Sensors and Materials, Vol. 38, No. 4 (2026) 1991–2023
MYU Tokyo

S & M 4420

Radio-frequency-identification-based Specimen Transportation Device Monitoring by Support Vector Machine Method

Wen-Tung Hsu,[1] Jing-Xiang Zhang,[2] Chen-Hsuan Ma,[2] and Cheng-Yu Peng[2*]

[1]Taichung Armed Forces General Hospital
No. 348, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung City 406, Taiwan (R.O.C.)
[2]National Chin-Yi University of Technology,
No. 57, Sec. 2, Zhongshan Rd., Taiping Dist., Taichung City 411, Taiwan (R.O.C.).

(Received December 16, 2025; accepted April 6, 2026)

Keywords: specimen transportation, RFID, embedded system, support vector machine

In this study, we developed a system integration of a mobile intelligent specimen transport box to build a smart specimen reading and transmission device using an embedded system. For environmental monitoring experiments, the sending unit can set sample information and temperature requirements, while the receiving unit can display the arriving sample's information and the temperature recorded during transport. Throughout the delivery process, the system can obtain real-time sample information and temperature data. The data algorithm of support vector machine (SVM)-based event detection is used to issue alerts during data monitoring, ensuring that temperature and humidity are properly controlled during specimen transport.

1. Introduction

Radio frequency identification (RFID) is a rapidly developing wireless communication technology, and its primary function is tag identification. In our society, RFID is widely used across various industrial and commercial systems. Different frequency bands have clearly defined specifications regarding their physical characteristics, communication methods, and application approaches. Because of these standards, the applications derived from different frequency bands also vary. For example, high-frequency (HF)-band RFID is typically used for short-range contactless devices, so it is commonly found in commercial applications such as e-wallets and ticketing systems. In contrast, ultrahigh-frequency (UHF)-band RFID can operate over longer distances and is therefore used in parking lots, positioning systems, highway toll collection, and more. Although there are clear differences in frequency bands and applications, their primary operation and working principles are largely similar—they all utilize electromagnetic induction to enable access permission and perform data transmission. The main topics of interest involve how RFIDs with different antenna frequencies operate in various environments and how to obtain optimal signal stability for different applications.

The major drawback of RFID lies in interference between its components and interference caused by nonconductive materials. To address these technical challenges, Papapostolou and

*Corresponding author: e-mail: peng@ncut.edu.tw
https://doi.org/10.18494/SAM6123

ISSN 0914-4935

Chaouchi[1] explored the suitability of RFID tags for indoor location sensing as an important feature in pervasive computing applications. In this study, the authors investigated how several types of interference affect RFID performance and considered three RFID positioning schemes. By analyzing a case in which deployed tags determine the location of a moving terminal equipped with a reader extension, the authors derived mathematical models describing major interference types and their impact on positioning accuracy and response time. Extensive simulation analyses were conducted to evaluate the practicality and effectiveness of RFID. The final results confirmed RFID's potential in location sensing and emphasized the need for careful design in RFID-based positioning systems. Mahfouz *et al.*[2] proposed a positioning technique combining the received signal strength indicator (RSSI) with the K-means algorithm. In their study, they developed a wireless sensor network localization algorithm based on radio fingerprinting and kernel methods. Using real-world data collected in noisy indoor environments affected by various interferences, the authors analyzed the behavior of different algorithms under realistic conditions.

To date, RFID has become a mature and standardized technology, primarily used in wireless applications involving traceability, logistics, and access control. Duroc and Tedjini[3] elaborated on RFID's inherent advantages, such as unique identification, wireless communication, and low tag manufacturing costs, which provide decisive benefits and drive new developments in concepts and applications. These trends have been validated by market forecasts and by RFID's practical use in various fields. In the first part of their article, they introduced the basic concepts of RFID technology and its relationship with wireless sensing, summarizing current performance and technological advancements. In the second part, they discussed RFID's actual impact on society, with emphasis on applications in autonomous medical care and disability support. In the final section, they presented prospects and directions for future RFID applications. With the rapid adoption of RFID technology, RFID-based sensors have attracted interest across many fields. Owing to the limitations of RFID tags, most existing RFID-based temperature sensing applications require hardware modifications, increasing costs and hindering deployment. Wang *et al.*[4] proposed the RFThermometer, a remote temperature-sensing system using commercial UHF RFID tags. They first examined the effect of temperature on the RFID phase. To mitigate accuracy loss caused by missing phase measurements, a tensor-completion method was introduced, along with a Gaussian process model to construct a phase-temperature map during the offline stage. In the online stage, unknown temperatures were simulated using dynamic time-warping techniques, effectively overcoming RFID's limitations in temperature-sensing scenarios.

Previous research studies[5–11] have shown that a smart specimen transport box integrating RFID, positioning technologies, and IoT offers significant advantages over traditional barcode technology. Existing barcode systems require manual, passive, and single-action scanning, which may lead to missed or lost specimens. RFID, being wireless and contactless, effectively mitigates these issues. To clarify the functional advancements in recent literature, Table 1 systematically shows the dimensions—specifically RFID, positioning, and IoT—achieved by these related studies. In the early stages of integrating these technologies into medical workflows, studies often focused on single functional dimensions. For instance, Shim *et al.*[5]

Table 1
System functions achieved in research studies.

	Application		
	RFID	Positioning	IoT
Shim *et al.*[5]	▓		
Ohashi *et al.*[6]		▓	
Urbano *et al.*[7]		▓	▓
Kurnianingsih *et al.*[8]		▓	
Khan *et al.*[9]		▓	▓
Monteleone *et al.*[10]		▓	▓
Le *et al.*[11]	▓	▓	▓
This study	▓	▓	▓

utilized RFID technology to prevent phlebotomy errors by authenticating patient identities and blood tubes at the point of care. While this system highly optimized the RFID identification process, it did not incorporate continuous cloud monitoring or geographic tracking. To address the need for tracking, Ohashi *et al.*[6] advanced location awareness by proposing a smart medical environment that auto-tracks clinical interventions. By deploying sensors to monitor the real-time presence of patients, nurses, and laboratory specimen tubes, their system fulfilled the positioning dimension in Table 1 to prevent misidentifications, although it remained confined to indoor applications.

As the demand for cross-site medical logistics grew, researchers shifted their focus toward combining geographic tracking with continuous data transmission. Urbano *et al.*[7] developed a monitoring system specifically for the transport of blood components, heavily utilizing IoT architecture and global positioning system (GPS) positioning to ensure the integrity of the cold chain over long distances. Similarly, studies by Kurnianingsih *et al.*,[8] Khan *et al.*,[9] and Monteleone *et al.*[10] further explored the integration of positioning and IoT technologies within medical tracking and cold chain logistics. These studies successfully established a solid foundation for connected healthcare infrastructure; however, they primarily focused on macro-level environmental tracking rather than item-level RFID verification during the handover process.

Achieving a fully comprehensive system requires the simultaneous integration of all three dimensions: RFID, positioning, and IoT. Le *et al.*[11] successfully proposed a smart specimen transport box that combines these elements. On the sending side, an RFID tag is attached to each specimen tube, and a reader writes patient information alongside the GPS coordinates of the receiving laboratory. For the IoT and positioning components, an Arduino Mega 2560 is installed inside the transport box with an SHT10 temperature sensor and a GPS module. Finally, narrow band internet of things (NB-IoT) is used to upload real-time sensor and geographic data to an IoT cloud platform, allowing users to view specimen status instantly. Upon arrival, an RFID reader retrieves the stored data and compares it with the GPS data to verify correct delivery. In recent years, the paradigm of medical logistics has rapidly evolved toward "Smart Healthcare IoT" and intelligent anomaly detection. For instance, Sarkar *et al.*[12] explored the integration of the Internet of Medical Things (IoMT) to achieve autonomous and proactive healthcare monitoring. To address the computational challenges of real-time monitoring, Cruz Castañeda

and Bertemes Filho[13] and Hennebelle *et al.*[14] proposed AI-driven Edge-IoT architectures that distribute machine learning tasks for predictive smart healthcare. Within the specific domain of cold chain logistics, Gillespie *et al.*[15] developed a real-time IoT anomaly detection system to identify equipment failures during the transport of temperature-sensitive goods. Furthermore, Mirkhail and Xinyou[16] demonstrated the efficacy of deep learning methods for anomaly detection in securing IoT healthcare systems. While these recent studies highlight the growing trend of deploying artificial intelligence within Smart Healthcare IoT, our proposed system builds upon these foundations by strategically adopting a centralized, server-side SVM architecture. This design leverages the robust data transmission capabilities of modern IoT to perform highly accurate dynamic 1D time-series anomaly detection, effectively moving beyond traditional static thresholds without imposing computational burdens on the edge transport boxes.

While the system proposed by Le *et al.* provides a complete tracking framework across all three dimensions, as shown in Table 1, it still relies on traditional, static threshold-based conditions for environmental alerts. This study builds upon the complete integration of RFID, positioning, and IoT by introducing an embedded system. By integrating a support vector machine (SVM) algorithm, the proposed system is able to learn dynamic 1D time-series waveform features, replacing rigid static thresholds with highly accurate anomaly detection and instant push notifications.

For this work, the smart specimen transport box integrates an embedded system with RFID communication technology to create a self-developed mobile IoT-based specimen delivery device. On the sending side, the system allows users to configure specimen information and set temperature monitoring requirements for the transport box. Upon arrival at the destination, the receiving side can display the delivered specimen information along with the recorded temperature history during transportation. In addition, throughout the delivery process, the system can provide real-time access to both specimen data and the temperature conditions experienced during transit.

2. Research Method: Using RFID Technology as a Smart Specimen Reading and Transmission Device

2.1 Construction of an embedded system for a smart specimen transport box

A smart specimen transport box was developed on the basis of an embedded system and RFID communication technology. This self-designed mobile specimen transport box incorporates intelligent IoT communication. The temperature monitoring system is divided into a specimen sending unit, a specimen receiving unit at the destination, and real-time specimen information during transport. This ensures full temperature control throughout the specimen transport process, as shown in Fig. 1. The planned implementation methods and functions of the specimen transport box are as follows.

A self-made mobile specimen transport box that integrates embedded systems with intelligent IoT communication is developed. The specimen sending unit can set sample information and the

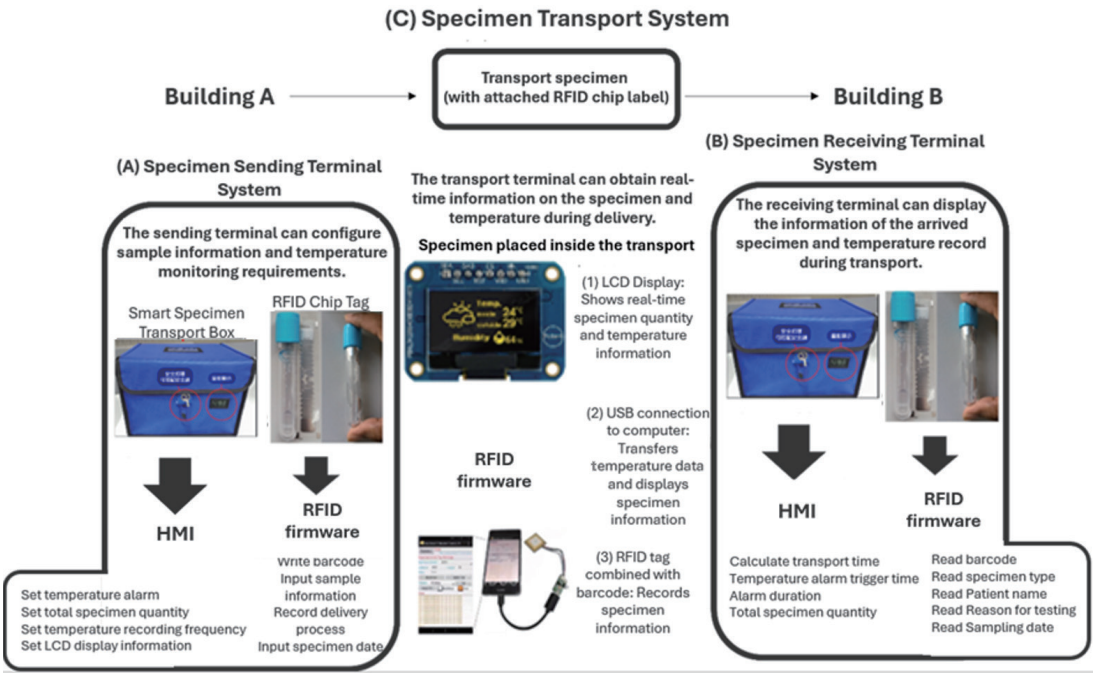


Fig. 1. (Color online) Implementation method and functions of the transport box.

required temperature monitoring parameters. Upon arrival, the specimen receiving unit displays the received sample's information and temperature records during transport. During transport, the system can provide real-time specimen information and temperature status to ensure complete temperature control throughout the process. The planned implementation steps of the self-made smart specimen reading and transmission device are as follows.

