Mathematical Biosciences and Engineering

MBE, 20(10): 18792–18819.
DOI: 10.3934/mbe.2023833
Received: 05 August 2023
Revised: 17 September 2023
Accepted: 19 September 2023
Published: 09 October 2023

http://www.aimspress.com/journal/MBE

Research article

Edge computing-based intelligent monitoring system for manhole cover

Liang Yu[1,2], Zhengkuan Zhang[3], Yangbing Lai[1], Yang Zhao[4,5] and Fu Mo[4,*]

[1] College of Computer Science, Guangdong University of Science and Technology, Dongguan 523000, China
[2] AIoT Edge Computing Engineering Technology Research Center of Dongguan City, Guangdong University of Science and Technology, Dongguan 523000, China
[3] R&D Department, Kingsun Optoelectronics Co., Ltd., Dongguan 523565, China
[4] College of Mechanical and Electrical Engineering, Guangdong University of Science and Technology, Dongguan 523000, China
[5] Intelligent Manufacturing and Environmental Monitoring Engineering Technology Research Center of Dongguan City, Guangdong University of Science and Technology, Dongguan 523000, China

* **Correspondence:** Email: mofu@gdust.edu.cn.

Abstract: Unusual states of manhole covers (MCs), such as being tilted, lost or flooded, can present substantial safety hazards and risks to pedestrians and vehicles on the roadway. Most MCs are still being managed through manual regular inspections and have limited information technology integration. This leads to time-consuming and labor-intensive identification with a lower level of accuracy. In this paper, we propose an edge computing-based intelligent monitoring system for manhole covers (EC-MCIMS). Sensors detect the MC and send status and positioning information via LoRa to the edge gateway located on the nearby wisdom pole. The edge gateway utilizes a lightweight machine learning model, trained on the edge impulse (EI) platform, which can predict the state of the MC. If an abnormality is detected, the display and voice device on the wisdom pole will respectively show and broadcast messages to alert pedestrians and vehicles. Simultaneously, the information is uploaded to the cloud platform, enabling remote maintenance personnel to promptly repair and restore it. Tests were performed on the EI platform and in Dongguan townships, demonstrating that the average response time for identifying MCs is 4.81 s. Higher responsiveness and lower power consumption were obtained compared to cloud computing models. Moreover, the system utilizes a lightweight model that better reduces read-only memory (ROM) and random-access memory (RAM), while maintaining an average identification accuracy of 94%.

Keywords: manhole cover (MC); edge computing; lightweight machine learning model; edge impulse platform (EI); LoRa; average response time; pedestrian security

1. Introduction

As part of urban infrastructure, MCs safeguard drainage systems, power supply networks, communication transmission and safe gas supply [1]. Therefore, their management is crucial. However, the movement, loss, damage and flooding of MCs due to various reasons can present significant safety hazards to pedestrians and vehicles on passing roads [2]. Currently, many organizations rely on manual inspections and have limited use of technology to monitor and manage MCs. This approach is time-consuming and labor-intensive and has a long recovery time for detecting abnormalities. Additionally, the intelligent recognition system has low precision, and the overall process is inefficient and lacks real-time capabilities [3]. Once they are observed, MC abnormalities are usually addressed with temporary maintenance methods. For instance, an ice-cream cone-shaped structure is placed around the outer circle of the MC, along with notices and flashing lights at night, to alert pedestrians and car owners. However, this approach poses significant safety risks. Hence, it is crucial to leverage technology and data to empower the monitoring of MCs. An intelligent manhole cover monitoring system should possess the following advantages:

1) Intelligent perception. The manhole cover detection device, as an embedded device that integrates acquisition, control and communication, is capable of automatically sensing the state of the MC in real-time with high precision and accuracy. It can consistently detect whether the MC is moving, damaged, missing or flooded, and it also includes a localization function.

2) Efficient transmission. Low-power wireless WAN communication (LPWAN) technology facilitates fast and efficient transmission of MC status data to edge gateway devices and cloud servers.

3) Real-time response and alarm. The intelligent manhole cover monitoring system can promptly detect various conditions such as missing or displaced MCs. It utilizes localization facilities to trigger alarms in the form of sound, light, image, video or voice, thereby alerting pedestrians and vehicles on the road. Additionally, municipal administrators can remotely monitor the abnormal state of MCs and promptly notify maintenance personnel for repairs and restoration.

4) Lowering management costs. Information technology enables a significant reduction in human and material resources. By utilizing fewer human resources and minimizing system operating costs, it becomes possible to effectively manage a large number of MCs in the city [4].

5) Higher recognition accuracy. This study proposes the use of cloud-based technologies, artificial intelligence and big data to design an intelligent manhole cover monitoring system. By optimizing the system's cost-effectiveness, it aims to achieve higher precision and accuracy in recognizing MCs, thereby reducing misjudgments and labor costs. Ultimately, this system aims to ensure the safe passage of pedestrians and vehicles.

We propose an intelligent manhole cover monitoring system based on edge computing to meet the needs of intelligent manhole cover monitoring and management. The system utilizes an MC terminal monitoring device to collect the dynamic state of the MC. The collected data is then transmitted to the edge gateway of the urban roadside wisdom pole using LoRa communication technology. Through artificial intelligence processing and prediction, the system can detect abnormalities in the MC. If an abnormality is detected, the system provides a real-time response by displaying the abnormal state of the MC and sounding a voice alarm to alert pedestrians and vehicles.

Conversely, if no abnormalities are detected, no action is taken. The edge gateway device uploads the abnormal data to the cloud KitLink platform (an IoT cloud platform developed by Kingsun Optoelectronics Co., Ltd.) via LoRa. The cloud platform then sends the recorded abnormal information to remote mobile municipal administrators and maintainers, enabling timely localization, repair and restoration of the abnormal MC.

In this paper, we propose a cloud-edge-end collaboration solution for MCs. The main contribution of this paper can be summarized as follows:

1) EC-MCIMS. It proposes an edge computing-based MCIMS system. The system includes the design of both hardware and software components, as well as the development of the edge gateway and deployment of the KitLink cloud platform. This system enables efficient real-time management of all MCs in the city with intelligent capabilities.

2) Strong real-time performance and higher accuracy. The system achieves edge computing by deploying tiny machine learning models in the edge gateway device on the wisdom pole. The model proposed in this paper demonstrates better performance compared to previous work. Furthermore, by drawing analogies with the cloud computing model and the cloud-edge cooperative computing mode, it is evident that the cloud-edge cooperative computing mode can effectively enhance the response speed, reduce response time and minimize the power consumption of the MC detecting device. This improvement in efficiency leads to an extended lifespan of the device without compromising prediction accuracy. This enables the system to promptly detect MC abnormalities with high accuracy and minimal response time, ensuring the safety of pedestrians and vehicles on the road.

3) Lower storage space usage. The machine learning model for edge computing utilizes a compact one-dimensional convolutional neural network (1D-CNN) [5]. This model is then quantized and compressed to int8 using the EON compiler of the edge computing EI platform [6]. Experimental results confirm that the compressed and quantized compact model more widely reduces the storage space required for the model flash and the peak memory usage during model runtime, without compromising the model's accuracy. This is particularly crucial for embedded edge gateway devices with limited resources.

The remainder of this paper is organized as follows. Section 2 reviews and analyzes the related work. Section 3 provides an overall description of the proposed EC-MCIMS system. Section 3.1 describes the MCIMS architecture and design of the hardware and software of the system. Section 3.2 particularly presents the principles of tiny machine learning models for edge computing. Sample results of data collected with the proposed system are presented and discussed in Section 4. Section 5 concludes with the future work plan.

2. Related work

As a crucial component of urban municipal engineering, MCs play a vital role in managing urban underground water, power transmission, communication transmission security and natural gas supply security. The management of MCs ensures the safety of people's daily lives. Consequently, countries worldwide have conducted extensive research on the monitoring and management of MCs.

The management of a large number of MCs in the city has evolved from manual timed inspections to a more advanced approach that involves informatization, digitalization and intelligent monitoring. This transformation has made the process more efficient and accurate, transitioning from time-consuming and inaccurate methods to real-time, networked and convenient solutions. The informatization and digital monitoring of MCs primarily rely on the application of Internet of Things

(IoT) technology, sensor technology, embedded technology and various communication technologies. Aly, Hesh et al. [7] conducted a study to assess the effects of automated and non-automated monitoring systems on the MC structure. They also proposed an IoT solution for designing a fully automated monitoring system for the MC. Fu [8] proposed a regionalized MC intelligent safety management system that is both detectable and maintainable. The system utilizes multiple sensors installed in the MC to monitor its condition in real-time. Through the use of an MCU, RF wireless data communication module and upper computer, the system can understand and control the MC. This enables real-time monitoring of urban MCs and automatic alarm triggering. Undoubtedly, this system has the potential to enhance the management ability of MCs and greatly improve the safety of people traveling. Nallamothu et al. [9], Rakesh et al. [10] and Saadnoor et al. [11] proposed the use of sensor technology to acquire the status of MCs. This approach aims to build an IoT system that enables monitoring and alarming of MCs. Ram et al. [12] and Muragesh et al. [13] primarily discussed the application of IoT technology in the design and implementation of the underground drainage monitoring system (UDMS) for MC. These IoT systems are designed to detect the status of MCs using different types of sensors. However, the system designs are relatively simple and do not take into account factors such as accuracy, precision, real-time monitoring and cost. Liu et al. [14] proposed a method for MC identification using RFID tags. The process involves handheld read-write devices to read and identify the tags, which are then uploaded to the cloud using mobile communication technology or low-power wireless wide area network technology for monitoring and locating the MC. However, this approach is considered to be more expensive and less efficient due to the requirement of additional handheld equipment and inspections [15]. Mankotia et al. [16] proposed using an Arduino integrated source microcontroller as the main control device for the MC monitoring and management system to simplify the system design, considering the cost of system monitoring. To ensure stable transmission and reliable operation of the system information, various types of wireless communication technologies are adopted. These technologies include, as used by Nataraja et al. [17], GSM mobile communication technology, Narrowband Internet of Things (NB-IoT) technology adopted by Guo et al. [18], Zhang et al. [19], and LoRa wireless communication technology adopted by Sun [20], Zhang et al. [21], Liu et al. [22] and Li et al. [23]. These communication technologies have reached a higher level of maturity and are capable of effectively monitoring large areas remotely. However, mobile communication technology faces challenges such as expensive tariffs and high power consumption. NB-IoT technology requires the pre-installation of base stations in the region, making it more difficult to implement in underdeveloped areas. Additionally, it requires a special SIM card and has limitations in terms of coverage, cost and real-time performance. On the other hand, LoRa communication technology offers better solutions to the aforementioned problems. However, its system requires the establishment of additional gateways, which necessitates careful consideration of the convenience and reliability of the gateway setup location. With the advancement of artificial intelligence technology, various methods such as image processing, LiDAR and localization techniques have been employed to enhance monitoring accuracy. For instance, Yu et al. [24] and Wei et al. [25] proposed a collaborative approach that combines mobile LiDAR point cloud and ultra-high resolution ground images to enable intelligent monitoring of MCs. Vishnani et al. [26] proposed a method for detecting MCs based on Google location data. Their approach can be enhanced by avoiding MC paths during emergencies and adverse conditions. Andrijašević et al. [27], Krishnan et al. [28], Zhang et al. [29] and Thakur et al. [30] proposed image processing and deep learning methods for intelligent identification, monitoring and prediction of MCs, all of which yielded better results. However, these intelligent methods solely focus on improving the accuracy of MC monitoring and control intelligence, resulting in the need for a large amount of image processing equipment and pre-trained data. Consequently, this leads to a complex system, high costs and challenges in terminal power supply.

