Published online on the page : https://ejournal.uinbukittinggi.ac.id/index.php/ijokid/

Knowbase : International Journal of Knowledge in Database

| ISSN (Print) 2798-0758 | ISSN (Online) 2797-7501 |

Lightweight Hybrid SVM-LSTM Edge AI for Real-Time Smart Greenhouse Anomaly Detection

Irfan Fahmi Ahmadi[1,], Seno Adi Putra[2], and Atam Rifa'i Sujiwanto[3]*

[1,2] *Telkom University, Indonesia*
[3] *PLN UP3 Tolitoli, Indonesia*

Article Information

Article History:

Submitted: March 13, 2026
Revision: April 30, 2026
Accepted: May 15, 2026
Published: June 28, 2026

Keywords

smart greenhouse
edge AI
lightweight machine learning
hybrid SVM-LSTM
real-time detection

Correspondence

E-mail: irfanfahmiahmadi@student.telkomuniversity.ac.id *

A B S T R A C T

Smart greenhouse systems require an artificial intelligence model that can transform sensor records into real-time abnormal-condition decisions while remaining practical for edge computing devices. This paper proposes a lightweight Hybrid SVM-LSTM Edge AI model for real-time smart greenhouse anomaly detection. The model integrates packet validation, deterministic feature fusion, SVM-based discriminative scoring, and LSTM temporal sequence learning to classify greenhouse states as normal or abnormal. The feature representation combines current sensor values, short-term sensor changes, sampling-gap information, and daily periodic patterns so that the model can capture both instantaneous conditions and recent environmental transitions. Using the IoT Agriculture 2024 smart greenhouse dataset, the evaluation compares several Hybrid SVM-LSTM and LSTM-only configurations under proxy labels for actuator-relevant abnormal states. The refined water-related proxy achieved 0.979 accuracy, 0.763 precision, 0.690 recall, and a 0.725 F1-score with 1.142 ms P95 inference time. The broader NPK-gap proxy achieved 0.983 accuracy, 0.886 precision, 0.549 recall, and a 0.678 F1-score with 1.448 ms P95 inference time. These results indicate that the proposed Hybrid SVM-LSTM model is sufficiently compact and accurate for real-time anomaly detection at the edge, especially when proxy-label design and threshold selection are aligned with the target greenhouse condition.

This is an open access article under the CC-BY-SA license

1. Introduction

Smart greenhouse systems keep plants within a controlled microclimate by continuously observing environmental and nutrient-related variables. In a real greenhouse, prediction output is not merely a monitoring result: it can directly trigger physical actions such as activating a fan, opening irrigation, or adjusting nutrient delivery. [1], [2], [3]. Earlier smart farming platforms further show that greenhouses benefit from layered local, edge, and cloud components rather than a single centralized server [4].

This layered requirement is consistent with the edge computing paradigm, which processes data near its source to reduce dependence on remote infrastructure, lower bandwidth burden, and support response-time-sensitive IoT operation [5]. In precision agriculture, edge and cloud-edge-device studies emphasize localized processing, distributed intelligence, and the separation between local real-time tasks and cloud-based management functions [6], [7], [8]. Edge AI is therefore increasingly viewed as an enabler of sustainable agriculture, improving production efficiency while reducing cloud dependency, particularly for processing sensor data in constrained field conditions [9], [10].

Smart greenhouse research has also moved from passive monitoring toward prediction and actuator-oriented decision support, including real-time warning prediction [11], temporal modeling of greenhouse temperature [12]edge-based climate control with fault detection and energy-aware actuation [13] autonomous

IoT climate-control platforms [14] and reinforcement-learning-based climate optimization [15] In the Indonesian context, MQTT-based node architectures have been studied for bandwidth-efficient greenhouse communication [16]while cloud-based management platforms still depend on remote infrastructure for several functions [17]. Deep learning studies on actuator prediction further show that actuator states involve temporal environmental patterns and class imbalance, making the problem more complex than rule-based control [18].

However, these studies generally position prediction as monitoring support, warning support, or heavy deep-learning control. The machine learning model itself is rarely examined as a compact edge-side inference workload evaluated through feature design, proxy-label construction, threshold policy, anomaly metrics, and computational cost during inference - even though edge-enabled agriculture frameworks [19] and TinyML and lightweight edge-AI literature [20], [21] confirm that such local inference is both feasible and beneficial in resource-constrained settings. The research gap addressed in this paper is therefore the absence of a focused lightweight Hybrid SVM-LSTM Edge AI model for actuator-oriented greenhouse abnormal-condition detection.

To fill this gap, this paper develops and evaluates a Lightweight Hybrid SVM-LSTM Edge AI controller for smart greenhouse actuation. The proposed controller integrates packet validation, feature fusion, SVM discriminative scoring, LSTM temporal learning, and normal-or-abnormal output generation within a compact edge-side pipeline, while preserving greenhouse-specific temporal information through current sensor features, short-term changes, sampling-gap features, and daily periodic patterns. Its feasibility is assessed by evaluating multiple model configurations on accuracy, precision, recall, F1-score, MCC, false-positive rate, false-negative rate, and P95 inference time. Therefore, this paper focuses on the feasibility of the proposed AI model as a compact edge-side decision component for smart greenhouse actuation.

2. Method

2.1 Research Design

This study uses a simulation-based experimental design to evaluate a lightweight Hybrid SVM-LSTM Edge AI controller for smart greenhouse actuation. The evaluation focuses on model feasibility by examining whether the proposed workload can detect actuator-relevant abnormal conditions using compact feature engineering and low computational cost during inference. The AI component uses the IoT Agriculture 2024 smart greenhouse dataset as the experimental data source [22]. Since greenhouse sensor records are time-dependent, the experimental design follows temporal modeling principles used in greenhouse environmental prediction studies, where sequence order, short-term transitions, and time-aware features are important for reliable prediction [23], [24]. The study does not evaluate a wireless communication protocol or a cloud-computing architecture. Instead, the experimental focus is limited to the AI pipeline that can be embedded in an edge-capable greenhouse controller.

