Computer Modeling in Engineering & Sciences

Tech Science Press

https://doi.org/10.32604/cmes.2026.081254

ARTICLE

FICNet: A Deep Learning Framework for Intrusion Detection in Agricultural Internet of Things

Md. Fahmid-Ul-Alam Juboraj[1], Fahmid Al Farid[2,3], Mahe Zabin[4], Jia Uddin[5], Muhammad Iqbal Hossain[1,*] and Sarina Mansor[2,*]

[1]Department of Computer Science and Engineering, BRAC University, Dhaka, Bangladesh
[2]Centre for Image and Vision Computing (CIVC), Centre of Excellence for Artificial Intelligence, Faculty of Artificial Intelligence and Engineering (FAIE), Multimedia University, Cyberjaya, Selangor, Malaysia
[3]Berlin School of Business & Innovation (BSBI), Berlin, Germany
[4]Human and Digital Interface Department, JW Kim College of Future Studies, Woosong University, Daejeon, Republic of Korea
[5]AI and Big Data Department, Woosong University, Daejeon, Republic of Korea

*Corresponding Authors: Muhammad Iqbal Hossain. Email: iqbal.hossain@bracu.ac.bd; Sarina Mansor. Email: sarina.mansor@mmu.edu.my

Received: 26 February 2026; Accepted: 12 May 2026; Published: 27 July 2026

ABSTRACT: The integration of Internet of Things (IoT) technologies in agriculture enables precision farming but introduces significant cybersecurity vulnerabilities. This paper presents FICNet (Feature Integrated Convolutional Network), a lightweight deep learning architecture for intrusion detection in agricultural IoT environments. Evaluated on the Farm-Flow AG-IoT security dataset, FICNet achieves 100% binary classification accuracy and 81.25% multiclass accuracy (macro F1: 80.43%, precision: 91.26%, ROC-AUC: 96.78%) across 8 traffic categories. A multi-dimensional component analysis confirms the contribution of each architectural component: multi-scale convolutions provide 5.3% noise robustness advantage, squeeze-and-excitation attention controls per-class detection trade-offs, and the full architecture achieves 8% data efficiency advantage over traditional baselines. Interpretability analysis via Integrated Gradients identifies header size, byte counts, and directional ratios as primary discriminative features. Comparative evaluation against 15 baselines including Transformer and GNN architectures validates FICNet's effectiveness with only 147,092 parameters (1.81 MB), demonstrating suitability for resource-constrained agricultural IoT deployments.

KEYWORDS: Agricultural IoT; intrusion detection; deep learning; attention mechanisms; FICNet; cybersecurity; smart farming

1 Introduction

Cyberbiosecurity (CBS) has emerged as a critical consideration for all stakeholders in the agricultural ecosystem. Cyberattacks on agricultural systems have both direct and indirect impacts across multiple domains. Manipulated sensor data and automated systems can cause crop failure, with covert attacks significantly delaying detection [1]. Smart Farming (SF) and Precision Agriculture (PA) use IoT to optimize inputs and improve efficiency. However, they face critical security challenges due to heterogeneous devices, resource constraints, and real-time requirements. Cyberattacks can compromise system integrity and data, directly affecting livestock health, feed management, and overall farm productivity. Ensuring secure operation is therefore essential for maintaining data integrity and operational reliability in SF and PA environments [2]. Attacks on autonomous equipment disrupt infrastructure and logistics, reducing operational efficiency.

Lastly, Agricultural IoT systems face critical data integrity risks due to resource constraints and direct impacts on crop yields. These incidents underscore agriculture's systemic interconnectedness and frame cybersecurity as essential to food security [1]. Intrusion Detection Systems (IDS) play a pivotal role in smart farming by identifying and analyzing security incidents, providing agricultural enterprises with quantitative insights into attack patterns and threat landscapes [3]. Smart farming increasingly relies on IoT technologies to monitor crop health and optimise resource use; however, the large number of resource-constrained devices makes these systems vulnerable to cyberattacks, making effective intrusion detection systems (IDS) critical for securing IoT-based agricultural networks. Recent approaches use deep learning architectures, to identify threats with high accuracy, highlighting the importance of advanced IDS in modern agriculture [4]. ML and AI have emerged as powerful tools for analyzing large-scale agricultural datasets, detecting anomalies indicative of malicious activity, and identifying complex threats within the AG-IoT landscape [3,5]. These algorithms can learn from historical data to detect patterns and anomalies, enabling proactive threat detection and automated response capabilities. However, the efficacy of machine learning-based IDS is fundamentally constrained by the availability of representative and comprehensive training data, which remains a critical limitation in real-world AG-IoT deployments [6]. Despite recent advances, significant research gaps persist: the foundational work by Ferreira et al. on the Farm-Flow dataset [7], while pioneering in providing a benchmark for AG-IoT intrusion detection, relied primarily on traditional machine learning models, leaving the potential of lightweight deep learning architectures unexplored for this domain.

In this paper, we make the following contributions:

(1) We propose FICNet, a lightweight deep learning architecture optimized for intrusion detection in agricultural IoT networks, combining convolutional layers with fully connected networks for effective feature extraction and classification.
(2) We conduct comprehensive evaluation on the Farm-Flow AG-IoT security dataset, demonstrating exceptional performance in both binary and multiclass classification scenarios.
(3) We present a multi-dimensional ablation study demonstrating that each architectural component serves a distinct role: multi-scale convolutions provide noise robustness, squeeze-and-excitation attention controls per-class detection behavior, and residual connections enable stable gradient flow, collectively achieving 8% data efficiency advantage over a plain CNN baseline.
(4) We provide interpretability analysis using Integrated Gradients, identifying the most influential network flow features for each attack category, and conduct root-cause analysis of challenging class confusions through feature-space similarity analysis.
(5) We perform extensive robustness analysis including adversarial attack resilience (FGSM, PGD, Carlini-Wagner), noise tolerance (Gaussian, uniform, salt-and-pepper).
(6) We provide detailed comparative analysis against baseline traditional models.
(7) We analyze computational complexity, inference latency, and scalability characteristics to validate FICNet's suitability for real-world deployment.

This paper is organized as follows. Literature reviews are presented in Section 2, then the methodology is described in Section 3, which includes data pre-processing Section 3.1 and the discussion on the proposed architecture is in Section 3.2. The metrics used for the evaluation of the model were discussed in Section 3.3 along with the strategies used for the robustness and efficiency. The detailed results for the experimentation with the model and training configuration were shown in Section 4 which includes both binary and multi-classification results. Also the comparative analysis were shown in Section 4.6 with the existing and baseline models. Lastly, the detailed discussion in Section 5 and we end with the Section 6.

2 Related Works

This section reviews existing IDS approaches relevant to agricultural IoT, organised by architectural paradigm. Numerous research efforts have focused on developing advanced intrusion detection mechanisms to mitigate cybersecurity risks in IoT-based smart agriculture systems. For instance, Pasca et al. [8] revealed counterintuitive fault–attack interactions in a vulnerable-by-design IoT sensor framework: spike faults increased BOLA attack detectability by 95.9%, whereas dropout faults reduced command injection detection by 18%. Their LSTM-based evaluation achieved moderate recall (0.5473) and high precision (0.8285), indicating that integrated sensor health and security monitoring is essential for preserving data integrity in smart agriculture. Similarly, the CBCTL-IDS method integrates transfer learning with pre-trained CNNs and the Black Kite Algorithm for adaptive hyperparameter optimisation, achieving >99% accuracy on the ToN-IoT, Edge-IIoTset, and WSN-DS benchmarks. However, its increased model complexity and computational demands underscore the need for lightweight architectures suited to resource-constrained IoT environments [9]. Kethineni and Pradeepini [10] proposed a three-tier smart agriculture IDS that integrates IoT sensors, fog computing, and cloud storage, employing a CNN-BiGRU model with attention optimised via the Wild Horse Optimization algorithm. Their system achieves accuracy exceeding 99% on the ToN-IoT and APA-DDoS datasets with superior detection and precision metrics compared to existing methods.