Addressing the aforementioned issues, this paper presents a proposal for a manhole cover intelligent monitoring system based on edge computing, which integrates the wisdom pole in the smart city. The system utilizes cloud-edge-end synergy to efficiently manage urban MCs. The edge gateway employs embedded tiny machine learning algorithms to preprocess and analyze the data transmitted by terminal sensor nodes. Real-time processing of localized data and a small amount of anomalous MC data are uploaded to the cloud [31]. This enables real-time abnormal alarm maintenance of the MC, effectively ensuring road traffic safety.

3. Proposed EC-MCIMS architecture

3.1. MCIMS Framework

The EC-MCIMS is a typical cloud-side-end collaboration architecture, which consists of three main layers, as shown in Figure 1. The actual structure of the system is schematically shown in Figure 2.

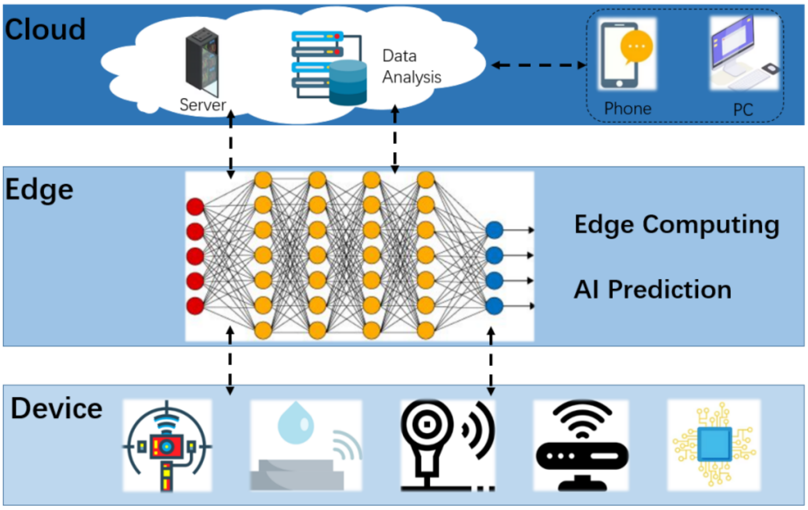


Figure 1. EC-MCIMS cloud-edge-end collaboration architecture.

The terminal node is an MC detection device that is enclosed in a disc-shaped shell. This shell can be easily installed under the MC, and its shape and working location can be observed in the field test described in Subsection 4.1, as shown in Figure 22(a). The shell is made of non-electromagnetically shielded plastic material, which is waterproof and dustproof, ensuring safe and stable detection of the MC status. The node consists of several components, including a sensor group, microcontroller, power module and LoRa communication module [32]. The sensor group primarily detects movement, being missing, waterlogging and localization information of the MC using a three-axis acceleration sensor, waterlogging sensor and positioning sensor. The microcontroller utilizes the low-power and high-performance STM32 series embedded controller to process real-time data collected by the sensor group [33]. It then sends this data to the edge gateway of the wisdom pole through the LoRa communication module. The circuit schematic design of the main modules of the manhole cover detection device is shown in Figures 3 and 4.

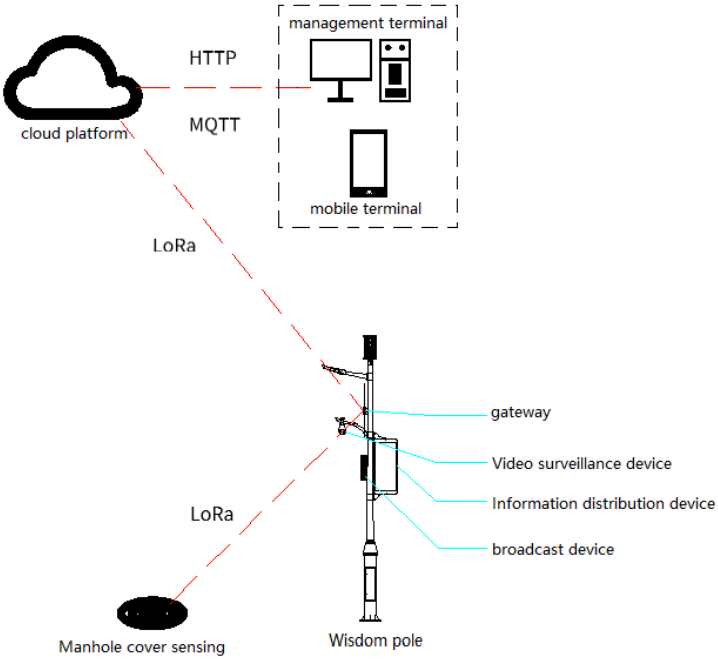


Figure 2. Schematic diagram of the actual structure of the MCIMS system.

After receiving the uploaded MC status and localization data from the terminal node, the edge gateway performs inference and prediction using the tiny machine learning model deployed inside the ESP32 [34]. This allows it to determine the current status of the MC and take immediate localized actions. The actions include displaying a reminder on the wisdom pole and broadcasting a voice message five times to alert pedestrians and vehicles passing by. This real-time approach ensures the safety of the public, especially at night. Additionally, the ESP32 module calculates and stores the effective data at the edge for backup purposes. It also provides a basis for future big data analysis. The circuit design of one of the edge gateway ESP32 is shown in Figure 5.

The edge gateway will upload the predicted conclusion situation to the cloud KitLink platform, and the abnormal status and localization information of the manhole cover will be presented on the cloud platform, as shown in Figure 6. This allows management personnel to understand any abnormal status of the MC and track it for decision-making. Additionally, it will push information about MCs that require timely maintenance to the management and maintenance personnel of both mobile and PC terminals through HTTP and MQTT protocols. This facilitates prompt maintenance and repair of MCs, thereby enhancing the social service capability of municipal departments and ensuring the safety of people's daily travels.

According to the working requirements and working mode of the system, the overall workflow diagram of EC-MCIMS is designed as shown in Figure 7. Figure 7(a)–(c) presents the workflow diagrams for MC status detection, edge gateway edge computing and KitLink cloud, respectively. EC-MCIMS collects and transmits information locally through the terminal nodes of the MC status detection device. It then uses edge computing at the edge gateway to process the data and provide real-time local alarms for pedestrians and vehicles. The system also uploads the MC abnormality status information to the KitLink cloud for storage and visualization, enabling data mining and regulatory decision-making in the future. Additionally, the information is sent to users of mobile and PC terminals, allowing relevant personnel to be notified promptly for maintenance and repair based on the abnormal status of MCs. This helps to minimize potential risks caused by cover abnormalities.

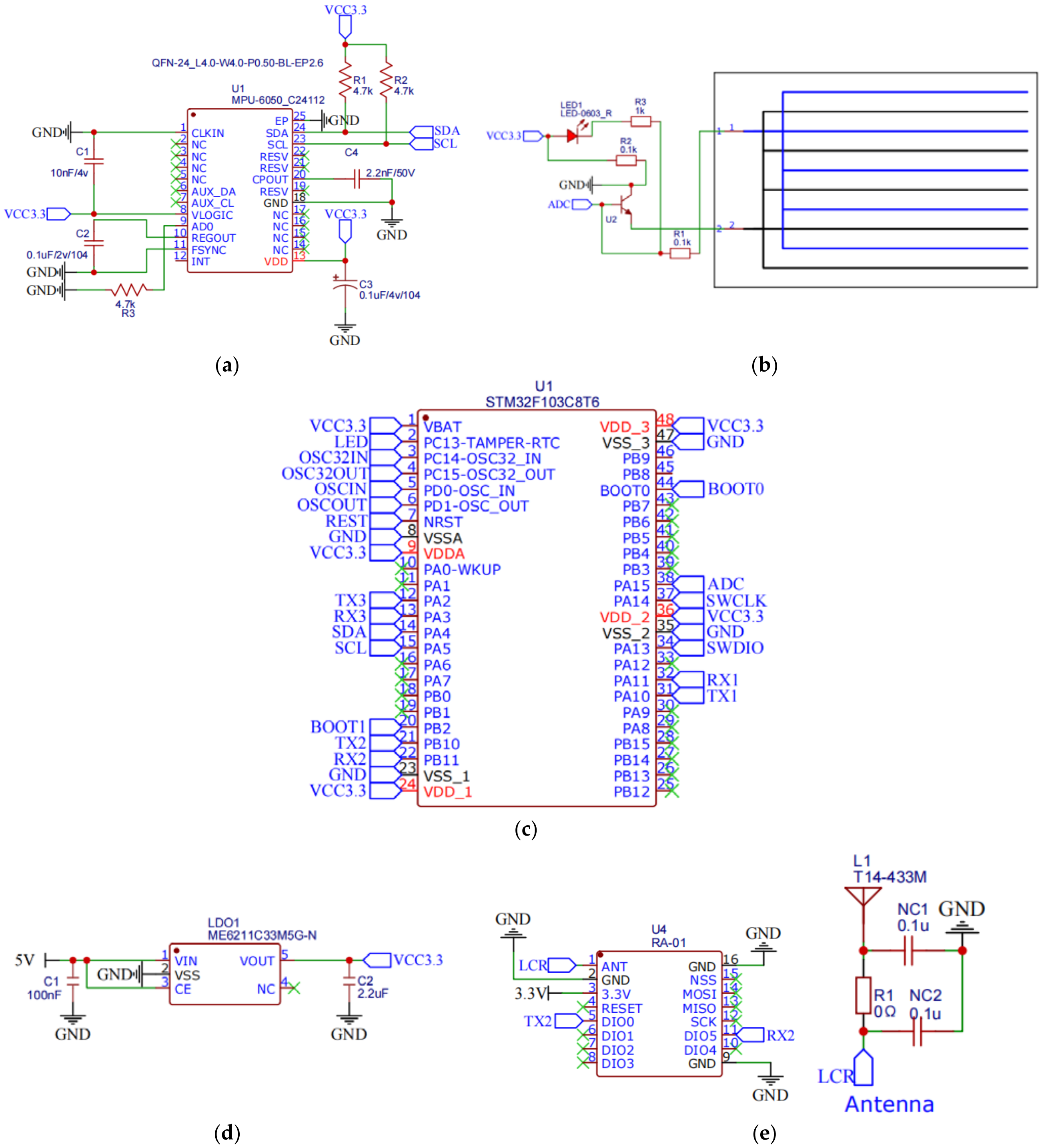


Figure 3. (a) Triaxial acceleration sensor circuit design; (b) Water sensor circuit design; (c) Microcontroller circuit design for manhole cover detection device; (d) Power supply circuit design; (e) LoRa communication circuit design.

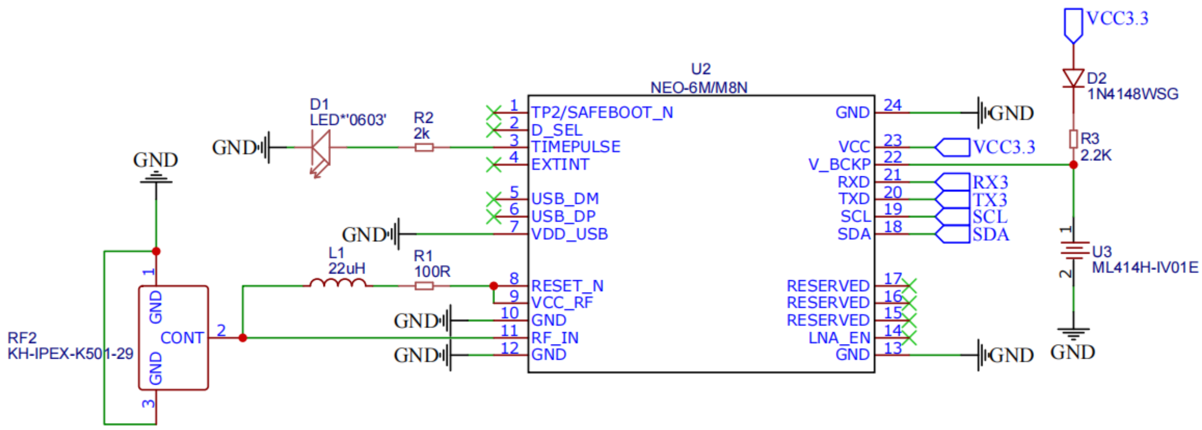


Figure 4. GPS positioning circuit design.