The proposed controller is treated as a local intelligence component that can be placed on a gateway, microcomputer, industrial edge box, or other edge-capable device near the greenhouse field. Sensor readings are assumed to arrive at the controller as packets or records. The controller then validates the record, builds fused features, applies the Hybrid SVM-LSTM model, and produces a normal or abnormal decision. This design allows the paper to examine the AI workload itself, including how feature fusion and threshold selection affect abnormal-condition detection under imbalanced proxy labels.

2.2 *Proposed Lightweight Hybrid SVM-LSTM Edge AI Controller*

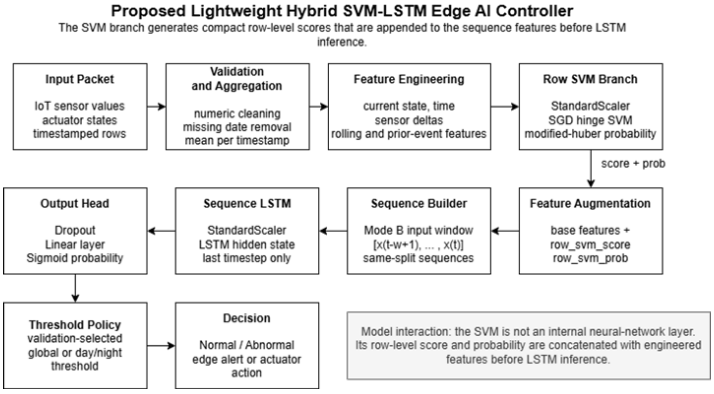

Figure 1. Proposed Lightweight Hybrid SVM-LSTM Edge AI controller

Figure 1 illustrates the proposed Lightweight Hybrid SVM-LSTM Edge AI controller. The model starts with packet validation to check whether the incoming greenhouse record is structurally valid and can be trusted for inference. This stage is important because actuator-oriented decisions should not be triggered by corrupted packets, inconsistent records, or unsupported sources. After validation, the feature fusion stage combines the current sensor state with temporal-change information, sampling-gap features, and periodic time features. The fused feature vector is then passed to the Hybrid SVM-LSTM block. The SVM part produces a compact discriminative score, while the LSTM part captures short-term temporal dependency before the output layer generates a normal or abnormal decision.

The AI model in the proposed system is developed to be compatible with edge computing rather than relying on a cloud-centric prediction process. The model is adapted from the general idea of anomaly-aware IoT real-time prediction, but it is redesigned for smart greenhouse decision support. Instead of processing only raw sensor values, the model combines base sensor features with temporal-change features, such as sampling interval, time pattern, and changes in environmental variables. This feature design is important because greenhouse actuator decisions are influenced not only by the current sensor condition, but also by how quickly the condition changes. For example, a rapid increase in temperature or a sudden decrease in water level may require different actuator interpretation than a stable condition with similar absolute values.

The controller is intentionally lightweight. The packet validation stage uses simple integrity and source checks. The feature fusion stage uses deterministic transformations that can be computed directly from the current record and recent historical context. The SVM component provides a compact discriminative representation without requiring a large deep neural network. The LSTM component is limited to short sequence windows so that the model can preserve temporal information while remaining practical for edge-side deployment. In this way, the proposed model emphasizes the balance between anomaly-detection capability and computational simplicity.

2.3 *Dataset and Preprocessing*

The raw dataset consisted of 37,923 timestamped sensor records covering variables relevant to the greenhouse control loop, especially water availability, nutrient-related information, temperature, humidity, and timestamped sensor conditions. Because the dataset does not provide ground-truth actuator commands, the experiment formulates proxy labels that represent abnormal greenhouse states requiring actuator-oriented attention. Before feature construction, the dataset was chronologically sorted to preserve temporal dependency and reduce temporal leakage. Invalid records, duplicated timestamps, and incomplete sensor readings were removed, reducing the dataset from 37,923 raw records to 28,682 valid records (a removal of 9,241 records, or 24.4% of the raw data). After preprocessing, 28,682 timestamped records remained. These records were divided chronologically into 20,077 training samples (70%), 4,302 validation samples (15%), and 4,303 test samples (15%). The training subset covered records from November 27, 2023, to January 29, 2024.

The validation subset covered January 29 to February 13, 2024, and the test subset covered February 13 to March 30, 2024.

The remaining records were transformed into edge-compatible features including current sensor values, temporal-change variables, sampling-gap information, and daily periodic features. This preprocessing strategy is aligned with greenhouse temporal-prediction studies that treat environmental records as ordered sequences rather than independent samples [23]Training, validation, and test separation followed the chronological order so that future records were not used to construct past predictions. Threshold values for proxy labels were estimated from the training subset and then consistently applied to validation and test samples.