Table 1 summarises representative IDS approaches across AG-IoT and related network security benchmarks, highlighting each method's dataset, architecture, and key limitations.

Table 1: Comparison of the related works.

Ref.	Dataset	Approach	Model(s)	Contribution	Remarks/Limitations
[7]	Farm-Flow (AG-IoT)	Network flow-based IDS	DT, LR, NB, RF, DNN	First realistic AG-IoT IDS benchmark using flow features from a real smart farm testbed, achieving 92.67% binary and 76.37% multiclass accuracy.	Limited dataset size; multiclass performance gap is large; restricted to shallow classifiers.
[11]	IoT smart irrigation (custom)	Hybrid deep IDS	SpinalFGNet + Fuzzy Logic + GoogLeNet	Novel spinal network connections fused with fuzzy reasoning and GoogLeNet for smart irrigation IDS; improved classification robustness.	The GoogLeNet backbone is computationally intensive for low-power nodes; no AG-specific attack traffic; and no multiclass evaluation.
[12]	Custom AG-IoT testbed	Requirements analysis	N/A (framework proposal)	First systematic analysis of operational IDS challenges in agriculture: heterogeneous devices, energy limits, and operator-guided response.	Does not present a fully trained/evaluated system; limited to requirements and recommendations.

(Continued)

Table 1 (continued)

Ref.	Dataset	Approach	Model(s)	Contribution	Remarks/Limitations
[13]	CSE-CIC-IDS2018, AG-IoT	Federated + DRL	Deep Reinforcement Learning + FL	Combines dynamic DRL policy adaptation with FL privacy guarantees; addresses offline learning limitation of static IDS.	DRL convergence is computationally expensive; no evaluation on flow-based AG-IoT datasets; model poisoning risks unaddressed.
[14]	IoT network datasets	Graph AI + Federated Learning	GNN + Pearson feature selection + FL	Adaptive session-based graph models IoT topology; 97%+ accuracy, <2% FPR across 25–200 nodes; privacy-preserving federation.	Communication overhead grows with federation scale; evaluated on generic IoT—not AG-specific traffic.
[15]	TON-IoT	Hybrid metaheuristic + DL	LSTM + MAFA (Memetic Adaptive Firefly)	Memetic firefly algorithm auto-optimises LSTM hyperparameters; 99.99% accuracy on TON-IoT.	Binary classification only; continuous retraining required for evolving attacks; scalability to AG edge devices not shown.
[16]	CIC-IDS2018, ToN-IoT, Edge-IIoTset	Edge-deployed deep IDS	BiGRU-LSTM + TBPTT	99.82% accuracy; designed explicitly for extreme agricultural environmental conditions; truncated BPTT for long IoT sequences.	High computational demand; retraining required per network topology; not evaluated on Farm-Flow or AG-specific flow data.
[17]	NSL-KDD	Deep learning IDS	LSTM, CNN	Deep models exceed traditional ML; F1 = 0.89, AUC = 0.94; demonstrates benefit of learned temporal and spatial features for IDS.	NSL-KDD is a legacy benchmark; binary classification only; does not reflect contemporary IoT or AG-IoT traffic distributions.

(Continued)

Table 1 (continued)

Ref.	Dataset	Approach	Model(s)	Contribution	Remarks/Limitations
[18]	CIC-IDS2017, CSE-CIC-IDS2018, LUFlow	Comparative ML study	DT, RF, SVM, NB, ANN, DNN	Systematic comparison across three datasets; ~99% accuracy; Naive Bayes yields lowest FPR—practically important for alert fatigue.	All datasets are generic; no IoT or AG-IoT traffic; no deep attention or residual architectures assessed.

3 Methodology

The proposed methodology consists of five primary stages: data collection, data preprocessing, FICNet architecture design, and training with evaluation as shown in Fig. 1. Each stage is systematically designed to ensure reproducibility, robustness, and optimal performance for intrusion detection in agricultural IoT environments.

Figure 1: Workflow diagram.

3.1 Data Preprocessing

The data preprocessing stage transforms raw PCAP files into structured feature representations suitable for machine learning analysis. The process begins with PCAP splitting using editcap with a 5-s interval parameter, segmenting continuous traffic into discrete time windows that correspond to network flow boundaries. This temporal segmentation ensures that each sample represents a coherent communication session rather than arbitrary packet sequences. The data processing pipeline, as mentioned in the Farm-Flow paper [7] from the raw PCAP to the tabular format is shown in the Fig. 2.

Two parallel processing pipelines extract complementary features from the segmented PCAP files. The Zeek analysis pipeline processes the PCAP files to generate conn.log files containing Layer 4 connection-level statistics. Zeek extracts fundamental network flow characteristics including connection duration, protocol information, packet counts, byte counts, and connection states. Simultaneously, the Flowmeter pipeline generates flowmeter.log files with detailed flow statistics including bidirectional packet rates, header sizes, payload characteristics, and temporal flow patterns.

The merge and label stage performs a UID-based join operation to combine features from both Zeek and Flowmeter outputs, creating a unified feature vector for each network flow. IP-based labeling is applied using predefined mappings that associate source/destination IP addresses with attack categories or normal traffic based on the controlled testbed environment. This labeling strategy ensures accurate ground truth annotations for supervised learning.

The resulting Farm-Flow dataset contains 30 features spanning multiple categories: flow-level statistics (missed_bytes, orig_pkts, orig_ip_bytes, resp_pkts, resp_ip_bytes), temporal characteristics (fwd_pkts_per_sec, bwd_pkts_per_sec, flow_pkts_per_sec), directional metrics (down_up_ratio), header information (fwd_header_size_tot, fwd_header_size_min, fwd_header_size_max, bwd_header_size_tot, bwd_header_size_min, bwd_header_size_max), and payload statistics like (fwd_pkts_payload.tot,

fwd_pkts_payload. avg, bwd_pkts_payload.tot, bwd_pkts_payload.avg). The dataset encompasses 8 attack classes representing different threat categories plus normal traffic, with approximately 1.3 million instances providing substantial data for model training and evaluation.

