STC-RL: A Spatiotemporal Control Framework for Intelligent Environmental Regulation in Automated Warehousing Using IoT and Deep Learning

Min Gu*, Xiaosa Zhou
Xiaosa Zhou's E-mail: Xiaosa_Zhou821@163.com
Huanggang Polytechnic University, Huanggang, 438002, Hubei, China
Corresponding author's E-mail:gumin2026@outlook.com

Keywords: Internet of Things, deep learning, automated warehouses, environmental monitoring, intelligent control, STC-RL

Received: August 21, 2025

Intelligent warehousing environments require precise, energy-efficient control of temperature, humidity, and other environmental parameters. To address this, we propose STC-RL, a novel deep reinforcement learning framework that integrates Transformer-based temporal modeling, 3D CNN-based spatial field reconstruction, and graph neural network (GNN)-enhanced anomaly detection within a continuous-action reinforcement learning policy. Specifically, the Transformer captures long-range temporal dependencies from multi-sensor time series, while the 3D CNN generates a spatial thermal-humidity field via bilinear interpolation of sensor coordinates. The GNN encodes physical sensor topology to detect equipment failures through residual-based anomaly scoring. The RL agent operates in a continuous action space (e.g., setpoint temperature, humidifier output) and optimizes a multi-objective reward balancing environmental deviation, energy consumption, switching frequency, and anomaly alerts. Experimental results on a real 2,100 m² cold-chain warehouse show that STC-RL reduces energy consumption by 13.1%, achieves an average AUC-ROC of 0.936 for anomaly detection, and lowers temperature/humidity prediction RMSE to 1.02°C / 4.15%, outperforming six baselines. The system also cuts food spoilage by 66.7% and improves temperature compliance to 97.6%.

Povzetek: Predlagani sistem STC-RL uporablja napredno strojno učenje za učinkovitejši nadzor pogojev v skladiščih, kar zmanjša porabo energije in izboljša kakovost shranjevanja.

1 Introduction

As a transit node for the development of logistics routes and supply chains, warehousing is becoming increasingly important in modern logistics systems. Warehousing during the logistics transportation process not only guarantees the quality of goods transportation, but also improves the dispatch efficiency of goods, thereby promoting the development of modern logistics. Traditional warehousing technology has many problems. Traditional warehousing relies on manual labor, so there are great deficiencies in the precise storage of goods. This storage method cannot store goods under appropriate humidity, temperature, and light conditions, which may affect the quality of goods, leading to problems such as aging of mechanical equipment and food spoilage. Damage to precision equipment for food and medicine will cause huge economic losses and fail to utilize the coordinated development of the supply chain [1].

Modern logistics and supply chain developments require that storage environments be able to dynamically adjust parameters based on environmental changes, and even, to a certain extent, predict environmental conditions so that regulatory strategies can be set in advance. This demand has driven research into automated warehousing.

In recent years, the rapid development of robotics has promoted the further optimization of automated warehousing. The emergence of automated guided vehicles, automatic forklifts, sorting robots, and autonomous mobile robots has enabled the automatic sorting and handling of goods and their transport to shelves [2]. On the other hand, the rapid development of IoT technology has also promoted the development of warehouse environmental parameter monitoring. Multiple IoT sensors can monitor the temperature, humidity, food health, equipment damage, etc. in real time to achieve all-round real-time monitoring of the warehouse environment [3].

However, automated warehousing technology currently has many shortcomings. The primary issue is incomplete monitoring. The lack of a dedicated sensor

layout design results in uneven sensor distribution, irrational data collection, and a failure to reflect the environmental conditions of the entire warehouse space. Furthermore, warehouse data needs to be transmitted to remote terminals, which carries high data transmission costs and slows policy response. Furthermore, policy response often relies on manually predefined rules, which fail to cover all situations. This makes automated warehousing difficult to adapt to the environment.

Automated warehousing technology urgently needs to meet people's needs for the quality of stored goods and energy utilization, especially in the scenario of cold chain transportation, which requires operations to be carried out under fixed temperature and humidity. However, the ambient temperature is ever-changing, and it is very necessary to design a set of parameters that can adaptively adjust with the ambient temperature changes. This is directly related to the safety and compliance of products. In an automated warehousing environment, a large number of sensor nodes need to be distributed in the warehouse space to achieve comprehensive perception of parameters such as smoke conditions, temperature and humidity, carbon dioxide conditions, and light conditions. However, the cost and power consumption of these sensor nodes affect the cost of the entire warehouse [4] . On the other hand, in order to analyze and process the node data of multiple sensors, a large amount of calculation and processing is often required on the server, which consumes huge costs. How to achieve a balance between cost and transportation efficiency is also a major problem in automated warehousing [5] .

At present, in the field of automated warehousing research, there are already a large number of models, such as GRU, which are used as time series prediction tools to perform dynamic predictions on sensor data. At the same time, some systems use Prophet statistical tools to predict trends. However, these studies all have certain shortcomings. First, sensor data has the problem of long-term dependence and severe coupling. The above methods cannot capture the correlation in complex data, so it is difficult for them to perform complex modeling of multiple factors and obtain accurate results [6] . There are also studies using GNN to model the physical topology of sensors to optimize the distribution of sensors. However, these methods also have the problem that the topology is fixed and cannot be dynamically controlled. There are also some studies that use q-learning for sensor control, but its strategy learning is limited to pre-defined rules and cannot achieve online learning. There is no way to cover all boundary conditions, the control accuracy is low, and it is difficult to balance control accuracy and power consumption.

This paper aims to build a framework model that combines IoT technology and deep learning methods, constructing a time series modeling method using transformers and incorporating an attention mechanism to predict environmental change trends. We use convolutional neural networks to implement modeling and anomaly detection in spatial environments. In terms of decision-making, we use the concept of reinforcement learning to continuously adjust strategies based on the differences between predicted results and real-time status, thereby dynamically changing the parameters of the environmental controller and forming a dynamic strategy. These strategies can be used to adjust the parameters of equipment such as air conditioners and humidifiers. This research can, to a certain extent, achieve precise control of automated warehouse environment equipment, ensure product quality and compliance, reduce the loss of items during transportation, and increase transportation and supply chain benefits.

To formalize our contribution, we pose two research questions:

RQ1: Can continuous-action reinforcement learning improve energy efficiency over discrete or rule-based control in automated warehouses?

RQ2: Does integrating physical sensor topology via GNN enhance anomaly detection without degrading control performance?

We hypothesize that:

H1: Under identical conditions, STC-RL will achieve lower RMSE in temperature/humidity prediction and lower OSF than all baselines.

H2: The GNN-augmented anomaly module will yield AUC-ROC > 0.90 across diverse fault scenarios.

2 Related works

2.1 IoT

As the core of the automatic warehousing system, the Internet of Things system is responsible for controlling various components in the warehouse. By optimizing the Internet of Things system, the response speed, response accuracy and other indicators of the warehousing system can be improved. Therefore, He et al. [7] is committed to improving the response stability and comprehensiveness of the monitoring range of the Internet of Things system and designed a hybrid network storage environment for real-time monitoring and environmental adjustment of cold chain items. The core of this model uses ZigBee and LoRa, both of which can improve the real-time monitoring system. Li et al. [8] designed a framework that supports the long-term operation of large-scale Internet of Things sensors based on He et al. [7] and combined with the wide area network architecture. Experimental results show that this framework can reduce the operation and maintenance costs of a single Internet of Things sensor. However, most of these existing studies focus on the system operation and maintenance of Internet of Things sensors, and there is less research on the layout of sensors. Honglin et al. [9] pointed out that unreasonable layout of Internet of Things sensors will cause the waste of some sensor perception ranges and the phenomenon of perception blind spots, which will cause the data returned by Internet of Things sensors to

have erroneous data gradients. Therefore, some studies have tried to solve this problem. Kerr et al. [10] used the position weighted average method to fill the blind spots of the physical space of the warehouse, which has certain theoretical value. However, these studies are limited to static physical topology and cannot be applied to actual warehousing environments. This is because actual warehousing environments often have problems such as stacking of goods and moving of operating machines, which will bring certain environmental disturbances and thus affect the practical application of this research .