2.4 *Feature Fusion and Proxy Label Construction*

The task is defined as binary real-time detection, where the positive class represents abnormal greenhouse conditions that may require actuation support and the negative class represents normal greenhouse states. Similar anomaly-oriented agriculture studies show that temporal sensor anomalies can be detected using deep learning approaches when the model is trained to identify abnormal behavior in sequential IoT data [25]. The engineered feature vector is represented as follows.

$$x_t = [s_t, \Delta s_t, g_t, c_t] \tag{1}$$

Where x_t denotes the fused edge feature vector, s_t denotes the current greenhouse sensor vector, Δs_t denotes the temporal-change vector, g_t denotes the sampling-gap feature, and c_t denotes the daily periodic feature vector. The temporal-change and periodic components are computed as follows.

$$\Delta s_t = s_t - s_{t-1} \tag{2}$$

$$c_t = [sin\left(\frac{2\pi m_t}{1440}\right), cos\left(\frac{2\pi m_t}{1440}\right)] \tag{3}$$

where m_t represents the minute index within a day. These features allow the model to represent both short-term environmental transitions and daily cyclic patterns while remaining suitable for edge-side computation. The design is also compatible with greenhouse control logic because abrupt changes in water, temperature, humidity, or nutrient-related indicators may carry different operational meaning from stable sensor readings.

Because the dataset does not contain manually annotated abnormal-condition labels, this study constructs proxy labels from operationally meaningful sensor-change rules. The proxy-label design is based on the assumption that sudden changes in temperature, humidity, water level, nutrient indicators, or sampling gaps may indicate unstable greenhouse conditions that require controller attention. Therefore, for each timestep t, an abnormal proxy label is assigned when at least one sensor-change or gap condition exceeds its data-driven threshold.

For each sensor variable j, the absolute temporal change is defined as follows.

$$\mathrm{d}_{j,t} = |\mathrm{s}_{j,t} - \mathrm{s}_{j,t-1}| \tag{4}$$

The threshold for each sensor is obtained from the training split only.

$$\tau_j = \mathrm{Qq}(\mathrm{d}_{j,t}), \quad \mathrm{d}_{j,t} > 0 \tag{5}$$

where Qq denotes the empirical quantile. Zero-change values are excluded so that stable repeated readings do not dominate the threshold estimation. The sampling-gap threshold is defined as follows.

$$\tau_g = \max(5, \mathrm{Qqg}(\mathrm{g_t})) \tag{6}$$

where 5 minutes is used as the minimum expected sampling interval. This prevents very small gap values from being selected as abnormal thresholds.

The positive proxy label is then defined as follows.

$$\mathrm{y_t} = 1 \text{ if any sensor} - \text{change event or sampling} - \text{gap event is true; otherwise } \mathrm{y_t} = 0 \tag{7}$$

For water level, the implementation uses $d_{water} \geq \tau_{water}$, while temperature, humidity, N, P, K, and sampling gap use strict greater-than rules. The final abnormal label is therefore an any-event proxy, meaning that one or more abnormal reasons are sufficient to mark a timestep as abnormal.

To improve reproducibility and avoid arbitrary threshold selection, the threshold values are not manually fixed. Instead, candidate sensor quantiles {0.94, 0.95, 0.96, 0.97, 0.98} and gap quantiles {0.80, 0.85, 0.90, 0.95} are evaluated. Candidate thresholds are considered valid when the training positive rate remains between 3% and 8%, while the validation positive rate remains within a relaxed stability range. Among valid candidates, the selected threshold set minimizes the distance from the target abnormal rate of 5% and penalizes instability between training and validation distributions. The selection objective is as follows.

$$J = |r_{train} - 0.05| + |r_{val} - 0.05| + 2|r_{train} - r_{val}| \quad (8)$$

where r_{train} and r_{val} denote the positive-label rates in the training and validation splits. The held-out test split is not used to select proxy thresholds and is used only for final reporting. This procedure improves validity by ensuring that the proxy labels represent rare but meaningful abnormal events, while also improving reproducibility because the thresholds are derived from explicit quantile rules and chronological train-validation selection.

Table 1. Proxy-label definition and threshold-selection rationale.

Proxy-label component	Implemented rule	Justification for validity and reproducibility
Sensor-change event	$d_{j,t} = \lvert s_{j,t} - s_{j,t-1} \rvert$; event if $d_{j,t} > \tau_j$ for temperature, humidity, N, P, and K.	Represents rare abrupt changes rather than stable readings. Thresholds are fitted from the training split only.
Water-level event	$d_{water} \geq \tau_{water}$	Preserves boundary cases where water-level change exactly reaches the selected threshold.
Sampling-gap event	$g_t > \tau_g$, where $\tau_g = \max(5, Qqg(g_t))$.	The 5-minute minimum reflects the expected interval and avoids marking small gaps as abnormal.
Threshold grid	Sensor quantiles: {0.94, 0.95, 0.96, 0.97, 0.98}; gap quantiles: {0.80, 0.85, 0.90, 0.95}.	Avoids arbitrary fixed thresholds and makes the proxy construction repeatable.
Candidate selection	Training positive rate must remain 3%-8%; validation rate must remain stable. The selected candidate minimizes J.	Keeps abnormal labels rare but not too sparse. The test split remains report-only.
Final binary label	$y_t = 1$ when at least one event condition is true; otherwise $y_t = 0$.	The any-event rule matches edge-controller operation, where any critical sensor transition may require attention or actuation support.

2.5 *Hybrid SVM-LSTM Edge AI Workload*

In this study, the SVM-LSTM model is used as a representative lightweight AI workload for evaluating the proposed Edge AI controller. The purpose of including the AI model is to provide a realistic inference task at the edge, not to position the model as a universal or state-of-the-art predictor for all greenhouse conditions. The SVM component produces a discriminative score that separates normal and abnormal feature patterns, while the LSTM component models the temporal sequence built from recent fused feature vectors. The choice of a hybrid SVM-LSTM structure is supported by greenhouse and environmental modeling studies that combine temporal learning with machine learning classifiers for proactive climate management and time-series prediction [26], [27].