Prepare RFID sensing chips, which may be designed as adhesive tags or water-resistant types, for storing specimen data. Integrate an embedded system inside the specimen transport box. A computer-based human–machine interface retrieves specimen quantities and temperature information from the memory module. The hardware is then combined with firmware design to develop the software monitoring system for the transport box. The system includes the following:

(A) Specimen sending-end system: The sending unit can set sample information and temperature monitoring requirements.
(B) Specimen receiving-end system: The receiving unit displays the arrived specimen's information and the temperature experienced during transport.
(C) Specimen transport-end system: The system displays real-time sample information and temperature conditions during transport to ensure precise temperature control.

Figure 1 illustrates the implementation method and functions of the transport box.

2.2 IoT-based smart temperature monitoring system

The IoT-based temperature monitoring system is divided into a specimen sending unit, a specimen receiving unit at the destination, and real-time specimen information during delivery,

ensuring accurate temperature control throughout the process, as shown in Fig. 2. To develop a smart specimen transport box that ensures temperature management during specimen movement, in this project, we focused on system integration for specimen monitoring. The system aims to achieve the following: (1) a mobile specimen transport box transmission system and (2) a specimen tracking system that accompanies the transport box. The operational functions of both the transport box and the specimen tag are described below.

(A) Specimen sending-end system: The sending unit can set sample information and temperature requirements.
→A sending-end monitoring and configuration software system will be developed to integrate with the transport box and RFID devices, enabling temperature parameter settings and specimen data writing.

(B) Specimen receiving-end system: The receiving unit can display the arrived specimen's information and the temperature recorded during transport.
→A receiving-end data analysis software system will be developed to integrate with the transport box and RFID devices, exporting temperature monitoring data and reading specimen information.

(C) Specimen transport-end system: The system provides real-time access to sample information and temperature during transit.
→A transport-end monitoring software system will be created to coordinate with the transport box and display monitoring status via that liquid crystal display panel, RFID, mobile device, computer, and microcontroller interface.

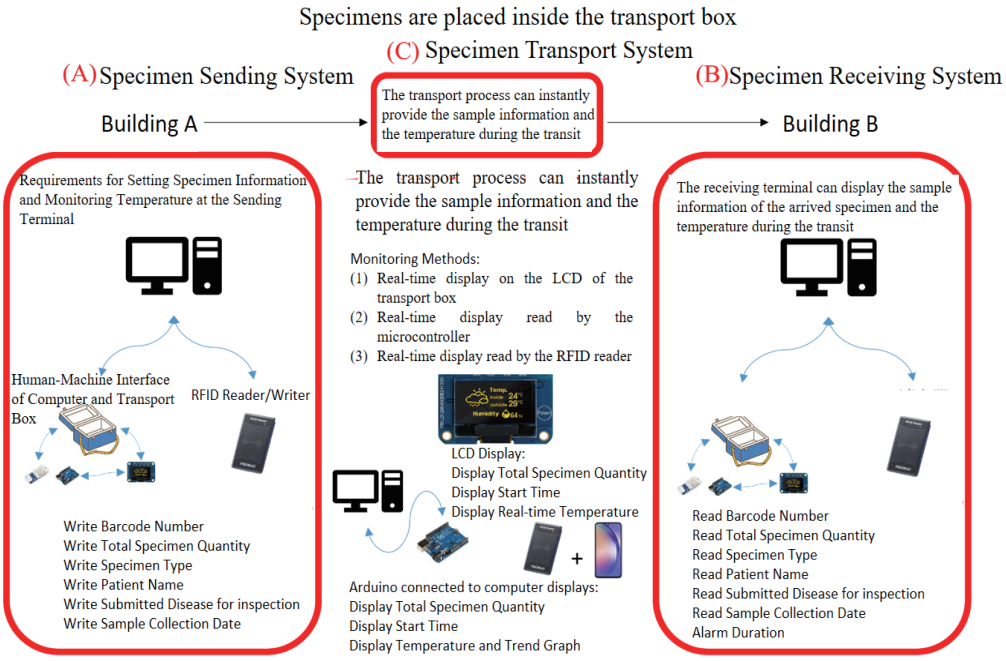


Fig. 2. (Color online) IoT-based smart temperature monitoring system.

2.3 Description of research device and materials for smart specimen transport box

The TTGO T18 V3.0 motherboard[17] was selected for this system on the basis of three main considerations. First, it is equipped with an ESP32-WROOM-32 MCU, which offers strong computing capability and expandability, and it includes a built-in Wi-Fi module that fully meets the wireless communication needs of this project. Second, the board features a built-in 18650 lithium battery holder with integrated boost and voltage-regulation circuitry, enabling long-term operation while reducing the cost and complexity of external power modules. Finally, its I/O expandability is excellent, providing three UART interfaces, two I²C channels, and two SPI channels, allowing flexible integration with various sensors and peripheral devices. The system uses the DHT22 temperature and humidity sensor[18] for environmental measurement owing to its high accuracy, stability, and ease of use. The DHT22 includes an internal digital signal processing unit capable of measuring both temperature and relative humidity, and outputs digital data through an I²C interface, greatly simplifying its connection with the microcontroller.[19] The DHT22 relies on a capacitive humidity sensing element and a negative temperature coefficient (NTC) thermistor. The capacitive element detects changes in dielectric constant caused by water molecule adsorption, whereas the NTC thermistor measures temperature changes through inversely proportional resistance variations. These analog signals are then converted by the internal microprocessor. It supports a temperature range of –40 to +80 °C and a humidity range of 0–100% RH, meeting the needs of typical environmental monitoring. In addition, its low power consumption and long-term stability make it ideal for continuous monitoring or battery-powered systems.

To ensure precise and stable positioning during transport, the system adopts the u-blox NEO-M8N 8th-generation multi-global navigation satellite system (GNSS) module, which supports BeiDou. Compared with modules supporting only a single GNSS system, the NEO-M8N can simultaneously receive signals from GPS, GLONASS, Galileo, and BeiDou, providing superior positioning accuracy and stronger anti-interference capability.[20] Its advantages in update rate and time-synchronization accuracy also enhance the timeliness and reliability of transmitted data. For identification, the system uses the RC522 RFID module,[21] selected for its low cost, compact size, and easy integration with microcontrollers. It supports 13.56 MHz contactless cards and provides fast, stable identification performance.

The system includes a GME12864-77Y organic light-emitting diode (OLED) display (1.3”, 128 × 64 resolution, I²C communication, SH1106 driver),[22] chosen for its clear display, low power consumption, and compact size—ideal for embedding into the smart specimen transport box as a real-time information interface. Through the OLED screen, users can directly view internal temperature/humidity, operating mode, and other system status information, improving usability and monitoring efficiency. Three operating modes were designed to support varying specimen preservation requirements. The current mode is displayed on the screen so operators can ensure specimen quality and safety during transport.

(1) Room temperature mode: Suitable for general specimens; temperature maintained between 24 and 28 °C, and humidity below 60%.
(2) Refrigerated mode: For low-temperature specimens; temperature maintained between 5 and 7 °C, and humidity below 60%.

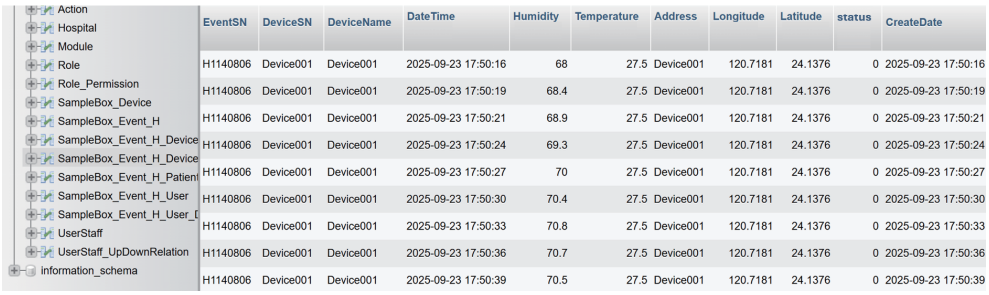
(3) Frozen mode: For frozen specimens; temperature maintained between –18 and –12 °C, humidity below 60%.

The system enclosure was designed using Autodesk Inventor. The structure consists of one main body and four external covers. The main body houses all circuit boards and wiring, providing structural support, while the covers protect and secure the electronic components. All parts were 3D-printed using polylactic acid (PLA) material. The covers' mounting points incorporate M3 heat-set threaded inserts embedded in the main body to enhance fastening strength and structural stability. The system uploads sensing data to the cloud database via an application programming interface (API), using the JSON format for easy parsing and storage by the server. The data transmitted by the device include basic device information, real-time sensor data, and system status parameters. The major fields are listed below.

API Data Upload Fields
DeviceId: Device identification code
Temperature: Real-time temperature (°C), rounded to one decimal place
Humidity: Real-time related humidity (RH), rounded to one decimal place
Longitude/latitude: Geographic coordinates of the device
Timestamp: Time when the data were generated
EventSN: Event serial number, the unique identifier of each record
DeviceSN/DeviceName: Device serial number and name (linked to registered device information)
Address: Text description of device location
Status: Device operational status code
1 = active or alert state
0 = normal
CreatUser: Identifier of the uploader or system account
CreateDate: Time the data were uploaded to the server

This API is responsible solely for uploading and writing data. It does not perform logical processing or decision-making. After transmission, the server stores the data in the SampleBox_Event_H_Device_Status table within the Hospital database for later querying, analysis, and visualization—corresponding to the SQL database operations described later. All recorded sensor and event data can be downloaded from the backend database. Users may export data as .csv files for analysis or .xlsx files for easier organization and review as shown in Fig. 3.

A SQL database was implemented as the core of the web system's front-end and back-end data management. The database is managed and accessed through phpMyAdmin; its API is responsible solely for the receiving data sent from the front end and writing it into the database, without performing additional data processing. All data are stored in a database named Hospital, within which the table SampleBox_Event_H_Device_Status records all device operations and status information. The table currently contains the following fields: EventSN, DeviceSN, DeviceName, DateTime, Humidity, Temperature, Address, Longitude, Latitude, Status, CreateDate, and CreateUser. These fields comprehensively represent each device's historical operating records and environmental monitoring data, corresponding to the API data structure described earlier.