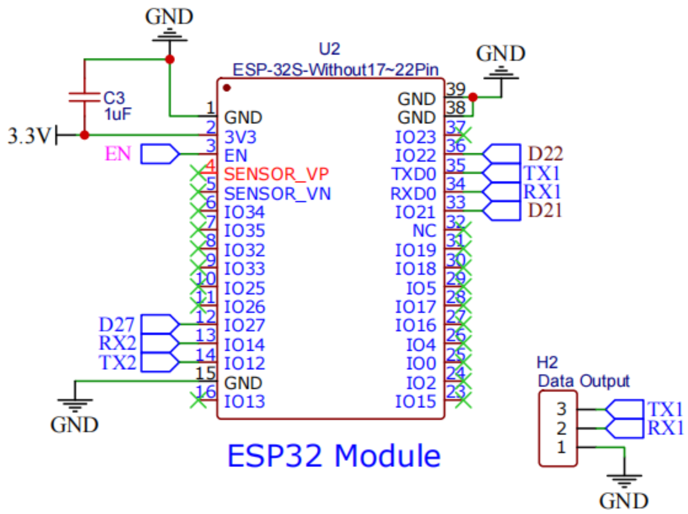


Figure 5. Circuit schematic design of edge gateway ESP32 for manhole cover.

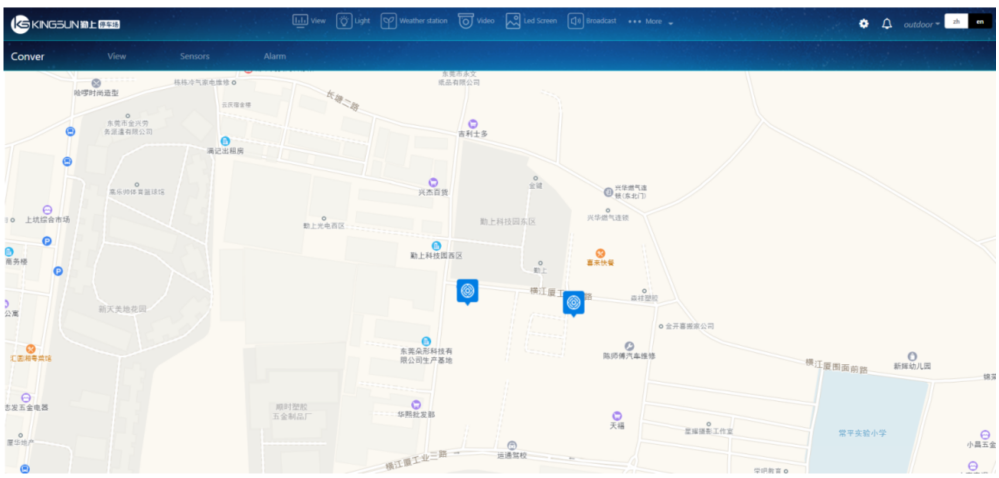


Figure 6. Monitoring and positioning status of manhole cover on KitLink platform.

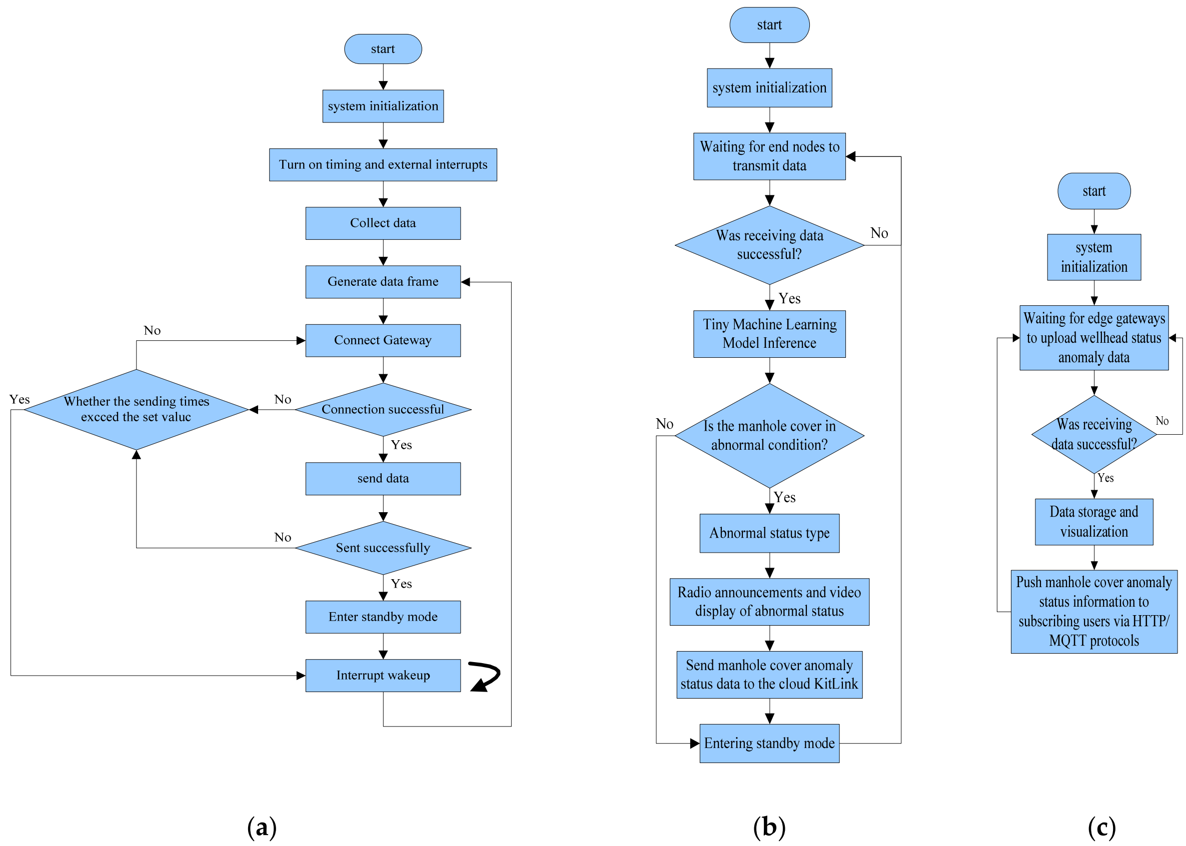


Figure 7. (a) Workflow diagram of manhole cover terminal node state detection; (b) Edge gateway based on edge computing workflow diagram; (c) KitLink cloud workflow diagram.

3.2. Edge computing-based monitoring system

Smart cities face the challenge of monitoring a vast number of devices, each requiring a real-time understanding of their status and efficient response to data requests. These devices, including access, exit and update devices, come from different manufacturers and require a robust system platform and network. The extensive data in smart cities creates significant strain on network bandwidth, resulting in issues such as delay and coordination [35,36]. Moreover, network interruptions pose a substantial risk to the stability of the system's operation. Additionally, traditional cloud computing methods are insufficient to handle the diverse scenarios presented by the vast amount of data in smart cities. MC monitoring is a crucial aspect of smart city development and is closely linked to the safety of citizens and transportation infrastructure. Consequently, it requires high levels of real-time monitoring. This paper focuses on the implementation of smart city construction using wisdom poles. These poles are used to monitor MCs in real time and address security issues through an edge computing model. The model enables efficient collaboration and linkage within a local area, especially when the equipment and facilities are disconnected from the network. This allows for quick response to business needs at the scene, reducing data interaction with the cloud and avoiding excessive bandwidth and time costs. The ultimate goal is to provide timely alerts to pedestrians and vehicles, ensuring road traffic safety and operational efficiency [37].

The edge computing gateway device performs four main tasks: 1) collecting and managing edge data, 2) preprocessing the collected edge data, which includes data cleaning, data coding and decoding and semantic integration and is achieved using a tiny machine learning model, 3) monitoring the collection process of the edge data and managing message queues for edge data transmission and 4) monitoring and storing the preprocessing process of the edge data.

3.2.1. Edge computing workflow

Edge computing involves deploying computing, storage and other network resources closer to the terminal device [35]. This approach facilitates a seamless integration of mobile applications, content and networks, minimizing intermediate transmission processes. As a result, edge computing enables faster and more real-time data processing, reducing network latency and pressure while enhancing real-time response and reliability.

When the system detects that the MC is opened or moved, the edge gateway will automatically trigger a series of actions. These actions include controlling nearby display equipment and voice equipment, as well as releasing warning messages through public broadcasting and information release screens. The purpose of these actions is to serve as a reminder or deterrent. The cloud platform also receives the data and can send real-time warning messages to managers through phone calls, SMS, email or WeChat, allowing them to intervene in the management process. The operation process of the specific edge computing gateway device is shown in Figure 8.

The storage component of the edge computing layer device stores configuration information and management policy information. This allows the device to continue executing its saved operation and management policy even when disconnected from the network of the cloud computing layer device. This helps avoid interruptions in the operation of the entire system due to network abnormalities. Additionally, when the edge device is overloaded or lacks computing power, it can proactively send commands to request collaboration with the cloud. This collaboration enables continuous and stable monitoring of well covers as shown in Figure 9.

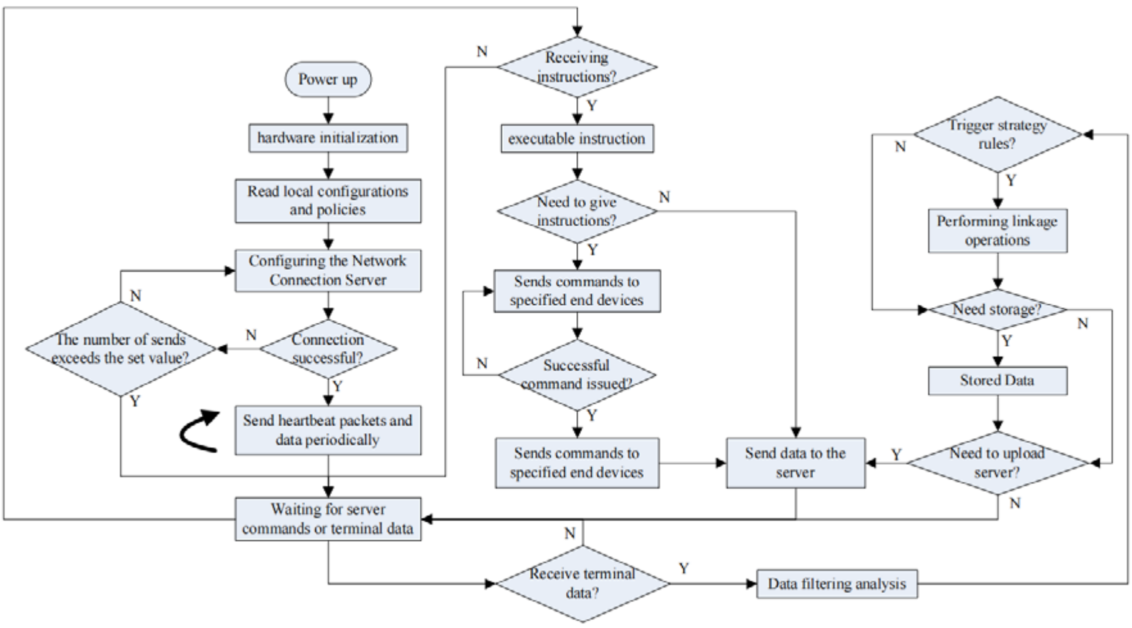


Figure 8. Flowchart of the edge computing gateway device operation method.