Additional support for this workload comes from lightweight anomaly-detection and evaluation studies. Lightweight LSTM-based anomaly detection has been used for multivariate time-series monitoring in industrial control systems, showing that compact recurrent models can be used for anomaly-oriented sequential data [28]. Because abnormal greenhouse states are expected to be less frequent than normal states, the evaluation also uses imbalanced-classification metrics. Recent studies on imbalanced classification emphasize that performance assessment should not rely only on accuracy, but should also consider precision, recall, F1-score, MCC, false-positive behavior, and false-negative behavior [29], [30].

After feature construction. the SVM component produces a compact discriminative score p_t that represents the instantaneous separation between normal and abnormal greenhouse states. This score is then combined with the temporal feature sequence and passed to the LSTM component to estimate the abnormal-condition probability. Given a sequence window of length k. the input to the LSTM is represented as follows.

$$Z_t = [x_{t-k+1}, p_{t-k+1}], \ldots, [x_t, p_t] \tag{9}$$

The final abnormal-condition probability is computed from the LSTM hidden representation as follows.

$$\hat{y}_t = LSTM(Z_t) \tag{10}$$

The final binary decision is obtained by comparing $\hat{y}_t$ with an optimized threshold selected from the validation set. The hybrid design remains suitable for edge deployment because the SVM component provides compact discriminative information. while the LSTM component captures short-term temporal dependency without requiring a large-scale deep learning architecture.

2.6 *Evaluation Metrics and Model Configuration*

The model evaluation uses accuracy, precision, recall, F1-score, MCC, false-positive rate, false-negative rate, and P95 inference time. Accuracy shows overall correctness, but it is not sufficient for imbalanced abnormal-condition detection. Precision indicates how many abnormal predictions are truly abnormal, which is important for avoiding false actuator triggers. Recall indicates how many abnormal states are successfully detected, which is important for avoiding missed abnormal conditions. The F1-score summarizes the trade-off between precision and recall, while MCC provides an additional balanced measurement under class imbalance. These metric choices follow the need for balanced evaluation in imbalanced classification settings [29][30]. P95 inference time is used as an indicator of inference efficiency because it shows the upper-tail processing time required by the AI workload.

For the reported hybrid SVM-LSTM configuration, the LSTM used a sequence length of 12, a hidden dimension of 96, and a dropout rate of 0.25. Training was performed with the Adam optimizer using a learning rate of (7×10^{-4}), a weight decay of (1×10^{-5}), and a batch size of 2,048. The remaining Adam parameters followed the default PyTorch settings. A class-weighted binary cross-entropy loss was applied to account for the imbalance between normal and abnormal samples, and the gradient norm was clipped at 2.0 during optimization.

Training was limited to a maximum of 16 epochs. The learning rate remained fixed throughout training, without an additional decay scheduler. Early stopping was applied when the validation selection score did not improve for four consecutive epochs. This score combined the validation PR-AUC and MCC as (PR – AUC + 0.5 x MCC), with MCC calculated at a probability threshold of 0.5. The model parameters from the epoch with the highest validation score were restored before final evaluation.

Table 2. Selected Hybrid SVM-LSTM configuration for final proxy tasks.

Proxy Task	Feature Set	Window	Hidden Dim	Dropout	Threshold Policy
Water-related proxy	Hybrid	12	96	0.25	Day=0.830; night=0.815; fallback=0.815
NPK-gap proxy	Hybrid	12	96	0.25	Day=0.840; night=0.885; fallback=0.855

Table 2 lists the selected final Hybrid SVM-LSTM configurations derived from the experimental spreadsheet. Both final proxy tasks use a hybrid feature set with a sequence window of 12, hidden dimension of 96, dropout of 0.25, learning rate of 0.0007, and batch size of 2048. The difference is mainly in the threshold policy, because the water-related proxy and the broader NPK-gap proxy require different operating points.

This threshold design allows the same lightweight architecture to be adapted to different actuator-relevant abnormal conditions.

3 Results and Discussion

3.1 Model Performance

The first evaluation focuses on the capability of the proposed Hybrid SVM-LSTM workload. This evaluation is necessary because the edge controller is meaningful only if the workload placed on it can produce reliable actuator-relevant abnormal-condition detection. Table 3 summarizes the performance of the model configurations that were selected to be displayed in the paper, based on the experimental spreadsheet. The table includes early high-recall configurations, constrained paper-ready configurations, LSTM-only baselines, and the refined Hybrid SVM-LSTM proxy models.

Table 3. Performance comparison of model configurations

Model	Accuracy	Precision	Recall	P95 Inference (ms)	F1-score
hybrid_w8_h64_d0.2	0.055	0.032	0.985	0.919	0.062
hybrid_w12_h64_d0.2	0.062	0.032	0.978	0.927	0.062
hybrid_w12_h64_d0.2	0.912	0.132	0.311	0.915	0.185
lstm_only_w12_h64	0.921	0.140	0.283	0.914	0.188
hybrid_w8_h48_d0.1	0.924	0.152	0.297	0.901	0.202
lstm_only_w8_h64_d0.15	0.929	0.143	0.239	0.920	0.179
hybrid_svm_water_related_proxy	0.979	0.763	0.690	1.142	0.725
hybrid_svm_NPK_gap_proxy	0.983	0.886	0.549	1.448	0.678

The initial Hybrid SVM-LSTM and LSTM-only configurations were not sufficient for reliable abnormal-condition detection under class imbalance. Although several configurations achieved high accuracy or high recall, their low precision and F1-scores indicate weak anomaly-class discrimination. This means that a model can appear useful from one metric while still being unsuitable for actuator-oriented decision support. For example, the high-recall configurations detected most positive samples, but their extremely low precision would create too many false abnormal predictions.