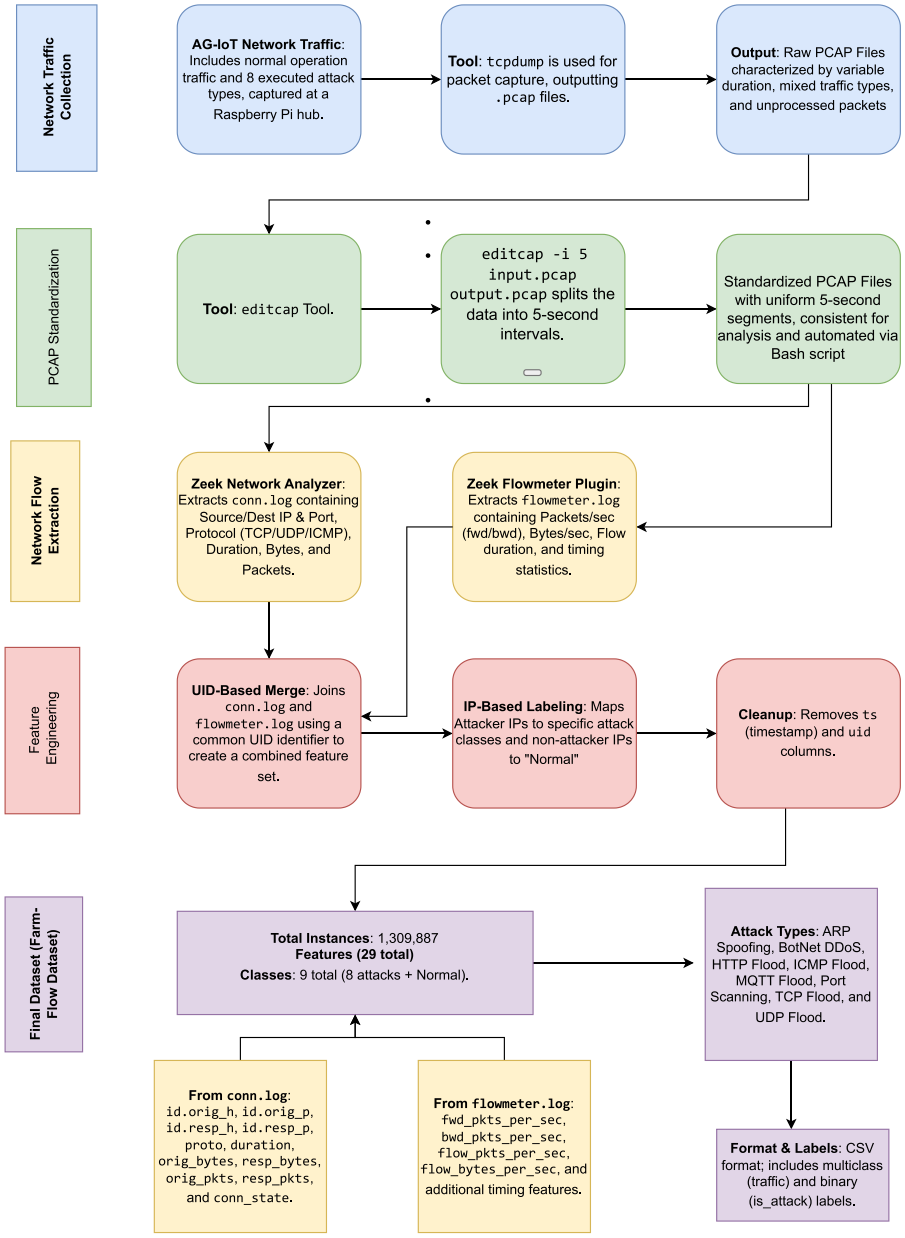

Figure 2: Detailed data preprocessing.

The final pre-processing stage prepares the Farm-Flow dataset for neural network training via stratified splitting, normalization, encoding, and reshaping. An 85/15 stratified train–test split is applied with random_state=42 to ensure reproducibility. Stratification preserves class distribution in both sets, which is essential given the inherent class imbalance in network intrusion detection datasets, where attack traffic typically represents a minority of total flows.

Standardization is applied using Standard Scaler from scikit-learn, transforming all 30 features to have zero mean ($\mu = 0$) and unit variance ($\sigma = 1$). This z-score normalization is essential for deep learning convergence as it prevents features with larger scales from dominating the gradient updates and ensures that the optimization landscape is well-conditioned. The scaler parameters (mean and standard deviation) are computed exclusively on the training set and applied to both training and test sets to prevent data leakage. To rigorously verify the absence of data leakage, we confirm: (i) zero row-level overlap between the training and test partitions for both binary and multiclass tasks; (ii) the StandardScaler parameters (mean and standard deviation) are computed exclusively on the training set and applied without re-fitting to the test set; and (iii) Kolmogorov-Smirnov tests on all 29 features confirm statistically distinct distributions between partitions (27/29 features with $p < 0.05$; mean KS statistic: 0.040), providing strong evidence of proper data separation. Label encoding converts categorical attack labels into numerical format suitable for neural network outputs. For binary classification, labels are encoded as 0 (normal traffic) and 1 (attack traffic). Following Ferreira et al. [7], the BotNet DDoS class (64 instances) was excluded from both partitions due to insufficient sample size for reliable oversampling, resulting in 8 classes (7 attack types + Normal).

The final preprocessing step reshapes the feature vectors to the format $(n, \text{features}, 1)$, where n is the number of samples and features equals 29. This reshaping operation prepares the data for Conv1D (1-dimensional convolutional) layers, treating each feature as a sequence element analogous to time-series or sequence modeling tasks. This representation enables the convolutional layers to learn local patterns and correlations among adjacent features in the network flow representation.

3.2 Model Architecture

FICNet (Feature Integrated Convolutional Network) is designed as a lightweight yet effective deep learning architecture specifically optimized for intrusion detection in resource-constrained agricultural IoT environments. The architecture begins with a multi-scale convolutional input layer. This layer processes the 29-dimensional feature vector through three parallel Conv1D operations with kernel sizes $k = 1, 3, 5$. Each branch employs 32 filters to capture patterns at different scales: kernel size 1 learns point-wise feature transformations, kernel size 3 captures local dependencies among adjacent features, and kernel size 5 learns broader contextual patterns spanning multiple features.

The outputs from all three branches are concatenated along the channel dimension to create 96 feature maps, followed by batch normalization to stabilize training dynamics. A dropout layer with rate 0.2 is applied for regularization, followed by a channel reduction layer consisting of Conv1D with 96 filters and kernel size 1, along with batch normalization, which serves to consolidate the multi-scale features into a unified representation. The architecture then incorporates two residual blocks, each containing two Conv1D layers (96 filters, kernel size 3) with batch normalization after each convolution, implementing skip connections formulated as $y = F(x) + x$ where $F(x)$ represents the learned transformation, followed by ReLU activation applied after the addition operation. Residual Block 1 includes an additional dropout layer with rate 0.2 after the ReLU activation, while Residual Block 2 omits this dropout to create an asymmetric design that balances regularization with feature learning capacity.

Following the residual blocks, a Squeeze-and-Excitation (SE) mechanism with reduction ratio 8 implements channel-wise attention through global average pooling that reduces each channel to a single descriptor, followed by two fully connected layers (96 to 12 and then again to 96 channels) with ReLU and sigmoid activations respectively, producing attention weights in the range $[0, 1]$ that are multiplied element-wise with the feature maps to recalibrate channel importance dynamically. Global average pooling then performs spatial reduction by computing the mean of each channel across all positions, transforming the feature tensor into 96 aggregated features that provide translation invariance while significantly reducing the

number of parameters compared to fully connected layers operating on the entire feature map. The pooled features are processed through two dense layers: the first contains 128 neurons with ReLU activation, followed by batch normalization and dropout with rate 0.4, while the second contains 64 neurons with ReLU activation and dropout with rate 0.3, learning progressively abstract and task-specific representations through this dimensionality reduction pathway. L2 regularization with coefficient 1×10^{-4} is applied to all convolutional and dense layers throughout the network to prevent overfitting by penalizing large weight values.

The model is trained for 50 epochs with batch size 256 using the Adam optimizer with learning rate 0.0001, employing binary cross-entropy loss for binary classification and sparse categorical cross-entropy loss for multiclass classification, with early stopping monitoring validation loss to terminate training when no improvement is observed for a specified patience period. The complete architecture contains 146,637 parameters for binary classification and 147,092 parameters for multiclass classification, with a compact model size of 1.81 MB, making it highly suitable for deployment on resource-constrained edge devices and embedded systems common in agricultural IoT infrastructure while maintaining strong classification performance through its carefully designed multi-scale feature extraction, residual learning, channel attention, and progressive abstraction capabilities. The architecture diagram is shown in the Fig. 3.

Architectural Novelty: While multi-scale convolutions, residual learning, and squeeze-and-excitation attention are individually established techniques, FICNet's contribution lies in their purposeful integration for tabular network flow-based intrusion detection in AG-IoT environments. Unlike image-oriented architectures, FICNet treats each flow feature as a sequence element, enabling Conv1D kernels of sizes $k \in \{1, 3, 5\}$ to learn point-wise, local, and contextual feature interactions simultaneously. The multi-dimensional ablation study in Section 4.7 confirms that each component serves a distinct role: multi-scale convolutions provide 5.3% noise robustness advantage at $\sigma = 0.10$, SE attention fundamentally controls per-class detection behavior (Section 4.7), and the full architecture achieves 8% data efficiency advantage over a plain CNN baseline at 10% training data. These results demonstrate synergistic design rather than trivial assembly of existing blocks.