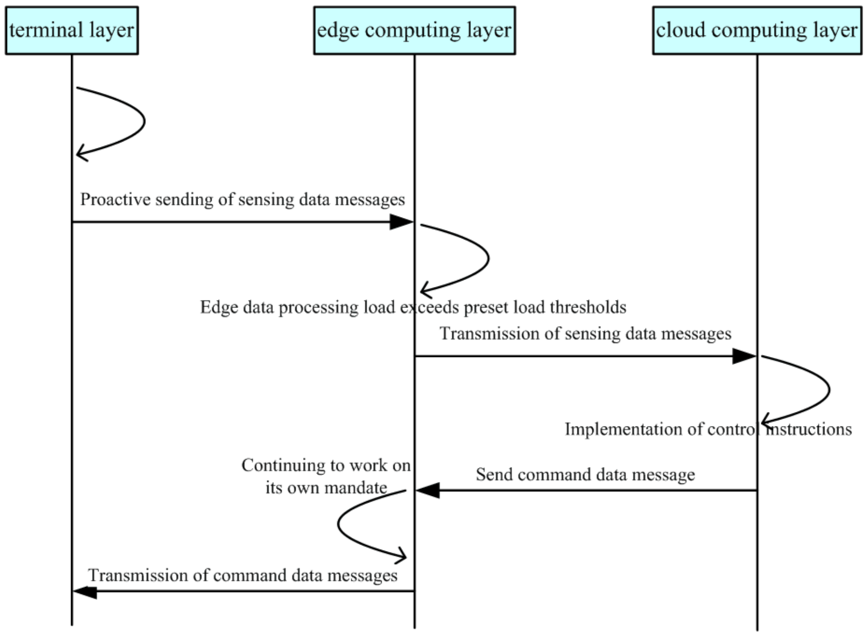


Figure 9. This is a schematic diagram illustrating the method of requesting computing power from the cloud computing layer when the edge computing gateway device is overloaded.

3.2.2. Tiny machine learning model principle

The tiny machine learning models in this paper are trained using the EI edge platform, which is developed based on TensorFlow Lite [38]. The EI platform offers powerful automation, low code and custom extensions, enabling developers to create and optimize embedded machine learning applications using various types of sensor, audio or visual data. Additionally, it provides a Python SDK that allows developers to optimize and convert their models into C++ libraries for any edge device [39]. The development process for tiny machine learning models on this platform consists of five main steps, as illustrated in Figure 10: preprocessing, learning, evaluation, conversion and prediction.

1) Preprocessing. It involves labeling the large amount of raw data collected, dividing the dataset into training and test sets and performing dimensionality reduction. These steps are necessary to prepare for the extraction of data features.

2) Learning. We utilize the EI platform for deep learning to construct a convolutional neural network (CNN) model. The optimization hyper-parameters are iteratively tuned using a validation set to generate an improved machine learning model [40].

3) Evaluation. The performance of the trained optimal learning model on the test set is evaluated and examined using the test dataset. This evaluation helps guide further optimization of the model to reach its optimal state.

4) Conversion. It is necessary to optimize the machine learning model for hardware support and efficient operation. Although the high-performance ESP32 device is used in the gateway device at the edge end, it is still insufficient compared to running the model on a PC. This leads to a decrease in edge computation efficiency and real-time performance. To address this, model quantization and compression are typically performed on the trained model of the EI platform. One method is to reduce the precision of data expression, such as converting float32 to int8, which better reduces the hardware

requirements for the model. This allows the model to be easily exported and deployed on the embedded edge computing gateway ESP32.

5) Prediction. The edge computing gateway device enables predictive processing of actual MC detection data using a deep learning model deployed within it. This allows for localized actions to improve real-time efficiency in event processing.

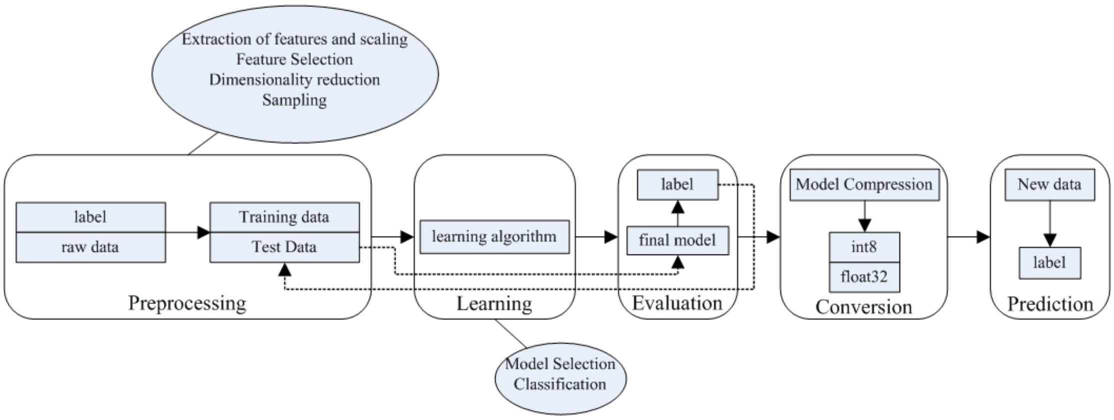


Figure 10. Tiny machine learning model workflow.

The construction of the neural network model structure on the edge computing EI platform is illustrated in Figure 11. It primarily comprises two phases: the feature extraction phase conducted by the convolutional layer, and the classification and identification phase performed by the fully connected layer. During the feature extraction phase, the input data consists of a one-dimensional time series of sensor readings. The sensor set includes three-dimensional acceleration data from triaxial acceleration sensors, converted current data from flooding sensors and latitude and longitude data from the localization chip. The sensor set has a sampling frequency of 30 Hz, resulting in 30 sampling points per second and a data matrix of size 6 × 30. By applying two one-dimensional convolution and pooling operations using the 1D-CNN method [41,42], multiple one-dimensional feature maps can be generated. During the classification and recognition phase, the input is first spread into a one-dimensional array of 42 features using the flattened layer. Then, the classification and recognition outputs are obtained sequentially through the implicit layer with 32 and 64 neurons, respectively. These outputs are used to determine the six MC states in the output layer. The parameter settings of the 1D-CNN model are shown in Table 1.

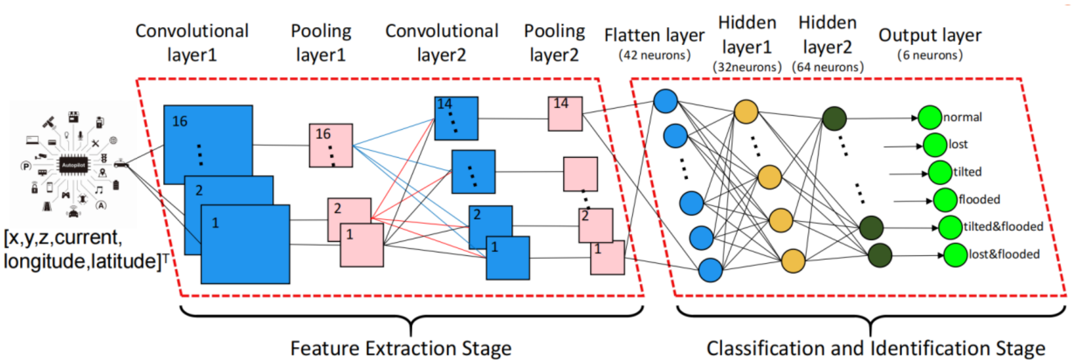


Figure 11. Structure of 1D-CNN model based on the edge computing EI platform.

Table 1. 1D-CNN model parameters.

Layer Name	Layer Parameter	Layer Output Size
Input Layer	inputs are 6-dimensional features with 30 sampling points	(6, 30)
Convolutional Layer1	16 convolutional kernels with kernel size of 11	(20, 16)
Pooling Layer1	maxpooling is used, with a kernel of 2	(10, 16)
Dropout Layer1	node loss probability is 0.5	(10, 16)
Convolutional Layer2	14 convolutional kernels with kernel size of 5	(6, 14)
Pooling Layer2	maxpooling is used, with a kernel of 2	(3, 14)
Dropout Layer2	node loss probability is 0.5	(3, 14)
Flatten Layer	multi-dimensional feature maps flattened to one dimension	(42)
Dense Layer1	the neurons are 32 and the activation function is relu	-
Dense Layer2	the neurons are 64 and the activation function is relu	-
Output Layer	the neurons are 6 and the activation function is softmax	(6)

As the data collected from the sensor is in the form of time series, this study utilizes a one-dimensional convolutional neural network model to extract features. The model selects different convolutional kernels to extract various features from the input signal. The output of the l convolutional layer can be represented by Eq (1).

$$x_j^l = f\left(\sum_{i=1}^{N} k_i^{l-1} * w_{ij}^l + b_j^i\right) \tag{1}$$

The variables used in the context are as follows: j represents the number of convolution kernels, k denotes the convolution kernel, N indicates the number of channels of input x^{l+1}, w represents the weight of the $(l-1)$ layer, b represents the $(l-1)$ layer bias, $f(\bullet)$ represents the activation function (in this case, the ReLU function is used with the expression as in Eq (2)), and $*$ denotes the convolution operator.

$$f(x) = \max(0, x) \tag{2}$$

After the multiple feature maps are extracted by the convolutional layer, the pooling layer is used to achieve dimensionality reduction. This helps in reducing the complexity of the model computation, improving computational speed and reducing overfitting. Common pooling operations include maximum pooling and mean pooling. In this paper, the maximum pooling operation (expressed as Eq (3)) is selected, in which H represents the convolution kernel width.

$$x_j^l = \max_{(j-1)H \le t \le jH} \left(x_t^{l-1} \right) \tag{3}$$

After completing the feature extraction stage, the information from multiple feature maps is spread out into a one-dimensional array of data. This array is then passed to the fully connected layer for classification and recognition output. The classification output is represented by Eq (4).

$$y_i^l = f\left(w^{l-1} \cdot x^{l-1} + b^{l-1} \right) \tag{4}$$

In this paper, we consider a six-classification problem and use the $soft\max$ activation function, for the output layer. The activation function is expressed as Eq (5), where w represents the weight and b represents the bias.

$$soft\max(y)_i = \frac{\exp(y_i)}{\sum_{j=1}^{n} \exp(y_j)} \tag{5}$$

The error between the output of the forward propagation output layer and the expected result is continuously updated through backpropagation using the stochastic gradient descent method (SGD). This method updates the weights and bias of each neuron until the error value is less than the set threshold or the maximum number of iterations is reached. The training of the model is considered complete when the error value meets the defined criteria. The error is calculated based on Eq (6), in which y_i^l represents the actual output of the model and y_a^* represents the target output of the vector a. The process of updating the weights and bias through backpropagation has been extensively studied by various scholars [43–45], and the details will not be reiterated here.

$$E_a = \sum_{i=1}^{N_l} \left(y_i^l - y_i^* \right)^2 \tag{6}$$

In addition, the monitoring data for MCs is inherently unbalanced. This paper addresses a six-classification problem, taking into account the data imbalance. It is not sufficient to only consider the recognition precision of the data. Evaluation of the machine learning model also requires assessing the recall and $F_{1\text{-}score}$ metrics, such as the confusion matrix for model evaluation. The recognition precision $P\text{recision}$, recall $R\text{ecall}$ and the average of precision and recall $F_{1\text{-}score}$ metrics are defined in Eqs (7)–(9) respectively.

$$P\text{recision} = \frac{TP}{TP + FP} \tag{7}$$

$$R\text{ecall} = \frac{TP}{TP + FN} \tag{8}$$

$$F_{1-score} = 2 * \frac{P\text{recision} \bullet R\text{ecall}}{P\text{recision} + R\text{ecall}} \tag{9}$$

where TP, FP and FN in Eqs (7) and (8) denote true positives, false positives and false negatives, respectively.

