EAI Endorsed Transactions
on Energy Web

Research Article **EAI.EU**

Implementation and Research on Neural Network-Based Monitoring System for Preventing Battery-Related Fire Hazards in Indoor Environments

Xin Li*, Hanyu Shi and Shiqi Zhou

School of Electronic and Optical Engineering, Nanjing University of Science and Technology Zijin College, Nanjing 210023, China

Abstract

With the widespread use of electric bicycles and batteries, fire accidents caused by batteries have become increasingly serious, especially in closed indoor environments. Traditional fire prevention methods often rely on static monitoring and simple sensors, which can suffer from delayed responses or fail to accurately identify fire hazards. To address this issue, this paper proposes an innovative monitoring system for preventing fire hazards caused by batteries or electric bicycles in indoor environments, based on an STMicroelectronics 32-bit Microcontroller (STM32) and Open-source Machine Vision module (OpenMV). The system uses deep learning to train a neural network to recognize the image information of batteries or electric bicycles. The key innovation of this system lies in several aspects: firstly, it utilizes the OpenMV module for real-time image processing, enabling efficient and accurate recognition of batteries and electric bicycles; secondly, the integration with the STM32 microcontroller enhances the system's data processing capabilities and enables flexible communication and responses with external devices; finally, the system features high-efficiency serial communication, ensuring the real-time transmission and processing of monitoring data for swift responses to potential fire risks. Experimental results show that the system can accurately identify batteries or electric bicycles in indoor environments and respond in a timely manner, significantly reducing fire hazards. In addition, the system's design is not limited to preventing battery-related fire hazards in indoor environments. Compared to traditional methods, this study's innovation lies in combining deep learning and embedded control technology for fire prevention, providing a practical and scalable solution for battery-related fire risk prevention.

Key words: openMV, STM32 microcontroller, deep learning; neural network

Received on 20 January 2026, accepted on 31 March 2026, published on 07 April 2026

doi: 10.4108/ew.11622

1. Introduction

Battery-related fire hazards have been a growing concern, particularly with the increasing adoption of lithium-ion batteries in consumer electronics and electric vehicles. These fires pose a significant risk to indoor environments, particularly in residential buildings and public spaces. A failure to detect early-stage thermal runaway or malfunctions

*Corresponding author. Email: 31449299@qq.com

in battery charging devices could lead to severe safety incidents and property damage, affecting public safety and energy reliability.
With the widespread use of electric bicycles, mobile power banks, and other lithium battery devices, these high-energy-density products have become common in daily life. However, when these devices experience overcharging, incorrect wiring, collisions, or aging, they are prone to catastrophic "thermal runaway." This process rapidly generates high-temperature jets reaching thousands of degrees Celsius, transforming living rooms, hallways, and elevator cabins into dangerous, high-pressure fire zones [1-2]. Public data indicates that in the past three years, there have been over ten thousand fires across the country caused by battery devices entering households, with fatalities occurring at rates 3-4 times higher than in ordinary residential fires [3]. Traditional fire safety solutions, such as smoke detectors and temperature sensors, typically react only after combustion products have already spread, making them too late to prevent danger[4]. Additionally, these systems often fail to distinguish between hazardous situations, such as battery charging or boiling water in a pot, leading to frequent false alarms and missed detections [5]. This makes it clear that property management and fire departments urgently need a reliable method to "detect hazards half a minute in advance" and automatically intervene before a fire breaks out [6].
The existing methods of fire prevention and monitoring in such scenarios are often limited by their inability to address the unique risks posed by lithium batteries and electric vehicles. Previous studies have focused on traditional fire detection systems, relying on smoke, heat, and gas sensors. However, these systems are ill-equipped to handle the specific and rapidly developing nature of battery fires. While there have been advancements in smoke detection and temperature monitoring, these technologies often suffer from limitations, such as slow response times, inability to detect early-stage risks, and high false alarm rates. Moreover, few studies have proposed effective, real-time intervention systems that address both detection and automatic response, particularly for lithium battery hazards.
This paper proposes a novel, visual priority-based embedded solution to fill this gap. The system uses OpenMV as a "security guard" to continuously monitor elevator doors and corridors. OpenMV is equipped with an internal, lightweight Convolutional Neural Network (CNN) model, only 312 KB in size, capable of performing a "battery/vehicle/non-battery" three-class classification in just 33 milliseconds. Once the confidence level exceeds 0.85 for three consecutive frames, the system sends the category code and coordinates to the STM32F407 microcontroller via serial communication. The STM32 then performs three levels of interlocking actions in just 200 milliseconds: ① triggering a high-decibel alarm to prompt residents to "Please take the battery downstairs"; ② activating a relay to cut off the power supply, forcibly stopping charging; and ③ linking to an electromagnetic lock or elevator light curtain to prevent further movement of the identified target. Additionally, event logs and thumbnail images are synchronized to a property management platform for post-event investigation.
The system is cost-effective, with an estimated price under 220 yuan and a power consumption of less than 1.2 W. It is plug-and-play and does not require any modifications to the existing electrical wiring. After 216 hours of multi-scenario testing, the recognition accuracy rate reaches 96.7%, with a false alarm rate of only 0.8%. Notably, it can detect fire hazards 30-60 seconds earlier than traditional smoke detectors, making it an ideal solution for preventing battery-related fire risks in communities, schools, shopping malls, and rental properties.[7]
The main contribution of this research is the development of an innovative fire prevention system that integrates advanced image recognition, real-time hazard detection, and automatic intervention mechanisms. By addressing the specific risks associated with lithium battery devices, this system offers a scalable and replicable model for fire prevention in diverse environments. The significance of this research lies in its potential to provide a reliable, early-warning fire detection system that could save lives and reduce property damage, paving the way for safer communities.
In this paper, the term "electric bicycle" is used consistently to represent battery-powered two-wheeled vehicles, including those commonly referred to as e-bikes or electric bicycles.