After proxy-label refinement and SVM-based discriminative scoring, the Hybrid SVM-LSTM model showed a substantial improvement. The water-related proxy achieved the highest F1-score of 0.725, with 0.979 accuracy, 0.763 precision, 0.690 recall, and a P95 inference time of 1.142 ms. This result indicates the best balance between detecting abnormal water-related conditions and avoiding excessive false alarms. The NPK-gap proxy achieved the highest precision of 0.886 and the highest accuracy of 0.983, with 0.549 recall, 0.678 F1-score, and a P95 inference time of 1.448 ms. This configuration is more conservative and is suitable when false abnormal predictions should be minimized.

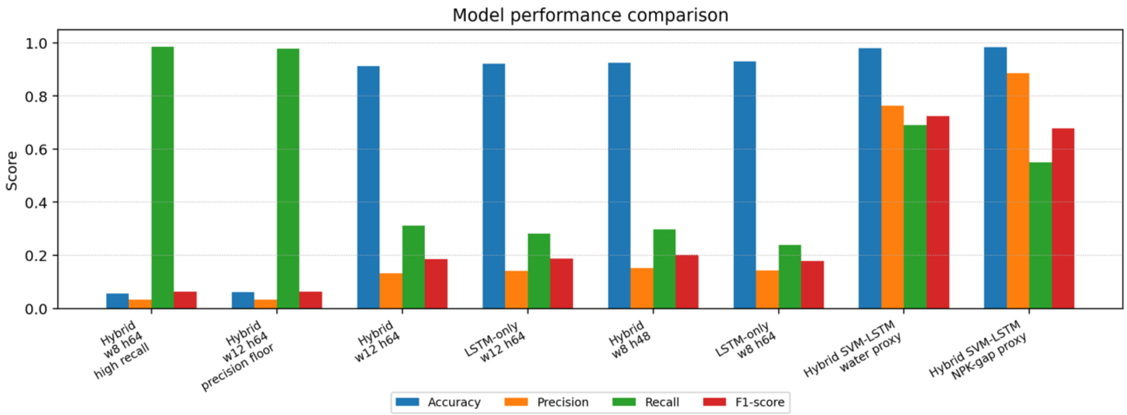

Figure 2. Performance metric comparison of paper-selected models.

Figure 2 visualizes the performance difference among the selected configurations. The refined Hybrid SVM-LSTM proxy models clearly outperform the earlier candidate models in precision and F1-score. This improvement is important because actuator-oriented abnormal-condition detection cannot rely only on accuracy. A greenhouse model must also avoid excessive false alarms while still detecting relevant abnormal states. The figure also shows that LSTM-only baselines remain weaker than the refined Hybrid SVM-LSTM proxy models, which supports the usefulness of adding the SVM discriminative score to the temporal model.

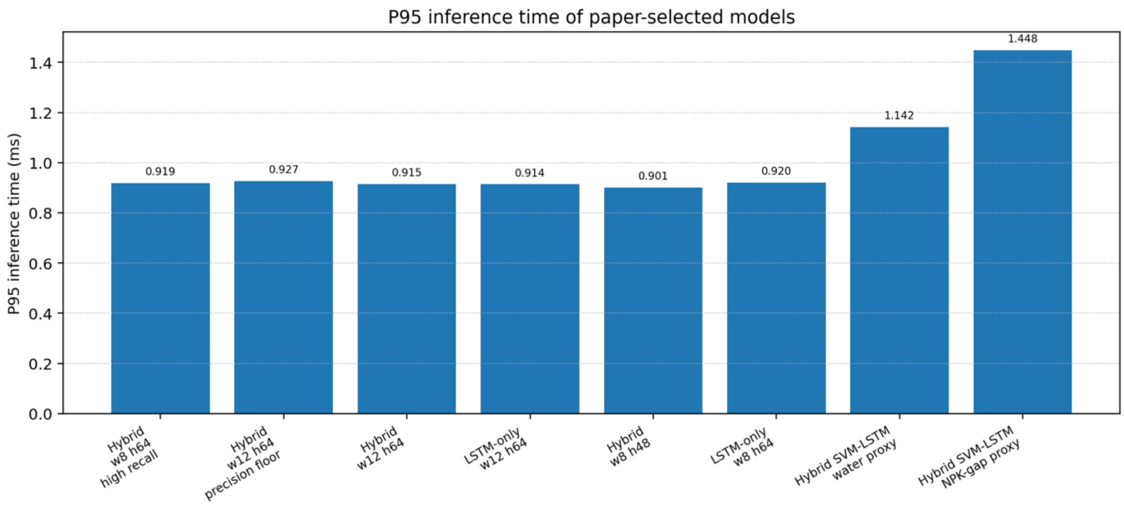

Figure 3. P95 inference time comparison of paper-selected models

Figure 3 shows that all selected configurations have P95 inference time in the millisecond range. The earlier candidate models are slightly below 1 ms, while the refined water-related and NPK-gap proxy models reach 1.142 ms and 1.448 ms, respectively. For context, the greenhouse autoencoder in [31] reported a detection time of 0.0585 s, whereas the edge-based spatio-temporal correlation approach in [32] reported 0.47 s for its big-data experiment. The proposed model's measured latency is numerically lower, but hardware, batch size, software stack, and timing boundaries differ across studies; therefore, these figures do not establish universal speed superiority. Within the controlled experiment, however, a P95 below 1.5 ms supports the model's lightweight real-time objective.