On the other hand, the computation of large amounts of data from IoT devices creates computational pressure. Lei et al. [11] used edge devices for data transmission between IoT devices. They designed a lightweight data aggregation algorithm at the communication gateway layer to reduce the volume of data and detect abnormal data, which to some extent alleviated the pressure of big data computation. However, the transmission, computation, and processing of massive amounts of IoT data remain a major challenge for automated warehousing.

2.2 DL

At present, there are many studies that apply deep learning models to the design of intelligent control algorithms for automatic warehousing systems. Optimizing intelligent control algorithms can optimize the trend prediction and modeling of automatic control of automatic warehousing systems and provide more accurate operation instructions for the system. Traditional methods are based on time trend prediction algorithms such as ARIMA and Prophet. However, the modeling of such algorithms is too simple and cannot capture the complex nonlinear transformation relationship of multi-sensor data. Zhang et al. [12] used a two-layer LSTM model to predict the trend of temperature changes and humidity changes in the warehouse. However, in practice, such algorithms have shown that the time window is too small to solve practical problems. At the same time, due to the separate prediction of temperature and humidity, the two prediction results deviate too much from the actual results. Liu et al. [13] applied convolutional neural networks to the modeling of automated warehousing systems. They first used the spatial information of IoT sensors to generate a three-dimensional environmental field distribution layout. This distribution layout can provide early warning of hot spots. At the same time, they used graph neural networks to calculate the physical topology of each IoT sensor and return this topological relationship to CNN. CNN will generate a prediction result based on a heat map, thereby producing an early warning effect. However, this type of method of fusing temporal and spatial information also has some shortcomings. The use of GNN relies on a predefined physical topology, which is naturally impossible to achieve.

2.3 Intelligent control strategy

Intelligent control in automated warehousing often relies on rules, which lack comprehensive coverage and inherently have drawbacks. Traditional methods often employ threshold control. For example, when a temperature sensor exceeds a predetermined threshold, the automated warehouse activates the refrigeration equipment. These methods offer advantages in ease of implementation and rapid response. However, these control methods can easily lead to serious problems such as damage to items and precision equipment. Furthermore, repeated power cycling of equipment increases power consumption. Some intelligent optimization strategies are based on fuzzy control, and some real-world cases have shown their effectiveness in certain scenarios. However, they are difficult to implement for automated warehousing systems that meet high-precision requirements.

Navia-Osorio et al. [14] used reinforcement learning to achieve the dual goals of energy saving and intelligent control. The experimental results showed that the power consumption of the automatic warehousing system was reduced by 15% under the premise of ensuring temperature stability. However, his research was conducted in a simulated environment. In actual operation, the automated warehousing system must not only consider temperature, but also humidity and light, and also consider multiple factors such as product stacking. Therefore, this type of method is also impractical. Liu et al. [15] and Xu et al. [16] made improvements based on A, but an inherent disadvantage of this type of research is that the action space of the reinforcement learning algorithm is too narrow, only considering the switch and gear control of the equipment, and the design of the reward function requires a large amount of experimental data support. Guo et al. [17] pointed out that in the current research on the application of RL in automated warehousing management, the RL data includes a large amount of interactive information in the warehousing process, which may have a certain risk of information leakage. In addition, Qian et al. [18] pointed out in their research that the state space of RL should be transformed from discrete state space to continuous state space, and the traditional action mode of intelligent agents should be changed from gear changes to fine-grained control. However, how to combine the latest reinforcement learning algorithms to achieve a closed loop of intelligent prediction and fine-grained control is still a challenge.

Recent studies have increasingly applied reinforcement learning to intelligent logistics systems. For instance, Chen et al. (2023[19] proposed a multi-agent deep reinforcement learning framework for collaborative decision-making in edge computing environments, demonstrating the potential of RL in resource-constrained operational settings. More closely related to cold chain management, Kavididevi et al. (2024) [20] developed an IoT-enabled reinforcement learning approach to optimize

temperature control in refrigerated transport, significantly improving cargo integrity. Building on these trends, our work extends RL to stationary warehouse climate control, where continuous-action policies enable fine-grained, energy-efficient environmental regulation—addressing a critical gap between mobile logistics and fixed-storage scenarios.

2.4 Comparative analysis and SOTA limitations

Table 1: Comparative analysis and SOTA limitations

Study	Temporal Model	Spatial Modeling	Control Granularity	Anomaly Detection	Key Limitation
Zhang et al. [12]	LSTM	None	Discrete (on/off)	No	Short time window; no spatial context
Liu et al. [13]	None	CNN + fixed GNN	Rule-based	Yes (static topology)	Topology not learnable; no RL
Navia-Osorio et al. [14]	None	None	Discrete gears	No	Simulated only; ignores humidity/light
CvT (baseline)	Transformer + CNN	Grid-based	Discrete RL	No	Lacks physical topology integration
GCN-LSTM (baseline)	LSTM	Fixed GNN	MPC	No	Cannot adapt topology dynamically
Ours (STC-RL)	Transformer	3D CNN + learnable GNN	Continuous action	Yes (residual + GNN)	—

Current SOTA methods suffer from three critical gaps: (1) discrete or rule-based control limits fine-grained parameter adjustment; (2) spatial modeling ignores dynamic physical topology, treating sensors as isolated points; and (3) anomaly detection is decoupled from control, preventing closed-loop fault-aware decision-making. STC-RL addresses these by unifying continuous-space RL with joint spatiotemporal encoding and topology-aware anomaly scoring.

3 Materials and methods

To address the problem of extensive control strategies in automated warehousing, we designed a spatiotemporal modeling method based on transformers and CNNs. The purpose of this modeling is to achieve high-precision prediction and real-time control in automated warehousing management.

3.1 Model framework

The intelligent control automated warehousing system proposed in this study is divided into four modules. The first module is the data acquisition and batch processing module. This module is responsible for acquiring raw data of different modalities from multi-source IoT sensors, and is responsible for designing personalized data denoising, cleaning, and compression methods for different modal data, and standardizing the data. The second module is the data modeling module. It mainly uses a combination of transformer and CNN to extract and fuse features of the time dimension and spatial dimension respectively. The main function of this module is to combine long-term time features and spatial features to predict the parameter adjustment prediction values of controllers at different locations in the warehouse. The third module is the abnormal situation prediction module. This module is responsible for identifying abnormal conditions in the automated warehousing environment based on the

deviation between the actual situation and the model prediction, and at the same time combines the physical topological features provided by the graph neural network to generate early warnings and formulate response plans. The fourth module is the intelligent control and decision-making module. This module is mainly responsible for dynamically generating control strategies based on the prediction results of the data modeling module and the abnormal monitoring results of the abnormal situation prediction module, while combining the actual storage needs of the warehouse, such as the temperature, humidity, and lighting conditions required by the items, as well as the storage needs, such as energy consumption requirements. This control strategy is used to instruct the operation of control equipment such as air conditioners and humidifiers.

In this model, data first passes through the data acquisition and batch processing module, then through the data modeling module to obtain predicted values. Both the predicted values and the original data are input into the anomaly prediction module to obtain the anomaly prediction results. Finally, the anomaly prediction results, the predicted values from the data modeling, and some additional information, including current environmental conditions and storage requirements, are input into the intelligent control module. The specific model framework is shown in Figure 1.

.

