logic layer, and human-machine interface layer (as shown in Figure 2).

The sensing layer contains sensor monitoring nodes. The monitored data are transmitted to the edge computing gateway through wireless transmission protocols such as LPWAN and 802.11 g. The transport layer adopts heterogeneous networking rules based on LoRaWAN and Wi-Fi and

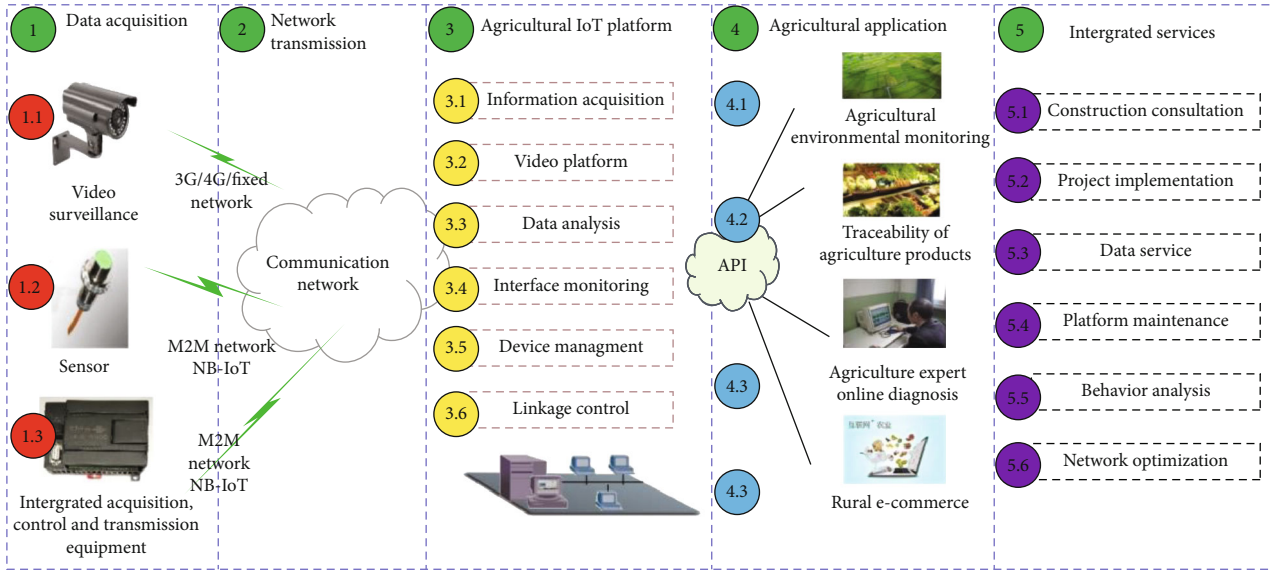

Figure 1: The overview of the agricultural IoT platform workflow.

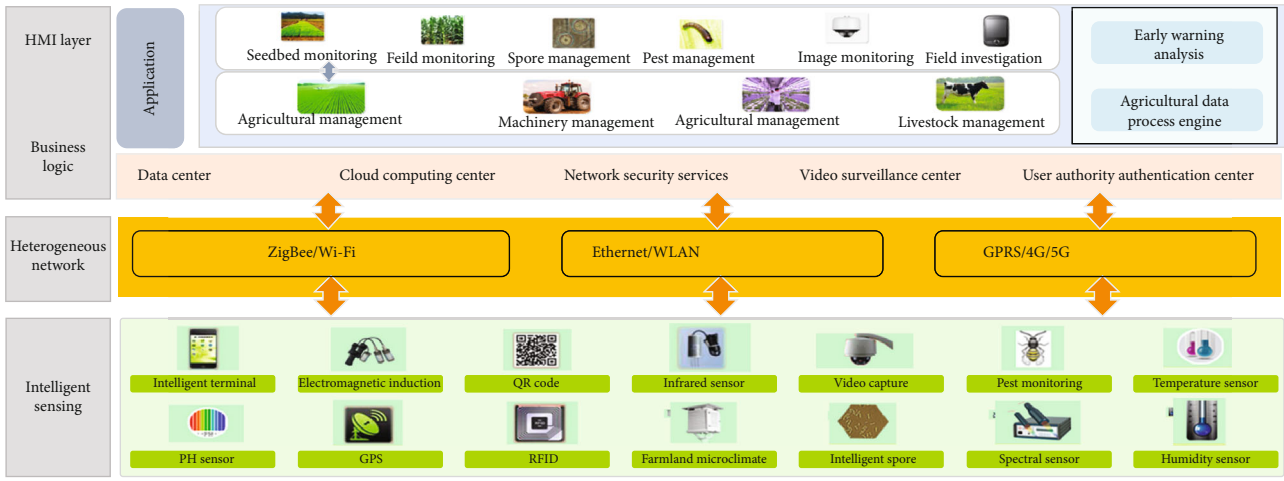

Figure 2: The hierarchical business logic architecture of the proposed system.

applies star topology with low energy consumption, wide coverage, and high bandwidth. The edge computing layer is able to perform online real-time data processing on each measurement parameter of agricultural environment using artificial intelligence and data fusion algorithms at the information collection site. Each functional module is encapsulated through container technology, and information is transmitted between each functional module and between the cloud and the edge through the edge messaging middleware server. Compared with the traditional cloud architecture monitoring system, it can reduce the transmission delay, improve the system response speed, and reduce the pressure on the server side. The application layer implements an agricultural environment monitoring visualization platform to effectively manage and apply the data collected by multiple types of nodes and to make expert decisions and early warnings based on environmental factors. It is designed for the platform resource management and collaboration. Resource sharing and resource control policy can be made among edge cloud systems to achieve controlled sharing of resources and service convergence.