2. Whole scheme design

2.1 Project design

This design utilizes the STM32 microcontroller as the core of the system control, and employs the OpenMV camera to identify the battery model. The overall design block diagram is shown in Figure 1. The image acquisition module uses the OpenMV camera, responsible for capturing real-time image information of the monitoring area, providing data support for the subsequent identification work.
This design mainly consists of the following modules:
Single-chip microcontroller control system: with the STM32 microcontroller as the core, it receives the identification results transmitted by the OpenMV module, processes the data, and responds according to the set logic to control other modules.
Sensor module: includes a temperature sensor, used to monitor the environmental temperature, assisting in the judgment of whether there is a fire hazard.
Alarm module: composed of a buzzer and an LED light, when the system identifies that the battery or electric bicycle enters the room, it emits sound and light alarms to alert relevant personnel to pay attention.
Display module: used to display the working status of the system, the identification results, and environmental temperature and other information, facilitating users to view.
Laser ranging module: used to measure the distance between the object and the monitoring device, assisting in the judgment of whether the battery or electric bicycle has entered the specific monitoring area.

Servo control module: can drive the servo to achieve certain mechanical actions, such as closing the channel, etc., further preventing the battery from entering the room, according to the control instructions of the system. Power supply module provides stable power supply for all modules of the entire system to ensure the normal operation of the system.

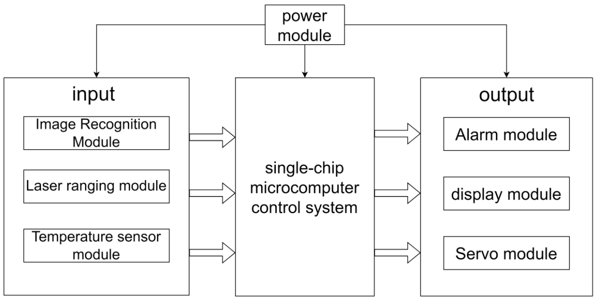

Figure 1. Overall System Block Diagram

2.2 Main hardware design of the system

2.2.1 Selection of STM32 Microcontroller

The STM32 series of microcontrollers feature high performance, low power consumption, and a rich set of peripherals. They are produced by ARM and have been widely used in embedded systems[8].This system selects the STM32F103C8T6 microcontroller as the control core. This model of microcontroller is based on the ARM Cortex-M3 core, with a working frequency of up to 72 MHz, 64KB of Flash and 20KB of SRAM. It has multiple general-purpose timers, Serial Peripheral Interface (SPI), Inter-Integrated Circuit (I2C) communication protocol, Universal Synchronous/Asynchronous Receiver/Transmitter (USART) and other communication interfaces, which can meet the data transmission and control requirements between the system and each module. At the same time, its smaller package size also facilitates the design of a smaller system.