4. Results and discussion

4.1. Data processing and analysis

By selecting 240 raw data samples from over 3 years of MC status monitoring data, the sensor data was located in two places: Changping Town, Dongguan City, Guangdong Province, Kingsun Optoelectronics Company Limited - South Gate, and Hengli Town, Dongguan City, Guangdong Province, Kingsun Optoelectronics Company Limited. Each place had 120 data points. If the flood sensor detects water-logging, it triggers circuit conduction and generates current. The sample was then randomly divided into training data and test data, with a ratio of 0.79:0.21. The specific details of the raw sample situation are shown in Figure 12. The distribution of the original data on the edge computing EI platform is shown in Figure 13.

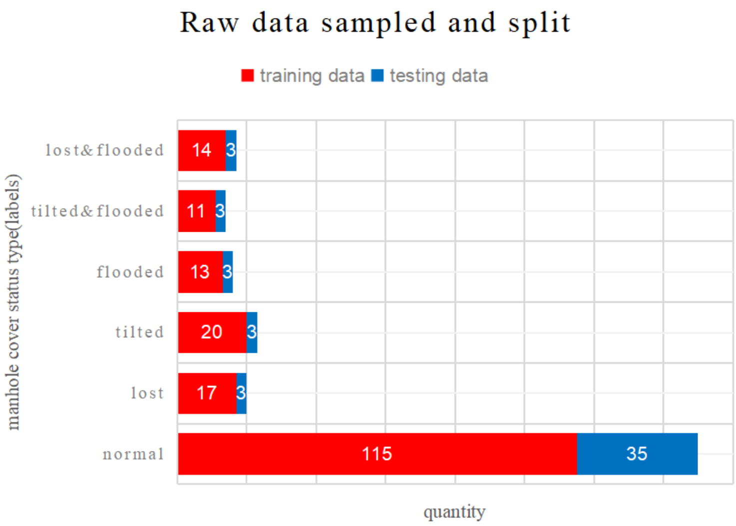


Figure 12. Raw data sampled and divided.

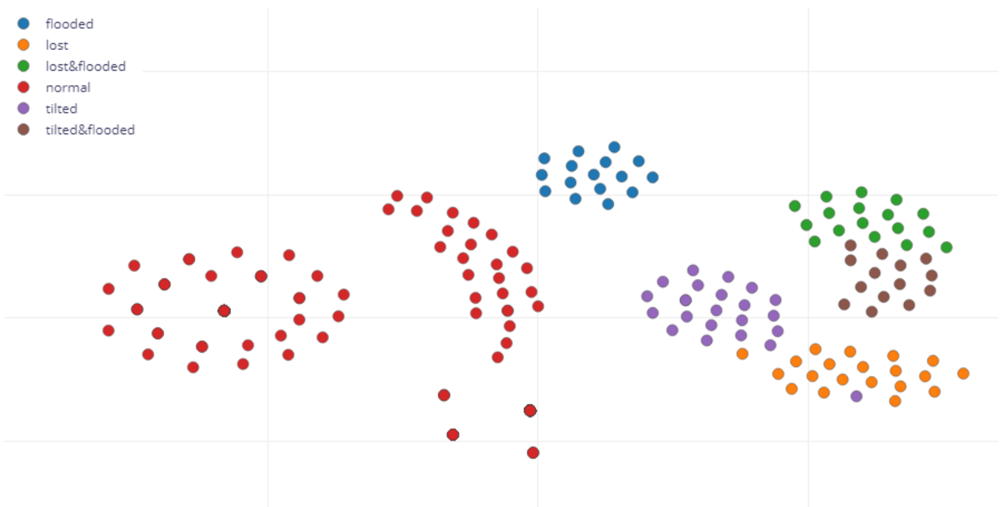
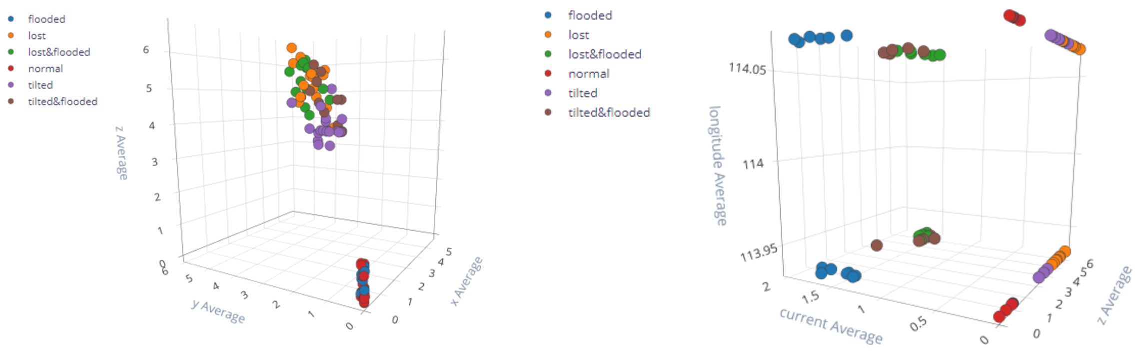


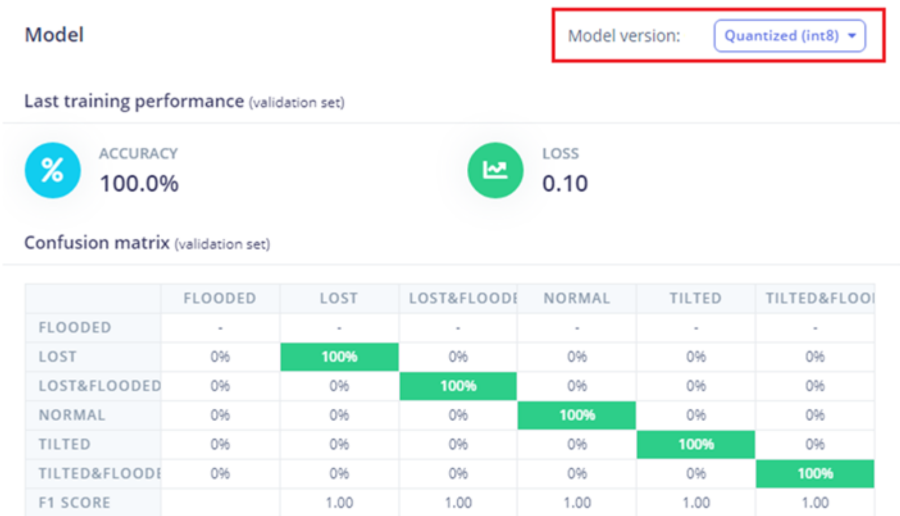
Figure 13. Raw data distribution of edge computing EI platforms.

To evaluate the feature extraction ability of the 1D-CNN model on the original time-domain signal, we employed principal component analysis (PCA) to downscale and visualize the features learned by the model. The results are presented in Figure 14, which shows (a) and (b) as the principal components of the three-axis acceleration sensor data and the features of the flooding sensor and localization sensor data after dimensionality reduction, respectively. The data belonging to the same state of the MC can be effectively aggregated, while the data representing different states exhibit better separability. This confirms that the 1D-CNN model can adaptively extract valuable features from the original time-domain signals, enabling accurate monitoring of the MC state.



(**a**) (**b**)

Figure 14. (a) Principal component analysis of learning features for triaxial acceleration sensors; (b) principal component analysis of learning features for flood sensors and localization sensors.



	FLOODED	LOST	LOST&FLOODE	NORMAL	TILTED	TILTED&FLOO
FLOODED	-	-	-	-	-	-
LOST	0%	100%	0%	0%	0%	0%
LOST&FLOODED	0%	0%	100%	0%	0%	0%
NORMAL	0%	0%	0%	100%	0%	0%
TILTED	0%	0%	0%	0%	100%	0%
TILTED&FLOODE	0%	0%	0%	0%	0%	100%
F1 SCORE		1.00	1.00	1.00	1.00	1.00

Figure 15. Performance on the validation set after the last training of the model.

The 1D-CNN network model is configured with an epoch of 300, a batch size of 32, a learning rate of 0.0005, a validation set size share of 20%, a random loss probability of neurons in the dropout layer of 0.5 and a model quantization compression of int8 type. This compression can be used to compare the model training results with the original uncompressed quantization of float32 type. Please refer to Table 2 in Subsection 4.2 for more details. All the model parameters are set and then trained. The model's performance is verified after each epoch, and the hyperparameters are continuously optimized using the verification set. The best training results are shown in Figure 15, where it can be observed that the model achieves 100% accuracy on the verification set after the last training with only 0.10 error. The confusion matrix shows that the value of $F_{1\text{-}score}$ is 1, indicating that the model has a strong training effect and good generalization ability. Furthermore, the training model demonstrates high recognition accuracy on all training sets, as depicted in Figure 16. By calculating the recognition situation in the figure, we can determine that the accuracy rate is approximately 96.71% = (round(240*0.79)*(1-20%)-5)/(round(240*0.79)*(1-20%)).

Figure 16. Classification of test results for training data.

Model testing results

ACCURACY
94.00%

	FLOODED	LOST	LOST&FLOO	NORMAL	TILTED	TILTED&FLO	UNCERTAIN
FLOODED	100%	0%	0%	0%	0%	0%	0%
LOST	0%	100%	0%	0%	0%	0%	0%
LOST&FLOODED	0%	0%	66.7%	0%	0%	0%	33.3%
NORMAL	0%	0%	0%	100%	0%	0%	0%
TILTED	0%	0%	0%	0%	100%	0%	0%
TILTED&FLOODE	0%	0%	0%	0%	0%	33.3%	66.7%
F1 SCORE	1.00	1.00	0.80	1.00	1.00	0.50	

Figure 17. Confusion matrix for test data test results.

The trained model was tested on the test set and achieved an accuracy rate of 94%, demonstrating good generalization, as shown in Figure 17. The confusion matrix analysis of the test results revealed that the model accurately detected single states of the MC, such as normal, tilted, lost and flooded, with 100% accuracy. There was only a small probability of confusion between the tilted and flooded states, as well as between the lost and flooded states. Importantly, the model never identified an abnormal state of the manhole cover as normal in all the data, indicating its high recognition accuracy. This finding, as shown in Figure 18, has minimal impact on promptly understanding the state of the MC for timely rescue and repair. Moreover, the occurrence of these two confusion situations is relatively small. Therefore, it is feasible to apply the trained model to the ESP32 edge gateway of the MC.

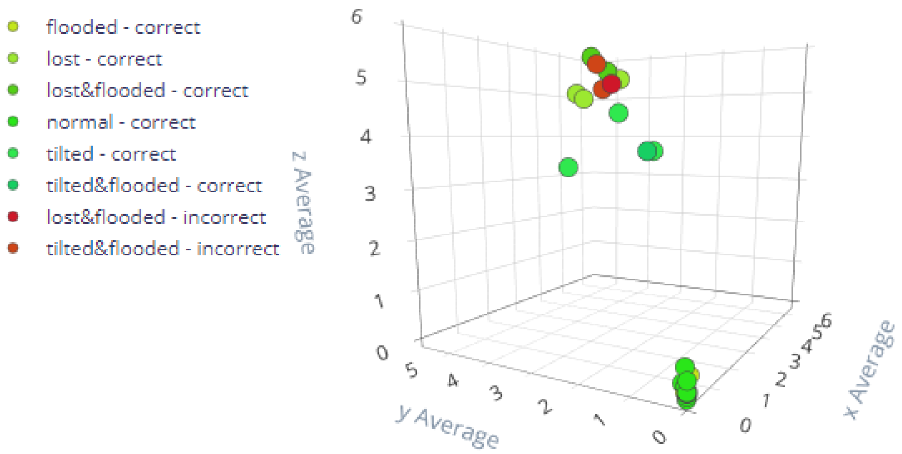


Figure 18. Test data test results recognition.