More precisely (as illustrated in Figure 3), the cloud computing centre adapts to access all edge clouds and provides unified services to the external through the conversion of resources. The sensing devices sense the information of the physical world (such as temperature, humidity, and pressure) and transmit the collected data to the corresponding servers through the network. The execution devices execute the received instructions from the upper layer or their control logic to realise the operation of the physical world. The cloud-edge collaboration refers to the access to the corresponding edge cloud system according to the functional requirements and geographic location. The edge system shields the differences of sensing devices to realise the

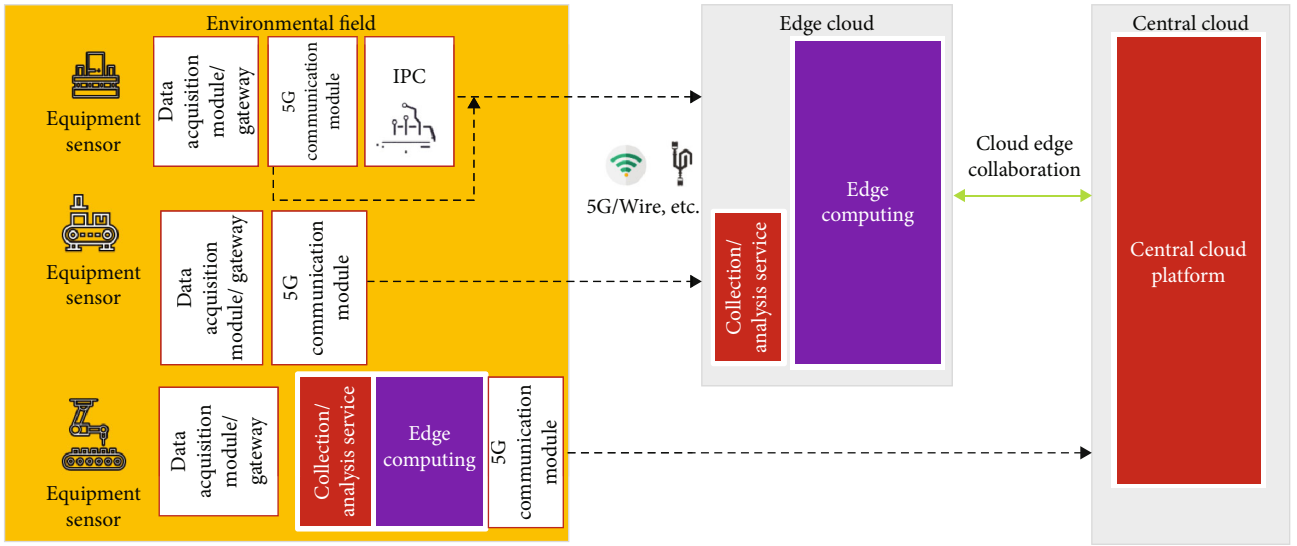

Figure 3: The workflow of cloud-edge collaborative management.

unified management and control of multisource heterogeneous devices. The edge cloud system provides device access and management services to the perception layer and provides a unified resource open interface to the business logic layer. The customised services of the cloud-edge collaboration meet the diversity of the underlying device access, and the edge cloud system deployed at the data source can effectively utilise the network bandwidth to provide high-quality services for the sensing layer devices and IoT applications. The heterogeneous network layer adopts different network communication technologies (such as WLAN, 5G, NB-IoT, and LoRa).

The data stored in the platform are mainly (1) text data collected (such as data collected by temperature and humidity sensors), (2) audio and video data, and (3) system data (such as users, service call permissions, system configuration). It is difficult to store these massive heterogeneous data using a structured database, so we use Redis (https://redis.io) as the persistent data repository. A cloud-based file storage system (https://os.iot.10086.cn) is employed to store IoT audio and video data and uses a content distribution network to distribute the resources according to the regional nature of the data. We employed SQL Server 2015 to store entity data. The structured database supports complex conditional statement queries to meet the platform application requirements. The platform uses read-write separation to operate the database to improve access performance, making data read and write in different instances.

3.2. Cloud-Edge Service and IoT Data Security. The IoT edge cloud collaboration applies to large-scale heterogeneous scenarios, using the cloud computing centre to connect edge cloud systems at the edge of the distributed network and manage and control the edge cloud systems to achieve cross-edge cloud service collaboration. The relationship between the cloud computing centre and the edge cloud systems is shown in Figure 3. The cloud computing centre is a key component of the edge cloud collaborative architecture, which connects each edge cloud system through the cloud computing centre, links the data resources between each edge cloud system, and realises cross-edge cloud collaborative services. The cloud computing centre provides management and control functions for each edge cloud, such as resource access policy, resource identification, instantiation deployment of cloud-edge microservices, security monitoring, and user management. Most of the data is stored in the edge cloud, and some resources frequently requested are stored in the cloud computing centre to reduce the dependency. In addition, the cloud computing centre integrates the platform resources and provides a unified service interface to the developers, making the underlying heterogeneous system transparent to the developers.

The edge cloud system is customised according to the application and device requirements of the scenario where it is located and can be adapted to various network communication protocols and data formats in the underlying layer. It uses a unified external interface in the upper layer to communicate with the business logic layer. The deployment of the edge cloud system requires digital certificates issued by the cloud computing centre as the legal proof of identity and the key for resource sharing. The service interface uses RESTful specification (https://restfulapi.net/), HTTP protocol for synchronous data exchange, and RabbitMQ (https://www.rabbitmq.com/) message queue for asynchronous data exchange, etc. The data generated by the edge cloud system is processed, analysed, and stored locally. The edge cloud system administrator has the highest management control over the data in the edge cloud and can decide which data are open to the public and which data can only be used in the current edge cloud system. The edge cloud system is an integral part of the IoT edge cloud collaborative architecture, which is implemented according to the service requirements of the scenario, mainly consisting of device access middleware, data storage centre, data analysis and

the processing module, and service provision centre and resource sharing and exchange module, and the edge cloud architecture is shown in Figure 3.