2.2.2 OpenMV Camera module

OpenMV is an open-source machine vision module based on MicroPython, equipped with an STM32H743VI microcontroller and an OV7725 camera[9-10].It supports the deployment of various image processing algorithms and deep learning models. It is produced by StarTone Technology Company. In this system, OpenMV is responsible for real-time processing and analysis of the captured images and uses a trained neural network model to identify batteries or electric bicycles. The OpenMV module has the advantages of small size, low power consumption, and fast processing speed, which can meet the real-time requirements of the system. It communicates with the STM32 microcontroller via a serial port and sends the recognition results in a specific format to the STM32.

2.2.3 Temperature sensor module

The DS18B20 temperature sensor is adopted as the environmental temperature monitoring component. DS18B20 is a single-bus digital temperature sensor. It can be directly connected to the General-Purpose Input/Output (GPIO) port of the STM32 microcontroller and transmits data through the single-bus protocol, simplifying the circuit design. The temperature sensor continuously monitors the environmental temperature and transmits the temperature data to the STM32 microcontroller. When the temperature exceeds the set threshold, the system will take corresponding warning measures in combination with the image recognition results.

2.2.4 Alarm module

The alarm module consists of a buzzer and an LED light. The buzzer is an active buzzer that can emit sound by applying a certain voltage to its two ends. It’s on/off state is controlled by the GPIO port of the STM32 microcontroller. The LED light uses a high-brightness red LED, which is also controlled by the STM32 microcontroller. When the system triggers an alarm, the buzzer emits a harsh sound and the LED light flashes to achieve the effect of audible and visual alarm, promptly alerting relevant personnel. The program flowchart is shown in Figure 2:

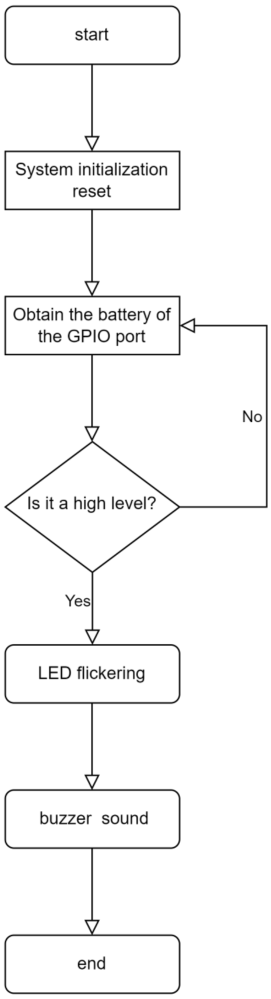

Figure 2. Alarm Module Flowchart

2.2.5 display module

The OLED12864 display screen is selected as the display module of the system. The Organic Light-Emitting Diode (OLED) display has the advantages of self-illumination, low power consumption, high contrast ratio, and fast response speed, which can clearly display various information of the system. It is connected to the STM32 single-chip microcontroller through the I2C, and real-time displays the current recognition results (whether a battery or electric bike is detected), environmental temperature, system working status, etc., making it convenient for users to understand the operation of the system at any time .

2.2.6 Laser ranging module

The VL53L0X laser ranging sensor is adopted. This sensor is based on the Time of Flight (ToF) principle and can measure distances within a range of 2 meters.[11] The VL53L0X communicates with the STM32 microcontroller via the I2C interface, and continuously measures and monitors the distance between objects in the monitoring area and the sensor. When an object is detected to be within a distance lower than the set threshold, combined with the image recognition results, it is determined whether it is an electric battery or an electric bicycle entering the room, thereby triggering the corresponding control strategy. The program flowchart is shown in Figure 3.