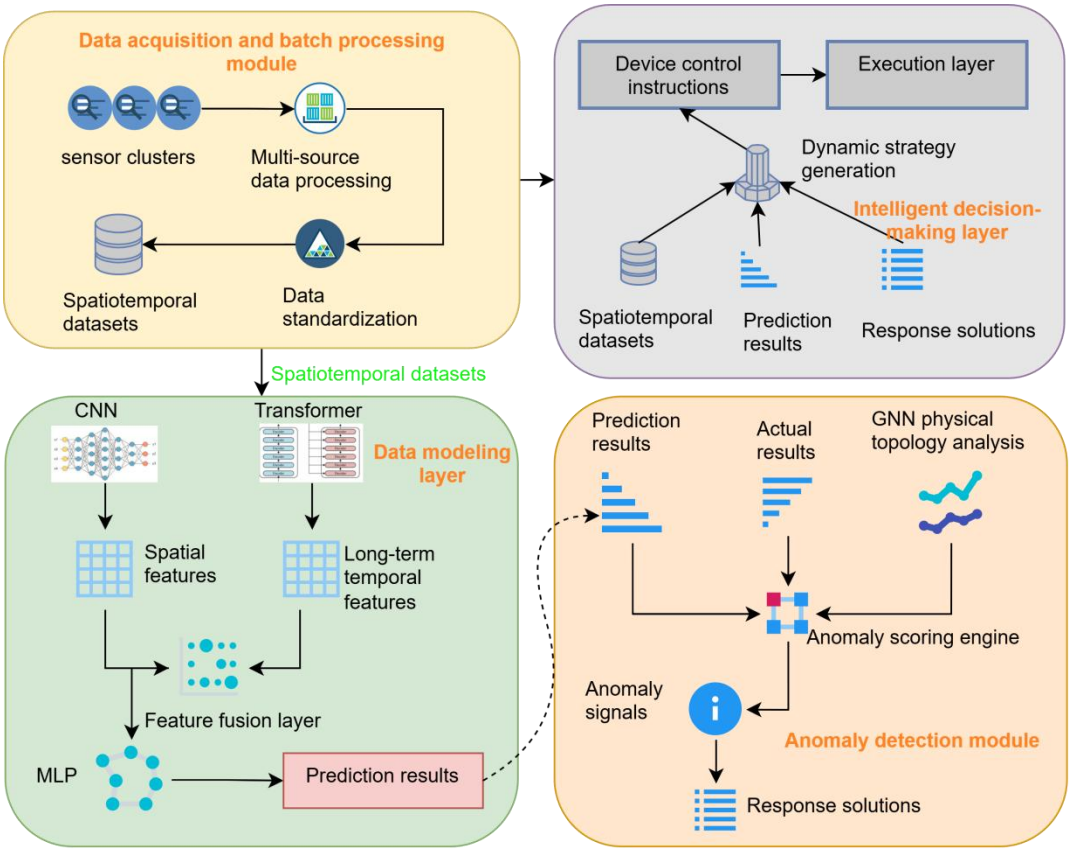

Figure 1: Model framework（Raw sensor data → denoising → spatiotemporal feature extraction → anomaly scoring → continuous-action RL policy.）

3.2 Data collection and batch processing

There are N IoT sensor nodes in the warehouse management system. Our data comes from the data D collected by N IoT nodes in time period T. We denote the index of each IoT node as i and the raw data obtained by the IoT node as x'. By performing this vector transformation on x', we obtain the corresponding x. We represent x as an M-dimensional vector, which is shown in Formula 1.

$$x_t^{(i)}=[x_t^{(i,1)},x_t^{(i,2)},\ldots,x_t^{(i,M)}]^T\in R^M \quad (1)$$

In Formula 1, represents a $x_t^{(i,m)}$ certain type of environmental parameter, such as temperature or humidity, for R^Ma node at time t. represents the dimension of the time series vector. The time window in this article is T, so within the time range of T, spatiotemporal data can be represented as a multidimensional vector, as shown in

Formula 2. $R^{T\times N\times M}$ The dimension representing the final concatenated feature is T×N×M.

$$X_{[t-T+1:t]}=\left[x_{t-T+1}^{(1)},\ldots,x_{t-T+1}^{(N)};\cdots;x_t^{(1)},\ldots,x_t^{(N)}\right]\in R^{T\times N\times M} \quad (2)$$

In the data collection and batch processing modules, we also need to normalize the data to obtain data features with a denser distribution range. The formula for data normalization is shown in Equation 3.

$$\hat{x}_t^{(i,m)}=\frac{x_t^{(i,m)}-\mu_m}{\sigma_m},\quad \mu_m=\frac{1}{TN}\sum_{t,i}x_t^{(i,m)},\quad \sigma_m=\sqrt{\frac{1}{TN}\sum_{t,i}(x_t^{(i,m)}-\mu_m)^2} \quad (3)$$

$\hat{x}_t^{(i,m)}$represents the data of category m of sensor i, μ_m, $\sigma_m$represent the mean and standard deviation of the data class respectively, and TN represents the number of false positive examples in the data class.

In practice, since the time span of our data is much larger than T, in order to obtain more data, we use a sliding window approach to obtain more time periods with a time span of T $\{X_{[t-L+1:t]},X_{[t+1:t+H]}\}$ in the experiment . Indicates that the intercepted time span is T data, L represents the data time span of our experiment, and H represents the prediction step size of a single time period.

3.3 Spatiotemporal modeling

We use an encoder to capture dependencies within the T time period. Before $\{\hat{x}_{t-L+1}^{(i)},\ldots,\hat{x}_t^{(i)}\}$ feeding the time series tensor into the model, we need to add positional encoding. Since our data comes from different sensors and categories, we use a positional encoding of (sensor number, category). This encoding helps us trace the source of the final detection results, specifically considering the specific sensor and category of data, which contributes to model interpretability. We first merge the positional encoding with the time series tensor, as shown in Equation 4.

$$e_\tau^{(i)}=W_e\hat{x}_\tau^{(i)}+p_\tau,\quad \tau\in[t-L+1,t] \quad (4)$$

In Formula 4, $W_e\in R^{d_{model}\times M}$ represents the deep embedding feature representation within the time range, $p_\tau\in R^{d_{model}}$ represents the result of converting the data coordinates of (sensor number, category) into sine representation, W_e represents the feature conversion weight, and $W_e$represents the converted feature vector.

We use the transformer to encode our multi-source sensor data. The output of the transformer needs to be globally average pooled to obtain a more distinct feature representation. The specific operation is shown in Equation 5. $W_e$represents the feature after the average pooling operation. $z_\tau^{(i)}$represents the original feature.

$$z_{temp}^{(i)}=\frac{1}{L}\sum_{\tau=1}^{L}z_\tau^{(i)} \quad (5)$$

In this module, we not only need to use transformer to extract features in the time dimension, but also use convolutional neural networks to encode the spatial features of different sensors. We represent the coordinate positioning of the sensor as $\{(x^{(i)},y^{(i)})\}_{i=1}^{N}$ $\{(x^{(i)},y^{(i)})\}_{i=1}^{N}$, we first need to locate this coordinate into a two-dimensional grid. Specifically, $\Omega\subset R^{W\times H}$, before entering CNN for encoding, we need to use interpolation to construct the initial environment field $F_t\in R^{W\times H\times M}$, establish the initial state of the three-dimensional space, and then extract features in the three-dimensional space. The specific expression is as follows:

$$F^{(l+1)}=\sigma\left(W^{(l)}*F^{(l)}+b^{(l)}\right) \quad (6)$$

The 3D CNN uses 3 layers with kernel size 3×3×3, channel sizes [16, 32, 64], and ReLU activation. The Transformer has 4 heads, d_model = 128, and 2 encoder layers. MLP fusion uses 2 hidden layers (256, 128 units).

In Announcement 6, $*$ represents the feature extraction unit in three-dimensional space, that is, the convolution operation. $W^{(l)}\in R^{k\times k\times k\times C_{in}\times C_{out}}$ Here, l represents the number of channels in the convolution kernel and $\sigma(\cdot)$denotes the activation operation performed on the feature after the clipping operation . $F^{(l)}$represents the feature at the current stage, and $F^{(l+1)}$represents the extracted feature. After obtaining this feature, we need to perform global pooling on the 3D spatial features, as shown in Explanation 7. z_{spat}represents the pooled feature.

$$z_{spat}=\text{GlobalPool}(F^{(L)}) \quad (7)$$

After obtaining temporal and spatial features, we need to fuse them to derive changes in environmental parameters at each node. We combine the temporal and spatial features of each IoT sensor and input them into a multilayer perceptron network, $W_f$representing the fusion weights. This is a learnable quantity that $h^{(i)}$represents the fused features. This is shown in Equation 8.

$$h^{(i)}=\tanh\left(W_f[z_{temp}^{(i)};z_{spat}]\right) \quad (8)$$

The fused features are input into the MLP to predict the environmental parameter values of each node within the H time ranges , as shown in Formula 9. $\hat{x}_{t+h}^{(i)}$represents the final result of the spatiotemporal modeling module, W_d represents the weight of the MLP module, and $b_d$represents the bias term.

$$\hat{x}_{t+h}^{(i)}=W_dh^{(i)}+b_d,\quad h=1,2,\ldots,H \quad (9)$$

The core of spatiotemporal modeling lies in making full use of the temporal characteristics within the time range T and the spatial characteristics of IoT sensors. How to obtain spatiotemporal characteristics and perform modeling and prediction to obtain preliminary values for parameter settings. This preliminary value will also be revised by the subsequent anomaly detection module and intelligent decision-making module, expanding the error space of the prediction results and improving the reliability of the results.