The heterogeneity of edge clouds for IoT resources leads to the difficulty of resource sharing and service collaboration among edge clouds. The resource management module was aimed at managing and utilising heterogeneous resources efficiently. IoT identification is the basis of IoT resource management; due to the lack of unified identification system standards, it is challenging to share resources among systems. We propose an IoT identity mapping method to support various identity technology standards, adopt a customised identity method within the system, and decouple resource identification and resource location. It adopts Uniform Resource Name (URN) to identify platform resources and uses Uniform Resource Locators (URL) to search resources. It realises (1) the compatibility of various IoT identification standards through the resource identification mapping, (2) the unique identification of heterogeneous resources in each edge cloud, and (3) the unification of the platform resource description mode. It reduces the difficulty of resource expression format conversion in the service collaboration operation in each edge cloud and supports the sharing of resources in each edge cloud. The architecture of the identity mapping module is shown in Figure 4.

The identity mapping service is deployed in the cloud computing centre to provide an identity mapping information query service. The identity mapping information includes the platform virtual identification number, the resource's physical identification number, and the edge cloud system number where the resource is located. Through this module, the platform virtual identification number can be converted to the physical identification number of the resource and the edge cloud system number where the resource is located. When retrieving the platform resources, only the platform virtual identification number of the resource needs to be passed in to discover the edge cloud system where the resource is located. Its physical identification number achieves the resource search. The identity mapping management module is deployed in the cloud computing centre to provide services for creating, modifying, and deleting identity mapping information. An edge cloud resource, which could be shared externally, must be registered in this module to obtain the platform virtual identification number before it can be accessed externally. When the physical identification number of the resource changes, only the physical identification number of the resource in the identity mapping information needs to be modified, while the platform virtual identification number of the resource does not need to be changed, which avoids the modification of applications developed based on the resource and simplifies the work of platform application developers. The identity mapping cache is deployed in each edge cloud system to cache the resource mapping information to reduce redundant data requests.

3.3. Agricultural-Oriented Service Management. The platform provides the information collection function, with which we can view the real-time status of the plot on the map. The platform transmits the images collected by high-definition cameras to the data centre through the network and enables real-time preview and playback. The interface counter can monitor the total number of interface calls and the success rate of interface calls. The system manager can browse and manage the information of base stations in each block (including base station name, base station level, base station serial number, and base station map markers) and sensor information (such as sensor name, category, health value, display type, and setting the period of sensor upload data). The system can control the equipment on the farm, such as turning on/off the devices, including the fan, the fill light, the shade screen, and the automatic sprinkler irrigation. The system can also support the facilities' performance monitoring, such as the management of M2M cards and SIM card management.

Some of the functional interfaces of the system are shown in Figure 5. Figure 5(a) shows the environmental factor functional interface. In this interface, the left area shows the sensor's name and the current value obtained from the sensor. Below the sensor value is the selection of parameters such as soil moisture, air humidity, and light level. The middle of the interface is shown more visually by displaying the locations where the platform has been used on a map, marked by red dots.

On the right side, information such as device name, device information, status, and switches is displayed, giving an intuitive and convenient display. The video monitoring function is shown in Figure 5(b). The left side of the interface shows the completed campus monitoring. You can select the location or camera you need to view. The area's video monitoring will be displayed in the middle of the page after double-clicking. You can also adjust the direction of the surveillance camera in multiple paths through the operation area below to see the surrounding images without leaving any dead angle. You can also adjust the number of images displayed in the main interface by the number of window segments, up to 16 cameras simultaneously. Click the history video button in the upper left corner to query the history video; after selecting the camera location, enter the query period in the operation area below to query the history video; the function also has the parts of pause, resume, fast playback, slow playback, etc. Growth report real-time query as shown in Figure 5(c) is to query the real-time growth monitoring data returned by each sensor; the menu is divided into three levels, menu level 1 for the agricultural industry, menu level 2 for the company, and menu level 3 for the sensor. Intelligent control engine function as shown in Figure 5(d), the left menu of the interface to select the regional node, after selecting the middle of the interface shows the existing equipment, you can also enter the query conditions above to filter. The primary displayed device information is device name, device information, status, node name, and switch. The switch button can be used to adjust the operation and stop of the device. The system can set the threshold value to achieve automatic control. Suppose the air temperature and humidity are greater than 30 degrees. The fan will be automatically turned on to cool down and automatically turned off when the temperature is less than 25 degrees.