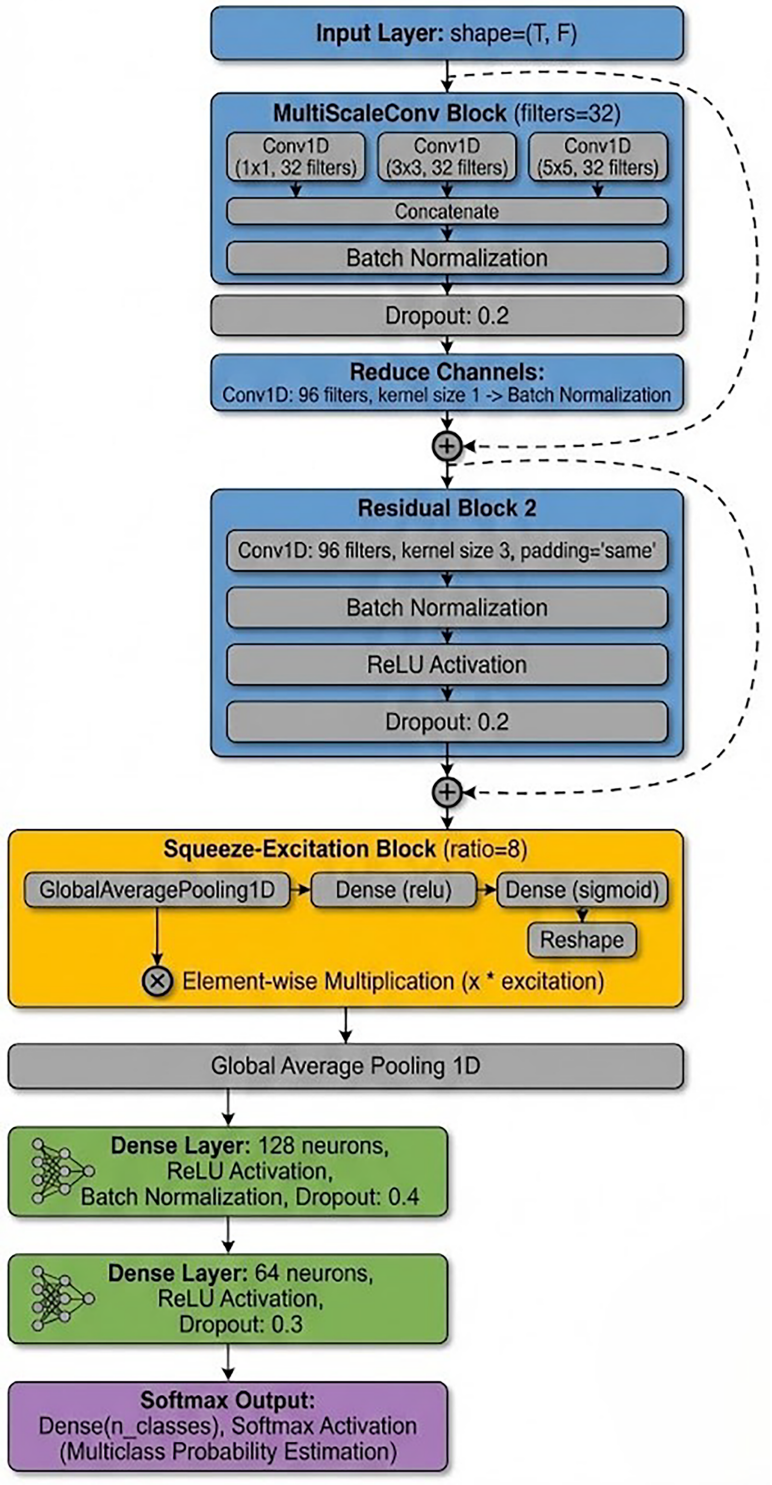

Figure 3: FICNet architecture.

3.3 Evaluation Metrics and Framework

For the evaluation of our model, we used the following metrics and frameworks for the noise and robustness testing.

3.3.1 Classification Metrics

We employ standard classification metrics to comprehensively evaluate FICNet's performance. Accuracy measures the proportion of correctly classified samples:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2}$$

Recall (sensitivity) measures the proportion of actual positives correctly identified:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3}$$

F1-Score provides the harmonic mean of precision and recall:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

Matthews Correlation Coefficient (MCC) offers a balanced measure even with class imbalance:

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{5}$$

Cohen's Kappa coefficient assesses agreement beyond chance:

$$\text{Cohen's Kappa} = \frac{p_o - p_e}{1 - p_e} \tag{6}$$

3.3.2 Robustness and Efficiency Evaluation

Performance is visualized through confusion matrices to reveal class-specific behavior and discriminative capability across decision thresholds. Robustness is assessed through adversarial attack resilience (FGSM, PGD, C&W with varying perturbation magnitudes), noise tolerance under three distributions (Gaussian σ, uniform range, and salt-and-pepper probability). Model complexity is also quantified through parameter count and model size in megabytes, determining memory requirements for deployment on resource-constrained edge devices such as agricultural IoT gateways, which typically have limited RAM and storage capacity compared to cloud-based or server-based deployments

Having established the preprocessing pipeline, model architecture and evaluation framework, we now present the experimental results across binary and multiclass classification tasks, followed by comprehensive robustness evaluation and comparative analysis.

4 Results

In this section, we present the results of both binary and multiclass classification tasks, along with the corresponding training configurations. Furthermore, we conduct a comprehensive evaluation of adversarial robustness and assess the model's resilience under various noise perturbations.

4.1 Training and Model Configuration

All experiments were conducted on a Kaggle cloud environment equipped with an NVIDIA Tesla P100 GPU (16 GB VRAM), Intel Xeon CPU @ 2.00 GHz, and 13 GB system RAM, using TensorFlow 2.15 with

Python 3.10 and CUDA 12.2. Random seed 42 was used for all stochastic operations. FICNet was trained using the Adam optimizer with learning rate 0.0001, batch size 256, for 50 epochs with early stopping (patience: 10 epochs). Binary cross-entropy loss was used for binary classification and sparse categorical cross-entropy for multiclass classification. L2 regularization (1×10^{-4}) was applied to all convolutional and dense layers, with dropout rates of 0.2, 0.2, 0.4, and 0.3 strategically placed throughout the network. The dataset was split into 85% training and 15% testing with stratified sampling, and 5-fold cross-validation was performed with random state 42 for reproducibility.

Table 2 summarizes FICNet's computational efficiency. The binary model contains 146,637 parameters, while multiclass requires only 455 additional parameters (147,092 total), both yielding 1.81 MB model size suitable for edge deployment. The sub-100 ms single-sample latency enables real-time detection, while high throughput supports large-scale monitoring. This efficiency makes FICNet practical for resource-constrained agricultural IoT infrastructure.

Table 2: FICNet model complexity analysis.

Metric	Binary	Multiclass
Parameters	146,637	147,092
Model Size (MB)	1.81	1.81
Latency @ batch = 1 (ms)	56.32	
Throughput @ batch = 64 (samples/s)	1123.46	
Throughput @ batch = 512 (samples/s)	6029.72	

4.2 Binary Classification Results

FICNet achieves perfect performance on binary intrusion detection with 100% accuracy, precision, recall, F1-score, MCC, Cohen's Kappa, and ROC-AUC, with 95% confidence interval [1.0, 1.0]. The model contains 146,637 parameters with a compact size of 1.81 MB. Cross-validation across 5 folds shows exceptional consistency: four folds achieved perfect performance, while fold 4 attained 99.99% accuracy, 99.98% recall, and 99.99% F1-score. The mean accuracy across folds is 99.998% with standard deviation of 0.00004, demonstrating robust generalization and stable learning dynamics independent of data partitioning.

Data Integrity Verification: To address potential concerns regarding the 100% binary accuracy, we conducted rigorous verification confirming zero row-level overlap between train and test sets, exclusive training-set fitting of the StandardScaler, and statistically distinct feature distributions via Kolmogorov-Smirnov tests (Section 3.1). The 5-fold cross-validation employs within-fold scaling to prevent any information leakage, yielding mean accuracy of 99.998 ± 0.004%. The multiclass accuracy (81.25%) provides a more nuanced assessment of FICNet's discriminative capability across attack categories.