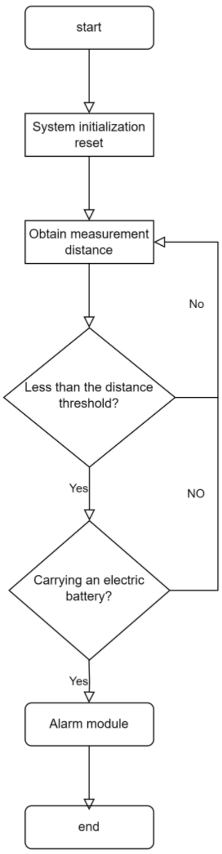

Figure 3. Flowchart of Laser Distance Measurement Module

2.2.7 Servo control module

The servo motor selected is the SG90 servo motor. It is a small-sized servo motor with moderate torque and simple control. The SG90 servo motor is controlled through Pulse-Width Modulation (PWM) signal[12]. The timer of the STM32 microcontroller can output PWM signals, and by changing the duty cycle of the PWM, the rotation angle of the servo motor can be controlled. When the system detects that the battery or electric bicycle attempts to enter the room, the STM32 microcontroller can control the servo motor to rotate, driving the related mechanical structures (such as the barrier) to close the passage and prevent it from entering.

In addition to the image recognition component, the system's real-time sensor data processing, including temperature and distance measurement, provides a comprehensive approach to monitoring potential hazards.

3. System software design

3.1 Overall software architecture

The software design of the system adopts the modular design concept, dividing the entire system's functions into multiple independent modules. Each module is responsible for a specific function, and the modules interact with each other through interfaces to exchange data. The overall software architecture is shown in Figure 4, which mainly includes the image acquisition and recognition module, the data communication module, the sensor data processing module, the control and alarm module, and the display module.

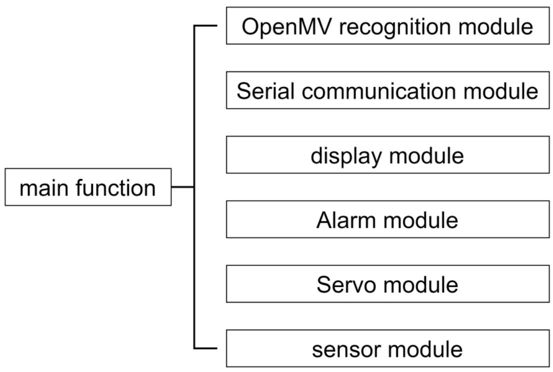

Figure 4. Software Framework

3.2 Software Design of Image Acquisition and Recognition Module

The core of the image acquisition and recognition module is the program running on the OpenMV module, which mainly accomplishes image acquisition, preprocessing, feature extraction, and classification recognition based on neural networks.

Before being fed into the neural network, the captured images undergo several preprocessing operations to enhance computational efficiency and robustness. Specifically, region-of-interest (ROI) cropping is applied to remove irrelevant background regions, followed by resizing to match the network input resolution. Grayscale conversion is adopted to reduce computational complexity while preserving essential structural features relevant to object classification.
After training, the model was converted to TensorFlow Lite (.tflite) format and deployed onto the OpenMV platform for real-time inference. The inference result is transmitted to the STM32 microcontroller via serial communication once the confidence score exceeds the predefined threshold. Figure 5 illustrates the training convergence performance of the proposed model.

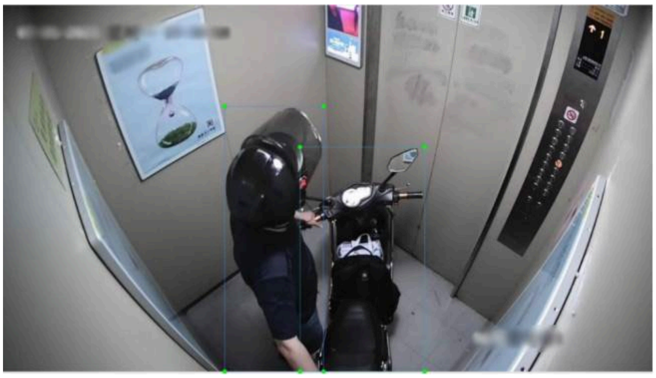

Figure 5. Model Training Effect

3.2.1 Dataset Description and Experimental Setup

To ensure reliable training and evaluation of the neural network model, a dedicated dataset was constructed for this study. A total of 3,200 images were collected under diverse environmental conditions, including variations in lighting (daytime, medium night lighting, low night lighting), temperature, humidity, angle, distance, and background complexity.
The dataset contains three categories:
- Battery: 1,200 images
- Electric bicycle: 1,000 images
- Non-target objects (negative samples such as suitcases, backpacks, and ordinary bicycles): 1,000 images