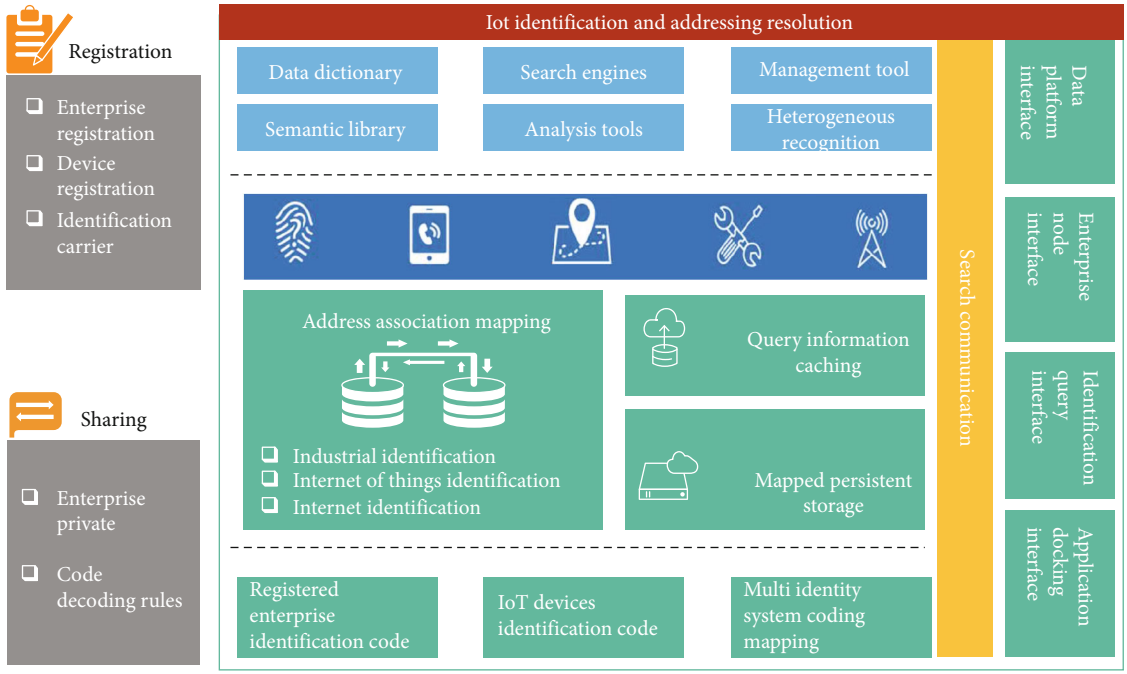

Figure 4: The architecture of IoT identity mapping service.

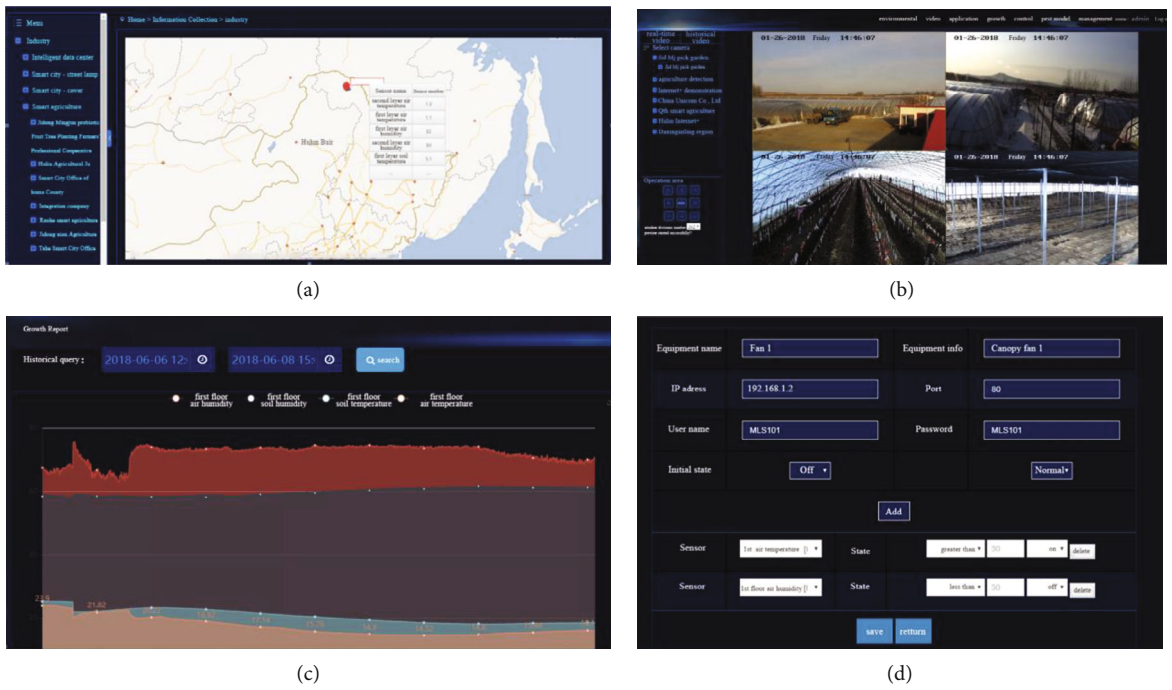

Figure 5: The snapshots of the system functions: (a) Environmental factor function. (b) Video monitoring function. (c) Growth report query. (d) Intelligent control engine function.

3.4. Environmental Predictive Computing Service. RNNs are mainly used to process temporal data, such as speech and text. In temporal data, the output of the current time point is related to the input of the previous time point, and traditional neural networks cannot capture this back-and-forth dependency, while RNNs learn this relationship by adding periodic connections to the neurons in the hidden layer. Figure 6(a) shows the basic structure of an RNN, where x and y denote the input and output vectors, respectively, H denotes the hidden layer, W_1, W_2, and W_3 are the weight matrices, respectively, and h denotes the output vector of the hidden layer. Figure 6(a) shows the circular structure of RNN, and Figure 6(b) shows the expanded structure of Figure 6(a). We find that the output y_t at the current time

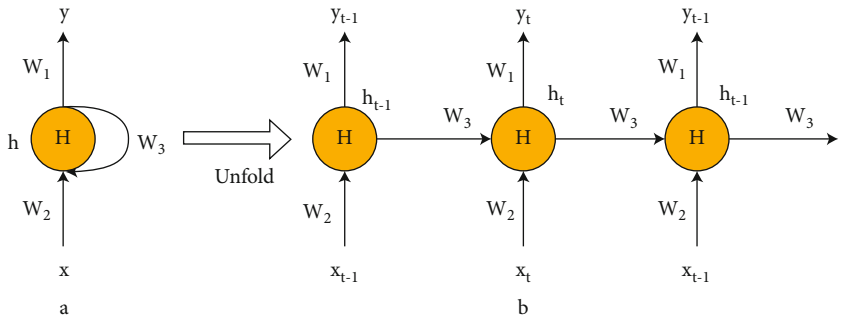

Figure 6: The structure of RNN.

point is jointly determined by the output h_{t-1} of the hidden layer at the previous time point and the current input x_t, and h_{t-1} contains the output information of the hidden layer at all previous time points. The RNN at time point t can be represented by the following equation.

$$\begin{aligned} h_t &= \tanh\left(W_3 h_{t-1} + W_2 x_t\right), \\ y_t &= W_1 h_t. \end{aligned} \tag{1}$$

The expanded RNN structure can be regarded as a feedforward neural network with N intermediate layers, so it can be trained using the backpropagation algorithm. However, in the process of backpropagation, the continuous multiplication of W_3 and h_{t-1} tends to cause gradient disappearance and gradient explosion, which makes it difficult for the RNN to learn the forward and backward information dependence at a long distance.