To ensure the validity and accuracy of the model, we implemented the same model using the edge computing platform and also programmed it in Anaconda3. The training and testing were conducted using the same raw data and divisions. The results are shown in Figure 19, which compares the verification accuracy with the training accuracy curve. Additionally, the verification error versus training error curve is depicted in Figure 20. Finally, Figure 21 presents the model testing results from the confusion matrix, which are consistent with those obtained from the model constructed by the edge computing platform.

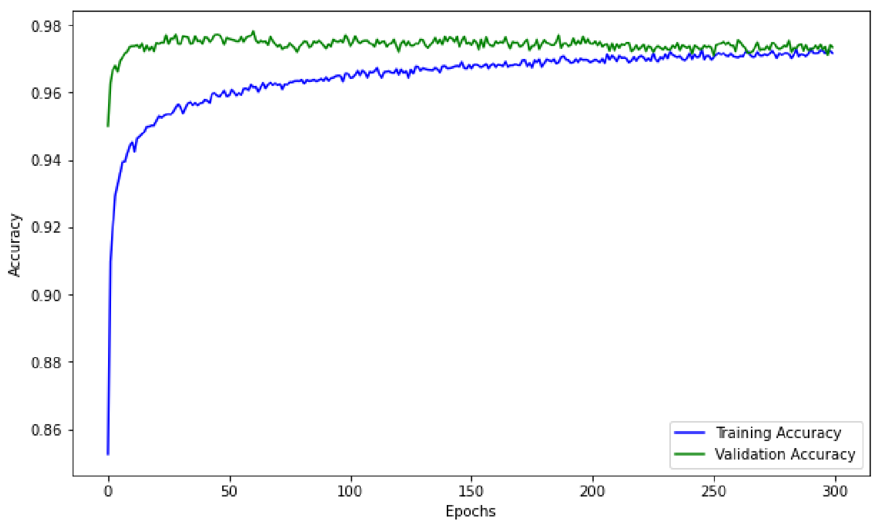


Figure 19. Model training accuracy vs. test accuracy network learning curve.

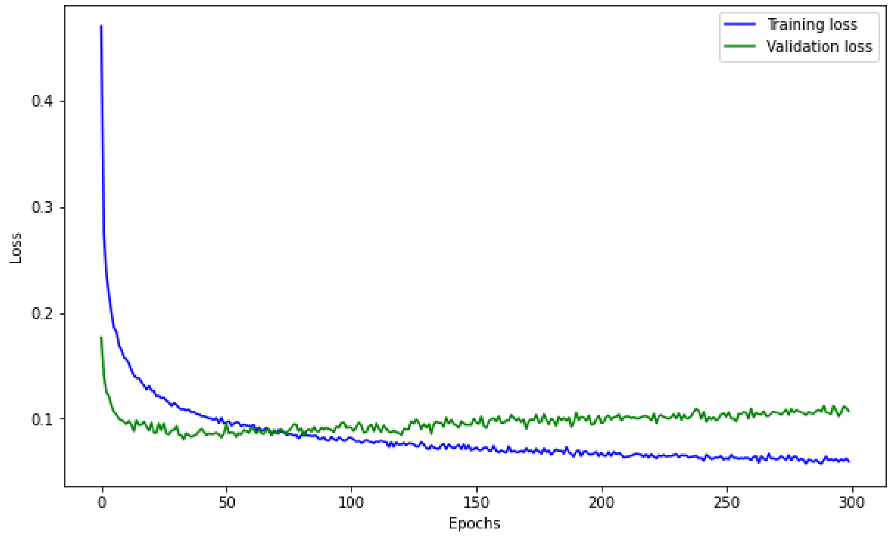


Figure 20. Model training error vs. testing error network learning curves.

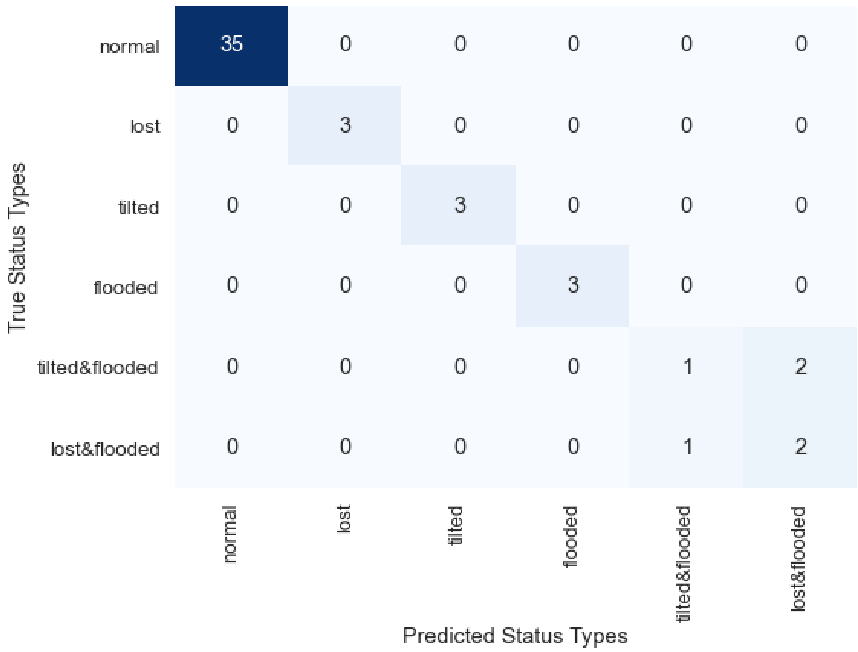


Figure 21. Confusion matrix for model test results.

To further verify the effectiveness and accuracy of the model, we downloaded and loaded the tiny machine learning model trained on the platform into the edge computing gateway ESP32 on the smart-pole. This setup was then subjected to an on-site field test (located in Changping Town, Dongguan City, Guangdong Province, China Kingsun Optoelectronics Co., Ltd), as shown in Figure 22. The results of the test revealed that when the MC is tilted, the display on the wisdom pole near the local MC immediately and correctly shows the state of the MC being turned on. Additionally, the system broadcasts the state of the MC in real time, indicating that the system has good accuracy and real-time performance.

(a) (b)

Figure 22. (a) Embedded inspection device for MC; (b) MC monitoring system field test situation.

4.2. Data transfer time, data storage analysis and energy power consumption

In order to run the machine learning model on the edge computing ESP32 microprocessor, it is necessary to convert the model. For the conversion of the tiny machine learning model, we utilize the EON compiler provided by the EI platform. This adds another layer of optimization to the machine learning model by quantizing and compressing it. The float32 types are converted into int8, which reduces the storage space required by the model and decreases the inference time. However, this reduction in size and time does not better affect the prediction recognition accuracy. The actual results are presented in Table 2. The test performance of the EI platform on the device is illustrated in Figure 23. These figures demonstrate that compared to the model before quantization and compression, there are improvements in terms of inference time, peak device memory usage and flash memory usage.

Table 2. Inference time, storage and accuracy in the case of two model versions, int8 and float32 types.

Model Version	Inference Time	Peak RAM Usage	Flash Usage	Accuracy
int8	2 ms	1.3 K	17.6 K	94.00%
float32	9 ms	1.5 K	24.6 K	94.57%

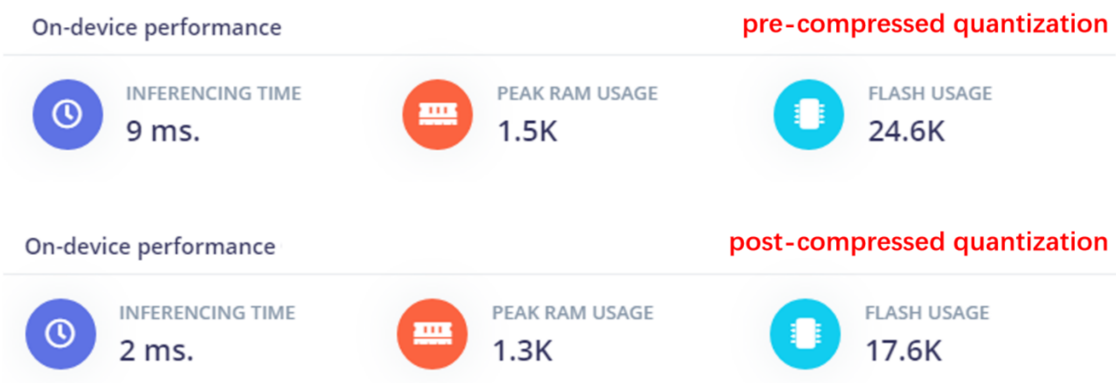


Figure 23. Performance on the device before and after model compression quantization.

The proposed model for the edge computing-based manhole cover intelligent monitoring system was evaluated through 320 real test experiments. The performance index of the model is presented in Table 3. When compared with other research results, the model demonstrates a smaller average response time and average latency time while achieving mean average precision (mAP). Thus, the system model has certain advantages.

Table 3. Comparison of data transfer performance across system models.

Method	Average Response Time	Average Latency Time	mAP
Gangyong Jia et al. [15]	5.13 s	9.2 ms	—
Haotian Ren et al. [46]	—	—	0.883
Wesam Moneer Rasheed et al. [47]	4.72 s	—	0.85
He-sheng Zhang et al. [21]	—	—	0.95
Our model	4.81 s	8 ms	0.94
Our model using only cloud computing	7.46 s	21.3 ms	0.957

This study also compares the power consumption, response time and accuracy of manhole cover intelligent monitoring in cloud computing mode and cloud-edge cooperative computing mode. Due to the manhole cover detection device being packaged into a disc-type shell installed under the manhole cover, it is challenging to directly measure its power consumption during operation. Therefore, we employed a conversion method to evaluate the power consumption of the two computing modes. The test was conducted on the industrial road in front of the South Gate of Kingsun Optoelectronics Co., Ltd. in Changping Town, Dongguan City, Guangdong Province, China. The cloud-based KitLink server was set up in Kingsun Optoelectronics Co., Ltd. in Hengli Town, Dongguan City, Guangdong Province, China, which is approximately 13 kilometers away from the test site. As depicted in Figure 24, it was observed that during the operational hours of the two computing modes, the cloud computing mode exhibited a shorter lifespan for the manhole cover detection device, exhausting its power within 3 months and rendering it incapable of functioning properly. Conversely, when the same test was conducted using the collaborative computing mode on the cloud side, it was found that the device could operate normally for over a year. This indicates that the cloud-edge collaborative computing mode offers more than four times the working hours compared to the cloud computing mode, thereby reducing the frequency of replacing the power supply for monitoring a large number of urban manhole covers. Furthermore, a rough calculation can be made to support these findings. The manhole cover detection device utilizes a 3000 mAh battery with a power supply voltage of 3.3 V. The transmission power of LoRa transmission is dependent on the transmission distance and typically ranges from 1 mW to 10 mW. By applying Ohm's law, we can easily calculate the working time for the cloud computing mode as tc = (3000*3.3)/4.5 = 2200 hours, which is a difference of at least 3 months. On the other hand, the working time for the cloud edge collaboration computing mode is calculated as tce = (3000*3.3)/1.1 = 9000 hours, equating to over 1 year.