All images were manually annotated to ensure labelling accuracy.
The dataset was randomly divided into training, validation, and test sets with a ratio of 70% : 15% : 15%. The independent test set was used for final performance evaluation.
To enhance the generalization ability of the model, data augmentation techniques were applied to the training set. [13] These included:
- Random horizontal flipping (probability = 0.5)
- Random rotation within ±15 degrees
- Brightness adjustment within ±20%
- Random cropping and scaling

These augmentation strategies effectively increased dataset diversity and improved robustness under varying environmental conditions.
The MobileNet model was trained using the TensorFlow framework. During training, the cross-entropy loss function and Adam optimizer were employed. The learning rate was initially set to 0.001 and gradually reduced using a step decay strategy. Training was conducted for 50 epochs until convergence was achieved on the validation set.
After training, the model was converted into TensorFlow Lite (.tflite) format for deployment on the OpenMV platform.
The baseline accuracy of 96.2% (before compression) and 95.1% (after pruning and quantization) were both evaluated on the same independent test set to ensure fair comparison. This guarantees that performance degradation after compression is objectively measured under identical conditions.[14]
After deployment, the OpenMV module performs real-time inference. Once the confidence score of the predicted category exceeds the predefined threshold, the recognition result (e.g., "Battery detected", "Electric bicycle detected", or "No target detected") is transmitted to the STM32 microcontroller via serial communication.

3.2.2 Real-Time Inference and Decision Logic

After deployment on the OpenMV platform, the optimized MobileNet model performs real-time inference at approximately 33 ms per frame. Each captured frame is processed individually, and the model outputs a probability distribution over the three defined classes.
To enhance system robustness, a confidence threshold of 0.85 is applied. Only when the predicted class confidence exceeds this threshold is the detection considered valid. Furthermore, a three-frame consecutive confirmation mechanism is implemented to suppress transient misclassification caused by motion blur or environmental noise.
Once the target category (battery or electric bicycle) is confirmed, the OpenMV module transmits the classification result and corresponding spatial coordinates to the STM32 microcontroller via serial communication. The STM32 then executes predefined control actions, including alarm triggering and servo actuation.
This closed-loop perception – decision – action mechanism ensures fast response while maintaining low false alarm rates.
The confidence threshold of 0.85 was determined empirically through repeated experiments to balance false positives and missed detections. Lower thresholds (e.g., 0.7) increased false alarm rates, while higher thresholds (e.g., 0.9) resulted in occasional missed detections under low-light conditions.
To further improve system robustness, a three-frame continuous detection mechanism was introduced to mitigate transient misclassifications caused by motion blur or environmental interference. Experimental comparison showed that using one frame increased false triggers, while five consecutive frames increased response delay. Therefore, three frames were selected as the optimal trade-off between reliability and response speed.
Once the confidence score exceeds the threshold and the three-frame confirmation is met, the classification result is

transmitted to the STM32 microcontroller via serial communication for real-time action.

3.3 Model Compression and Optimization of MobileNet

To ensure efficient operation of the neural network on the resource-constrained embedded platform OpenMV, the Mobile Convolutional Neural Network (MobileNet) model was compressed and optimized. MobileNet inherently adopts Depthwise Separable Convolution, which significantly reduces parameters and computation while maintaining high recognition accuracy, making it suitable for deployment on edge devices with limited computing and storage resources.
The optimization process involved:

1. Pruning: Redundant weights and insensitive channels were removed, reducing the number of parameters from approximately 4.2M to 1.1M.
2. Quantization:Model weights were quantized to 8-bit integers, reducing memory usage and computation cost. The model size was reduced from 1.6 MB to about 312 KB, satisfying OpenMV's memory constraints.
3.Acceleration: After optimization, the inference speed improved from ~60 ms/frame to 33 ms/frame (about 1.8× faster), while recognition accuracy only decreased slightly (from 96.2% to 95.1%)[15-16].