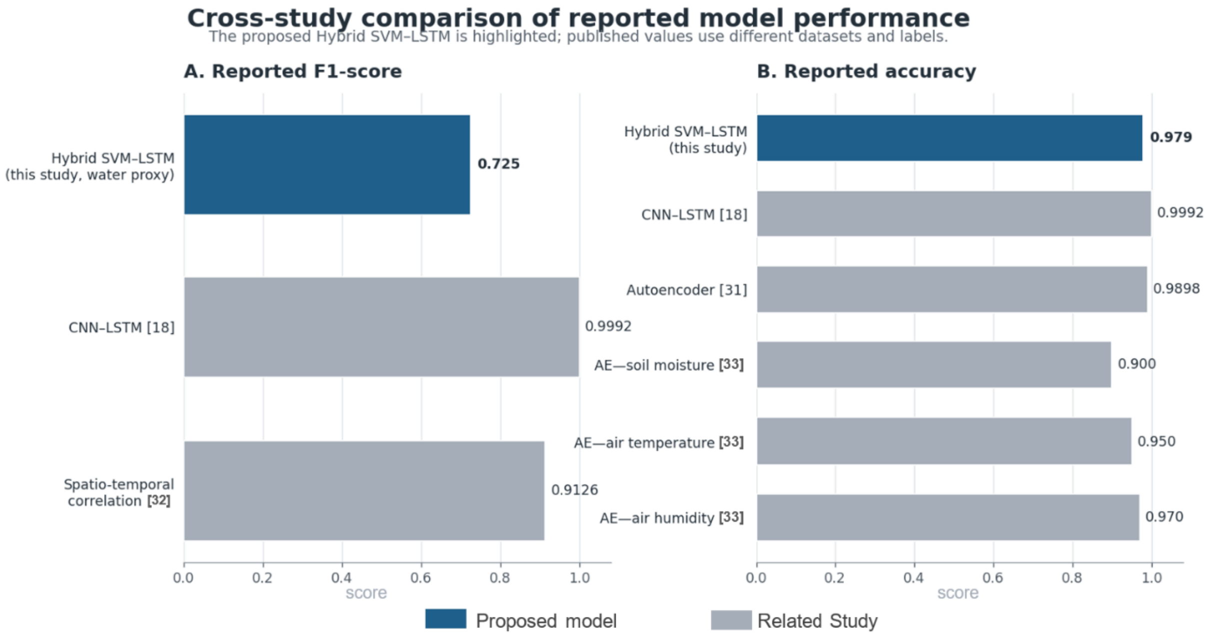

Figure 4. Final Hybrid SVM-LSTM proxy comparison

Figure 4 compares the proposed Hybrid SVM-LSTM with alternative architectures reported in related research. In the F1-score panel, the proposed water-related proxy achieves 0.725, compared with 0.9992 for CNN-LSTM using direct fan-actuator labels [18] and 0.9126 for the spatio-temporal correlation approach [32]. In the accuracy panel, the proposed model reaches 0.979, compared with 0.9992 for CNN-LSTM [18], 0.9898 for a greenhouse autoencoder [31], and 0.900–0.970 for smart-irrigation autoencoders across soil moisture, air temperature, and air humidity [33]. Because [31] and [32] do not report the same complete precision-recall metrics, their F1-scores are not estimated and their accuracy results are shown separately. The comparison therefore does not establish a direct model ranking: the studies use different datasets, label sources, anomaly definitions, and test environments. In particular, CNN-LSTM uses verified actuator states, whereas the proposed model learns actuator-relevant proxy conditions. Although its F1-score is lower, the proposed Hybrid SVM-LSTM retains millisecond-level P95 inference and remains applicable when direct actuator labels are unavailable.

3.2 *Discussion on Edge AI Model Feasibility*

The results show that the proposed Hybrid SVM-LSTM model is feasible as a lightweight Edge AI workload for real-time smart greenhouse anomaly detection. The strongest evidence is the combination of high accuracy, improved anomaly-class precision, balanced F1-score, and millisecond-level P95 inference time measured in the software evaluation environment. These results align with earlier greenhouse prediction studies, which similarly found that abnormal-condition detection requires balancing sensitivity to rare abnormal events against the risk of false alarms [11], [18]. The model does not only classify the majority normal class, but also improves abnormal-condition discrimination after proxy-label refinement. This is important because a real-time greenhouse anomaly detection model should identify abnormal sensor patterns quickly while avoiding excessive false alarms that may lead to unnecessary operational responses.

The proposed controller also illustrates why feature fusion is important for real-time anomaly detection. A model that only uses current sensor values may miss the speed and direction of environmental change. By including short-term changes, sampling gaps, and periodic patterns, the controller can distinguish between stable and transitional greenhouse states. The SVM score further compresses discriminative information before it is processed by the LSTM sequence model. This hybrid structure gives the model a practical balance between compactness, temporal awareness, and real-time detection capability.

The comparison between the water-related and NPK-gap proxies also shows that real-time smart greenhouse anomaly detection should be designed according to the target abnormal condition. If the objective is water-related anomaly detection, a balanced F1-oriented model is more useful because both missed abnormal states and false alarms matter. If the objective is broader nutrient-gap or multi-indicator abnormality, a precision-oriented model may be preferable because unnecessary abnormal predictions can create inefficient operational responses. Therefore, the proposed model should not be interpreted as a single fixed classifier, but as a lightweight Edge AI framework whose threshold and proxy target can be adjusted to the detection requirement.

Nevertheless, the feasibility evaluation in this study is limited to the model level. All experiments were conducted in a software-based evaluation environment, so the reported inference times reflect algorithmic latency rather than measurements on an actual embedded edge device. Hardware implementation aspects, such as deployment on microcontroller or single-board-computer platforms, memory footprint, CPU utilization, energy consumption, and behavior under real greenhouse network conditions, are beyond the scope of this study. The millisecond-level P95 inference time should therefore be interpreted as an indicator of computational compactness that supports edge deployability, not as a guaranteed on-device performance figure.