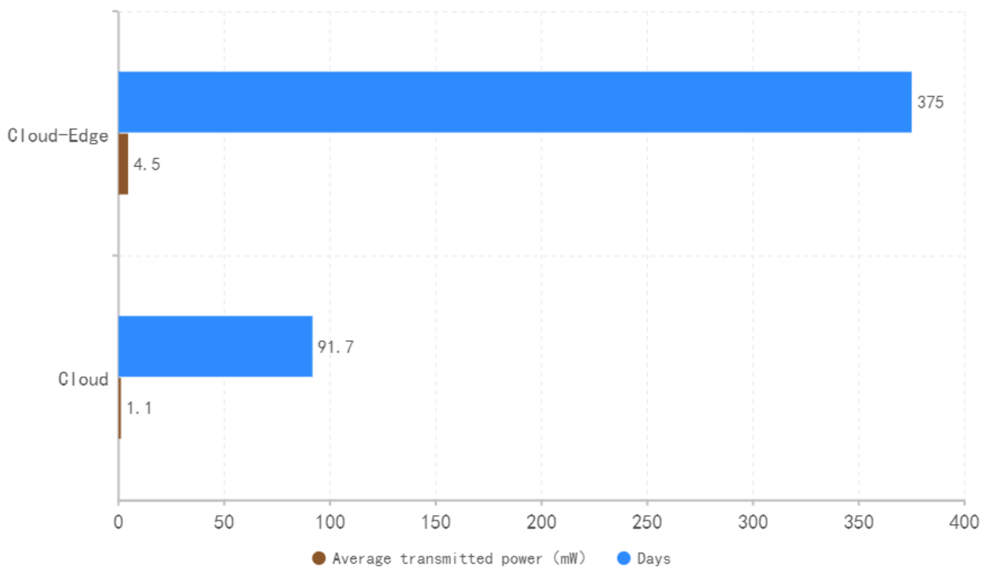


Figure 24. Comparison of electrical energy power consumption between cloud computing and cloud-edge co-computing models.

In analyzing the response time and accuracy test results as shown in the last two rows of Table 3, it is evident that the cloud computing mode yields slightly higher recognition accuracy compared to the cloud-edge collaborative computing mode. However, the real-time response speed of the former is significantly lower than that of the latter. This poses a potential risk as it may result in delayed alerts for abnormal states of MCs, failing to promptly notify the authorities. Thus, it is recommended to opt for the cloud-edge cooperative computing model for intelligent monitoring of MCs. In addition, a detailed analysis of the response time for the two computing modes revealed that the response process includes three key times: 1) transmission time for MC state data, 2) processing time for state data and 3) transmission time for decision execution. Figure 25 illustrates that the cloud-side collaborative computing model offers significant advantages in terms of data collection transmission time and decision execution transmission compared to the cloud computing model. However, it has a slight disadvantage due to limited resources in model inference recognition. Overall, it performs better. In scenarios that require real-time monitoring of manhole covers, the cloud-edge collaborative computing model is recommended as it enables computationally intensive tasks to be offloaded to the edge and the cloud for collaborative processing [48]. This approach greatly enhances the system's overall performance and serves as a suitable choice for the computing model.

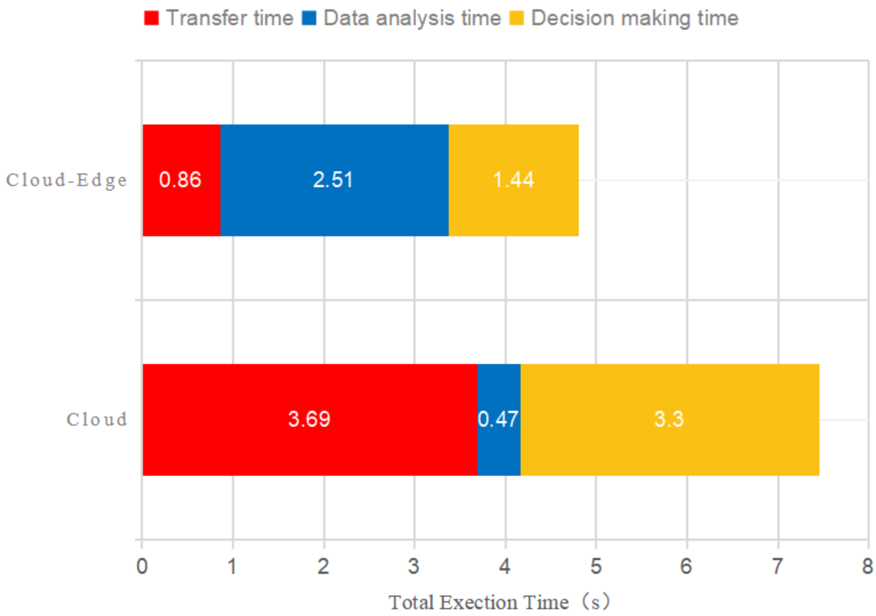


Figure 25. Comparison of response times for cloud computing and cloud-edge co-computing models.

5. Conclusions and outlooks

In this paper, we propose a cloud-edge-end collaborative system called MCIMS (Manhole Cover Intelligent Monitoring System) that utilizes edge computing. The system involves embedding a monitoring device in each MC in the city, which can effectively detect various statuses such as normal, tilted, lost, flooded, inundated and moving. The device transmits the status data, along with positioning data, to the edge gateway ESP32 microcontroller module located on the nearby wisdom pole using low-power wireless LoRa communication. The edge gateway is equipped with a pre-loaded lightweight machine learning model that has been trained, validated and tested by the edge computing EI platform. The model is compressed and quantized to ensure high reliability and stability. The MC status is derived by analyzing the real-time timing signals received by the edge gateway through this tiny machine learning model. If the MC status is abnormal, the ESP32 immediately performs real-time localization processing to determine the type of abnormality. It then sends a control signal to the display and voice module, which displays the abnormal status of the MC and broadcasts a voice alarm on the big screen. This serves as a reminder for pedestrians and vehicles passing by. Simultaneously, the information is sent to the KitLink cloud platform, allowing remote municipal management personnel to understand the locations and types of MC anomalies. Mobile municipal managers and maintenance personnel can access this information through the HTTP and MQTT protocols, respectively, enabling them to quickly and efficiently address the maintenance and repair needs of the abnormal MC. This ensures the safety of public travel, enhances the service and safety of municipal departments and improves their efficiency. If the MC is normal, no further action is taken. The system model has been applied to various towns and streets in Dongguan City, Guangdong Province. The results indicate that the average response time of the system is less than 5 seconds, with an average delay time of less than 10 milliseconds. Furthermore, by quantizing and compressing the model into a lightweight version, it better reduces the storage space and operating memory required. Despite this, the system maintains an average recognition accuracy of 94%. Overall, the system demonstrates commendable performance. The MC is an integral part of urban life and its control and management can be challenging due to its large volume, decentralization and complexity. While this paper focuses on using edge computing for abnormal manhole cover detection, alarm and rapid repair, there is also a need for in-depth exploration and research in the intelligent management of manhole covers from the following four aspects.

1) Accuracy of MC abnormal state detection and identification. The accurate detection of the MC state requires the use of reliable and precise sensing equipment. This can include high-precision sensors, RFID, video equipment or a combination of multiple modalities. Various types of artificial intelligence algorithms, such as deep learning algorithms, can then be employed for high-precision identification. Different types of research on different modal data can be conducted using these algorithms. For instance, real-time time sequence signals from sensors can be analyzed using 1D-CNN, while video image signals are suitable for 2D-CNN. Additionally, RNN, LSTM or YOLO series algorithms can be utilized for multi-target object detection in images as per the requirements. These methods are reasonable approaches to enhance the accuracy of data recognition.

2) The response speed of MC abnormalities. It is crucial to ensure public safety during travel. When MCs are tilted, moved, lost or flooded, it poses significant safety risks and requires immediate handling and repair. To solve these risks, it is essential to enhance real-time response to MC abnormalities. This can be achieved through research in edge computing, edge intelligence and other areas to improve localized and timely response. Additionally, improvements in the remote response mode in the cloud can help reduce transmission distance and mitigate delays caused by congested transmission channels, especially when dealing with large amounts of data. By addressing these challenges, the lag in processing abnormal MCs can be better reduced. The introduction of edge computing can better enhance the response speed of abnormal MC processing, improving real-time processing efficiency. Additionally, further research is necessary to enhance the efficiency of edge computing, addressing issues such as cache allocation and task job scheduling capabilities.

3) The power consumption of the MC status detection device. To ensure waterproofing, anti-electricity and dustproof capabilities, the detection device is typically sealed and installed as a whole unit within the MC. As a result, it primarily relies on dry batteries for power. To minimize power consumption, low-power, high-performance microcontrollers and low-power wireless transceiver protocols such as NB-IoT, LoRa and ZigBee technologies can be utilized. Additionally, strategies such as putting the device into sleep mode during periods of inactivity, optimizing data transmission and allocation of transmission time slots, can further reduce power consumption and extend the device's survival time.

4) Cost of the system. Cost considerations are crucial for both the development and implementation of the system, particularly with the accuracy of abnormal state detection of the MC. By applying the principle of logical equivalence between hardware and software, a balanced solution can be achieved through technological empowerment, integration and software optimization.

Use of AI tools declaration

The authors declare they have not used artificial intelligence (AI) tools in the creation of this article.

Acknowledgments

This work was supported by 2020 Dongguan Science and Technology Specialist Program (Grant No. 20201800500792), 2021 Guangdong Basic and Applied Fundamental Youth Fund Project (Grant No. 2021A1515110834), 2019 Guangdong Youth Characteristic Innovation Project (Grant No. 2019KQNCX227), 2021 Research Program of Guangdong Institute of Science and Technology (Grant No. GKY-2021KYZDK-5), Dongguan Sci-tech Commissioner Program (Grant No. 20231800500572), Dongguan Science and Technology of Social Development Program (Grant No. 20221800905802), Special Projects in Key Fields of Colleges and Universities in Guangdong Province (Grant No. 2021ZDZX1017), Key Scientific Research Projects of Guangdong University of Science and Technology (Grant No. GKY-2021KYZDK-1 & XJ2022003401), Young Innovative Talents Projects in Colleges and Universities in Guangdong Province (Grant No. 2022KQNCX120), Guangdong Provincial Key Construction Discipline Scientific Research Capacity Improvement Project (Grant No. 2022ZDJS148) and Teaching and Learning Mutual Benefit Project Team under Science and Education Innovation of Guangdong University of Science and Technology (Grant No. GKJXXZ2023030).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. W. Liu, D. Y. Chen, P. C. Yin, M. Y. Yang, E. Z. Li, M. Xie, et al., Small manhole cover detection in remote sensing imagery with deep convolutional neural networks, *ISPRS. Int. J. Geo-Inf.*, **8** (2019), 913–924. https://doi.org/10.3390/ijgi8010049
2. B. D. Zhou, W. J. Zhao, W. H. Guo, L. C. Li, D. J. Zhang, Q. Z. Mao, et al., Smartphone-based road manhole cover detection and classification, *Autom. Constr.*, **140** (2022), 104344–104355. https://doi.org/10.1016/j.autcon.2022.104344
3. R. Hubaut, R. Guichard, J. Greenfield, M. Blandeau, Validation of an embedded motion-capture and EMG setup for the analysis of musculoskeletal disorder risks during manhole cover handling, *Sensors*, **22** (2022), 436–451. https://doi.org/10.3390/s22020436
4. V. Albino, U. Berardi, R. M. Dangelico, Smart cities: Definitions, dimensions, performance, and initiatives, *J. Urban Technol.*, **22** (2015), 3–21. https://doi.org/10.1080/10630732.2014.942092
5. X. Y. Liu, Y. Han, Y. H. Du, IoT device identification using directional packet length sequences and 1D-CNN, *Sensors*, **22** (2022), 8337–8356. https://doi.org/10.3390/s22218337
6. S. Hymel, C. Banbury, D. Situnayake, A. Elium, C. Ward, M. Kelcey, et al., Edge Impulse: An MLOps platform for tiny machine learning, preprint, arXiv:2212.03332.
7. H. H. Aly, A. H. Soliman, M. Mouniri, Towards a fully automated monitoring system for manhole cover: Smart cities and IOT applications, in *2015 IEEE First International Smart Cities Conference (ISC2)*, (2015), 1–7. https://doi.org/10.1109/ISC2.2015.7366150
8. X. R. Fu, Manhole cover intelligent detection and management system, in *2016 6th International Conference on Electronic, Mechanical, Information and Management Society (ICEMIMS)*, (2016), 986–988. https://doi.org/10.2991/emim-16.2016.203