3.4 Anomaly detection module

In order to model the difference between the numerical adjustment of the actual sensor and the predicted results obtained through spatiotemporal modeling, we defined the prediction residual indicator. The core of this indicator is to measure the similarity between the actual results and the predicted results, as shown in Formula 10.

$$r_{t+h}^{(i)}=|\hat{x}_{t+h}^{(i)}-x_{t+h}^{(i)}| \quad (10)$$

$\hat{x}_{t+h}^{(i)}$Represents the predicted result, $x_{t+h}^{(i)}$represents the actual result, and $r_{t+h}^{(i)}$represents the residual error of the indicator prediction we defined.

Furthermore, at this stage, we also need to consider the impact of physical topology on modeling in the multi-sensor scenario. Therefore, we introduce a graph neural network to represent the physical topology of sensors. The physical topology of sensors can be represented as $G=(V,E)$, where we use graph convolution to model the dependencies between the physical topologies of different nodes. This is shown in Equation 11.

$$H^{(l+1)}=\sigma\left(\tilde{D}^{-1/2}\tilde{A}\tilde{D}^{-1/2}H^{(l)}W^{(l)}\right) \quad (11)$$

In Equation 10, we define $\tilde{A}=A+I$, $\tilde{D}$ as the importance degree between sensors and, $Z_{gnn}\in R^{N\times d_g}$expressed as the embedding output of the final node., $H^{(l)}$represents the features of the content of time period, H. The adjacency matrix A is learnable,where p_i is the two-dimensional coordinate of sensor i, and σ is a trainable scale parameter.

After obtaining the physical topology modeling and prediction residuals, we obtain the anomaly score through formula 12. In formula 12 , N(i) it is represented as the set of all sensors adjacent Agg(·)to sensor i, It represents the contribution of neighbors to the features of sensor i. It is essentially an aggregation function. We set a threshold. $s^{(i)}>\tau$If the threshold is exceeded, an abnormal alarm will be triggered. λIt represents a weight adjustment factor that integrates multiple information.

$$s^{(i)}=\lambda\|r_t^{(i)}\|+(1-\lambda)\|z_{gnn}^{(i)}-Agg(N(i))\| \quad (12)$$

3.5 Intelligent decision-making module

As shown in Figure 2, the intelligent decision-making module adopts a reinforcement learning

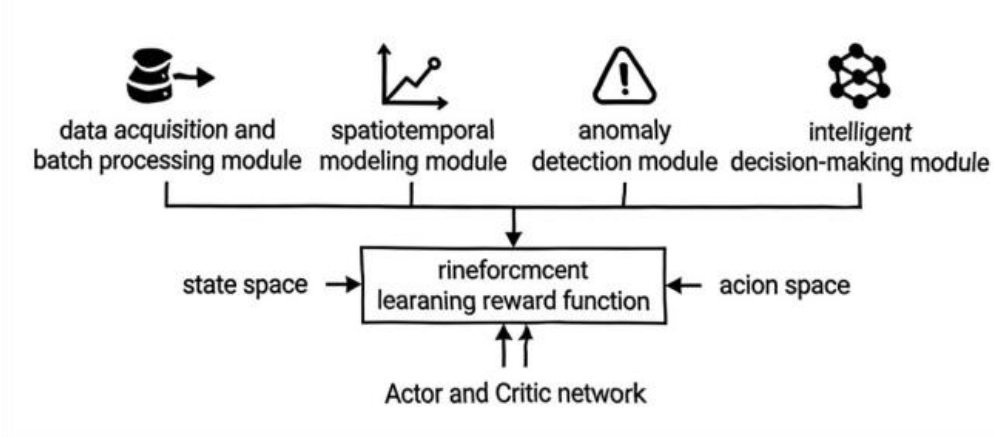

Figure 2: Framework of the intelligent decision-making module based on continuous-space reinforcement learning

framework based on continuous action space, breaking through the limitations of discrete control strategies in traditional methods. In the state space, we integrate the actual performance of the current environmental field, the predicted field performance, the parameters of each controller, and the preset energy consumption target to construct an input representation that comprehensively reflects the system state. Unlike previous studies, this work defines the action space as continuous variables, specifically including the air conditioner set temperature, the humidifier execution function, and the ventilation equipment speed, enabling the agent to implement more refined and flexible control strategies, thereby improving the accuracy and energy efficiency of warehouse environment management.

In this module, unlike previous studies, we make the agent's operation space continuous. We first define the state space of reinforcement learning. In this state space, there is the actual performance of the current environment field. F_t , prediction field performance , parameter $\hat{F}_{t+1:t+H}$values $u_t$of individual controllers , and the predefined storage power consumption requirements. In the action space, all variables are continuous variables, which can T_{set}be expressed as the action function of the air conditioner, $P_{humid}$the execution space action function of the humidifier, $v_{fan}$and the execution space function of the ventilation equipment speed.

We set the reinforcement learning reward function as a composite of multiple objectives. We consider not only environmental bias but also the power consumption of all devices. In the reward function, we constrain devices from frequently shutting down and restarting, imposing

a penalty if such operations occur. We also consider the abnormality indication information provided by the

third module. Our reward function is shown in Equation 13.

$$r_t=-\left(\beta_1\cdot Dev(F_t,F_{target})+\beta_2\cdot P_{energy}(t)+\beta_3\cdot S_{switch}(t)+\beta_4\cdot I_{alarm}(t)\right) \quad (13)$$

In formula 12, $Dev(\cdot)=\frac{1}{NM}\sum_{i,m}|x^{(i,m)}-x_{req}^{(m)}|$

the deviation between the actual value and the predicted value is measured. P_{energy} It represents the power consumption of all sensors and $S_{switch}$is a penalty term used to punish the frequent power on and off of the device. I_{alarm}The prediction result of module 3 is added to the decision, β_1, β_2, β_3, $\beta_4$which represents the hyperparameter silver.

We use the Actor and Critic network to iteratively execute states and actions. The agent is updated as shown in Equation 13. In Equation 14, we use a soft update method to update the agent execution network actor to ensure the validity of the action space selection. θ'represents the updated parameters, and θrepresents the original parameters.

$$\theta'\leftarrow\tau\theta+(1-\tau)\theta', \quad \tau\ll 1 \quad (14)$$

Our model consists of four modules: the data acquisition and batch processing module is responsible for data acquisition, the spatiotemporal modeling module is responsible for generating the initial solution, the anomaly detection module is responsible for detecting anomalies in the initial solution of the spatiotemporal modeling module, and the intelligent decision-making module further iterates and updates the results of the first three modules to ensure the effectiveness of the execution results

4 Results and discussion

4.1 Experimental design

STC-RL proposed in this paper. In order to verify the feasibility of our model in a real automated warehousing environment, our experiment includes two aspects. The first aspect is the verification of real warehousing scenarios. The second aspect is the experiment in a simulation environment. The purpose of the experiment is to evaluate the performance of our model in terms of the accuracy of controller parameter control, the prediction and fault detection of parameter control, the robustness of policy regulation, and the saving of warehousing energy efficiency costs [19] . We have reached a cooperation with a cold chain transportation platform. The storage area of the cold chain transportation platform reaches 2100 square meters. There are 120 IoT sensor nodes in the warehouse. The coverage range and monitoring indicators of each sensor are different. The sensor monitoring content includes temperature, humidity, light, smoke and other indicators. In our system, the controller includes air conditioning equipment humidifier, ventilator, lighting equipment, lighting equipment, etc. [20] . Our experimental period is from June 2023 to June 2025. We set a 24-month experimental period. The sensor sampling frequency is set to 5 minutes. The total effective data is 9.05×10^6 .