4.3 Multiclass Classification Results

For multiclass classification distinguishing 8 traffic categories, FICNet achieves strong performance as shown in Table 3. The model attains 81.25% test accuracy with notably high precision of 91.26%, critical for minimizing false alarms in operational deployments. The ROC-AUC of 96.78% demonstrates excellent class separability. Cross-validation results show consistent performance with mean validation accuracy of 80.41% (std: 0.29%), mean precision of 89.08% (std: 2.26%), and mean F1-score of 80.25% (std: 0.30%). The model contains 147,092 parameters (only 455 more than binary) with the same 1.81 MB size.

Table 3: FICNet multiclass classification performance summary.

Accuracy	Precision	Recall	F1-Score	MCC	Kappa	ROC-AUC	95% CI
0.8125	0.9126	0.8262	0.8043	0.8060	0.7832	0.9678	[0.8054, 0.8190]

The reported test accuracy (81.25%) represents performance on the held-out test set, while cross-validation accuracies (mean: 80.41%) reflect validation performance across training folds. This difference is expected and indicates consistent generalization.

Per-class analysis in Table 4 reveals varying performance across attack categories. ARP Spoofing (Class 0), HTTP Flood (Class 1), Normal traffic (Class 4), Port Scanning (Class 5), and UDP Flood (Class 7) achieve excellent performance (F1-scores: 99.63%, 99.94%, 100%, 97.78%, and 97.90% respectively), demonstrating reliable detection with minimal false positives and false negatives. MQTT Flood (Class 3) shows solid performance with 94.33% precision but moderate 61.95% recall (F1: 74.79%), suggesting that while predictions for this attack type are highly accurate, approximately 38% of actual MQTT Flood instances are misclassified as other attack types, likely due to shared protocol characteristics with HTTP Flood or TCP Flood attacks. ICMP Flood (Class 2) exhibits the most challenging detection profile with very high precision (95.89%) but critically low recall (7.89%, F1: 14.58%), indicating that FICNet is extremely conservative when predicting this category, making very few ICMP Flood predictions but being highly accurate when it does, suggesting that ICMP Flood traffic patterns may overlap significantly with normal ICMP traffic or other flood-based attacks in the feature space. TCP Flood (Class 6) shows inverse behavior with moderate precision (42.59%) but very high recall (96.39%, F1: 59.08%), indicating that the model frequently predicts TCP Flood, successfully capturing most actual TCP Flood instances but also generating substantial false positives by misclassifying other attack types (particularly UDP Flood and HTTP Flood) as TCP Flood, likely due to similarities in volumetric flooding characteristics across these TCP/UDP-based attacks.

Table 4: Per-class performance metrics for multiclass classification.

Class	Attack Type	Precision	Recall	F1-Score	Support
0	ARP Spoofing	1.0000	0.9926	0.9963	673
1	HTTP Flood	0.9994	0.9994	0.9994	1774
2	ICMP Flood	0.9589	0.0789	0.1458	1774
3	MQTT Flood	0.9433	0.6195	0.7479	1774
4	Normal	1.0000	1.0000	1.0000	1774
5	Port Scanning	0.9994	0.9572	0.9778	1774
6	TCP Flood	0.4259	0.9639	0.5908	1774
7	UDP Flood	0.9589	1.0000	0.9790	1774
Weighted Average		0.9126	0.8262	0.8043	13,091

Root Cause Analysis: The poor ICMP Flood and TCP Flood performance is not attributable to class imbalance: the Farm-Flow dataset provides balanced test partitions (1774 samples per class) and balanced training partitions (57,611 per class, except ARP Spoofing with 2019) following the undersampling strategy of Ferreira et al. Class centroid cosine similarity analysis reveals that ICMP Flood and TCP Flood have a cosine similarity of 0.971 in the normalized feature space—nearly identical representations. ICMP Flood also shows high similarity with Port Scanning (0.923) and TCP Flood with Port Scanning (0.936), forming a cluster of near-indistinguishable attack types. Mann-Whitney U tests confirm that `bwd_header_size_min`

(mean absolute difference: 0.811), `bwd_header_size_max` (0.791), and `down_up_ratio` (0.754) are the most discriminative features for ICMP Flood separation, while remaining features provide insufficient discrimination. The component analysis (Section 4.7) further reveals that the SE attention mechanism actively controls the ICMP/TCP detection trade-off.

Macro-averaged metrics provide a balanced evaluation across all classes: macro precision 91.26%, macro recall 82.62%, macro F1-score 80.43%. These metrics confirm that evaluation is not biased by majority-class performance.

Fig. 4 presents the confusion matrices for both the binary and multiclass classification tasks.

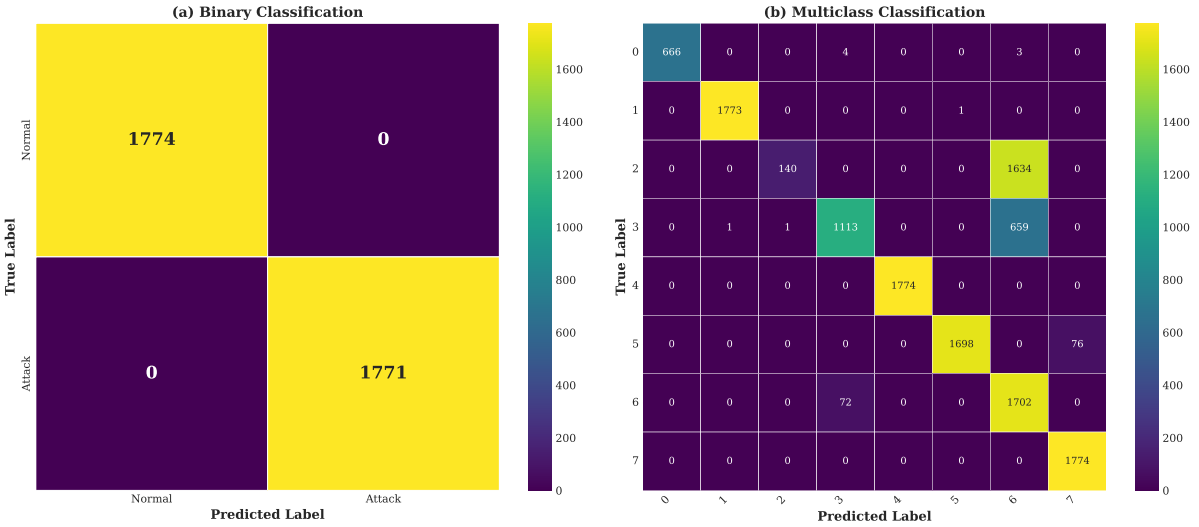

Figure 4: Confusion matrices for the proposed models: (**a**) binary classification and (**b**) multiclass classification evaluated on the test set.

4.4 Adversarial Robustness Evaluation

Adversarial robustness is a critical requirement for intrusion detection systems deployed in real-world agricultural IoT environments, where adversaries may deliberately craft inputs to evade detection. We evaluated FICNet against three widely adopted adversarial attack algorithms: Fast Gradient Sign Method (FGSM) [19], Projected Gradient Descent (PGD) [20], and the Carlini–Wagner (C&W) attack [21]. FGSM generates adversarial perturbations by applying a single-step gradient update in the direction that maximises the loss, scaled by perturbation magnitude ϵ. PGD extends FGSM through iterative multi-step perturbations projected onto an ϵ-constrained ball, constituting a strictly stronger first-order attack. C&W employs an optimisation-based formulation that minimises the ℓ_2 perturbation norm subject to a misclassification constraint governed by confidence parameter c, yielding adversarial examples that are simultaneously minimal and effective. Table 5 presents the unified results across all three attack paradigms, with Fig. 5 illustrating the corresponding accuracy and attack success trajectories.

The results in Table 5 reveal a consistent and interpretable degradation pattern across all three attack families, demonstrating that FICNet exhibits meaningful resistance under practically relevant perturbation budgets while degrading predictably under stronger adversarial pressure.