4 Conclusion

This paper presented a Lightweight Hybrid SVM-LSTM Edge AI model for real-time smart greenhouse anomaly detection. The proposed model integrates packet validation, feature fusion, SVM discriminative scoring, and LSTM temporal learning into a compact AI pipeline that can be embedded in an edge-capable greenhouse controller. The model uses current sensor values, temporal-change features, sampling-gap information, and daily periodic patterns to classify greenhouse states as normal or abnormal. Experimental results show that the water-related proxy achieved 0.979 accuracy, 0.763 precision, 0.690 recall, and a 0.725 F1-score with 1.142 ms P95 inference time. The NPK-gap proxy achieved 0.983 accuracy, 0.886 precision, 0.549 recall, and a 0.678 F1-score with 1.448 ms P95 inference time. These findings indicate that the proposed Hybrid SVM-LSTM model is accurate, compact, and suitable for real-time anomaly detection in smart greenhouse environments.

In practical terms, the proposed framework allows greenhouse operators to detect abnormal water-related and nutrient-related conditions locally at the edge, without depending on continuous cloud connectivity. Early and low-latency detection of such conditions can support more timely actuator responses, reduce unnecessary irrigation and nutrient delivery caused by false alarms, and help maintain stable growing conditions even when network access is limited. Because the threshold and proxy target can be adjusted, the framework can also be adapted to different crops, greenhouse layouts, and management priorities without redesigning the entire pipeline. Future research should validate the proxy labels using real actuator data, deploy the model on physical edge hardware, and compare the proposed hybrid approach with other lightweight machine learning models under continuous greenhouse operation.

References

[1] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, "A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming," 2019, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2019.2949703.

[2] C. Bersani, C. Ruggiero, R. Sacile, A. Soussi, and E. Zero, "Internet of Things Approaches for Monitoring and Control of Smart Greenhouses in Industry 4.0," *Energies (Basel)*., vol. 15, no. 10, p. 3834, May 2022, doi: 10.3390/en15103834.

[3] C. Maraveas, D. Piromalis, K. G. Arvanitis, T. Bartzanas, and D. Loukatos, "Applications of IoT for optimized greenhouse environment and resources management," *Comput. Electron. Agric.*, vol. 198, p. 106993, Jul. 2022, doi: 10.1016/j.compag.2022.106993.

[4] M. A. Zamora-Izquierdo, J. Santa, J. A. Martínez, V. Martínez, and A. F. Skarmeta, "Smart farming IoT platform based on edge and cloud computing," *Biosyst. Eng.*, vol. 177, pp. 4–17, Jan. 2019, doi: 10.1016/j.biosystemseng.2018.10.014.

[5] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016, doi: 10.1109/JIOT.2016.2579198.

[6] Sapna, H. Gauttam, V. Chauhan, K. K. Pattanaik, A. Trivedi, and H. Ghosh, "A comprehensive review of Edge Computing empowered smart agriculture: Trends, Opportunities and Future directions," *Comput. Electron. Agric.*, vol. 241, p. 111252, Feb. 2026, doi: 10.1016/j.compag.2025.111252.

[7] P. Yu, F. Teng, W. Zhu, C. Shen, Z. Chen, and J. Song, "Cloud–edge–device collaborative computing in smart agriculture: architectures, applications, and future perspectives," *Front. Plant Sci.*, vol. 16, Oct. 2025, doi: 10.3389/fpls.2025.1668545.

[8] S. Iftikhar *et al.*, "AI-based fog and edge computing: A systematic review, taxonomy and future directions," *Internet of Things*, vol. 21, p. 100674, Apr. 2023, doi: 10.1016/j.iot.2022.100674.

[9] M. El Jarroudi *et al.*, "Leveraging edge artificial intelligence for sustainable agriculture," *Nat. Sustain.*, vol. 7, no. 7, pp. 846–854, Jun. 2024, doi: 10.1038/s41893-024-01352-4.

[10] M. N. Akhtar, A. J. Shaikh, A. Khan, H. Awais, E. A. Bakar, and A. R. Othman, "Smart Sensing with Edge Computing in Precision Agriculture for Soil Assessment and Heavy Metal Monitoring: A Review," *Agriculture*, vol. 11, no. 6, p. 475, May 2021, doi: 10.3390/agriculture11060475.

[11] J. Morales-García, D. Padilla-Quimbiulco, M. Cantabella, B. Ayuso, A. Muñoz, and J. M. Cecilia, "GreenhouseGuard: Enabling real-time warning prediction for smart greenhouse management," *J. Ambient Intell. Smart Environ.*, vol. 16, no. 3, pp. 389–405, Sep. 2024, doi: 10.3233/AIS-230359.

[12] B. Ravelo, M. Guerin, L. Rajaoarisoa, and W. Rahajandraibe, "Low-Pass NGD Digital Circuit Application for Real-Time Greenhouse Temperature Prediction," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 9, pp. 3709–3713, Sep. 2023, doi: 10.1109/TCSII.2023.3276389.

[13] L. Zeping, N. Abdullah, and M. K. Ishak, "Smart greenhouse climate control with real-time fault detection and energy-aware automation," *Smart Agricultural Technology*, vol. 13, p. 101707, Mar. 2026, doi: 10.1016/j.atech.2025.101707.

[14] B. Toskov and A. Toskova, "AgroNova: An Autonomous IoT Platform for Greenhouse Climate Control," *Sensors*, vol. 26, no. 6, p. 1861, Mar. 2026, doi: 10.3390/s26061861.

[15] M. Platero-Horcajadas, S. Pardo-Pina, J.-M. Cámara-Zapata, J.-A. Brenes-Carranza, and F.-J. Ferrández-Pastor, "Enhancing Greenhouse Efficiency: Integrating IoT and Reinforcement Learning for Optimized Climate Control," *Sensors*, vol. 24, no. 24, p. 8109, Dec. 2024, doi: 10.3390/s24248109.