EventSN	DeviceSN	DeviceName	DateTime	Humidity	Temperature	Address	Longitude	Latitude	status	CreateDate
H1140806	Device001	Device001	2025-09-23 17:50:16	68	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:16
H1140806	Device001	Device001	2025-09-23 17:50:19	68.4	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:19
H1140806	Device001	Device001	2025-09-23 17:50:21	68.9	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:21
H1140806	Device001	Device001	2025-09-23 17:50:24	69.3	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:24
H1140806	Device001	Device001	2025-09-23 17:50:27	70	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:27
H1140806	Device001	Device001	2025-09-23 17:50:30	70.4	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:30
H1140806	Device001	Device001	2025-09-23 17:50:33	70.8	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:33
H1140806	Device001	Device001	2025-09-23 17:50:36	70.7	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:36
H1140806	Device001	Device001	2025-09-23 17:50:39	70.5	27.5	Device001	120.7181	24.1376	0	2025-09-23 17:50:39

Fig. 3. (Color online) SQL database operations of the smart temperature monitoring system by phpMyAdmin.

2.4 Support vector machine theory and classification algorithm

The objective of SVM is to perfectly separate data belonging to different classes and to identify a decision boundary that maximizes the margin between two classes. The decision boundary is not necessarily linear; in many cases, it can be nonlinear. The following SVM equations are derived based on existing literature.[23] The SVM algorithm assumes perfect reparability and attempts to find an optimal decision boundary (hyperplane) under this assumption by adjusting parameter weights so that the margin between two classes is maximized. We first assume the hyperplane equation

$$W^T X + b = 0 \cdot \tag{1}$$

Here, W is the normal vector to the hyperplane, which is perpendicular to the plane. Therefore, the unit normal vector is $\frac{W}{|W|}$. Consider a data point located at X and another arbitrary point X_0 on the hyperplane. The vector from X_0 to X is $X - X_0$. The projection of this vector onto the unit normal vector gives the shortest distance from the data point to the hyperplane:

$$d = \frac{\left|W \cdot \left(X - X_0\right)\right|}{|W|}. \tag{2}$$

Since X_0 lies on the hyperplane, it satisfies $W^T X_0 + b = 0$, and substituting yields

$$d = \frac{|W \cdot X + b|}{|W|}. \tag{3}$$

For a set of linearly separable binary data, we want the hyperplane to be as far away from both classes as possible, creating a margin that contains no data points. Let the boundaries of this margin be

$$W^T X + b = \pm 1 \cdot \tag{4}$$

By appropriately scaling W and b, the hyperplane equation remains unchanged, $W^T X + b = 0$. From the distance formula above, the margin between the hyperplane and the boundary is

$$margin = \frac{1}{|W|}. \tag{5}$$

Data points that lie exactly on the margin boundary satisfy

$$y_n \times \left(W^T X_n + b\right) = 1 \quad (\text{Support vector}). \tag{6}$$

For all data points, the constraint becomes

$$y_n \times \left(W^T X_n + b\right) \geq 1 \quad (\text{All data}). \tag{7}$$

In summary, the goal of Hard-Margin SVM is to maximize the margin ($\frac{1}{|W|}$) while satisfying the constraint $y_n \times (W^T X_n + b) \geq 1$, $n = 1$. This is equivalent to minimizing ($\frac{W^T W}{2}$). A kernel function enables an implicit mapping into an infinite-dimensional feature space $Z_n(X_n) \times Z_m(X_m)$. Instead of explicitly performing a nonlinear transformation from the original space X to a new space Z, the kernel computes the inner product directly.

$$K\left(X_n, X_m\right) = Z_n(X_n) \times Z_m(X_m) \tag{8}$$

This avoids the computational complexity associated with explicit feature transformations. Common kernel functions include the following:

$$\text{Linear:} \quad K\left(X, X'\right) = X^T X, \tag{9}$$

$$\text{Polynomial:} \quad K_Q\left(X_n, X_m\right) = \left(\zeta + \gamma {X_n}^T + X_m\right)^Q, \tag{10}$$

$$\text{Gaussian (RBF):} \quad K\left(X_n, X_m\right) = \exp\left(-\gamma \left|X_n - X_m\right|^2\right). \tag{11}$$

The Gaussian kernel provides infinite-dimensional nonlinear transformation capability using only the original features. Physically, the kernel represents the "similarity" (inner product) between two vectors in the transformed space. When two vectors are identical ($X_n = X_m$), the Gaussian kernel yields a maximum value, indicating high similarity.

Under the constraints $a_n \geq 0$ and $\sum a_n y_n = 0$, the kernel hard-margin SVM solves for the Lagrange multipliers a_n by minimizing the dual objective function. Specifically, this optimization problem is defined to minimize the equation $\left[\frac{1}{2}\sum_n\sum_m \alpha_n\alpha_m y_n y_m K\left(X_n, X_m\right) - \sum_n \alpha_n\right]$.

Once a_n values are obtained, the hyperplane parameters W and b can be computed. Only support vectors contribute ($a_n > 0$).

$$W = \sum\nolimits_{n=sv} \alpha_n y_n Z_n \tag{12}$$

$$b = y_{sv} - \sum\nolimits_n \alpha_n y_n K\left(X_n, X_{sv}\right) \tag{13}$$

Simulated results show that the decision boundary changes with different $K\left(X_n, X_m\right) = \exp\left(-\gamma\left|X_n - X_m\right|^2\right)$ values.

However, the hard-margin SVM is highly susceptible to overfitting because it does not tolerate noise. Therefore, the soft-margin SVM is introduced. The soft-margin SVM introduces slack variables ξ_n, which represent the degree to which a data point violates the margin constraints. These violations are incorporated into the cost function. A penalty parameter C controls tolerance to noise: a large C presents low tolerance (strict classification), and a small C presents high tolerance (more flexible).

The optimization problem becomes $\left[\frac{1}{2}\sum_n\sum_m \alpha_n\alpha_m y_n y_m K\left(X_n, X_m\right) - \sum_n \alpha_n\right]$.

This is identical to the hard-margin SVM except a_n is now upper-bounded by C.

The soft-margin SVM with Gaussian kernels can tolerate classification errors while avoiding overfitting when parameters are selected appropriately. To evaluate model performance, binary classification metrics are introduced before extending them to three-class classification. In the confusion matrix:

True Positive (*TP*): Model predicts “yes”, and true label is “yes”.

True Negative (*TN*): Model predicts “no”, and true label is “no”.

False Positive (*FP*): Model predicts “yes”, but true label is “no”.

False Negative (*FN*): Model predicts “no”, but true label is “yes”.

Accuracy is defined as

$$\text{Accuracy:} \quad \frac{TP + TN}{TP + TN + FP + FN}. \tag{14}$$

It is evident that the Gaussian kernel with the soft-margin SVM significantly improves classification accuracy compared with linear and polynomial kernels. Remaining errors are due to extremely close raw data points. Improving the training dataset can further enhance accuracy.

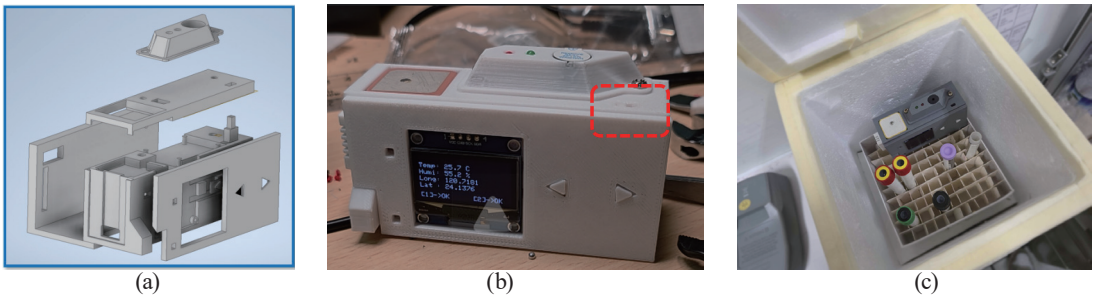
3. Operational Efficiency Analysis of the Smart Specimen Transport Box

3.1 System architecture and design workflow of the smart specimen transport box

The device box shown in Fig. 4 uses the TTGO T18 V3.0 (ESP32 development board) as its core, powered by an independent 18650 lithium battery (3200 mAh). On the hardware input side, the device is equipped with a DHT22 temperature and humidity sensor, a NEO-M8N GPS module, and an RC522 RFID module, which are used for environmental monitoring, location tracking, and identity verification during handover. All sensor signals are processed and integrated by the development board. The output components include a 128×64 OLED display, LED indicators, and a buzzer, which provide information display and on-site alert functions. Operation buttons support mode switching. All circuits and sensors are integrated into a 3D-printed PLA enclosure designed with one main body and four cover pieces. Sensors are mounted in fixed positions on the main body, with wiring routed through openings to the TTGO T18 V3.0. Once the covers are attached from front to back, all components are securely fixed. The 3D-printed enclosure emphasizes durability and the prevention of accidental activation.

- The power button is flush with a recessed groove to avoid unintended shutdown during transport.
- Operation buttons follow a fool-proof interaction design (double-click to switch modes, long-press to enter settings, short-press to adjust, long-press to apply), reducing the risk of operational errors that can disable alarms.

This design ensures continuous measurement and reliable alerts in real transport scenarios as shown in Fig. 5, supporting the research goals of traceability and real-time warning capability. The web front end includes four major functional tabs for administrators: (1) Basic settings: Add hospital personnel and devices to the database for later event binding. (2) Permission management: Adjust hierarchical access levels to prevent unauthorized personnel from viewing confidential data. (3) Event settings: Create new events by specifying time, location, personnel, and devices. All parameters related to specimen and device boxes are configured here. (4) Scan management: Require personnel identity verification. Once authorized, users may add new specimens and edit medical record information.



(a) (b) (c)

Fig. 4. (Color online) (a) Explosion diagram, (b) device, and (c) specimen transport box.

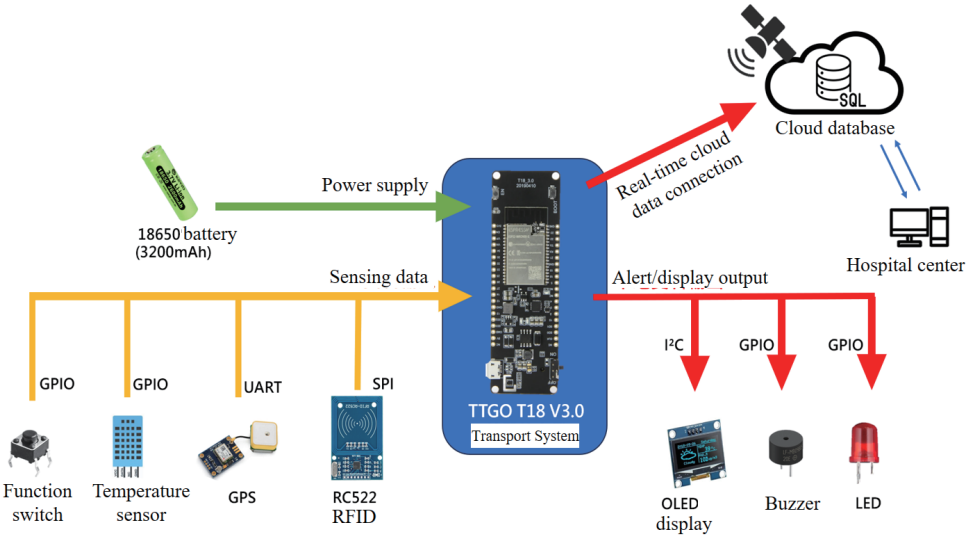


Fig. 5. (Color online) System operation structure of specimen transport box.