To solve this problem, long short-term memory networks (LSTM) are proposed. Similar to RNN, LSTM also consists of a set of repeated neural network modules, and such modules are called memory blocks. As shown in Figure 7, each memory block contains three gates, i.e., forgetting gate, input gate, and output gate. In contrast to RNN, which has only one state for recurrent transmission (output of the hidden layer), LSTM has two, namely, the hidden layer state (h_t) and the cellular state (s_t), that runs through the entire memory block. In Figure 2, s_{t-1} and h_{t-1} are the cell state and the hidden layer state at the previous time point, respectively, and x_t and y_t are the input and output at the current time point, respectively. The roles of the three gates are described in detail below.

The forgetting gate determines how much information is discarded from the cell state. The hidden state h_{t-1} at the previous time point and the input x_t at the current time point are fed into the memory block and after the activation function Sigmoid outputs a proportional value from 0 to 1, which represents the proportion of information retained from the cell state. Finally, the proportional value is multiplied by s_{t-1} to achieve the forgetting function. The forgetting gate can be represented as

$$f_t = \text{Sigmoid}\left(W_f \cdot \left[h_{t-1}, x_t\right] + b_f\right), \tag{2}$$

where W_f and b_f denote the weight and bias, respectively.

The input gate determines how much information from the current input is added to the cell state. First, consistent with the forgetting gate, the hidden state h_{t-1} from the previous time point and the input x_t from the current time point are input to the memory block, and after the activation function Sigmoid outputs a scale value from 0 to 1, which represents the proportion of information retained from the current input. At the same time, $h_{t-1}$and x_t are input to the memory block, and $x_t^{'}$ is output after the activation function tanh. Finally, the proportional value is multiplied with $x_t^{'}$ to realise the input function. The input gate can be represented as

$$\begin{aligned} i_t &= \text{Sigmoid}\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right), \\ x_t^{'} &= \tanh\left(W_C \cdot \left[h_{t-1}, x_t\right] + b_C\right), \end{aligned} \tag{3}$$

where W_i and W_C denote weights and b_i and b_C denote biases.

The output gate determines the output information for the current point in time. This is also done first by feeding the memory block with the hidden state h_{t-1} of the previous time point and the input x_t of the current time point, which is then passed through the activation function Sigmoid to obtain a proportion. Then, the output value of the cell state after the activation function tanh is multiplied by the proportion to get the output at the current time point. The output gate can be expressed as follows.

$$\begin{aligned} o_t &= \text{Sigmoid}\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right), \\ h_t &= o_t * \tanh\left(s_t\right), \end{aligned} \tag{4}$$

where W_o and b_o denote the weight and bias, respectively.

4. Experiment

The experiment was designed to verify the overall availability of the proposed system and the accuracy of its environmental prediction function.

The packet loss, throughput, and response time are frequently used metrics to describe the system availability. We used a group of data to test the packet loss (between transmission layers) of the system data transmission and in the environmental information data collection. The terminal

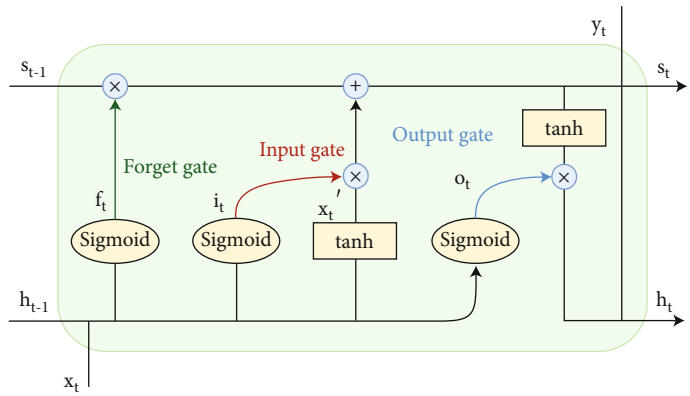

Figure 7: The structure of LSTM.

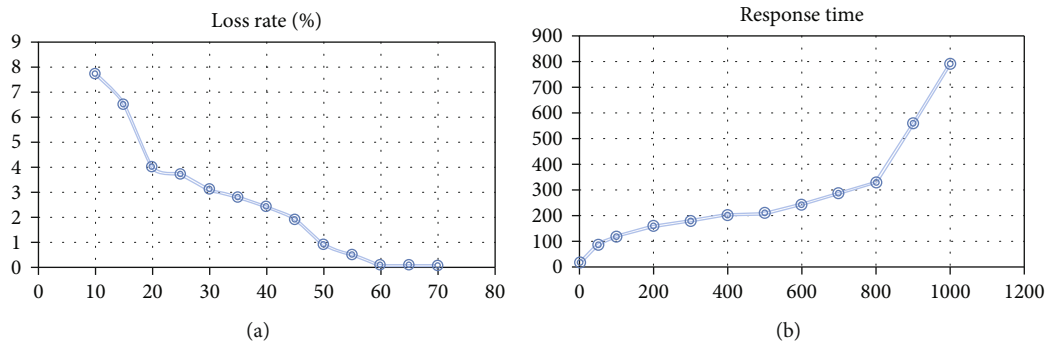

Figure 8: The experimental result regarding the performance metrics.

Table 1: Availability testing result.