The final optimized MobileNet model runs stably on OpenMV, achieving a balance between real-time performance and recognition accuracy.
Table 1 presents the performance comparison before and after optimization:

Table 1. Performance Comparison Before and After Model Optimization

Model Version	Model Size	Parameters	Inference Speed (ms/frame)	Accuracy (%)	OpenMV Compatibility
Original MobileNet	1.6 MB	4.2M	60	96.2	No
Pruned + Quantized	312 KB	1.1M	33	95.1	Yes

Model quantization was performed using TensorFlow Lite post-training integer quantization. Specifically, full integer quantization (INT8) was applied after model training. Representative dataset calibration was used to minimize accuracy degradation.
Quantization-aware training was not adopted in this study due to hardware constraints of the OpenMV platform.

3.4 Software Design of Data Communication Module

The data communication module mainly realizes serial communication between the OpenMV module and the STM32 microcontroller, as well as communication between the STM32 microcontroller and other modules (such as sensors, display modules, etc.).
The serial communication between the STM32 microcontroller and the OpenMV module adopts asynchronous communication mode. The baud rate is set to 115200 bps, and the data format is 8 data bits, 1 stop bit, and no parity bit. The STM32 microcontroller receives the recognition result sent by the OpenMV module through the USART1 interface. The received data is in a specific protocol format, such as "$DATA,RESULT, XXX#", where "XXX" represents the recognition result (0 indicates no detection, 1 indicates detection of an electric battery, 2 indicates detection of an electric bicycle). After receiving the data, the STM32 microcontroller first performs data verification to determine if the data is complete and correct, and then parses the recognition result.

3.5 Software Design of Sensor Data Processing Module

The sensor data processing module is mainly responsible for processing and analysing the data collected by the temperature sensor and the laser ranging sensor.
For the laser ranging sensor VL53L0X, the STM32 microcontroller reads its measured distance value through the I2C protocol. Similarly, the distance data is filtered to remove outliers. A distance threshold (such as 1m) is set. When the measured distance is less than this threshold, it indicates that an object has entered the monitoring area.

3.6 Software Design of Control and Alarm Module

The control and alarm module is the core control part of the system. The STM32 microcontroller makes judgments based on the image recognition results, temperature sensor data, and laser ranging sensor data according to the preset logic and controls the corresponding modules to respond.

The control logic of the system is as follows:

When the OpenMV module detects a battery or an electric bicycle (the recognition result is 1 or 2), and the distance measured by the laser ranging sensor is less than the set threshold (the object enters the monitoring area), regardless of the environmental temperature, the system triggers an alarm, the buzzer sounds, the LED lights flash, and the servo rotates, and the channel is closed.

When the OpenMV module does not detect a battery or an electric bicycle (the recognition result is 0), but the environmental temperature exceeds the set threshold, the system only issues a temperature abnormal alarm, the buzzer sounds, the LED lights are constantly on, and the servo does not move.

When the OpenMV module does not detect a battery or an electric bicycle and the environmental temperature is normal, the system is in a normal working state and does not issue an alarm.

In the software design, the above logic is realized by writing corresponding control functions. The timer is used to control the working state of the buzzer and the LED lights, and the PWM signal is used to control the rotation of the servo.

3.7 Display module software design

The display module software mainly realizes the initialization of the OLED display screen and the display function. The STM32 microcontroller communicates with the OLED display screen through the I2C protocol, sending control instructions and display data.

The display content of the screen includes:

System working status: such as "Normal Monitoring" "Alarm" etc.

Identification results: such as "Battery detected" "Electric bike detected" "Not detected" etc.

Environmental temperature: Real-time display of the current environmental temperature value.

Distance information: Display of the distance value measured by the laser ranging sensor.

In the software, by writing display functions, the above information is displayed on the OLED screen in the preset format, and the display content is updated every second.

4. System Experiment and Result Analysis

4.1 Setup of experimental environment

In order to verify the performance of the battery entry prevention monitoring system based on the STM32 microcontroller, the following experimental environment was set up:

1. Test Area: A simulated indoor entrance area of approximately 5m × 3m was selected, mimicking typical entry conditions found in residential buildings or offices. All system modules, including the OpenMV camera, STM32 microcontroller control board, temperature sensor, laser ranging sensor, buzzer, LED light, OLED display screen, servo motor, and voice recognition module, were installed at the entrance.