To clearly define the system's structural contribution, it is essential to emphasize that the proposed architecture operates as an "event-driven" monitoring system, rather than a mere passive data logger. Traditional data recording systems passively log environmental metrics (e.g., time, temperature, and humidity) without contextualizing the logistics workflow. In contrast, our system's web frontend, backend, and SQL cloud database are structurally driven by "Events". Upon initiating a transportation task, the system generates a unique Event Serial Number (EventSN). All subsequent logistical processes—including device binding (e.g., assigning a specific device to an event), personnel authorization, and handover procedures (e.g., transferring custody from one personnel to another)—are dynamically linked to this specific EventSN. This event-driven architecture ensures that every piece of uploaded sensor data is strictly contextualized within a verifiable chain of custody, transforming raw data recording into a comprehensive, accountable, and traceable medical logistics management system.

During event creation, the system automatically generates a unique event identification (ID), which can later be queried through the SQL database. Users must specify the event start time, end date, and event name; modifications are restricted to authorized users to prevent data errors by junior staff. The event description field allows personnel to record notes such as discrepancies between expected and actual specimen counts. Related personnel and devices may then be added, and after specimen collection, both the device and specimen boxes are placed together for transfer using the "Transfer Device" function. Before transfer, the "Add Patient" function allows users to input medical record data and notes to ensure that each specimen is properly documented prior to handover and transport.

The interface also provides a Device Status section where users can view real-time temperature, humidity, GPS location, and the travel path of each device, allowing the full monitoring of environmental conditions during transport. The specimen handover process is designed to ensure specimen traceability and clear accountability. Starting from event creation, the workflow includes adding personnel and devices, completing specimen collection, entering

medical record data, binding the device box with the specimen box, placing the specimen into the box, adding notes, and beginning the formal handover process. Handover can occur multiple times as needed until the specimen reaches the target hospital. To support flexibility across various operational scenarios, the system provides three types of operation interface (web via PC and mobile, LINE@ via mobile, and mobile app) and two scanning methods (RFID and QR Code). On the web interface, the system is already linked to the user account; handover is completed by scanning the personnel badge via RFID or a QR code scanner. For LINE@, the user must first bind the account; selecting "Start Handover" activates the device box RFID module, and scanning the personnel badge completes the update. The mobile app allows near-field communication (NFC) or QR code scanning, with authentication completed at first login. All handover records are stored automatically to ensure complete traceability.

To evaluate process efficiency, ten participants were divided into two groups of five to compare the traditional paper-based workflow with the proposed system. The paper-based method required manual recording before and after each step, whereas the proposed system used Google Forms with preset items and RFID-assisted automatic data filling. Results showed approximately 10% improvement in operational efficiency, with automation reducing the risk of missing records. To further highlight the operational advantages of the proposed system, it is essential to compare it with existing barcode-based tracking methods. As noted by Le *et al.,*[11] existing barcode systems require manual, passive, and single-action scanning, which may lead to missed or lost specimens. In the context of medical cold-chain logistics, printed barcodes are also highly susceptible to physical damage or becoming unreadable owing to surface condensation in low-temperature environments. Furthermore, employing a barcode system in the proposed smart transport box presents a practical challenge during the handover phase. Using barcodes would necessitate opening the transport box lid to scan each specimen individually. This continuous exposure to external ambient air would induce severe internal temperature and humidity fluctuations, directly disrupting the strict environmental integrity that our SVM model is designed to monitor. Therefore, as emphasized by Le *et al*.,[11] the integration of RFID provides a significant advantage over barcode systems because it is wireless and contactless. RFID not only improves operational efficiency by enabling simultaneous "blind batch reading" without opening the box, but it also ensures zero thermal disruption during the handover process, strongly reinforcing the system's reliability in medical specimen transportation. A short-distance walking test (laboratory → convenience store → return) was conducted to demonstrate workflow fluency and basic measurement continuity. Without ice packs, temperature increased outdoors, stabilized in air-conditioned environments, and rose again upon return—confirming successful real-time monitoring. The front end displays real-time temperature-humidity curves and GPS trajectories; when thresholds are exceeded, the device LED turns red and the buzzer activates. The database simultaneously marks the record as an abnormal event for auditing and traceability. In this dynamic transport experiment, data were primarily transmitted via a Wi-Fi connection provided by the experimenter's mobile hotspot (HTTP/HTTPS POST → PHP API → MySQL). While the system architecture is fully equipped to support NB-IoT modules for wide coverage and low power consumption, Wi-Fi was utilized as the current practical solution due to telecommunication regulations in Taiwan, which restrict

the purchase of NB-IoT subscriber identity module (SIM) cards to specific enterprise identities and lack permanent data plans.

The database structure hospital.Hospital includes deviceId, temperature, humidity, longitude, latitude, timestamp, and state (0 = normal, 1 = abnormal), with a composite index (deviceId, timestamp) for efficient event queries and time-series visualization. The system supports three monitoring modes (room temperature 24–28 °C, refrigerated 5–7 °C, frozen −18 to −12 °C) and a unified humidity limit of 60% RH. When temperature or humidity exceeds limits, the device flags an abnormal event, and the front-end query interface allows data export (CSV, TXT). To balance resolution and server load, the device uploads data every 3 s (adjustable). This maintains near-second-level visibility while reducing data volume by 66.7%, supporting the research goals: traceability, supervision, and immediate response.

The abnormal alert flowchart of the system in Fig. 6 shows that the mechanism relies on temperature and humidity data collected by the device box with three monitoring modes based on specimen requirements (room temperature 24–28 °C, refrigerated 5–7 °C, frozen −18–−12 °C). When the temperature exceeds the mode's range or the humidity exceeds 60% RH, the system identifies it as an abnormal condition. During monitoring, if an anomaly is detected, the blinking green LED changes to a steady red light, and the buzzer emits a high-frequency alarm. Meanwhile, the device continues uploading full sensor data to the server API. In the SQL database, the status field (State) switches from "0" to "1" to mark the event as abnormal. Users can later review these records via the "Event Query" or "Abnormal Event Query" functions and export the data for further analysis and reporting.

3.2 Long-distance transportation specimen temperature and humidity monitoring analysis

3.2.1 Experimental scenario setup and environmental switching analysis

In this experiment, we simulated the long-distance transportation of medical specimens, incorporating different transportation modes and environmental conditions. The long-haul

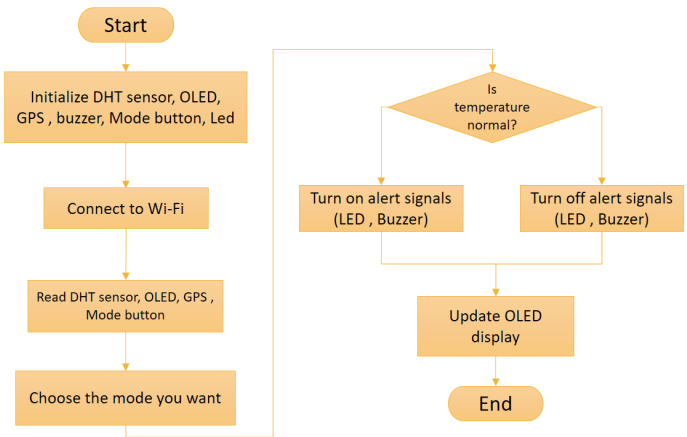


Fig. 6. (Color online) The abnormal alert flowchart relies on setting the temperature and humidity.

transportation and handover scenario was planned from the Industrial Technology Research Institute (ITRI) (Hsinchu, Taiwan) transferring between motorcycle and train, and ultimately arriving at National Chin-Yi University of Technology (NCUT) (Taichung, Taiwan). The experiment was intentionally conducted on a rainy day so that both rainfall and gusty wind introduced strong environmental humidity fluctuations, thereby amplifying environmental disturbances suitable for evaluating the device's sensing performance under real-world conditions. The mobile environmental monitoring device served as the core measurement system, with the primary objective of observing humidity variations across different environments (e.g., air-conditioned indoors, outdoor heavy rain, and train compartments). Figure 7 shows the humidity analysis for long-distance transportation specimens to reveal three environment variations (variations 1–3).

The experiment began at the ITRI dormitory in Hsinchu. The device was activated at 17:50 inside an air-conditioned indoor environment at approximately 27.5 °C. At 17:53, upon stepping outdoors into the rainy environment, the device was immediately exposed to rapidly increasing humidity conditions, resulting in a significant rise in measured humidity. At 18:41, the device entered a local train at Hsinchu Station. Owing to the train's air-conditioning system, humidity gradually decreased; brief humidity spikes were observed each time the train stopped and the doors opened, presumably caused by the rapid influx of moist outside air. The air-conditioning system then progressively restored humidity to a stable level. During train travel, environmental changes were relatively minor, producing smooth humidity adjustments. At 19:47 upon arriving in Taichung, the device was again exposed to heavy rain, causing another rapid humidity increase. At 19:51, the device was transported by motorcycle to NCUT and finally reached a non-air-conditioned indoor space at 20:23, where measurements stabilized.

For Fig. 8, this sequence of transitions—air-conditioned indoors → outdoor rain → air-conditioned train → heavy-rain exposure → non-air-conditioned indoors—provided a multilayered environmental cycle for examining sensor responsiveness, delay effects, and hysteresis. The data clearly showed substantial humidity elevation in outdoor heavy rain and gradual reduction in air-conditioned environments.

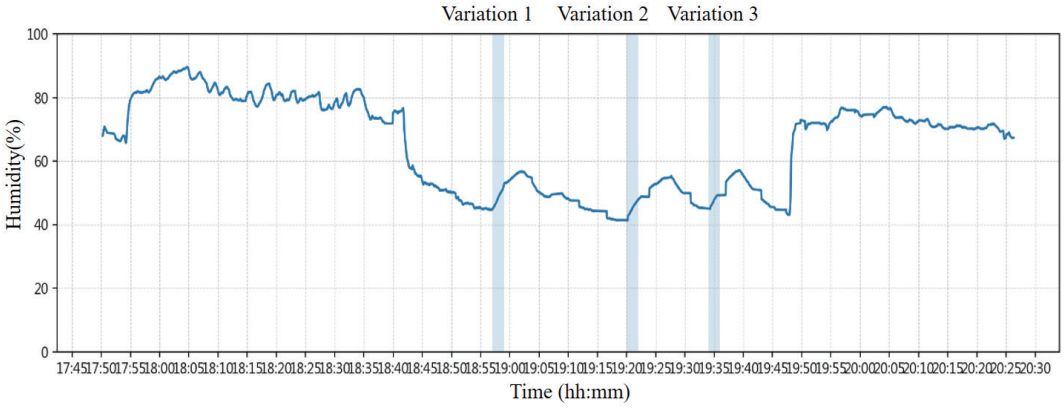


Fig. 7. (Color online) Humidity analysis for long-distance transportation specimens.

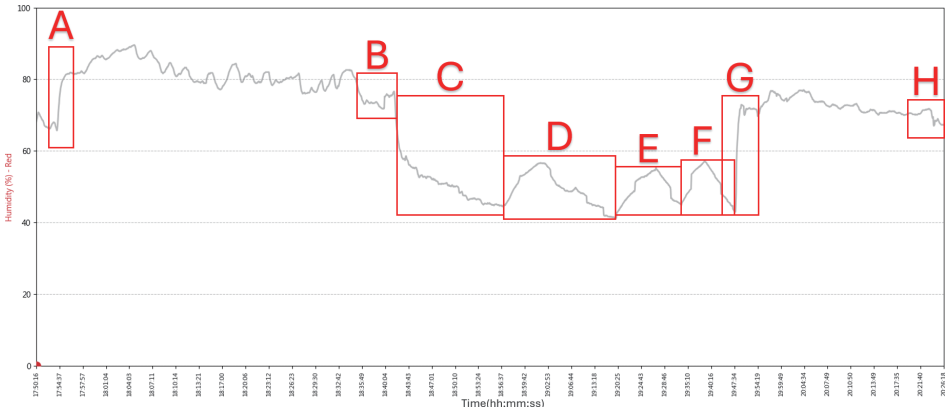


Fig. 8. (Color online) Scenario for the conditions of long-distance transportation specimen.