[16] U. Ristian, I. Ruslianto, H. Hasfani, and K. Sari, "Perancangan Arsitektur Node Nirkabel dalam Efisiensi Bandwidth Smart Greenhouse Berbasis Protokol MQTT," *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 9, no. 2, p. 218, Aug. 2023, doi: 10.26418/jp.v9i2.63885.

[17] A. Zaguia, "Smart greenhouse management system with cloud-based platform and IoT sensors," *Spatial Information Research*, vol. 31, no. 5, pp. 559–571, Oct. 2023, doi: 10.1007/s41324-023-00523-3.

[18] G. Airlangga, J. Bata, O. I. Adi Nugroho, and B. H. P. Lim, "Hybrid CNN-LSTM Model with Custom Activation and Loss Functions for Predicting Fan Actuator States in Smart Greenhouses," *AgriEngineering*, vol. 7, no. 4, p. 118, Apr. 2025, doi: 10.3390/agriengineering7040118.

[19] M. U. Tariq, S. M. Saqib, T. Mazhar, M. A. Khan, T. Shahzad, and H. Hamam, "Edge-enabled smart agriculture framework: Integrating IoT, lightweight deep learning, and agentic AI for context-aware farming," *Results in Engineering*, vol. 28, p. 107342, Dec. 2025, doi: 10.1016/j.rineng.2025.107342.

[20] S. Heydari and Q. H. Mahmoud, "Tiny Machine Learning and On-Device Inference: A Survey of Applications, Challenges, and Future Directions," *Sensors*, vol. 25, no. 10, p. 3191, May 2025, doi: 10.3390/s25103191.

[21] M. J. C. S. Reis, "Lightweight Signal Processing and Edge AI for Real-Time Anomaly Detection in IoT Sensor Networks," *Sensors*, vol. 25, no. 21, p. 6629, Oct. 2025, doi: 10.3390/s25216629.

[22] "Advanced IoT Agriculture 2024." Accessed: Jun. 14, 2026. [Online]. Available: https://www.kaggle.com/datasets/wisam1985/advanced-iot-agriculture-2024/data

[23] X.-B. Jin *et al.*, "Deep-Learning Temporal Predictor via Bidirectional Self-Attentive Encoder–Decoder Framework for IOT-Based Environmental Sensing in Intelligent Greenhouse," *Agriculture*, vol. 11, no. 8, p. 802, Aug. 2021, doi: 10.3390/agriculture11080802.

[24] Z. Guo and L. Feng, “Multi-step prediction of greenhouse temperature and humidity based on temporal position attention LSTM,” *Stochastic Environmental Research and Risk Assessment*, vol. 38, no. 12, pp. 4907–4934, Dec. 2024, doi: 10.1007/s00477-024-02840-x.

[25] W. Cheng, T. Ma, X. Wang, and G. Wang, “Anomaly Detection for Internet of Things Time Series Data Using Generative Adversarial Networks With Attention Mechanism in Smart Agriculture,” *Front. Plant Sci.*, vol. 13, Jun. 2022, doi: 10.3389/fpls.2022.890563.

[26] Y.-C. Tung, N. W. Syahputri, and I. G. N. A. S. Diputra, “Greenhouse Environment Sentinel with Hybrid LSTM-SVM for Proactive Climate Management,” *AgriEngineering*, vol. 7, no. 4, p. 96, Apr. 2025, doi: 10.3390/agriengineering7040096.

[27] T. Tanyıldızı Ağır, “Daily Global Solar Radiation Prediction with Hybrid LSTM-SVM:The Case of Nusaybin,” *Arab. J. Sci. Eng.*, Mar. 2025, doi: 10.1007/s13369-025-10322-7.

[28] D. Fährmann, N. Damer, F. Kirchbuchner, and A. Kuijper, “Lightweight Long Short-Term Memory Variational Auto-Encoder for Multivariate Time Series Anomaly Detection in Industrial Control Systems,” *Sensors*, vol. 22, no. 8, p. 2886, Apr. 2022, doi: 10.3390/s22082886.

[29] A. de la Cruz Huayanay, J. L. Bazán, and C. M. Russo, “Performance of evaluation metrics for classification in imbalanced data,” *Comput. Stat.*, Aug. 2024, doi: 10.1007/s00180-024-01539-5.

[30] J.-G. Gaudreault and P. Branco, “Empirical analysis of performance assessment for imbalanced classification,” *Mach. Learn.*, vol. 113, no. 8, pp. 5533–5575, Aug. 2024, doi: 10.1007/s10994-023-06497-5.

[31] M. Adkisson, J. C. Kimmell, M. Gupta, and M. Abdelsalam, “Autoencoder-based Anomaly Detection in Smart Farming Ecosystem,” in *2021 IEEE International Conference on Big Data (Big Data)*, IEEE, Dec. 2021, pp. 3390–3399. doi: 10.1109/BigData52589.2021.9671613.

[32] R. Zhang, L. Zhou, A. Mei, and Y. He, “Sensor Monitoring Techniques in Edge Computing Using Spatio-Temporal Correlation Anomaly Detection Algorithms,” *IEEE Access*, vol. 12, pp. 116516–116529, 2024, doi: 10.1109/ACCESS.2024.3444046.

[33] R. Benameur, A. Dahane, B. Kechar, and A. E. H. Benyamina, “An Innovative Smart and Sustainable Low-Cost Irrigation System for Anomaly Detection Using Deep Learning,” *Sensors*, vol. 24, no. 4, p. 1162, Feb. 2024, doi: 10.3390/s24041162.