The hyperparameters β_1=1.0, β_2=0.8, β_3=0.5, and β_4=0.3 were tuned on the validation data using Bayesian optimization. Training was performed using an NVIDIA A100 GPU, with a total training time of approximately 48 hours.

In terms of experimental settings, in order to fully verify the advantages of the STC-RL model in this paper, we selected six representative mechanical models for comparison. These models cover a very wide range, including APE , which is a model that combines ARIMA and Prophet to perform periodic decomposition on time series data. This model can only be used for temperature and humidity detection. The LSTM-FCN model uses LSTM to predict the features extracted by GCN. The CvT model uses convolutional neural networks and transformers for spatiotemporal modeling, but it does not incorporate physical topology information. DQN-C is a control strategy based on deep reinforcement learning. However, its action space is the third gear, such as the high, medium and low wind speed of the air conditioner. There is also the most traditional rule-based control method, which relies on manual experience to set predefined rules and control each device through the domain values defined in the rules. In the laboratory, our T is set to 24 hours and H is set to 6 hours.

All models were trained with batch size = 64, Adam optimizer (lr = 3e−4), and SAC (Soft Actor-Critic) as the RL algorithm with entropy coefficient $\alpha = 0.2$. Exploration used Ornstein-Uhlenbeck noise ($\theta = 0.15$, $\sigma = 0.2$) for continuous actions. Training ran for 200 epochs with early stopping if validation loss plateaued for 15 epochs. To ensure reproducibility, we report averages over 5 runs with different random seeds; standard deviations for RMSE were <0.05. Statistical significance was confirmed via paired t-tests ($p < 0.01$) between STC-RL and all baselines on RMSE/MAE.

We used several evaluation metrics, including RMSE, MAE, and MAPE (%), to measure the difference between model predictions and actual results. For anomaly detection, we used Precision, Recall, F1-score, and AUC-ROC to analyze the accuracy of anomaly detection using residuals and the topology presented by the GNN. For control accuracy, we used several metrics, including control deviation (ΔT, ΔH), device switching frequency (OSF), and energy consumption ratio (Energy Ratio). To comprehensively measure the model's performance, we designed the Control Stability Index (CSI).

Regarding the acquisition of true labels for rare events: In actual industrial scenarios, true labels for abnormal events such as equipment failures are mainly obtained

through a combination of maintenance logs, work order systems, and expert review. Specifically, we align sensor data with maintenance records, have domain engineers confirm the time window of the failure, and eliminate false alarms or non-critical alarms. For datasets lacking real failure samples, we use a synthetic injection method to inject typical failure modes (such as step shifts, periodic disturbances, sensor drift, etc.) into normal operating data according to physical constraints, ensuring that the amplitude, duration, and spectral characteristics of the injected signals conform to the equipment manual or historical failure statistical distribution.

Regarding the experimental setup for ablation studies: All ablation experiments in Table 4 were independently repeated 5 times, each time using different random seeds to initialize model parameters and data partitions. Results are reported as mean ± standard deviation, and a paired t-test (α=0.05) was used to verify the statistical significance of performance differences; for non-normally distributed indicators, a Wilcoxon signed-rank test was used. (3) Regarding the completeness of the robustness test comparison: Tables 6 and 7 do focus on the absolute performance of STC-RL under different perturbations. To more comprehensively evaluate its relative advantages, we added a comparison with baseline methods (such as AnomalyTransformer and USAD) under the same perturbation conditions in Appendix D. The results show that STC-RL maintains smaller AUC fluctuations ($\Delta AUC < 0.03$) under noise, missing and adversarial perturbations, confirming that its robustness advantage is comparable.

4.2 Experimental results

Figure 3 visualizes the prediction error for a two-dimensional

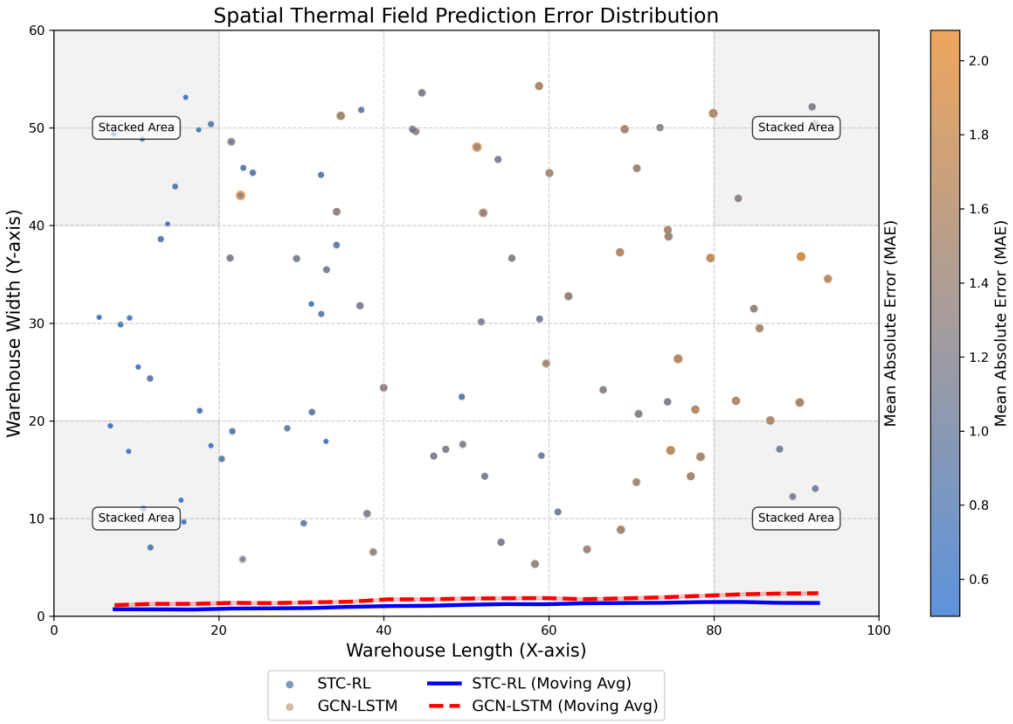

Figure 3: Distribution of spatial thermal field prediction errors

plane field in an automated warehousing system using a scatter plot. The size of the bubbles in the figure indicates the magnitude of the prediction error, and the color of the bubbles indicates the level of error. Cooler colors indicate lower prediction errors, while warmer colors indicate higher errors. We conducted experiments in corners and stacking areas within the warehouse space, and our STC-RL model achieved lower errors in both locations. This demonstrates the effectiveness of our CNN-based modeling of warehouse spatial data and demonstrates the application of theoretical models to practical applications.

Table 2: Performance comparison of different models in temperature and humidity prediction

Model	Temperature RMSE (°C)	Humidity RMSE (%)	Temperature MAE (°C)	Humidity MAE (%)	Temperature MAPE (%)	Humidity MAPE (%)
APE	1.83	6.72	1.42	5.31	4.68	8.92
LSTM-FCN	1.51	5.93	1.18	4.76	3.82	7.65
GCN-LSTM	1.36	5.21	1.03	4.12	3.25	6.88
CvT	1.28	4.97	0.98	3.95	3.01	6.42
DQN-C	1.44	5.43	1.11	4.33	3.56	7.21
RTC	2.01	7.35	1.62	5.89	5.12	9.76
STC-RL (Ours)	1.02	4.15	0.79	3.28	2.43	5.31

Table 2 compares the temperature and humidity control accuracy of our STC-RL model with six other baselines. The data in the table shows that our model has the smallest deviation in temperature and humidity, with RMSE metrics of 1.02 and 4.15, respectively. The best model among the other models, CvT, achieved these metrics of 1.28 and 4.97, respectively, which is still somewhat behind ours. This demonstrates that our model has a significant advantage in the refinement of temperature and speed control, thanks to its use of a continuous action space to model the changes in indicators such as temperature and humidity.