The plotted charts highlight humidity-change segments corresponding to train door openings to illustrate the impact of environmental switching. Throughout the trip, the device continuously uploaded humidity and GPS data, and the experimental schedule of environmental switching is shown in Table 2. During outdoor movement in the rain, relative humidity repeatedly approached saturation, while platform and train entry points revealed distinct environment switching. Time-series inspection confirmed multiple upward–downward humidity pulses consistent with the "door opening" events. To validate that these patterns resulted from air-mass exchange rather than random noise, a controlled experiment was conducted. The same monitoring device was placed inside a household refrigerator to simulate the low-temperature, low-humidity conditions of a train compartment. The refrigerator was located in an indoor environment (~26 °C) with humidity similar to the previous rainy-day outdoor condition. The B1–B3 regions in the chart correspond to three 2-min door-opening intervals. When the refrigerator door opened, humidity instantly increased and smoothly decreased after the door closed. When all segments were aligned at t = 0 and normalized to a 0–1 amplitude scale, their peak timing, rising slope, and relaxation behavior closely matched those observed in the real-world train data. This strong correspondence confirms that the humidity pulses captured in the cross-city experiment are reliable and reproducible physical phenomena, and the device is capable of detecting short-term disturbances such as train door openings.

After examining the one-dimensional time-series data, humidity curves showed characteristic "spike-and-decay" cone-shaped waveforms. Visual inspection alone, however, was insufficient to determine whether these waveforms were consistent and quantifiable.

3.2.2 Analysis of the physical characteristics of abnormal waveforms

(1) Environmental delay and thermal inertia: According to the official DHT22 datasheet, the sensor features excellent resolution (humidity 0.1% RH, temperature 0.1 °C) and repeatability (humidity ±1% RH, temperature ±0.2 °C). The phase lag and smoothing phenomena observed in the actual measurements are not numerical errors from the sensor itself. Instead, they are physical delays generated during environmental transitions, which are limited by the thermal insulation materials of the transport box and the internal air thermal resistance.

Table 2
Experimental schedule of environmental switching for the scenario conditions.

Time	Location/Action	Environmental conditions	Notes
17:50	dormitory power-up	indoor air-conditioned environment ~27.5 °C	initial state
17:52	leave dormitory and go outdoors	rainfall environment begins	(A) start exposure to high humidity environment
18:36	arrive at Hsinchu station rear exit, prepare to enter underpass	outdoor rain to semi-enclosed space	(B) semi-enclosed space during heavy rain
18:39	walk onto Hsinchu station platform	platform is an open environment	humidity gradually recovers/rises
18:41	enter train	enter air-conditioned compartment environment	(C) prepare to compare device readings with train display values
18:42	train departs from Hsinchu station	train instrument: 23.8 °C / 50.9% RH	(C) temperature and humidity difference shows the device has hysteresis and delay
18:56	arrive at Zhunan station	train doors open	(D) train doors open for 2 min, humidity increases
18:58	depart from Zhunan station	train doors close	(D) train doors close, air conditioning causes humidity to drop
19:08	arrive at Miaoli station	train doors open, train instrument: 23.1 °C / 51.5% RH	(E) train doors open for 2 min, humidity increases
19:10	depart from Miaoli station	train doors close	(E) train doors close, air conditioning causes humidity to drop
19:35	arrive at Fengyuan station	train doors open	(F) train doors open for 2 min, humidity increases
19:36	depart from Fengyuan station	train doors close	(F) train doors close, air conditioning causes humidity to drop
19:47	arrive at Taichung station, get off	experimenter gets off, device moves directly into outdoor heavy rain environment	(G) high-humidity rapid change time period
19:51	ride scooter to NCUT	continuous exposure to heavy rain	significant impact on housing moisture adsorption and sensing film hysteresis
20:23	arrive at NCUT (non-air-conditioned indoor)	naturally ventilated indoor environment	(H) enter stable observation phase

(2) Enclosure moisture absorption effect: During the design phase, the system mounted the sensor on the outermost surface of the 3D-printed enclosure, exposing it directly to airborne water molecules. However, when the transport box is moved from a high-humidity outdoor environment into a dry train compartment, the water molecules attached to and permeated within the "inner walls of the box" continuously evaporate into the air. This microscopic phase-change process inevitably elongates the humidity decay curve significantly.
(3) Sensing film hysteresis effect (hysteresis): The capacitive humidity sensing element of the DHT22 utilizes a polymer film. Because the film adsorbs water molecules rapidly, while the desorption (dehydration) process is relatively slow due to internal diffusion resistance, this inherent hysteresis characteristic of the material causes the measured waveforms to exhibit a severe "steep-rise and slow-decay" asymmetry.

3.3 Training samples and dataset construction

3.3.1 Dataset construction and data augmentation

In this study, our final experimental dataset consists of 40 sets, 10 of which were generated from completely repeated field experiments under identical fixed train schedules, routes, and rainy environments. To expand the original 10 sets converted into 40 training samples, we generated the additional extra 30 data sets by utilizing Python signal processing libraries (such as NumPy and SciPy) through three computational operations. During these field experiments, the transport box was consistently placed on a seat next to the train door. These 10 sets of field measurement data successfully captured the waveform characteristics of indoor/outdoor humidity variations and humidity spikes caused by train door openings. However, considering the substantial time and economic costs associated with frequent cross-city physical testing, and to prevent the machine learning model from overfitting due to overly homogenous samples, these significant variations were introduced to enhance the robustness of the training set.

The three operations included the following: (1) baseline shifting, which vertically shifted the generated data on the Y-axis while maintaining the original shape and variation trends of the feature waveforms; (2) time warping, which randomly extended or compressed the duration by 10 to 15%, thereby altering the gradual slope and time span of the waveforms; and (3) Gaussian noise injection, which randomly inserted minute values into the overall data to strengthen the model's tolerance to boundary spike noise.

Regarding data scale, the data trends of train arrivals were aligned on the basis of the railway arrival schedule. After time-axis alignment, the leading and trailing portions of the data were truncated using the duration of the first experiment (2 h, 36 min, and 2 s) as a baseline. This procedure was intended to eliminate the unequal time intervals between device power-up and the actual start of the experiment. Consequently, all data uniformly lasted for 2 h, 36 min, and 2 s per set across the 40 sets, resulting in a total duration of recorded data of 104 h, 1 min, and 20 s (374480 s). In terms of data acquisition frequency, the DHT22 sensor performed environmental sampling at a physical frequency of once every 2 s according to its hardware specifications. However, constrained by wireless network latency, the communication overhead of API port handshakes, and the dynamic load conditions of the cloud server, the effective recording frequency of the DHT22 sensor successfully uploaded and written to the database fell to approximately once every 3 to 4 s. After division, the total number of raw temperature and humidity samples obtained from the 40 iterations reached the total sample number of 101440. The system lacked an interpolation mechanism; thus, missing data caused by disconnections exceeding 1 min, which invalidated the buffer, could not be recovered. Nevertheless, the system implemented a filtering mechanism to filter out disconnection samples with intervals greater than 60 s, discarding invalid data (removing the sliding windows for those time segments) to maintain the integrity of sliding windows for normal data. When these raw data underwent the aforementioned sliding window technique with a window size of 180 s and a step size of 30 s, a single set of data produced 204 windows (excluding invalid data discarded by the filtering mechanism), yielding a total of 8160 feature window samples across the 40 sets of waveforms.

3.3.2 Data preprocessing and effective sample filtering

Regarding the acquisition frequency of the DHT22 sensor, environmental sampling was performed at a physical frequency of once every 2 s. However, constrained by wireless network latency, the communication overhead of API port handshakes, and the dynamic load conditions of the cloud server, the effective recording frequency successfully uploaded and written to the database fell to approximately once every 3 to 4 s.

Regarding the preprocessing prior to feature extraction, it is necessary to separate these three techniques, as they occupy completely different stages within our system pipeline.

Smoothing was implemented using the linear interpolation method. "Prior to" executing the sliding window technique and feature extraction, foundational smoothing and imputation were indeed performed on the raw time-series data. Because real-world IoT environments inevitably encounter brief communication blind spots or data loss, linear interpolation technology was introduced at the very front end. This repaired data gaps caused by disconnections. However, for data losses exceeding the buffer capacity (20 records), the system lacked a corresponding imputation technique to fill the void. Instead, it relied solely on a filtering mechanism to exclude disconnected samples with intervals greater than 60 s, thereby discarding invalid data (i.e., removing the sliding windows for that specific time period) to maintain the integrity of the sliding windows for normal data.

3.4 Annotation methodology and label generation

To define specific events occurring during transportation including the train and outdoor environments, the labeling process utilized a developed Python software program. The input consisted of raw data in CSV format, structured as [time, humidity, temperature], and the output file was also in CSV format structured as [Start Time, End Time, Label Name]. In contrast to the traditional approach of setting absolute temperature and humidity values as anomaly thresholds, this tool adopted a "time-segment annotation" strategy. The annotation method involved manual labeling. Before starting the annotation, users could add or select an established class (label name). By setting the time duration to generate a label, the "start time" was first set, followed by the "end time". For Fig. 9, the background color between these two time points on the interface would change to indicate a continuous and annotated segment, thereby completing a single annotation. This approach preserved the complete physical waveform variations of the events, enabling subsequent models to learn dynamic features with greater generalization capabilities.

The annotation rule was based on photographs taken using a mobile device (equipped with an internet connection for accurate time updates) immediately when special events (such as humidity spikes caused by train door openings) occurred during the data collection phase. During the annotation phase, the exact time points were determined by reviewing the photo information from the mobile device, which served as the basis for marking the start and end times. The smart specimen transport box retrieved the standard network time via a Wi-Fi connection to the internet during data collection (NB-IoT could also be used, but this experiment primarily relied on Wi-Fi). Because the mobile device and the monitoring equipment referenced

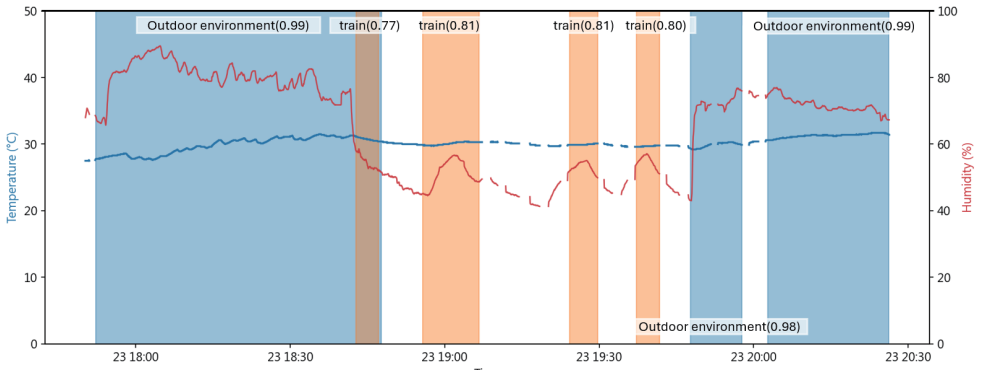


Fig. 9. (Color online) SVM detection results for the experimental data in Table 2: humidity (red line), temperature (blue line), annotated train environment (orange box), and annotated outdoor environment (blue box)

the same time standard, the photograph time was considered consistent with the occurrence time of the special events.