Transactions (hits)	Availability	Elapsed time (secs)	Response time (secs)	Transaction rate (trans/sec)	Concurrency	Successful transactions	Failed transactions
10000	100%	6.7	0.05	1492.5	200	10000	0
10000	100%	8.1	0.89	1234.5	300	10000	0
10000	100%	12.1	1.93	826.4	400	10000	0
10000	100%	15.4	2.01	649.3	500	10000	0
10000	100%	24.55	2.54	407.3	600	10000	0
10000	99%	47.06	2.87	210.0	700	9989	11
10000	96%	78.1	3.13	128.0	800	9602	398

Table 2: Samples from the selected dataset.

Date	Maximum temperature (°C)	Minimum temperature (°C)	Average temperature (°C)	Average humidity (%RH)	Yield (kg/mu)
2015-01-01	1.9	-0.4	0.7875	75	907.177044
2015-01-02	6.2	-3.9	1.7625	77.25	747.835779
2015-01-03	7.8	2	4.2375	72.75	740.097015
2015-01-03	8.5	-1.2	3.0375	65.875	760.081199

device collects data in a given period (e.g., an hour) and sends it to the automatic monitoring base station for aggregation at regular intervals (e.g., every 2 hours). The environmental monitoring wireless node collects data once a day and sends them to the monitoring base station for aggregation, and finally, all data are transmitted to the IoT platform. As shown in Figure 8(a), under different sampling frequencies, the computing unit performs 5000 data sends and

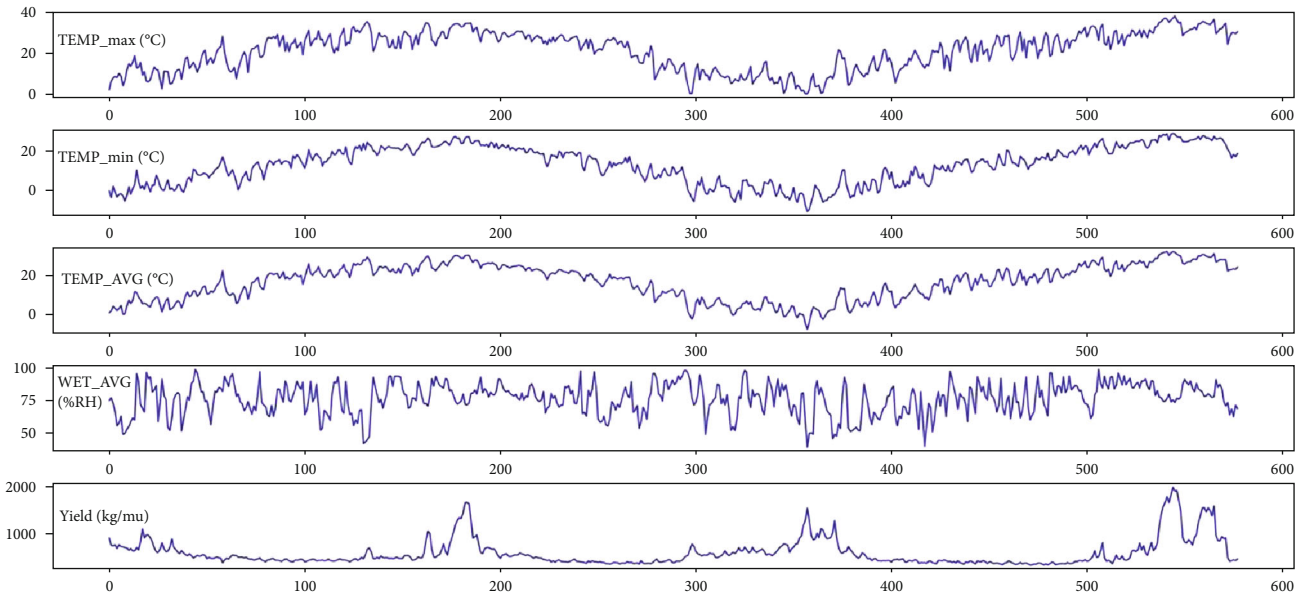

Figure 9: The visualisation of the selected experimental dataset.

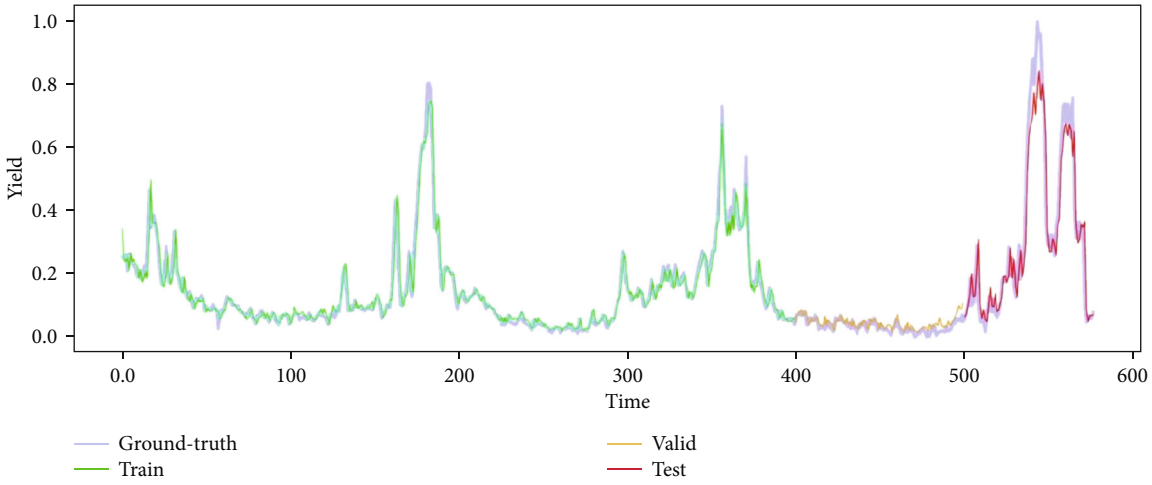

Figure 10: The visualized result of training data, valid data, and test data.

counts the number of packet losses, with the increase of sampling frequency, the overall packet loss rate declines sharply, and when the sampling frequency reaches 60 ms, the packet loss rate drops to 0.1%. The result falls in our expectation, and it indicates the proposed system could fulfil the requirement in practice.