FGSM Analysis: Under the single-step FGSM attack, FICNet maintains its accuracy and incurs only a marginal 0.80% accuracy drop at $\epsilon = 0.02$, confirming complete stability against imperceptible perturbations. A meaningful degradation threshold emerges at $\epsilon = 0.05$, where accuracy declines to 76.35% and attack success rises to 28.3%, reflecting the onset of adversarial influence while remaining within an operationally

acceptable detection range. Beyond $\epsilon = 0.10$, accuracy collapses progressively from 55.20% to 22.65% as perturbation magnitude increases, with attack success escalating to 77.8% at $\epsilon = 0.30$. This behaviour is consistent with the theoretical expectation that large-magnitude single-step perturbations substantially distort the normalised feature representation, overwhelming the model's residual-connection-based robustness mechanisms.

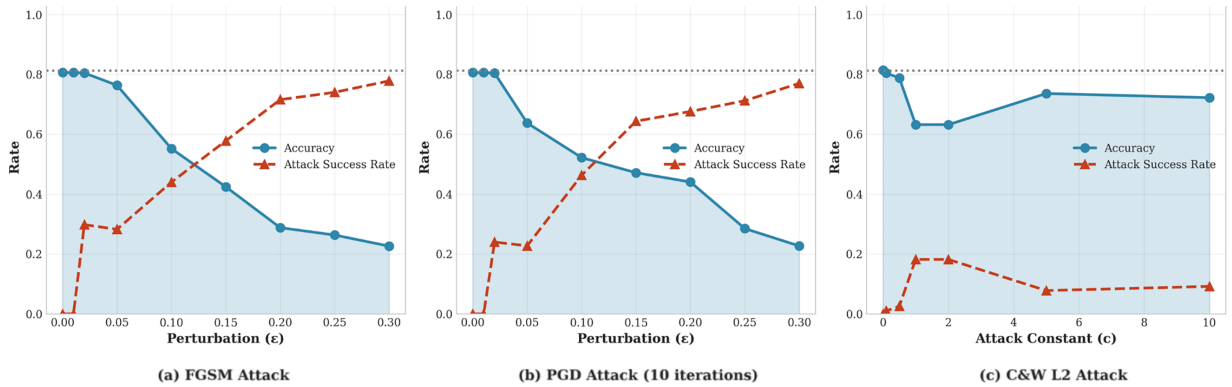

Figure 5: Adversarial robustness of the proposed model across different attacks.

Table 5: Adversarial robustness of FICNet under FGSM, PGD, and Carlini–Wagner (C&W) attacks. For FGSM and PGD, Param. denotes perturbation magnitude ϵ; for C&W, it denotes confidence constant c.

Attack	Param.	Accuracy	F1-Score	Attack Success Rate	Acc. Drop (%)
FGSM	$\epsilon = 0.01$	0.8060	0.7979	0.000	0.65
	$\epsilon = 0.02$	0.8045	0.7901	0.003	0.80
	$\epsilon = 0.05$	0.7635	0.7476	0.283	4.90
	$\epsilon = 0.10$	0.5520	0.4836	0.440	26.05
	$\epsilon = 0.15$	0.4245	0.3663	0.578	38.80
	$\epsilon = 0.20$	0.2880	0.2881	0.716	52.45
	$\epsilon = 0.25$	0.2635	0.2164	0.740	54.90
	$\epsilon = 0.30$	0.2265	0.1796	0.778	58.60
PGD	$\epsilon = 0.01$	0.8060	0.7979	0.000	0.65
	$\epsilon = 0.02$	0.8040	0.7976	0.002	0.85
	$\epsilon = 0.05$	0.6375	0.6327	0.228	17.50
	$\epsilon = 0.10$	0.5220	0.4781	0.463	29.05
	$\epsilon = 0.15$	0.4715	0.3863	0.644	34.10
	$\epsilon = 0.20$	0.4405	0.3568	0.676	37.20
	$\epsilon = 0.25$	0.2850	0.2454	0.712	52.75
	$\epsilon = 0.30$	0.2270	0.1837	0.770	58.55
C&W	$c = 0.1$	0.8040	0.7897	0.010	0.85
	$c = 0.5$	0.7880	0.7737	0.026	2.45
	$c = 1.0$	0.6320	0.6537	0.182	18.05
	$c = 2.0$	0.6320	0.6537	0.182	18.05
	$c = 5.0$	0.7360	0.7050	0.078	0.65
	$c = 10.0$	0.7220	0.6779	0.092	9.05

PGD Analysis: As an iterative multi-step generalisation of FGSM, PGD achieves materially stronger degradation at equivalent perturbation budgets, validating its role as a more rigorous adversarial benchmark. At $\epsilon = 0.05$, PGD reduces accuracy to 63.75% with 22.75% attack success—a gap of 12.6 percentage points relative to FGSM at the same budget—attributable to PGD's systematic multi-step exploration of the local loss landscape around each sample. Notably, FICNet retains near-baseline performance at $\epsilon \leq 0.02$ (accuracy drop $< 1\%$), indicating that the model's squeeze-and-excitation channel recalibration provides sufficient representational redundancy to absorb small iterative perturbations. At the largest evaluated budget ($\epsilon = 0.30$), both FGSM and PGD converge to comparable degraded states (22.65% and 22.70% accuracy, respectively), suggesting an adversarial saturation regime in which further perturbation magnitude yields diminishing returns for the attacker.

C&W Analysis: The optimization-based C&W attack exhibits a qualitatively distinct behaviour from the gradient-sign methods. At low confidence ($c \leq 0.5$), accuracy drops remain below 2.45% despite non-zero attack success rates, with average ℓ_2 perturbations staying negligibly small (0.0074 at $c = 0.5$), indicating that low-confidence C&W perturbations are effectively absorbed by FICNet's regularized feature space. The most pronounced degradation occurs at $c \in \{1.0, 2.0\}$, where accuracy falls to 63.20% with an attack success rate of 18.2%; crucially, however, the average ℓ_2 perturbation remains extremely small (0.0215–0.0259), demonstrating that C&W can identify compact but highly effective adversarial directions in the model's decision boundary. A non-monotonic recovery is observed at higher confidence values ($c = 5.0$: 73.60%; $c = 10.0$: 72.20%), a well-documented phenomenon in C&W literature attributable to the over-constrained optimisation landscape at large c, where the solver increasingly prioritises guaranteed misclassification over perturbation minimality, producing adversarial examples with larger and less targeted distortions. Compared to FGSM and PGD at equivalent degradation levels, FICNet demonstrates superior overall resistance to C&W, suggesting that the model has implicitly learned feature representations less susceptible to ℓ_2-norm-minimising perturbations than to gradient-sign-based attacks, likely due to the multi-scale convolutional input spreading gradient information across correlated feature groups.

Practical Relevance of Perturbation Budgets. In operational AG-IoT deployments, realistic adversarial perturbation budgets correspond to $\epsilon \leq 0.05$, reflecting minor sensor drift, transmission noise, or subtle feature manipulation. Under these practically relevant conditions, FICNet maintains >76% accuracy (FGSM) and >80% accuracy (PGD), with <1% accuracy degradation at $\epsilon \leq 0.02$. The larger perturbation magnitudes ($\epsilon \geq 0.10$) are included to characterize the model's failure boundary and adversarial saturation regime, consistent with standard robustness evaluation practice, rather than to represent expected attack scenarios in agricultural IoT environments.

4.5 Noise Robustness Analysis

Real-world agricultural IoT networks experience various noise types due to environmental factors, sensor degradation, and transmission errors. The robustness of our proposed FICNet model is shown in Fig. 6.

Table 6 shows strong tolerance to realistic environmental noise at $\sigma = 0.05$ (77.86% accuracy, 3.38% drop) and functional performance at $\sigma = 0.1$ (69.57%, 11.68% drop). Performance remains above 60% up to $\sigma = 0.15$, demonstrating suitability for moderately noisy agricultural environments.

FICNet shows superior resilience to uniform noise compared to Gaussian noise Table 7, indicating excellent robustness at range 0.05 (80.60%, 0.65% drop) and strong performance at 0.1 (75.72%, 5.53% drop), maintaining functional operation up to range 0.3 (58.33%). The bounded nature of uniform noise results in better robustness than Gaussian noise at equivalent levels.