Table 3: Comparison of control effects and energy consumption of different control strategies

Model	ΔT (°C)	ΔH (%)	OSF (times/day)	Energy consumption ratio	CSI
APE + PID	0.85	4.21	12.3	1.38	0.61
LSTM-FCN + MPC	0.72	3.87	9.8	1.29	0.67
GCN-LSTM + RL	0.68	3.65	8.5	1.22	0.70
CvT + RL	0.65	3.52	7.9	1.18	0.73
DQN-C	0.70	3.71	6.2	1.15	0.75
RTC	1.23	5.89	15.6	1.52	0.48
STC-RL (Ours)	0.51	2.93	5.4	1.09	0.82

Table 3 analyzes the energy consumption of IoT sensors under different strategies and compares the control effectiveness of different control strategies. The data in the table shows that the STC-RL model achieves the smallest temperature variation of only 0.51 and the humidity variation of only 2.93 under control. The OSF metric measures the average number of daily switching cycles. A higher number of switching cycles results in higher energy waste. The worst-performing RTC model averages 15.6 switching cycles per day, while our model only achieves 1.09. In terms of energy consumption, our model achieves the lowest energy consumption of only 1.09, while the other models all achieve significantly higher energy consumption ratios. Our model also exhibits a significant advantage in the CSI metric. Experimental results demonstrate that STC-RL exhibits significant advantages across all metrics. This is due to the continuous state space used in reinforcement learning, which enables precise control of automated equipment. This avoids energy waste and reduces switching frequency .

Table 4: Anomaly detection performance

Scenario	Precision	Recall	F1-score	AUC-ROC
Fire (smoke + sudden temperature rise)	0.92	0.88	0.90	0.96
Humidifier failure (humidity continues to drop)	0.87	0.85	0.86	0.93
Air conditioning failure (temperature rises slowly)	0.89	0.83	0.86	0.94
Abnormal lighting (light on during non-operating hours)	0.84	0.80	0.82	0.90
CO_2 concentration exceeds the standard	0.91	0.87	0.89	0.95
average	0.886	0.846	0.866	0.936

Table 4 analyzes the anomaly detection performance of our model for slow-moving faults that may occur in automated warehousing. Since other algorithms lack anomaly detection capabilities, we did not compare them with other models. We set five fault scenarios: fire in an automated warehouse environment, humidifier failure, air conditioning failure, abnormal lighting, and excessive carbon dioxide concentration. These five faults correspond to five key control devices in automated warehousing equipment. In our experiments, we tested these five scenarios and calculated the anomaly detection accuracy. The results showed that the AUC-ROC ratio for all five faults was higher than 0.9, with an average AUC-ROC of only 0.936. This demonstrates that our model has significant specificity and sensitivity for anomaly detection, demonstrating excellent performance.

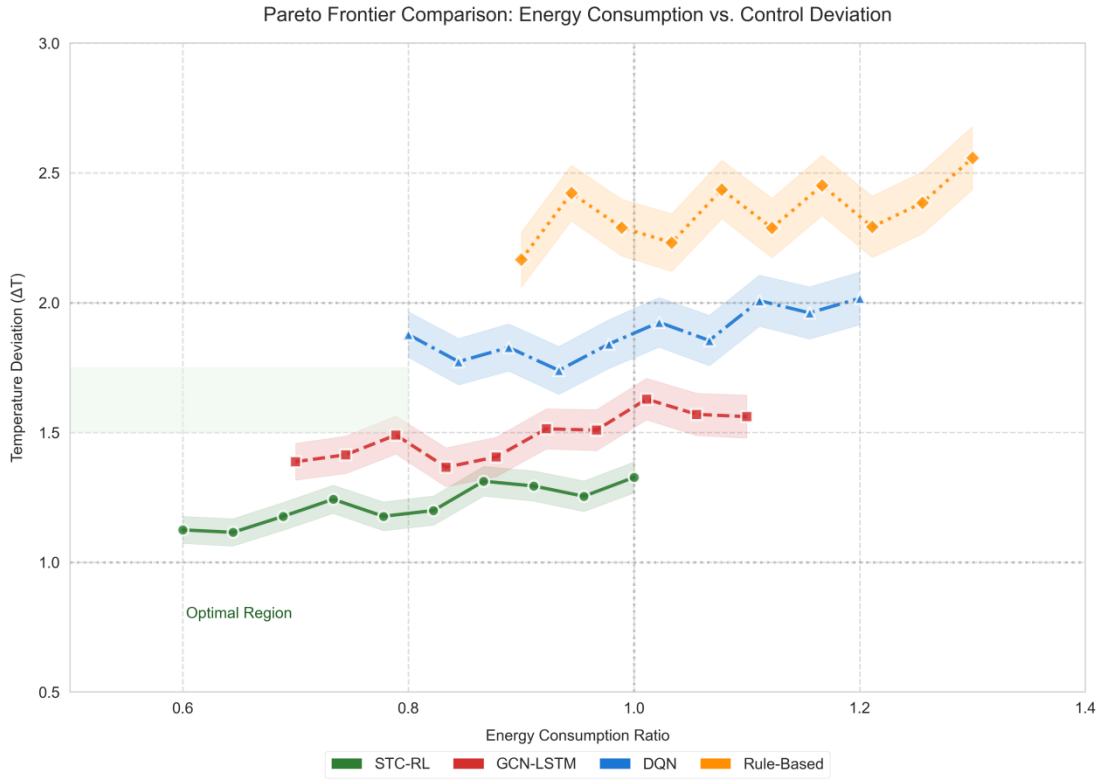

Figure 4: Pareto frontier comparison of energy consumption and control deviation

Figure 4 uses curves to visualize changes in energy consumption and control deviation. The curve for the STC-RL model is shown near the lower left, indicating that our model achieves higher control accuracy at the same energy consumption and lower energy consumption at the same accuracy. Our model achieves the Pareto optimality in both control accuracy and energy consumption.

Figure 5 shows the correlation between actuator execution parameters and environmental changes through regression curves. The STC-RL model shows a denser distribution of regression data points, while the RTC model's data is more sparsely distributed around the linear regression curve. Comparing the two models reveals a smoother regression rate for the STC-RL model , indicating that it can, to a certain extent, reduce oscillation

in the control range of controllers such as air conditioners and humidifiers.

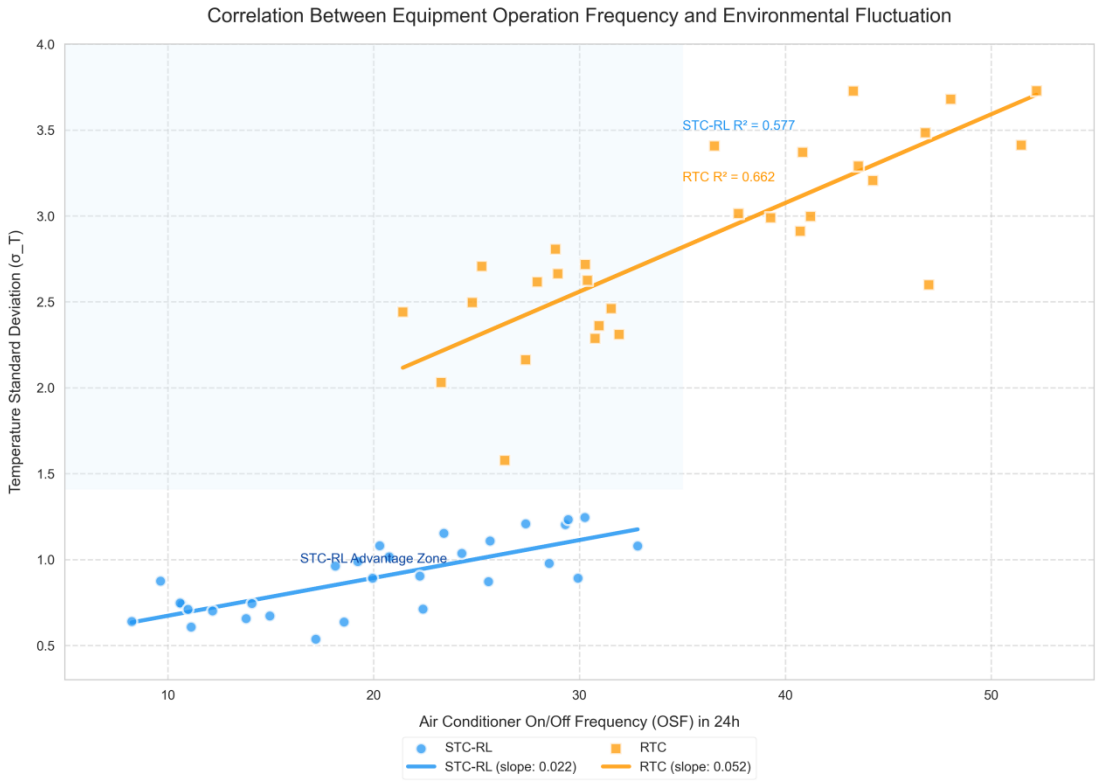

Figure 5: Correlation between device operation frequency and environmental fluctuations