The time-aligned method of the door-opening events was recorded by taking a photograph with the mobile device the moment the train arrived at the station and opened its doors; the generation time of this photograph served as the start time for the label. Subsequently, another photograph was taken when the RH displayed on the electronic temperature and humidity control panel on the left side of the train door stabilized at 25%, marking the end time. (The values displayed by the train's equipment were not used as ground truth but rather as a unified reference target. The purpose of this was to avoid setting the train door closure as the end time, which would result in training samples that were very short and featured overly homogenous characteristics, leading to a failure in training convergence.) This completed the labeling for the "train" environment. The label for the "outdoor environment" was established by taking a photograph with the mobile device the moment the smart specimen transport box left the air-conditioned indoor environment, using the photograph's generation time as the start time. Another photograph was taken the moment the train doors closed after entering the train, serving as the end time label.

However, considering that the established time points might not perfectly align with the starting points of the windows, the system introduced an "Overlap Threshold" mechanism, as shown in Fig. 10. When the thresholds of overlap ratio between the sliding window and the manually annotated interval exceeded 50%, the corresponding supervised label was assigned to that window. Windows that did not meet the overlap threshold were not directly discarded; instead, they were defined as the "Non-event" category. These non-event samples constituted a crucial negative sample space, enabling the SVM model to learn the environmental baseline of the transport box under stable conditions and establish decision boundaries between normal and abnormal states in the feature space, thereby effectively reducing the system's false positive rate.

For the number of labeled events versus non-event segments, the experiments including 40 datasets were repeated in the train's schedules and environments. As shown in Fig. 8, each dataset consistently contains three train (intervals D–F) and two outdoor (intervals A, B, G, and

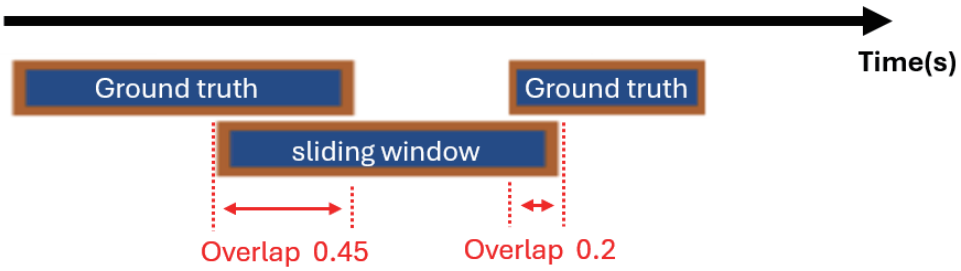


Fig. 10. (Color online) Illustration of the overlap threshold.

H) environment label annotations. (The 40 datasets produce a total of 120 train and 80 outdoor environment label boxes.) The remaining unannotated segments are all classified under the non-event label. Each label occupies a varying number of windows depending on its duration. The statistical ranges of window counts occupied by the three label categories in a single dataset (maximum and minimum values among the 40 datasets) are detailed as follows: For the train environment, the number of windows used in a single label box ranges from 9 to 26, the total number for an entire dataset (three labels) ranges from 41 to 49, and the total usage across all 40 datasets is 1793 windows. For the outdoor environment, the number of windows used in a single label box ranges from 52 to 89, the total number for an entire dataset (two labels) ranges from 138 to 169, and the total usage across all 40 datasets is 5654 windows. For non-event, the number of windows used in a single label box ranges from 1 to 12, the total number for an entire dataset ranges from 14 to 39, and the total usage across all 40 datasets is 713 windows. Altogether, the 40 datasets yield a total of 8160 feature windows.

3.5 Quantitative evaluation metrics and confidence level calibration of the SVM model

3.5.1 Waveform feature extraction and standardization process

Excluding terminal stations and peak hours, the time from a train door opening to closing is approximately 1 min. To ensure that the window of experimental samples can encompass complete features, the system's time window length (Window Size) was set by adding 1 min before and 1 min after the 1-min base duration, totaling 180 s. The sliding window step size (step size) was set to 30 s, utilizing a smaller step size to increase data resolution, as shown in Fig. 11. Concurrently, the NumPy library was utilized to perform low-level, high-performance matrix operations for the execution of sliding window segmentation. A single dataset slid 204 times, and the 40 sets of waveforms yielded a total of 8160 feature window samples.

For detrending, we intentionally avoided any detrending processing prior to feature extraction. In general signal processing, detrending is employed to eliminate long-term, slow variations. However, for the application scenario of this study, when the transport box moves from an outdoor heavy rain environment into an air-conditioned train compartment, the elongated waveform resulting from the slow decrease in humidity is precisely the most core feature of the environmental transition process. If we forcibly eliminated this long-term trend

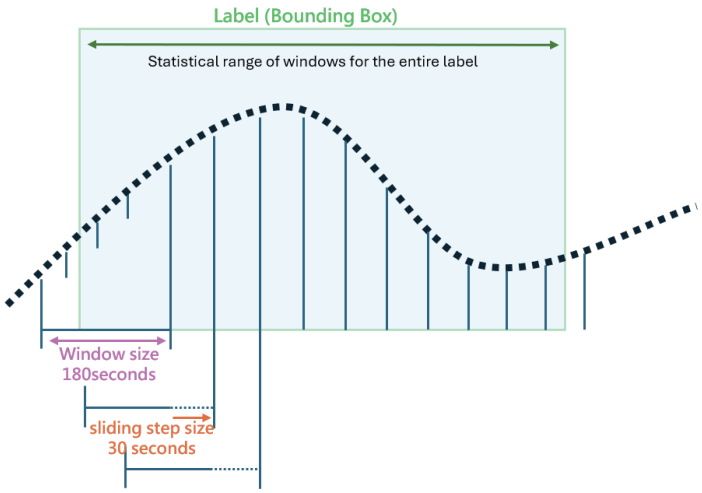


Fig. 11. (Color online) Illustration of the relationship among the label, window size, and step size.

prior to feature extraction, it would be equivalent to destroying the critical clues that the SVM model relies upon to identify environmental transitions. Therefore, preserving the true physical trend of the original signals is a crucial prerequisite for the system to successfully identify subtle environmental changes.

For the waveform feature extraction and standardization process, we adopted machine-learning methods for feature-learning analysis. SVM was used as the classifier. Humidity time-series data were treated as signals from which features such as local gradient changes, rising/falling slopes, and peak widths were extracted to construct feature vectors representing "abnormal humidity rise events". These vectors served as the training set. The most critical step is normalization/standardization. We indeed implemented this technique, but it was executed "after" feature extraction, rather than before. The system utilized the sliding window to calculate an eight-dimensional local feature vector from the original waveforms: (1) temperature mean, (2) humidity mean, (3) temperature standard deviation, (4) humidity standard deviation, (5) temperature local slope, (6) humidity local slope, (7) humidity peak-to-peak difference (difference between maximum and minimum values), and (8) Pearson correlation coefficient between temperature and humidity. Only after extracting these feature vectors did the system utilize the StandardScaler from the Scikit-learn library to standardize these "feature vectors", transforming the data into a distribution with a mean of 0 and a variance of 1. This step aims to unify the measurement scales of the data. Because the range disparity between absolute temperatures (larger numerical values) and variation slopes (extremely small numerical values) is substantial, standardization effectively eliminates this numerical difference. This ensures that when the SVM calculates feature distances, each feature dimension can exert an equal level of influence, thereby preventing the model from overly weighting features with larger numerical values and producing bias.

3.5.2 SVM model configuration and validation strategy

Software Framework: The entire data pipeline and model construction were based on Python 3, deeply integrating Pandas (data structuring and imputation), NumPy (matrix operations), Scikit-learn (machine learning pipeline and model construction), as well as Tkinter and Matplotlib (graphical annotation and visualization).

In this study, we utilized 8160 feature windows and followed an identical 8:2 split ratio. The training set (6528 samples) was used to construct the model boundaries, while the independent testing set (1632 samples) was used for performance validation. The program randomly extracted and isolated 20% of the data using the parameter (random_state = 42). Consequently, the 20% of the testing data became completely independent and never participated in the model's training phase. These 1632 entirely new samples were solely used in the final inference stage, where they unidirectionally applied the scaling ratio and decision weights derived from the training set.

Core Type and Hyperparameters: The system adopted the Gaussian radial basis function (RBF Kernel), with its hyperparameters set to $C = 1.0$ and $\gamma =$ "scale", and activated the class_weight = "balanced" mechanism to handle the highly imbalanced dataset. $C = 1.0$ successfully achieved an optimal balance between maximizing the classification margin and tolerating temperature and humidity noise, preventing the overfitting caused by a hard margin; meanwhile, $\gamma =$ "scale" automatically calculated the kernel width on the basis of the number of features and variance, ensuring that the high-dimensional mapped feature space was neither overly dispersed nor overly concentrated. This facilitated finding the nonlinear decision hyperplane capable of perfectly separating "train", "outdoor environment", and "non-event".

Within the remaining 80% of the training set, to obtain the "confidence score", the system further integrated a fivefold cross-validation. This 80% of the data was internally subdivided into five equal folds. The model iteratively utilized four of these folds for training, while on the remaining one fold, isotonic regression calibration was applied to convert the SVM's geometric margin into a probability to serve as the confidence score.

3.5.3 Confusion matrix and quantitative evaluation metrics

We input 1632 independent test data samples into the trained SVM model for prediction and cross-compared the predicted results with the ground truth labels to generate a 3×3 confusion matrix. To comprehensively present the quantitative performance of the model across each feature category, we converted the matrix into corresponding true positive (*TP*), false positive (*FP*), false negative (*FN*), and true negative (*TN*) values for each individual class. The specific empirical results for each metric are presented in Table 3.

On the basis of the generated confusion matrix mentioned above, we further utilized mathematical formulas to calculate the precision, recall, F1-score, and overall accuracy for each category. The calculation processes for these metrics and their practical significance in medical specimen transportation are detailed as follows.

Table 3
3×3 confusion matrix of SVM model.

	Predicted non-event	Predicted outdoor	Predicted train
True non-event	130 (94.9%)	0 (0.0%)	7 (5.1%)
True outdoor	30 (2.4%)	1210 (95.2%)	31 (2.4%)
True train	16 (7.1%)	1 (0.4%)	207 (92.4%)

- $\text{Precision} = \dfrac{TP}{TP + FP}$

Calculation process: Taking the "train" event as an example, the model made a total of 245 "train" predictions (the sum of the third column in the matrix: 7 + 31 + 207 = 245). Among these, the number of true "train" instances was 207 (*TP*). Substituting into the formula: 207 / 245 ≈ 0.84.

- $\text{Recall} = \dfrac{TP}{TP + FN}$

Calculation process: Taking the "train" event as an example, there were a total of 224 true "train" instances in the test set (the sum of the third row in the matrix: 16 + 1 + 207 = 224). Among these, the number successfully captured by the model was 207 (*TP*) and the number of missed detections misclassified as background was 17 (*FN*). Substituting into the formula: 207 / 224 ≈ 0.924.

- $\text{F1-score} = 2 \times \dfrac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$

Calculation process: The F1-score is the harmonic mean of precision and recall. Taking the "train" event as an example, substituting the previously calculated precision of 0.84 and the recall of 0.92 into the formula: 2 × (0.84 × 0.92) / (0.84 + 0.92) ≈ 0.878.

- $\text{Accuracy} = \dfrac{\text{sum of correct predictions on the diagonal (total } TP\text{)}}{\text{total number of samples in the test set}}$

Calculation process: The number of successful classifications by the model is derived from the sum of the diagonal in the confusion matrix: comprising the correctly predicted normal background (130) + correctly predicted outdoor (1210) + correctly predicted train (207) = 1547 records. Dividing this number by the total number of records in the test set (1632), which is 1547/1632 ≈ 0.9479, corresponds exactly to the 95% overall accuracy presented in the report.