2.Test Samples: The experiment involved using a diverse set of test samples to cover a wide range of scenarios:

Battery Samples: Different brands and models of lithium batteries (e.g., from mobile power banks, e-bikes, and laptops) were tested. A total of 10 battery samples were used, varying in size, shape, and power ratings to assess the system's adaptability.

Electric Bicycle Samples: Three different models of electric bicycles with varying battery sizes were tested.

Interference Samples: To evaluate the system's robustness against false alarms, a total of 15 non-battery-related items (such as suitcases, backpacks, and regular bicycles) were used as interference samples.

3. Conduct experiments under different lighting conditions (natural light during the day, lighting during the night) and temperature conditions (normal temperature of 25℃, high temperature of 45℃) to test the performance of the system in different environments[17-18].

4.2 Experimental content and methods

The experiment mainly includes the following aspects:

1.Accuracy rate test: In different environmental conditions, place the battery, electric bicycle and interference samples in the monitoring area, record the system's identification results, and calculate the system's identification accuracy rate for the battery and electric

bicycle. The formula for calculating the accuracy rate is as follows:
Recognition accuracy is defined as the percentage of correctly classified samples (including both positive and negative classes) over the total number of test samples:
Accuracy = (TP + TN) / (TP + TN + FP + FN) × 100%
2. Response time test: When the battery or electric bicycle enters the monitoring area, record the time from the system detecting the target to issuing the alarm signal (beeping of the buzzer, flashing of the LED light), testing the system's response speed.
3. Alarm reliability test: When the system identifies that the battery or electric bicycle has entered the monitoring area, observe whether the system can accurately trigger the alarm, and whether the system will generate false alarms when no battery or electric bicycle enters, testing the system's alarm reliability.

4.3 Analysis of experimental results

4.3.1 Analysis of Recognition Accuracy

Under different environmental conditions, the test results of the system's recognition accuracy for batteries and electric bicycles are shown in Table 2.

Table 2. Recognition Accuracy Rate (%) under Different Environmental Conditions

environmental conditions	Battery identification accuracy rate	Accuracy rate of electric bicycle recognition
Natural light during the day	97.8	94.5
Nighttime lighting (medium)	94.9	93.1
Nighttime lighting (low)	92.3	89.7
Room temperature (25°C)	95.1	94.8
High temperature (45°C)	94.7	94.1
Low temperature (5°C)	93.8	91.2
High humidity (85%)	94.2	92.5
Low humidity (50%)	95.3	94.9
Low background noise (40 dB)	97.5	95.3
High background noise (70 dB)	94.6	92.0

From Table 2, it can be observed that the recognition accuracy of the system for both batteries and electric bicycles is above 93% under all lighting and temperature conditions. This indicates that the system exhibits strong adaptability to environmental changes. The slight reduction in accuracy under nighttime lighting conditions could be attributed to lower lighting, which negatively affects image quality and the neural network's recognition accuracy.
To further evaluate the classification performance, a confusion matrix on the independent test dataset is presented in Table 3.

Table 3. Confusion Matrix on Test Dataset

	Predicted Battery	Predicted Electric Bicycle	Predicted Non-target
Actual Battery	168	4	3
Actual Electric Bicycle	6	155	5
Actual Non-target	2	3	172

From Table 3, it can be observed that the proposed model achieves high precision and recall across all categories. Most misclassifications occur between batteries and electric bicycles due to visual similarity in certain scenarios. However, the overall misclassification rate remains low, confirming the robustness of the proposed approach.

The overall precision, recall, and F1-score of the model are 95.4%, 95.1%, and 95.2%, respectively.

4.3.2 Response time analysis

The test results of the system's response time are shown in Table 4.

Table 4. System Response Time (ms)

Number of tests	1	2	3	4	5	Average
Response time	119	116	120	130	131	123.2

The average response time of the system is 123.2 ms, which satisfies the real-time requirement. The system can promptly send out an alarm signal once the battery or electric bicycle enters the monitoring area.