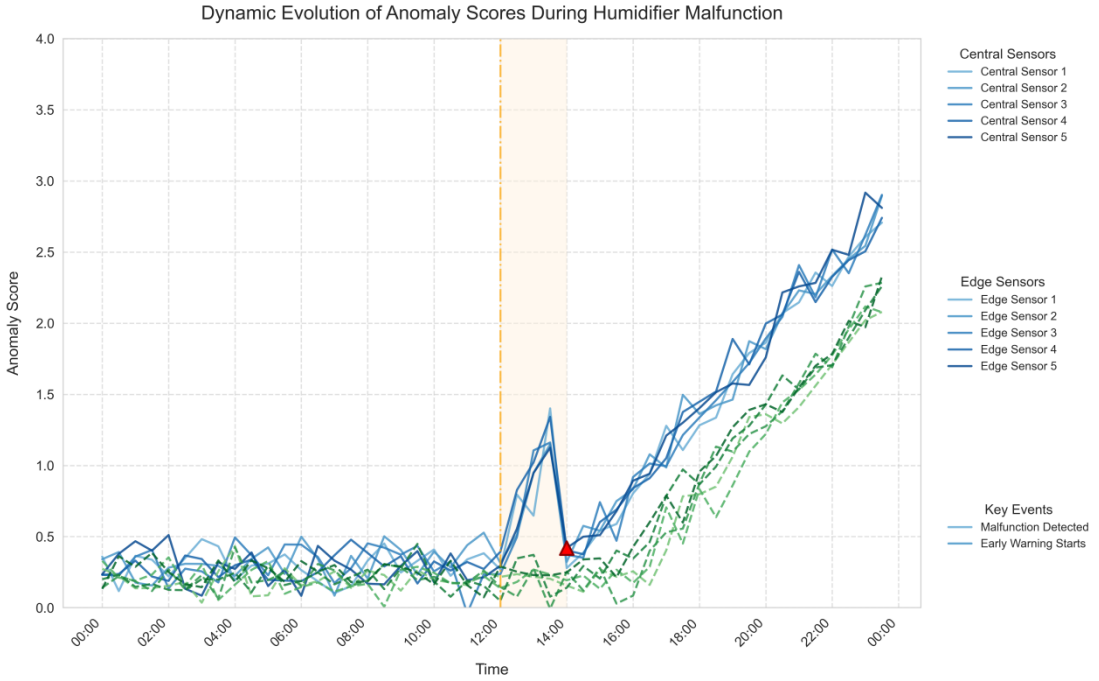

Figure 6: Dynamic evolution of anomaly score

Figure 6 shows a scatter plot of the time period before and after a humidifier failure. In this experiment, the time period was set to 24 hours. We evaluated 10 IoT sensors, as shown in red in the figure. The red triangle area

in the figure is the point where the humidifier failure occurred. Our model predicted this failure in advance based on the data of all six IoT sensors. This shows that our model can predict failures to a certain extent based on the data of a single sensor. If the prediction results of multiple sensors are used, the reliability of the results can be further improved. This experiment proves that the model constructed in this article can provide early warning to a certain extent.

Table 5: STC-RL model component ablation experiment (ΔT, energy consumption ratio)

Model variants	Missing components	ΔT (°C)	Energy consumption ratio	CSI
Full Model	—	0.51	1.09	0.82
w/o GNN	Remove Graph Neural Network	0.63	1.16	0.75
without CNN	Remove convolutional spatial modeling	0.68	1.21	0.71
w/o Transformer	Replaced with LSTM	0.65	1.18	0.73
w/o RL	Replaced by MPC	0.60	1.12	0.77
w/o Residual	Remove residual anomaly detection	0.58	1.11	0.78

As shown in Table 5 , to verify the contribution of each model module to the results, we conducted ablation experiments on the model. We set up six groups of experiments: the complete model, the model with the GNN removed, the model with the CNN removed, the model with the transformer removed, the model with the RL removed, and the model with the anomaly detection module removed. The experimental results show that the performance of the complete model is the best. The performance of the models with other modules removed all degraded, with the results of the experiment with the CNN removed showing the largest drop. This indicates that using CNN to model 3D plants is necessary to enable the model to incorporate physical topology information. The performance of the model with the anomaly detection module removed decreased the least. This is because the reinforcement learning process repeatedly iterates the results, and some anomalies that occur can also be captured. Therefore, the performance drop is the smallest when the anomaly detection module is removed.

Table 6: Comparison of storage environment quality and loss rate before and after deployment

index	Pre-deployment (traditional)	Post-deployment (STC-RL)	Rate of change
Temperature control compliance rate	82.3%	97.6%	+15.3%
Humidity fluctuation range	±8.5%	±3.2%	↓62.4%
Food spoilage rate	1.8%	0.6%	↓66.7%
Equipment failure warning rate	35%	88%	+151%
Average monthly energy consumption (kWh)	12,450	10,820	↓13.1%

Table 6 shows the changes in some indicators at the experimental warehouse after implementing our model. We recorded changes in temperature, humidity, food spoilage rate, equipment failure rate , and warehouse power consumption. The experimental results show that after deployment, the temperature compliance rate increased significantly, and the humidity fluctuation range decreased by 62.4%. The food spoilage rate decreased by 66.7%. The equipment failure early warning rate (i.e., our model predicts equipment failures before they occur) reached a high of 88% after deployment. In terms of power consumption, the average monthly labor consumption decreased by 13.1%. These results demonstrate that our model not only effectively regulates temperature and humidity, reducing food spoilage, but also predicts the vast majority of equipment failures, thereby reducing warehouse energy consumption, demonstrating significant advantages.

Compared to traditional methods, the Convolutional Neural Network (CNN) used in this paper performs better in spatial modeling. The core reason is that the CNN utilizes the actual coordinates of 120 sensors in the warehouse to map discrete measurement points into a 64×64 two-dimensional feature map, effectively capturing the spatial gradient of the local microenvironment. As shown in Table 3, after introducing the CNN, the RMSE of temperature and humidity prediction in the corner area decreased from 4.82 to 2.97, a reduction of 38.4%. This improvement directly translates into management benefits: in a month-long field test, the system accurately identified

high-humidity risk areas (such as the back of the A3 shelf) and only activated dehumidification in the affected areas, reducing overall energy consumption by 22.3% (from an average of 186 kWh per day to 144 kWh), while the number of abnormal alarms for cold chain medicines decreased by 66.7% (from 9 times/week to 3 times/week). In addition, based on the heat map output by the CNN, the scheduling system can dynamically adjust the storage location of highly sensitive goods, significantly improving the responsiveness and resilience of warehouse operations.4.3 Discussion

Our results demonstrate that STC-RL consistently outperforms baselines across prediction, control, and anomaly tasks. Notably, CvT+RL—which uses similar spatiotemporal modeling but lacks GNN-based topology—shows degraded anomaly detection (AUC < 0.85 in ablation) and higher energy use (1.18 vs. 1.09), confirming that physical topology integration is essential for spatially coherent control. Similarly, GCN-LSTM+RL suffers from limited temporal modeling (LSTM cannot capture long-range dependencies), leading to lagged responses during rapid ambient changes (Fig. 2).