The system is designed for large-scale heterogeneous scenarios, and the platform needs to guarantee service availability during high concurrent scenarios. The throughput and response time of the platform are verified by analysing the log data of the platform under different concurrency. We use the Pulsar tool (https://pulsar.apache.org/) to simulate the service requests to the platform, with different values of concurrent clients. The test result of workload is shown in Table 1, in which, the task queue indicates the percentage of backlogged tasks in the task queue. For each group in the experiment, 10000 transactions were executed with different size of terminals (i.e., concurrency). In the 6^{th} group, 11 transactions failed; that means the concurrency maximum is between 600 and 700. The relationship between the average response time and the number of concurrent clients is shown in Figure 8(b). The average response time increases with the number of concurrent threads. When the number of concurrent threads surpasses the maximum number of threads supported by the edge cloud, there is a significant increase in the average response time. As shown in Figure 8(b), the system throughput of the edge cloud system deployed with a single service reaches the maximum when the concurrency is 300, indicating that the system does not saturate with the number of tasks when the concurrency is less than 300; while the number of tasks saturates when the concurrency is more significant than

Table 3: Part of the result of the crop yield prediction.

Date	Maximum temperature (°C)	Minimum temperature (°C)	Average temperature (°C)	Average humidity (%RH)	Yield (kg/mu)	
					Actual	Predicted
2016-06-15	27.6	22.2	24.4875	72.5	408.97309	434.70498
2016-06-16	30.2	17.8	24.45	69.125	419.502002	427.01839
2016-06-17	32.8	20.1	26.875	61.125	463.757745	492.51727
2016-06-18	33.2	21.7	28.175	64.25	507.862592	553.84575
2016-06-19	32.8	23.9	27.6375	79.625	637.323462	684.95569
2016-06-20	31	21.5	26.425	81.875	492.604463	544.86529
2016-06-21	25.9	24.2	25.275	99	507.48092	541.65983
2016-06-22	32.6	25.1	28.6875	87.875	637.718308	711.57310
2016-06-23	34	26.3	29.625	84.5	809.730649	871.13022
2016-06-24	24.6	22.5	23.45	93.625	488.359556	499.60946

300, but the system throughput does not decrease significantly when it is less than 800. When the concurrency reaches 300, the task queue starts to have tasks piling up, but at this time, the system can still process in time, and the task processing error rate is 0. When the concurrency reaches 800, request processing exceptions start to occur.

We designed an experiment based on the computing service (detailed in Section 3.5) to predict crop yield based on meteorological, environmental factors. The data were collected from February 1, 2015, to August 31, 2016. They contained four environmental factors, i.e., the maximum temperature of the day, minimum temperature of the day, the average temperature of the day and average humidity of the day, and crop yield. The dataset contains 46 data from different data sources, and each data contains 578 datasets. Some of the data in the dataset are shown in Table 2.

The visualization of the dataset is shown in Figure 9. The five images from top to bottom in Figure 9 show the data changes of maximum temperature, minimum temperature, average temperature, average humidity, and yield of the day in chronological order. The horizontal axis represents the time, totalling 578 days, and the vertical axis represents temperature, humidity, and yield. As shown from Figure 9, the temperature varied with the change of the season, the humidity factor had no obvious pattern, and the work reached its highest in the harvest period (in late July).

We first normalized the temperature, humidity, and yield, dividing the dataset into a training set, validation set, and test set, containing 400, 100, and 78 sets of data, respectively, and trained 100 rounds with the mean square error as the loss function. At the end of the training, the training loss reached 0.0016, the validation loss reached 0.0004, and the test loss reached 0.0176. Figure 10 shows the performance of the model. The horizontal axis represents time and the vertical axis represents the yield (after normalisation). The blue curve indicates the actual yield value, the green curve indicates the predicted yield value output from the training set, the orange curve indicates the predicted yield value output from the validation set, and the red curve indicates the predicted yield value output from the test set. As shown in Table 3, the prediction results of LSTM on the training, validation, and test sets are consistent with the true values, but the predicted values are slightly higher than the actual values in the nonharvest period and slightly lower than the actual values in the harvest period.

5. Conclusion

In this paper, an agricultural environment monitoring system is built by integrating edge computing and artificial intelligence. This paper investigates the traditional architecture of agricultural IoT system, proposes a cloud-edge collaboration framework for agricultural environment monitoring, and implements agricultural environment prediction function based on LSTM. The system has been deployed in more than 10 large-scale farms. There are still some shortcomings in the research; e.g., for sensors, improper installation location may lead to inaccurate data acquisition, and instability could result in data collection changes. Moreover, there are some wireless sensors transmission signal distance is limited. The power supply for equipment is not easy-to-obtain: solar power supply may not provide sufficient power, and the adoption of AC power required relocating power wires on the site. Advancement of sensor technology is expected in future. In other hand, for LSTM structure, the training cost of its model is relatively high. In the future, we will introduce several improved versions of LSTM (e.g., coupled LSTM) and other methods on the agricultural environment prediction model proposed in this paper to enhance the training performance of the prediction model and improve the prediction accuracy.

Data Availability

The data that support the findings of this study are available on request from the corresponding author.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] G. Premsankar, M. D. Francesco, and T. Taleb, "Edge computing for the Internet of Things: a case study," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1275–1284, 2018.

[2] S. Gill, I. Chana, and R. Buyya, "IoT based agriculture as a cloud and big data service: the beginning of digital India," *Journal of Organizational and End User Computing*, vol. 29, no. 4, pp. 1–23, 2017.