4.3.3 Alarm Reliability Analysis

A total of 100 tests were conducted to evaluate the alarm reliability. Of the 50 tests with batteries or electric bicycles, the system accurately triggered the alarm in all cases without any missed alarms. For the 50 interference tests, the system did not generate any false alarms. This demonstrates that the system has a high level of alarm reliability.

4.3.4 Analysis of Voice Control Function

The voice control function was tested with 20 random voice commands, achieving an identification accuracy of 90%. The system was able to recognize 18 out of 20 commands correctly. However, background noise (especially at 60 dB) slightly affected the performance, causing minor drops in recognition accuracy.

5. Conclusion

This paper presents a preventive monitoring system for indoor battery entry based on an STM32 microcontroller and an OpenMV embedded vision module. By integrating a lightweight deep learning model with a closed-loop perception–decision–action mechanism, the proposed system enables real-time recognition of batteries and electric bicycles and executes automatic intervention when necessary.

Experimental results demonstrate that the system achieves high recognition accuracy, low false alarm rates, and fast response time under various environmental conditions. The combination of model compression, embedded deployment, and multi-sensor collaboration ensures both real-time performance and practical feasibility in resource-constrained environments.

The proposed system integrates image acquisition, neural network inference, environmental sensing, alarm control, and mechanical actuation into a compact and low-cost architecture. Its modular design and low power consumption make it suitable for deployment in residential buildings, office spaces, and other indoor public environments where battery-related fire hazards are of concern.

Future work may further optimize model efficiency and robustness under more complex environmental conditions and explore the extension of the proposed embedded AI framework to other safety monitoring scenarios.

References

[1] Spotnitz, R., Franklin, J. Abuse behavior of high-power, lithium-ion cells. Journal of Power Sources, 2003, 113(1): 81–100.

[2] Doughty, D. H., Roth, E. P. A general discussion of Li-ion battery safety. The Electrochemical Society Interface, 2012, 21(2): 37–44.

[3] Feng, X., Ouyang, M., Liu, X., Lu, L., Xia, Y., He, X. Thermal runaway mechanism of lithium-ion battery for electric vehicles: A review. Energy Storage Materials, 2018, 10: 246–267.

[4] Celik, T., Demirel, H. Fire detection in video sequences using a generic color model. Fire Safety Journal, 2009, 44(2): 147–158.

[5] Ko, B. C., Cheong, K. H., Nam, J. Y. Early fire detection algorithm based on smoke and flame color models. Fire Safety Journal, 2009, 44(3): 322–329.

[6] Toreyin, B. U., Dedeoglu, Y., Cetin, A. E. Wavelet based real-time smoke detection in video. Proc. ICIP, 2005: 213–216.

[7] National Fire Protection Association. NFPA 72: National Fire Alarm and Signaling Code. 2019 Edition.

[8] Howard, A., Sandler, M., Chu, G., et al. Searching for MobileNetV3. Proc. ICCV, 2019: 1314–1324.

[9] Howard, A. G., Zhu, M., Chen, B., et al. MobileNets: Efficient convolutional neural networks for mobile vision applications. arXiv:1704.04861, 2017.

[10] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.-C. MobileNetV2: Inverted residuals and linear bottlenecks. Proc. CVPR, 2018: 4510–4520.

[11] STMicroelectronics. VL53L0X Time-of-Flight Ranging Sensor Datasheet. Rev 2, 2016.

[12] Maxim Integrated. DS18B20 Programmable Resolution 1-Wire Digital Thermometer. Rev 5, 2015.

[13] Shorten, C., Khoshgoftaar, T. M. A survey on image data augmentation for deep learning. Journal of Big Data, 2019, 6: 60.
[14] Han, S., Mao, H., Dally, W. J. Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. Proc. ICLR, 2016.
[15] Jacob, B., Kligys, S., Chen, B., et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference. Proc. CVPR, 2018: 2704–2713.
[16] Reddi, V. J., Cheng, C., Cheng, Y., et al. MLPerf Tiny benchmark. arXiv:2106.07597, 2021.
[17] Han, S., Mao, H., Dally, W. J. Deep compression: Compressing deep neural networks with pruning, trained quantization and Huffman coding. Proc. ICLR, 2016.
[18] Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A. XNOR-Net: ImageNet classification using binary convolutional neural networks. Proc. ECCV, 2016: 525–542.