9. V. K. Nallamothu, S. Medidi, S. P. Jannu, IOT based manhole detection and monitoring system, in *2022 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, (2022), 1–6. https://doi.org/10.1109/ICDCECE53908.2022.9793287
10. R. Dronavalli, K. Seelam, P. Maganti, J. Gowineni, S. D. Challamalla, IoT-based automatic manhole observant for sewage worker's safety, in *2022 International Conference on Automation, Computing and Renewable Systems (ICACRS)*, (2022), 310–316. https://doi.org/10.1109/ICACRS55517.2022.10029252
11. S. Salehin, S. S. Akter, A. Ibnat, T. T. Anannya, N. N. Liya, M. Paramita, et al., An IoT based proposed system for monitoring manhole in context of Bangladesh, in *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT)*, (2018), 411–415. https://doi.org/10.1109/CEEICT.2018.8628091
12. C. S. Ram, C. N. Kumar, S. Abhilash, Automated street light control and manhole monitoring with fault detection & reporting system for municipal department, *Int. J. Sci. Res. Eng. Man*, **7** (2023), 9–15. https://doi.org/10.55041/IJSREM.17962
13. S. K. Muragesh, R. Santhosha, Automated internet of things for underground drainage and manhole monitoring system for metropolitan cities, *Int. J. Inf. Comput. Technol.*, **4** (2014), 1211–1220. https://doi.org/10.0974/IJICT.15634
14. Y. Liu, M. Y. Du, C. F. Jing, Y. Bai, Design of supervision and management system for ownerless manhole covers based on RFID, in *2013 21st International Conference on Geoinformatics (ICG)*, (2013), 1–4. https://doi.org/10.1109/Geoinformatics.2013.6626149
15. G. Y. Jia, G. J. Han, H. L. Rao, L. Shu, Edge computing-based intelligent manhole cover management system for smart cities, *IEEE Internet Things*, **5** (2018), 1648–1656. https://doi.org/10.1109/JIOT.2017.2786349
16. A. Mankotia, A. K. Shukla, IOT based manhole detection and monitoring system using Arduino, *Mater. Today: Proc.*, **57** (2022), 2195–2198. https://doi.org/10.1016/j.matpr.2021.12.264
17. N. Nataraja, R. Amruthavarshini, N. L. Chaitra, K. Jyothi, N. Krupaa, S. S. M. Saqquaf, Secure manhole monitoring system employing sensors and GSM techniques, in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, (2018), 2078–2082. https://doi.org/10.1109/RTEICT42901.2018.9012245
18. X. C. Guo, B. B. Liu, L. L. Wang, Design and implementation of intelligent manhole cover monitoring system based on NB-IoT, in *2019 International Conference on Robots & Intelligent System (ICRIS)*, (2019), 207–210. https://doi.org/10.1109/ICRIS.2019.00061
19. J. P. Zhang, X. L. Zeng, Design of intelligent manhole cover monitoring system based on narrow band internet of things, in *2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP)*, (2022), 1354–1357. https://doi.org/10.1109/ICSP54964.2022.9778462
20. W. Sun, Design and realization of LoRa-based manhole cover safety monitoring system, *J. Int. Things Technol.*, **9** (2019), 25–26,30. https://doi.org/10.16667/j.issn.2095-1302.2019.04.005
21. H. S. Zhang, L. Li, X. Liu, Development and test of manhole cover monitoring device using LoRa and accelerometer, *IEEE Trans. Instrum. Meas.*, **69** (2020), 2570–2580. https://doi.org/10.1109/TIM.2020.2967854
22. X. Liu, H. S. Zhang, L. Li, Research on LoRa communication performance in manhole cover monitoring, in *2019 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, (2019), 1–6. https://doi.org/10.1109/I2MTC.2019.8826898

23. L. Li, H. S. Zhang, X. Liu, Development of low power consumption manhole cover monitoring device using LoRa, in *2019 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, (2019), 1–6. https://doi.org/10.1109/I2MTC.2019.8826885
24. Y. Yu, J. Li, H. Guan, C. Wang, Automated detection of road manhole covers from mobile LiDAR point-clouds based on a marked point process, in *2013 Fifth International Conference on Geo-Information Technologies for Natural Disaster Management (ICGITNDM)*, (2013), 130–136. https://doi.org/10.1109/GIT4NDM.2013.23
25. Z. Y. Wei, M. M. Yang, L. Z. Wang, H. Ma, X. X. Chen, R. F. Zhong, Customized mobile LiDAR system for manhole cover detection and identification, *Sensors*, **19** (2019), 2422–2439. https://doi.org/10.3390/s19102422
26. V. Vishnani, A. Adhya, C. Bajpai, P. Chimurkar, K. Khandagle, Manhole detection using image processing on google street view imagery, in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, (2020), 684–688. https://doi.org/10.1109/ICSSIT48917.2020.9214219
27. U. Andrijašević, J. Kocić, V. Nešić, Lid opening detection in manholes using RNN, in *2020 28th Telecommunications Forum (TELFOR)*, (2020), 1–4. https://doi.org/10.1109/TELFOR51502.2020.9306668
28. R. Krishnan, A. Santhana, D. D. Kumari, N. Nandhini, G. Karpagarajesh, K. Narayanan, et al., A secured manhole management system using IoT and machine learning, *Rec. Adv. Int. Things Mach. Learn.*, **215** (2022), 3–22. https://doi.org/10.1007/978-3-030-90119-6_3
29. D. P. Zhang, X. C. Yu, L. Yang, D. Y. Quan, H. M. Mi, K. Yan, Data-augmented deep learning models for abnormal road manhole cover detection, *Sensors*, **23** (2023), 2676–2693. https://doi.org/10.3390/s23052676
30. K. Thakur, A. Adhya, C. Bajpai, P. Chimurkar, P. Kasambe, Manhole management using image processing and data analytics, in *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, **12** (2021), 1–5. https://doi.org/10.1109/ICCCNT51525.2021.9579541
31. W. S. Shi, J. Cao, Q. Zhang, Y. H. Z. Li, L. Y. Xu, Edge computing: Vision and challenges, *IEEE Internet Things*, **3** (2016), 637–646. https://doi.org/10.1109/JIOT.2016.2579198
32. F. Mo, L. Yu, Z. K. Zhang, Y. Zhao, Design and implementation of manhole cover safety monitoring system based on smart light pole, *Math. Probl. Eng.*, **2022** (2022), 1–12. https://doi.org/10.1155/2022/3081649
33. B. Ravelo, M. Guerin, W. Rahajandraibe, V. Gies, L. Rajaoarisoa, S. Lalléchère, Low-pass NGD numerical function and STM32 MCU emulation test, *IEEE Trans. Ind. Electron.*, **8** (2022), 8346–8355. https://doi.org/10.1109/TIE.2021.3109543
34. V. B. Vales, O. C. Fernández, T. D. Bolaño, C. J. Escudero, J. A. G. Naya, Fine time measurement for the internet of things: a practical approach using ESP32, *IEEE Internet Things*, **19** (2022), 18305–18318. https://doi.org/10.1109/JIOT.2022.3158701
35. W. S. Shi, X. Z. Zhang, Y. F. Wang, Q. Y. Zhang, Edge computing: Status and prospects, *J. Comput. Res. Dev.*, **56** (2019), 69–89. https://doi.org/10.7544/issn1000-1239.2019.20180760
36. W. S. Shi, H. Sun, J. Cao, Q. Zhang, W. Liu, Edge computing: A new computing model for the internet of everything era, *J. Comput. Res. Dev.*, **54** (2017), 907–924. https://doi.org/10.7544/issn1000-1239.2017.20160941

37. Y. F. Li, X. R. He, Y. Z. Bian, Task offloading of edge computing network and energy saving of passive house for smart city, *Mob. Inf. Syst.*, **2022** (2022), 1–11. https://doi.org/10.1155/2022/4832240
38. B. Pang, E. Nijkamp, Y. N. Wu, Deep learning with TensorFlow: A review, *J. Educ. Behav. Stat.*, **2** (2020), 227–248. https://doi.org/10.3102/1076998619872761
39. I. N. Mihigo, M. Zennaro, A. Uwitonze, J. Rwigema, M. Rovai, On-Device IoT-based predictive maintenance analytics model: Comparing TinyLSTM and TinyModel from edge impulse, *Sensors*, **22** (2022), 5174–5194. https://doi.org/10.3390/s22145174
40. L. Qing, K. Yang, W. Tan, J. Li, Automated detection of manhole covers in Mls point clouds using a deep learning approach, in *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, (2020), 1580–1583. https://doi.org/10.1109/IGARSS39084.2020.9324137
41. S. J. Tian, S. Wang, H. R. Xu, Early detection of freezing damage in oranges by online Vis/NIR computers and electronics in agriculture transmission coupled with diameter correction method and deep 1D-CNN, *Comput. Electr. Agric.*, **193** (2022), 106638–106659. https://doi.org/10.1016/j.compag.2021.106638
42. C. C. Che, H. W. Wang, X. M. Ni, R. G. Ning, M. L. Xiong, Remaining life prediction of aero-engine based on 1D-CNN and Bi-LSTM, *J. Mech. Eng.*, **57** (2021), 304–312. https://doi.org/10.3901/JME.2021.14.304
43. Y. Kim, Convolutional neural networks for sentence classification, preprint, arXiv:1408.5882.
44. H. J. Wang, Z. Y. Yi, Z. Z. Ke, Y. J. Guo, H. Y. Dong, Wear monitoring of spiral milling tools based on one-dimensional convolutional neural network, *J. Zhejiang Univ. (Eng. Ed.)*, **54** (2020), 931–939. https://doi.org/10.3785/j.issn.1008-973X.2020.05.010
45. L. Liu, J. C. Zhu, G. J. Han, Y. G. Bi, Bearing health monitoring and fault diagnosis based on joint feature extraction in one-dimensional convolution neural network, *J. Soft.*, **32** (2021), 2379−2390. https://doi.org/10.13328/j.cnki.jos.006188
46. H. T. Ren, F. Deng, Manhole cover detection using depth information, *J. Phys.: Conf. Ser.*, **1856** (2021), 1–7. https://doi.org/10.1088/1742-6596/1856/1/012037
47. W. M. Rasheed, R. Abdulla, L. Y. San, Manhole cover monitoring system over IOT, *J. Appl. Technol. Innov.*, **5** (2021), 1–6. https://doi.org/10.2600/JATI.245739682
48. S. Bouhoula, M. Avgeris, A. Leivadeas, I. Lambadaris, Computational offloading for the industrial internet of things: A performance analysis, in *2022 IEEE International Mediterranean Conference on Communications and Networking (MeditCom)*, (2022), 1–6. https://doi.org/10.1109/MeditCom55741.2022.9928770