3.5.4 Confidence score calibration and decision threshold setting

3.5.4.1 Mathematical principles of confidence score calculation

We have supplemented the text with the precise mathematical computation process. Initially, the underlying SVM model computes the "geometric margin" between the feature vector $\phi(x)$ and the decision hyperplane, expressed by the formula $f(x) = w^T\phi(x) + b$. Given that the numerical range of this margin spans [−∞, ∞] and does not constitute a probability value, the system introduces the non-parametric "isotonic regression calibration". In contrast to traditional Platt

scaling, which a priori assumes that the data follows a sigmoid logistic regression distribution, isotonic regression employs a non-parametric method. This makes it exceptionally suited for addressing the highly asymmetrical distribution of temperature and humidity features induced by hysteresis effects in this study. This algorithm fits a step mapping function by solving for the minimization of the mean squared error equation, $\min\sum(y_i - p_i)^2$, under the strict constraint of a monotonically increasing condition ($p_i \leq p_j$ for $f(x_i) \leq f(x_j)$). For instance, when the geometric margin is $f(x) = 0.5$, the function references the true distribution derived from cross-validation and objectively maps it to $p = 0.55$. This mathematical procedure successfully and rigorously transforms boundless geometric distances into true probability values bounded between 0 and 1, which serve as the confidence scores.

3.5.4.2 Basis for threshold selection

After obtaining the scores, we discarded the default machine learning threshold of 0.5 and strictly set the decision threshold to 0.6 on the basis of a bidirectional trade-off. The text details an analysis of two extreme risks: if the threshold is very low (e.g., 0.5), false alarms can easily be generated by minor waveform variations, leading to overly frequent alarm triggers and causing clinical "alarm fatigue". However, blindly pursuing zero false alarms by raising the threshold above 0.7 would cause subtle but authentic physical features, such as train door openings, to be entirely overlooked, leading to practically intolerable *FN*. In the practice of medical specimen transportation, the cost of missing a true environmental exposure risk far exceeds that of a system false alarm.

3.6 Baseline model comparative analysis: Performance comparison between decision tree and SVM

To justify the added complexity of the SVM model and compare its performance with simpler detection approaches, we included a simple classifier—specifically, a decision tree—as a baseline classifier model (with a maximum depth limit of 5). Table 4 presents a detailed comparison of precision, recall, F1-score, and overall accuracy between the two models. For the results, refer to Table 4.

Section 3.5.3 is about calculating precision, recall, and F1-score for the train of model type according to the formula. For evaluating the additional features of the train, outdoor environment, and non-event in Table 4, the precision metrics are 0.84, 1.00, and 0.74, the recall metrics are 0.92, 0.95, and 0.95, and the F1-score metrics are 0.88, 0.98, and 0.83, respectively. The number of successful classifications by the model is derived from the sum of the diagonal in the confusion matrix.

3.6.1 Precision comparison

The decision tree shows a higher precision in the "non-event" category (0.84) than the SVM (0.74). This difference highlights how each model processes data. The decision tree acts like a rigid, threshold-based detection method. When an abnormal event ends, it immediately cuts off

Table 4
Comparison of evaluation metrics between SVM and decision tree.

Model type	Evaluation metrics	Non-event	Outdoor environment	Train	Overall accuracy
Support	—	137 records	1271 records	224 records	Total: 1632 records (%)
Baseline	precision	0.84	0.98	0.86	95
	recall	0.88	0.97	0.88	
	F1-score	0.86	0.97	0.87	
SVM	precision	0.74	1.00	0.84	95
	recall	0.95	0.95	0.92	
	F1-score	0.83	0.98	0.88	

the classification, ignoring the natural delay (hysteresis) of the sensors. The SVM, however, understands that humidity drops slowly and flexibly accommodates this natural decay as a normal "non-event". While this practically correct judgment mathematically lowers the SVM's precision by increasing *FP* in a rigid scoring system, it better reflects physical reality. Furthermore, the SVM achieves a perfect 1.00 precision in the "outdoor environment" category, proving its superior ability to accurately isolate extreme environmental changes.

3.6.2 Recall comparison

The SVM demonstrates a clear advantage in recall, especially for the "non-event" (0.95 vs 0.88) and "train" (0.92 vs 0.88) categories. In medical specimen transportation, missing a true environmental risk (a *FN*) is practically intolerable. The decision tree's lower recall shows that its simple, threshold-based rules easily miss subtle but real physical features, such as the brief humidity spikes from train door opening. The SVM successfully captures these minor fluctuations, guarding normal conditions while reliably catching real anomalies.

3.6.3 F1-score comparison

The F1-scores are close because both are fundamentally capable classifiers. However, the SVM slightly outperforms the decision tree in identifying the most critical dynamic events ("outdoor environment" at 0.98 vs 0.97, and "train" at 0.88 vs 0.87). The SVM's slightly lower F1-score for non-events (0.83 vs 0.86) is a worthwhile trade-off. It sacrifices a small amount of background precision to maintain a much higher sensitivity to real anomalies.

3.6.4 Overall accuracy comparison and complexity justification

Both models achieve an identical overall accuracy of 95%. However, looking only at this 95% can be misleading because the test dataset has a very uneven distribution of event types. Specifically, out of the 1632 total test samples, a massive majority of 1271 samples (about 78%) belong to the "outdoor environment" category, while the "train" event only has 224 samples. Moving a transport box into heavy outdoor rain causes clear and extreme humidity changes. Because this feature is so prominent, both the SVM and the decision tree can easily and accurately identify this large group of data. This massive amount of "easy" data artificially pushes the overall average accuracy up to 95% for both models.

To truly compare the two models and justify the added complexity of the SVM, we must look at how they handle the much smaller group of data that contains subtle environmental changes. For the "train" event (only 224 samples), the SVM achieves a much higher recall (0.92) than the decision tree (0.88). This proves that the simpler, threshold-based rules of the decision tree mostly rely on the obvious outdoor data to obtain a high total score, but they struggle to catch minor physical variations, such as brief train door openings. In contrast, the SVM effectively handles sensor delays and subtle noise. Therefore, the SVM's 95% accuracy represents true stability and reliability in complex real-world situations, proving it as a much more practical choice than simpler detection approaches.

3.7 Visualization of anomaly detection and IoT edge inference results

Empirical results confirm that Gaussian kernels outperform other kernel functions. Thus, these kernels yield the highest overall control performance for the final model. Figure 9 shows the results of training the detect model using the SVM model for Fig. 7. The method conceptually mirrors the bounding-box feature extraction in You Only Look Once (YOLO) image detection. Whereas YOLO learns object contours and boundaries in 2D images, in this study, we learn the morphological characteristics of humidity waveforms in 1D time-series curves. More importantly, compared with image-based deep learning models such as YOLO, the 1D SVM classifier requires significantly less computational power and inference time. This low-overhead characteristic makes the SVM model highly suitable for resource-constrained IoT edge devices, providing rapid and efficient anomaly detection without the need for heavy graphical processing units (GPUs). After training, the model was used to detect unseen experimental data to evaluate whether it could still recognize the same humidity patterns across varying environments. Model output included confidence scores indicating the similarity of detected events to the characteristic "humidity spike caused by door opening".

This feature-learning and validation process allowed the study to exclude variations caused by random environmental fluctuation, measurement noise, or device delay, establishing an analysis based on quantifiable and reproducible waveform characteristics. As a result, the "cone-shaped humidity pulses" transitioned from subjective observations to model-recognizable physical phenomena. Three 2-min segments (B1, B2, B3) shown in Fig. 12 were used as labeled targets. The data-processing procedure utilized a sliding window technique with a window size of 180 s and a step size of 30 s. To ensure labeling quality, a minimum segment duration of 60 s was required, and a gap-merge threshold of 90 s was used to merge adjacent feature segments into a single event.

The detection stage used data with labels defined as "train" and "outdoor environment". "Train" corresponded to the cooling effects of the air-conditioned compartment, whereas "outdoor environment" corresponded to rapid humidity increases under heavy-rain conditions. The SVM detection results in Fig. 12 show dark-blue regions classified as outdoor and light-blue regions classified as train environments. The overall classification aligns closely with the trends, indicating that the model successfully captured each environment's characteristic patterns. Some discontinuities in detected feature regions were observed. This is preliminarily attributed

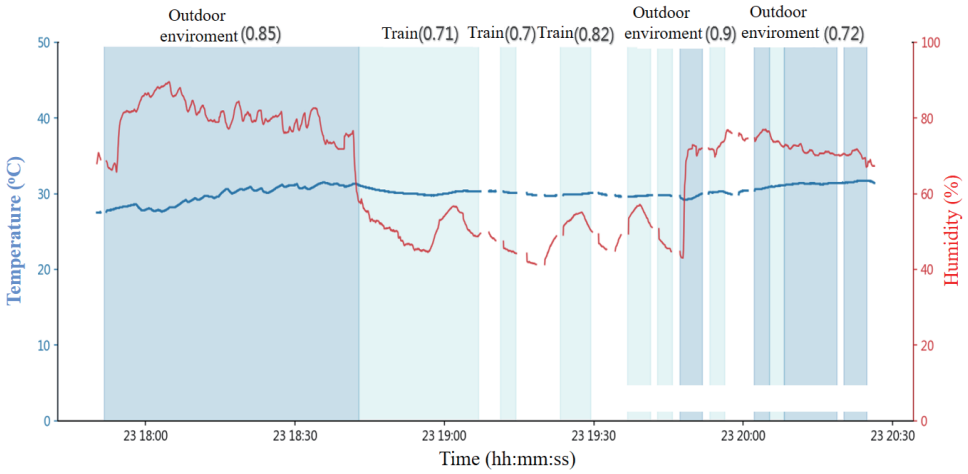


Fig. 12. (Color online) Results of training the detect model using SVM model according to Fig. 7.

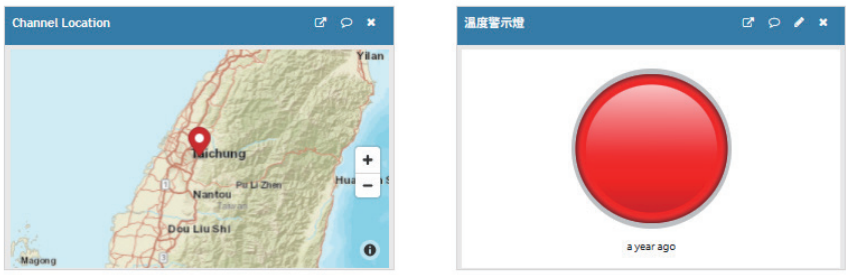


Fig. 13. (Color online) GPS track and alarm display.

to the gap-merge threshold being very low (90 s), causing segments belonging to the same event to be separated and resulting in discontinuous classifications.

The system's front end allows the real-time visualization of temperature/humidity curves and GPS tracks for each device as shown in Fig. 13. When a threshold is exceeded, the device immediately switches its LED from green flashing to solid red and activates the buzzer. The corresponding data entry is marked as an "event-level anomaly" in the database for auditing and traceability. The device currently uploads data via a Wi-Fi network (HTTP/HTTPS POST → PHP API → MySQL). However, the hardware is fully compatible with NB-IoT modules. Since NB-IoT provides wide coverage, strong indoor penetration, and low power consumption,[24] it remains the ideal communication architecture for future long-term continuous monitoring once telecommunication constraints are resolved.

4. Conclusions

The transport box contains an embedded system, and the computer uses a human–machine interface (HMI) to obtain the number of specimens and temperature information from the

internal module. The hardware is then integrated with the firmware designed for the chip to develop a software monitoring system for the specimen transport box. This system includes the following:

(A) Specimen sending system: Allows the configuration of specimen information and required temperature monitoring parameters.
(B) Specimen receiving system: Displays specimen information at arrival and the temperature history during transport.
(C) Specimen transport system: Provides real-time access to specimen information and the temperature conditions throughout the delivery process, ensuring effective temperature control during specimen movement.