[3] Z. Cai and Z. He, "Trading private range counting over big IoT data," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 144–153, Dallas, TX, USA, 2019.

[4] X. Zhou, X. Xu, W. Liang et al., "Intelligent small object detection for digital twin in smart manufacturing with industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 2, pp. 1377–1386, 2022.

[5] S. Li, L. Xu, and S. Zhao, "5G Internet of Things: a survey," *Journal of Industrial Information Integration*, vol. 10, pp. 1–9, 2018.

[6] R. K. Singh, M. Aernouts, M. De Meyer, M. Weyn, and R. Berkvens, "Leveraging LoRaWAN technology for precision agriculture in greenhouses," *Sensors*, vol. 20, no. 7, 2020.

[7] W. Qiu, K. Saleem, M. Pham et al., "Robust multipath links for wireless sensor networks in irrigation applications," in *The 3rd Intelligent Sensors, Sensor Networks and Information Processing Conference*, Melbourne, VIC, Australia, 2007.

[8] Z. Cai, Z. Xiong, H. Xu, P. Wang, W. Li, and Y. Pan, "Generative adversarial networks: a survey towards private and secure applications," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–38, 2021.

[9] K. Ashton, "That 'Internet of Things' thing," *RFID Journal*, vol. 22, no. 7, pp. 97–114, 2009.

[10] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): a vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[11] X. Zhou, X. Yang, J. Ma, and K. Wang, "Energy efficient smart routing based on link correlation mining for wireless edge computing in IoT," *IEEE Internet of Things Journal*, vol. 8, 2021.

[12] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, 2009.

[13] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *The proceedings of the first edition of the MCC Workshop on Mobile Cloud Computing*, pp. 13–16, Helsinki Finland, 2012.

[14] L. Ryan, "Edge Computing [EB/OL]," 2013, https://mafiadoc.com/edge-computingpacificnorthwest-national-laboratory_59d648481723dd08e35b7b77.html.

[15] W. Liang, Y. Hu, X. Zhou, Y. Pan, and K. Wang, "Variational few-shot learning for microservice-oriented intrusion detection in distributed industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 8, pp. 5087–5095, 2021.

[16] X. Zhou, X. Xu, W. Liang, Z. Zeng, and Z. Yan, "Deep-learning-enhanced multitarget detection for end-edge-cloud surveillance in smart IoT," *IEEE Internet of Things Journal*, vol. 8, no. 16, pp. 12588–12596, 2021.

[17] X. Zheng and Z. Cai, "Privacy-preserved data sharing towards multiple parties in industrial IoTs," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 968–979, 2020.

[18] Y. Guo, P. Corke, G. Poulton, T. Wark, G. Bishop-Hurley, and D. Swain, "Animal behaviour understanding using wireless sensor networks," in *2006 31st IEEE Conference on Local Computer Networks*, Tampa, FL, USA, 2006.

[19] N. Vijayakumar and R. Ramya, "The real-time monitoring of water quality in IoT environment," *International Journal of Science and Research (IJSR)*, vol. 4, no. 3, 2015.

[20] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 577–590, 2018.

[21] J. Chen, T. Cai, W. He et al., "A blockchain-driven supply chain finance application for auto retail industry," *Entropy*, vol. 22, no. 1, p. 95, 2020.

[22] S. Sendra, L. García, J. Lloret, I. Bosch, and R. Vega-Rodríguez, "LoRaWAN network for fire monitoring in rural environments," *Electronics*, vol. 9, no. 3, p. 531, 2020.

[23] A. Valente, S. Silva, D. Duarte, F. Cabral Pinto, and S. Soares, "Low-cost LoRaWAN node for agro-intelligence IoT," *Electronics*, vol. 9, no. 6, p. 987, 2020.

[24] Z. Cai and X. Zheng, "A private and efficient mechanism for data uploading in smart cyber-physical systems," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 2, pp. 766–775, 2020.

[25] S. K. Grange and D. C. Carslaw, "Using meteorological normalisation to detect interventions in air quality time series," *Science of the Total Environment*, vol. 653, pp. 578–588, 2019.

[26] N. Wang, N. Zhang, and M. Wang, "Wireless sensors in agriculture and food industry–recent development and future perspective," *Computers and Electronics in Agriculture*, vol. 50, no. 1, pp. 1–14, 2006.

[27] L. A. González, G. J. Bishop-Hurley, R. N. Handcock, and C. Crossman, "Behavioral classification of data from collars containing motion sensors in grazing cattle," *Computers and Electronics in Agriculture*, vol. 110, pp. 91–102, 2015.

[28] X. Zhou, W. Liang, S. Shimizu, J. Ma, and Q. Jin, "Siamese neural network based few-shot learning for anomaly detection in industrial cyber-physical systems," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 8, pp. 5790–5798, 2021.

[29] D. Zhu, G. Shen, D. Liu, J. Chen, and Y. Zhang, "FCG-ASpredictor: an approach for the prediction of average speed of road segments with floating car GPS data," *Sensors*, vol. 19, no. 22, p. 4967, 2019.

[30] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. Keogh, "Indexing multidimensional time-series," *The VLDB Journal*, vol. 15, no. 1, pp. 1–20, 2006.

[31] M. Iqbal and A. Naveed, "Forecasting inflation: autoregressive integrated moving average model," *European Scientific Journal*, vol. 12, no. 1, pp. 83–92, 2016.

[32] X. Zhou, W. Liang, W. Li, K. Yan, S. Shimizu, and K. Wang, "Hierarchical adversarial attacks against graph-neural-network-based IoT network intrusion detection system," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9310–9319, 2021.

[33] B. Sadeghi, "A bp-neural network predictor model for plastic injection molding process," *Journal of Materials Processing Technology*, vol. 103, no. 3, pp. 411–416, 2000.

[34] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 1, no. 2, 1998.