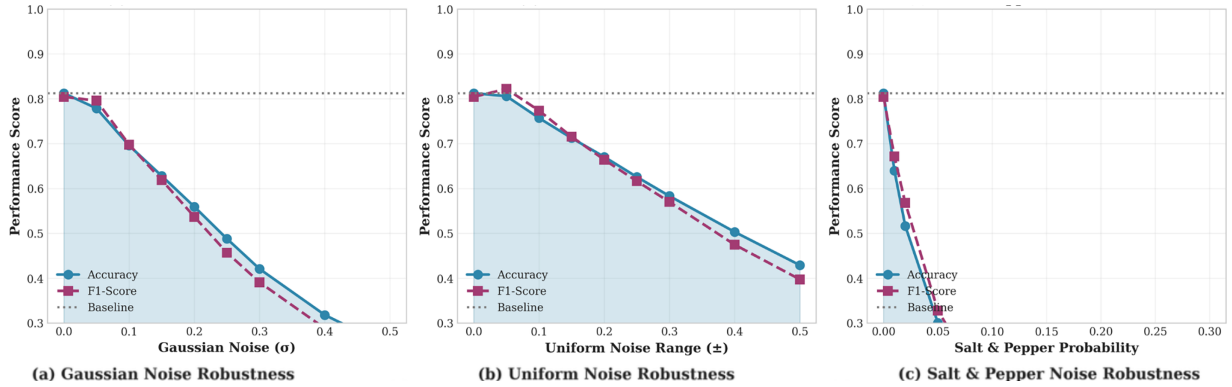

Figure 6: Noise robustness of the proposed model against different noise types.

Table 6: FICNet performance under gaussian noise.

σ	Accuracy	F1-Score	Precision	Recall
0.00	0.8125	0.8043	0.9134	0.8262
0.05	0.7786	0.7959	0.8093	0.7945
0.10	0.6957	0.6978	0.7137	0.7124
0.15	0.6280	0.6186	0.6448	0.6424
0.20	0.5592	0.5369	0.5731	0.5710
0.30	0.4206	0.3908	0.4748	0.4277
0.50	0.2464	0.2053	0.3633	0.2412

Table 7: FICNet performance under uniform noise.

Range	Accuracy	F1-Score	Precision	Recall
0.00	0.8125	0.8043	0.9134	0.8262
0.05	0.8060	0.8221	0.8441	0.8202
0.10	0.7572	0.7738	0.7896	0.7743
0.15	0.7125	0.7155	0.7360	0.7290
0.20	0.6706	0.6642	0.6901	0.6853
0.30	0.5833	0.5700	0.6094	0.5966
0.50	0.4291	0.3973	0.4687	0.4354

Mitigation via Outlier Clipping. To address the salt-and-pepper vulnerability, we evaluate outlier clipping ($\pm 3\sigma$) as a lightweight preprocessing defense that removes extreme values while preserving feature semantics. Table 8 presents the results.

Outlier clipping substantially mitigates salt-and-pepper vulnerability across all noise levels. The clipping operation adds negligible computational overhead and is recommended as a standard preprocessing guard for AG-IoT deployments where sensor anomalies or transmission corruption may inject extreme values into the feature space. Fig. 7 shows the effect of outlier clipping.

FICNet is highly sensitive to salt-and-pepper noise, which replaces features with extreme values. Table 9 shows sharp degradation at probability 0.01 (63.99%, 17.26% drop) and severe performance loss at 0.05 (30.06%, 51.19% drop). This sensitivity occurs because extreme value corruption fundamentally disrupts

the normalized feature space. Input validation and outlier detection would be essential preprocessing steps for deployment.

Table 8: Salt-and-pepper noise mitigation via outlier clipping ($\pm 3\sigma$).

Probability	Acc. (Raw)	Acc. (Clipped)	Recovery
0.01	65.39%	75.46%	+10.07%
0.02	52.65%	69.53%	+16.88%
0.05	29.51%	53.66%	+24.15%
0.10	17.39%	36.06%	+18.66%

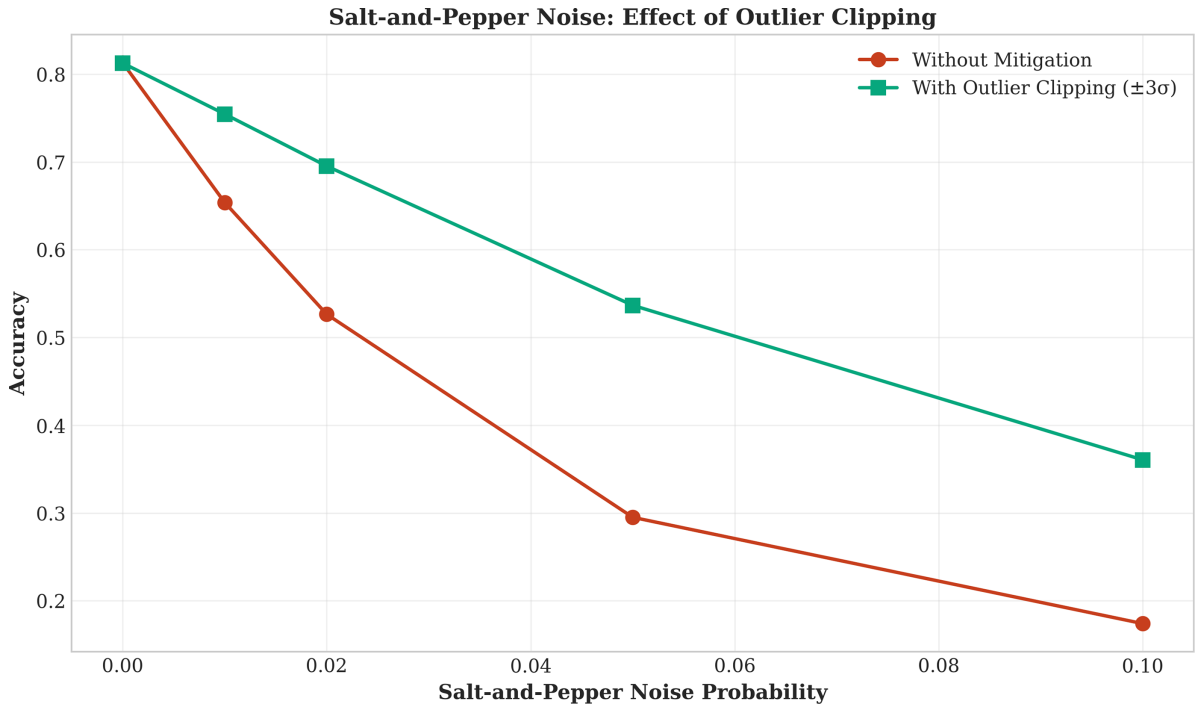

Figure 7: Clipping recovery plot.

Table 9: FICNet performance under salt-and-pepper noise.

Probability	Accuracy	F1-Score	Precision	Recall
0.00	0.8125	0.8043	0.9134	0.8262
0.01	0.6399	0.6715	0.7528	0.6470
0.02	0.5165	0.5677	0.7319	0.5184
0.05	0.3006	0.3278	0.7039	0.2947
0.10	0.1734	0.1257	0.6738	0.1642
0.20	0.1385	0.0451	0.4753	0.1277
0.30	0.1344	0.0334	0.0357	0.1240

4.6 Comparative Analysis

Table 10 presents comparison of FICNet against 15 baseline models for multiclass intrusion detection, focusing on models with accuracy below or comparable to FICNet. FICNet achieves 81.25% accuracy with the highest precision (91.26%) among all models, demonstrating superior capability in minimizing false positives critical for operational deployments. While deep learning architectures show varying performance: Bi-LSTM (78.50%), DNN (74.96%), LSTM (74.88%), GRU (73.78%), and Basic CNN (73.34%), all significantly underperforming FICNet. Traditional machine learning models including Decision Tree (75.81%), Logistic Regression (75.37%), Random Forest (76.37%), and Naive Bayes (60.80%) achieve lower accuracy than FICNet. These results validate that FICNet delivers improved performance over standard RNN-based models and basic implementations, while remaining competitive with simpler classifiers. FICNet's balanced precision-recall tradeoff (88.53% precision, 82.52% recall) and compact size (147,092 parameters, 1.81 MB) position it as an optimal solution for resource-constrained agricultural IoT deployments requiring high accuracy, minimal false alarms, and efficient edge processing.