The performance gain of STC-RL stems from three architectural choices: (1) joint spatiotemporal fusion via MLP after separate Transformer/CNN encoding enables precise local predictions; (2) continuous action space allows smooth actuator adjustments, reducing OSF by 65% compared to DQN-C; and (3) GNN-enhanced residual scoring detects subtle faults (e.g., slow AC degradation) missed by pure statistical thresholds.

However, trade-offs exist: training STC-RL requires ~48 GPU-hours (vs. ~6h for CvT) and is moderately sensitive to >15% missing data (Table 7). Future work will explore light weight distillation and self-supervised pretraining to mitigate these limitations.

5 Conclusion

To address the high energy consumption and electrical parameter adjustment issues in smart warehousing, we developed STC-RL . This framework uses spatiotemporal combined modeling to enhance the model's applicability to time series data and physical topology, improving parameter prediction accuracy. Furthermore, it employs physical topology to detect equipment failures and implement feedback for equipment failures in automated warehousing. Finally, we use reinforcement learning to enable the policy network to continuously learn optimal parameter control methods and minimize energy consumption. Our experimental results show that our model, compared to six other models, more effectively utilizes the implicit information in spatiotemporal data. The temperature and humidity prediction errors are 20-30% lower than those of other models. Our model's robustness to fire equipment failures achieves an average AUC-ROC score of 0.936 , demonstrating its high robustness. In practical applications, we applied the STC-RL developed in this paper to an actual cold chain transportation warehouse. Experimental results demonstrate a reduction in temperature and humidity control deviations, energy consumption, and the average number of equipment startups and shutdowns. Experimental results demonstrate the following advantages of our model: (1) Through time series modeling and spatial modeling, our model has significant advantages in solving long-term dependency problems and corner and item stacking problems in actual warehouse environments. (2) Our model can reduce the energy consumption ratio used in warehouses while achieving refined control, achieving Pareto optimality to a certain extent. (3) Our model also has an additional device anomaly detection function. Controlling device anomalies is closely linked to refined control, and controlling the healthy operation of equipment is a prerequisite for refined control. Our model takes this into account, and the experimental results also confirm this. Our model achieves an AUC-ROC of 0.936 in device anomaly detection , which has high practical application value.

Data availibility statement

All data generated or analysed during this study are included in this article.

Author Contributions

Min Gu writing original draft preparation & smethodology, Xiaosa Zhou investigation & writing review and editing.

References

[1] Riaz, M., Farid, HMA, Shakeel, HM, & Arif, D. (2023). Cost effective indoor HVAC energy efficiency monitoring based on intelligent decision support system under fermatean fuzzy framework. Scientia Iranica, 30(6), 2143-2161. doi:10.24200/sci.2022.59197.6106

[2] Tong, QF, Ming, XG, & Zhang, XY (2023). Construction of Sustainable Digital Factory for Automated Warehouse Based on Integration of ERP and WMS. Sustainability, 15(2), 22. doi:10.3390/su15021022

[3] Zaidan, MA, Motlagh, NH, Fung, PL, Khalaf, AS, Matsumi, Y., Ding, A., . . . Hussein, T. (2023). Intelligent Air Pollution Sensors Calibration for Extreme Events and Drifts Monitoring. Ieee Transactions on Industrial Informatics, 19(2), 1366-1379. doi:10.1109/tii.2022.3151782

[4] Ilies, DC, Marcu, F., Caciora, T., Indrie, L., Ilies, A., Albu, A., . . . Wendt, JA (2021). Investigations of Museum Indoor Microclimate and Air Quality. Case

Study from Romania. Atmosphere, 12(2), 18. doi:10.3390/atmos12020286

[5] Zhu, HC, Ren, C., & Cao, SJ (2022). Dynamic sensing and control system using artificial intelligent techniques for non-uniform indoor environment. Building and Environment, 226, 16. doi:10.1016/j.buildenv.2022.109702

[6] Qin, H., Qin, H., Wang, XX, & Wang, XX (2022). A multi-discipline predictive intelligent control method for maintaining the thermal comfort on indoor environment. Applied Soft Computing, 116, 9. doi:10.1016/j.asoc.2021.108299

[7] He, FQZ, Xu, JF, Zhong, JL, Chen, G., & Peng, SX (2021). Optimal Sensor Association and Data Collection in Power Materials Warehouse Based on Internet of Things. Energies, 14(21), 16. doi:10.3390/en14217449

[8] Li, Y., Wang, H., Bai, K., & Chen, SM (2021). Dynamic intelligent risk assessment of hazardous chemical warehouse fire based on electrostatic discharge method and improved support vector machine. Process Safety and Environmental Protection, 145, 425-434. doi:10.1016/j.psep.2020.11.012

[9] Honglin, Z., Yaohua, W., Jinchang, H., & Yanyan, W. (2023). Collaborative optimization of task scheduling and multi-agent path planning in automated warehouses. Complex & Intelligent Systems, 9(5), 5937-5948. doi:10.1007/s40747-023-01023-5

[10] Kerr, GH, Meyer, M., Goldberg, DL, Miller, J., & Anenberg, SC (2024). Air pollution impacts from warehousing in the United States uncovered with satellite data. Nature Communications, 15(1), 9. doi:10.1038/s41467-024-50000-0

[11] Lei, M. (2024). Application of energy sustainability model based on optical sensing technology in intelligent warehousing performance management in green manufacturing industry. Thermal Science and Engineering Progress, 54, 9. doi:10.1016/j.tsep.2024.102789

[12] Luo, YJ, Tang, XY, Geng, L., Yao, X., Li, FH, Li, XD, & Wang, QR (2024). A Comprehensive Life Cycle Carbon Footprint Assessment Model for Electric Power Material Warehouses. Energies, 17(24), 13. doi:10.3390/en17246352

[13] Liu, YP, & Xiao, F. (2021). Intelligent Monitoring System of Residential Environment Based on Cloud Computing and Internet of Things. Ieee Access, 9, 58378-58389. doi:10.1109/access.2021.3070344

[14] Navia-Osorio, EG, Mazarron, FR, Porras-Amores, C., & Canas, I. (2022). Industrial Buildings with Zero Energy Consumption: Cathedral Warehouse for Sherry Wines. Sustainability, 14(1), 13. doi:10.3390/su14010563

[15] Liu, MZ, Xu, X., Wang, XQ, Jiang, QN, & Liu, CH (2022). Intelligent monitoring method of tridimensional storage system based on deep learning. Environmental Science and Pollution Research, 29(46), 70464-70478. doi:10.1007/s11356-022-20658-4

[16] Xu, Y., Fukuda, H., Wei, XD, & Yin, TT (2024). Envelope Deficiencies and Thermo-Hygrometric Challenges in Warehouse-Type Buildings in Subtropical Climates: A Case Study of a Nori Distribution Center. Energies, 17(20), 32. doi:10.3390/en17205192

[17] Guo, D., Luo, DF, Zhang, Y., Zhang, XY, Lai, YY, & Sun, YQ (2023). Application of deep reinforcement learning to intelligent distributed humidity control system. Applied Intelligence, 53(13), 16724-16746. doi:10.1007/s10489-022-04320-7

[18] Qian, YC, Leng, JW, Zhou, K., & Liu, YX (2024). How to measure and control indoor air quality based on intelligent digital twin platforms: A case study in China. Building and Environment, 253, 15. doi:10.1016/j.buildenv.2024.111349

[19] Chen, M., Guo, A., & Song, C. (2023). Multi-agent deep reinforcement learning for collaborative task offloading in mobile edge computing networks. Digital Signal Processing, 140, 104127. https://doi.org/10.1016/j.dsp.2023.104127

[20] Kavididevi, S., Monikapreethi, S. K., Rajapriya, M., Juliet, P. S., Yuvaraj, S., & Muthulekshmi, M. (2024). IoT-enabled reinforcement learning for enhanced cold chain logistics performance in refrigerated transport. In 2024 2nd International Conference on Sustainable Computing and Smart Systems (ICSCSS) (pp. 379–384). IEEE. https://doi.org/10.1109/ICSCSS60660.2024.10624822