Currently, the smart specimen transport box uses Wi-Fi as the primary communication method, relying specifically on portable mobile hotspots provided by personnel during cross-city transit. This practical approach was adopted owing to current telecommunication regulations in Taiwan, which restrict the procurement of NB-IoT SIM cards to specific enterprise applications and lack permanent commercial data plans. Nevertheless, the proposed system architecture fully supports NB-IoT integration. In future commercial or hospital-wide deployments, transitioning to NB-IoT will allow the device to connect directly to the server through telecom networks without relying on specific Wi-Fi service set identifiers (SSIDs), thereby offering wider coverage, higher connection stability, lower power consumption, and enhanced reliability for cross-site operations.

The motherboard used in this system (TTGO T18 V3.0) is equipped with 4 MB of flash memory. In actual operation, the ESP32's dual-core operating system (FreeRTOS), Wi-Fi communication stack, and data acquisition firmware already occupy approximately 3 MB, leaving only about 1 MB (1024 KB) for the serial peripheral interface flash file system (SPIFFS) to serve as an offline data buffer. To ensure the integrity of disconnected data, the system allocates a fixed block of approximately 35 KB for each offline cache record, which includes sensor data, GPS sentences, and system logs. Therefore, a buffer capacity of 20 records occupies about 700 KB (35 KB × 20). This configuration deliberately reserves a safety margin of approximately 324 KB to prevent "Filesystem Overflow" or storage exhaustion when the buffer is full, thereby avoiding a system crash or I/O blockage. As mentioned in Sect. 3.3, constrained by network latency, the actual effective upload frequency of the system dynamically ranges from once every 3 to 4 s. Consequently, this 20-record buffer capacity, dictated by the hardware capacity limit, precisely provides the system with approximately 60 to 80 s (20 records × 3–4 s) of disconnection protection. This precise hardware resource allocation perfectly ensures that critical sensor data is fully preserved without loss when the device passes through temporary communication blind spots, such as elevators or basements.

Regarding the discussion of advantages and remaining limitations, when compared with traditional mercury and dial thermometers, the proposed device demonstrates the following significant advantages.

1. Real-time cloud monitoring to prevent temperature and humidity fluctuations from opening the box: traditional mercury and dial thermometers require constantly opening and closing the transport box just to check temperature variations during specimen delivery. This process

inevitably causes severe temperature and humidity fluctuations. In scenarios requiring strict temperature control, the advantages of the proposed device become fully apparent: it leverages Wi-Fi or NB-IoT technology to retrieve real-time temperature and humidity data directly from the cloud database without the need to open the lid.

2. Support for multi-mode environmental settings and instant anomaly push notifications: The specimen transport box features three temperature monitoring modes: room temperature mode (24 to 28 °C, humidity below 60% RH), refrigerated mode (5 to 7 °C, humidity below 60% RH), and frozen mode (–18 to –12 °C, humidity below 60% RH). When environmental values exceed the configured range of the current mode, the hardware immediately triggers an alarm (the LED flashes a high-intensity red light at a 1-s frequency, accompanied by a buzzer sound). Furthermore, when the SQL server receives the data packet and the "Status" in the database table switches from 0 to 1, the system triggers an API to send an instant notification to the user via the official message script service of "LINE" (a mainstream communication app in Taiwan). This mechanism effectively resolves the flaw of traditional thermometers placed inside the box, which fail to notify personnel of anomalies in the first instance.
3. Integration of the SVM algorithm for dynamic decision-making and low hardware load: Traditional microcontrollers mostly rely on if statements to set rigid threshold conditions. If the environmental changes triggering an event are irregular, it often leads to alarms that are difficult to trigger accurately or overly frequent false alarms. Here, the advantages of the SVM classifier are fully realized: through supervised learning training, the decision boundary for an event is no longer a rigid, fixed range but can flexibly adapt to complex scenarios (such as using SVM in this study to identify special events such as the "outdoor environment" and "train door openings"). This can also be widely applied to intelligent alarms for temperature and humidity anomalies in the future. In addition, as a 1D time-series-based classifier, the SVM model requires considerably shorter training and detection times than image-processing models such as YOLO. The hardware computational performance it demands is also far lower than those of the other image-based deep learning models, making it highly suitable for scalable deployment on centralized cloud servers without significant computational overhead. Despite these advantages, regarding the remaining hardware limitations, power consumption remains a challenge. Experimental results show that a fully charged 3200 mAh 18650 lithium battery provides a continuous operational runtime of exactly 11 h, 35 min, and 4 s. While sufficient for single-day inter-city transport, extending battery life will be a focus for future optimization.

Acknowledgments

This research work is supported by the research projects by Taichung Armed Forces General Hospital and National Science and Technology Council (NSTC) under the project grant (TCAFGH_D_114030 and NSTC 114-2637-E-167-010-), and is gratefully acknowledged.

Authors' contributions

Author	Authors' contributions
Wen-Tung Hsu	System and specimen container design, specimen handling process research and design, specimen delivery data analysis, project execution and schedule control, patent/paper writing
Jing-Xiang Zhang	Execution program development and writing, sensor and hardware fabrication, chip and memory configuration, temperature calibration, communication transmission, data storage and database, delivery and receiving end interfaces, data format conversion, hospital-side app interface, hospital-side program user interface, etc.
Chen-Hsuan Ma	Execution program development and writing, sensor and hardware fabrication, chip and memory configuration, temperature calibration, communication transmission, data storage and database, delivery and receiving end interfaces, data format conversion, hospital-side app interface, hospital-side program user interface, etc.
Cheng-Yu Peng	Research and design of RFID tags and temperature systems; analysis of temperature monitoring data; design and modification of specimen box software and hardware; patent/paper writing and progress management

References

1 A. Papapostolou and H. Chaouchi: J. Netw. Comput. Appl. **34** (2011) 902. https://doi.org/10.1016/j.jnca.2010.04.009
2 S. Mahfouz, F. Mourad-Chehade, P. Honeine, H. Snoussi, and J. Farah: 2013 IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC, 2013) 744–748. https://doi.org/10.1109/SPAWC.2013.6612149
3 Y. Duroc and S. Tedjini: C. R. Phys **19** (2018) 64. https://doi.org/10.1016/j.crhy.2018.01.003
4 X. Wang, J. Zhang, Z. Yu, S. Mao, S. C. G. Periaswamy, and J. Patton: IEEE Internet Things J. **6** (2019) 10715. https://doi.org/10.1109/JIOT.2019.2941023
5 H. Shim, Y. Uh, S. H. Lee, and Y. R. Yoon: J. Med. Syst. **35** (2011) 1403. https://doi.org/10.1007/s10916-009-9417-z
6 K. Ohashi, S. Ota, L. Ohno-Machado, and H. Tanaka: Comput. Biol. Med. **40** (2010) 545. https://doi.org/10.1016/j.compbiomed.2010.03.007
7 O. Urbano, A. Perles, C. Pedraza, S. Rubio-Arraez, M. L. Castelló, M. D. Ortola, and R. Mercado: Sensors **20** (2020) 1163. https://doi.org/10.3390/s20041163
8 Kurnianingsih, M. A. Helmy, A. S. Putra, D. Ernawati, and A. S. Prabuwono: 2015 2nd Int. Conf. Information Technology, Computer, and Electrical Engineering (ICITACEE, 2015) 449–453. https://doi.org/10.1109/ICITACEE.2015.7437848
9 H. A. Khan, R. Abdulla, S. K. Selvaperumal, and A. Bathich: Int. J. Electr. Comput. Eng. **11** (2021) 3300. https://doi.org/10.11591/ijece.v11i4.pp3300-3309
10 S. Monteleone, M. Sampaio, and R. F. Maia: 2017 IEEE Int. Conf. Service Operations and Logistics, and Informatics (SOLI) 205–210. https://doi.org/10.1109/SOLI.2017.8120995
11 N. T. Le, Thwe, M. M. T. Chit, T. L. Truong, A. Siritantikorn, N. Kongruttanachok, W. Asdornwised, S. Chaitusaney, and W. Benjapolakul: Sensors **23** (2023) 546. https://doi.org/10.3390/s23010546
12 M. Sarkar, T.-H. Lee, and P. K. Sahoo: Electronics **13** (2024) 2309. https://doi.org/10.3390/electronics13122309
13 W. A. Cruz Castañeda, and P. Bertemes Filho: Sensors **24** (2024) 7965. https://doi.org/10.3390/s24247965
14 A. Hennebelle, H. Materwala, and L. Ismail: Procedia Comput. Sci. **220** (2023) 331. https://doi.org/10.1016/j.procs.2023.03.043
15 J. Gillespie, T. P. da Costa, X. Cama-Moncunill, T. Cadden, J. Condell, T. Cowderoy, E. Ramsey, F. Murphy, M. Kull, R. Gallagher, and R. Ramanathan: Sustainability **15** (2023) 2255. https://doi.org/10.3390/su15032255
16 A. S. Mirkhail and Z. Xinyou: Int. Res. J. Multidiscip. Scope **6** (2025) 1480. https://doi.org/10.47857/irjms.2025.v06i02.03768

17 LilyGO. (n.d.). TTGO T-Energy (T18 V3.0) ESP32 Development Board—Technical Documentation. https://github.com/LilyGO/LILYGO-T-Energy
18 Aosong Electronics Co. (n.d.). DHT22 (AM2302) Temperature and Humidity Sensor Datasheet. https://files.seeedstudio.com/wiki/Grove-Temperature_and_Humidity_Sensor_Pro/res/AM2302-EN.pdf
19 Y. A. Ahmad, T. S. Gunawan, H. Mansor, B. A. Hamida, A. F. Hishamudin, and F. Arifin: 2021 8th Int. Conf. Computer and Communication Engineering (ICCCE, 2021) 131–134. https://doi.org/10.1109/ICCCE50029.2021.9467147
20 NEO-M8N Datasheet u-blox AG. (n.d.). NEO-M8N GNSS Module Datasheet. https://www.alldatasheet.com/html-pdf/1242657/U-BLOX/NEO-M8N/251/1/NEO-M8N.html
21 NXP Semiconductors. (n.d.). MFRC522 RFID Reader IC Datasheet. https://www.alldatasheet.com/html-pdf/346109/NXP/RC522/52/1/RC522.html
22 GME Technology. (n.d.). GME12864-77Y 128×64 Graphic OLED Display Module Datasheet. https://robu.in/wp-content/uploads/2019/12/GME12864-11-1-1.pdf
23 Support Vector Machine (SVM). https://www.ycc.idv.tw/ml-course-techniques_2.html
24 K. Mekki, E. Bajic, F. Chaxel, and F. Meyer: ICT Express **5** (2019) 1. https://doi.org/10.1016/j.icte.2017.12.005

About the Authors

Wen-Tung Hsu is currently a medical laboratory scientist at Taichung Armed Forces General Hospital. His research interests focus on medical laboratory science. (bio803@gmail.com)

Jing-Xiang Zhang is currently studying for his master's degree in electronic engineering at National Chin-Yi University of Technology, Taiwan. His research interests are in AIoT and smart sensors. (4B313019@ncut.edu.tw)

Chen-Hsuan Ma received his master's degree in electronic engineering from National Chin-Yi University of Technology, Taiwan. His research interests are in vision detection, tactile sensors, and model predictive control. (4B213018@ncut.edu.tw)

Cheng-Yu Peng received his Ph.D. degree from the Graduate Institute of Mechanical and Electrical Engineering of National Taipei University of Technology, Taiwan, in 2007. He was a senior researcher and department manager at Green Energy and Environment Institute, Industrial Technology Research Institute, from 2007 to 2017. Since 2017, he has been a faculty member in the Department of Electronic Engineering, National Chin-Yi University of Technology. His current research interests primarily involve smart supervisory control, intelligent robotics, automation and mechatronics, power engineering, solar energy, and engineering applications. (peng@ncut.edu.tw)