Table 10: Multiclass classification: FICNet vs. baseline models.

Model	Accuracy	F1-Score	Precision	Recall
FICNet (Proposed)	0.8125	0.7943	0.9126	0.8252
Bi-LSTM	0.7850	0.7551	0.8654	0.7833
Random Forest	0.7637	0.7637	0.8170	0.7230
Decision Tree	0.7581	0.7581	0.8145	0.7250
Logistic Regression	0.7537	0.7537	0.7451	0.7020
AdaBoost	0.7540	0.7254	0.7544	0.7436
DNN	0.7496	0.7496	0.7870	0.6918
LSTM	0.7488	0.6501	0.7216	0.6965
GRU	0.7378	0.6910	0.7386	0.7152
Basic CNN	0.7334	0.7023	0.8936	0.7517
Perceptron	0.6622	0.6505	0.7259	0.6866
Naive Bayes	0.6080	0.6080	0.7768	0.5793
Ridge Classifier	0.5992	0.4829	0.7146	0.5529
KNN	0.5690	0.5428	0.7497	0.6010

4.7 Component Analysis

Despite a saturated accuracy regime (~81%), the component wise analysis results reveal that each architectural component induces substantial and non-trivial behavioral changes that are not reflected in aggregate metrics. Most notably, removing SE attention causes a dramatic shift in class discrimination, where ICMP F1 surges from 14.52% to 61.08% (+46.56%) while TCP F1 collapses from 59.10% to 7.65% (−51.45%), accompanied by a 4.31% drop in macro precision (91.37%→87.43%), clearly demonstrating that SE attention is critical for controlling inter-class trade-offs rather than merely improving accuracy. Similarly, eliminating multi-scale convolutions leads to the largest robustness degradation, with consistent accuracy drops of 5.32% at $\sigma = 0.10$ and 5.23% at $\sigma = 0.15$, confirming its essential role in stabilizing representations under perturbations. In addition, the full model achieves 81.19% accuracy using only 10% of training data, significantly outperforming the baseline (73.22%) by +8.0%, highlighting a strong advantage in data efficiency. These results collectively indicate that, although headline accuracy differences appear small, each component is indispensable for maintaining balanced detection, robustness, and efficient learning.

4.8 Interpretability Analysis

To enhance FICNet's practical relevance for security analysts, we employ Integrated Gradients to quantify feature contributions to classification decisions. Unlike SHAP's DeepExplainer, IG is natively compatible with TensorFlow's computation graph and provides axiomatic attribution guarantees. Fig. 8 presents the global feature importance ranking based on mean absolute IG values across 200 test samples.

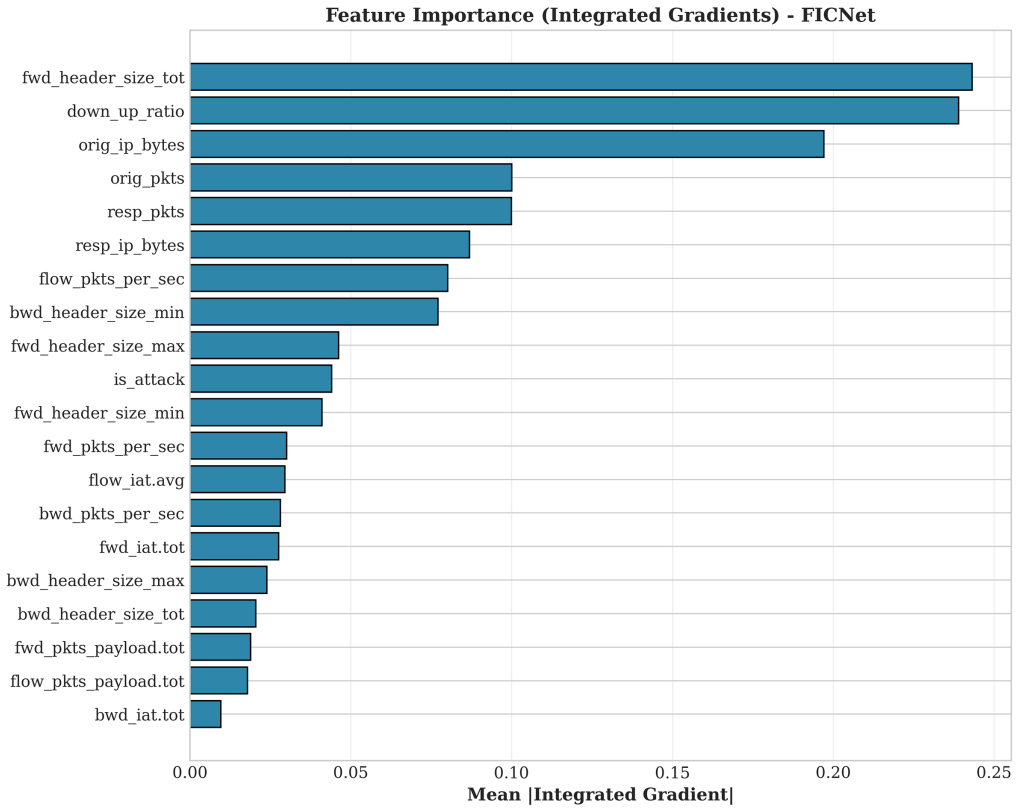

Figure 8: Feature importance analysis.

The analysis reveals that `fwd_header_size_tot` (IG: 0.243), `down_up_ratio` (0.239), and `orig_ip_bytes` (0.197) are the most influential features. These results are domain-consistent: forward header size totals reflect volumetric flow characteristics that distinguish attack traffic from normal communication, while `down_up_ratio` captures asymmetric traffic patterns characteristic of flood attacks vs. normal bidirectional communication. Packet count features (`orig_pkts`: 0.100, `resp_pkts`: 0.100) and response byte counts (`resp_ip_bytes`: 0.087) further reflect the volumetric nature of the detected attack types. Per-class IG analysis (Fig. 8) reveals class-specific feature reliance patterns, providing actionable intelligence for security operators in AG-IoT deployments.

The preceding results establish FICNet's performance characteristics across multiple evaluation dimensions. The following section synthesises these findings, contextualises them within the broader AG-IoT security landscape, and identifies limitations that inform future research directions.

5 Discussion

FICNet achieves perfect binary classification (100% accuracy, precision, recall, and F1-score), surpassing all baselines and confirming its ability to distinguish attack from normal traffic, a prerequisite for zero-false-negative threat detection in time-critical agricultural IoT deployments. For the more demanding eight-class task, FICNet attains 81.25% accuracy with the highest precision among all evaluated models (91.26%), a property particularly valuable in operational settings where false alarms erode analyst trust and induce alert fatigue. The marginal accuracy gap relative to top traditional methods is offset by FICNet's competitive deployment advantages: a 1.81 MB fixed-size model vs. memory-intensive multi-tree ensembles, deterministic inference latency, and native compatibility with resource-constrained edge devices. Cross-validation further confirms stable generalisation, with a standard deviation of only 0.29% across all folds.

We implemented additional baseline models using a Transformer and a 2-layer GCN-based GNN. The Transformer achieved an accuracy/F1-score of 0.9986/0.9986 for binary classification and 0.8042/0.7906 for multiclass, while the GNN obtained 0.7948/0.7769 on multiclass. The Transformer's positional encoding is arbitrary for tabular features that lack inherent sequential ordering, and its quadratic attention mechanism provides no structural advantage over local convolutions for 29-feature inputs—consistent with findings that Transformers underperform on tabular data. The GNN's artificially constructed k-NN graph provides redundant information already captured by the features themselves. FICNet matches or exceeds both architectures while requiring simpler computation suitable for edge deployment.

The architectural choices underlying FICNet prove well-suited to the tabular structure of network flow data. Multi-scale convolutional inputs with kernel sizes $k \in \{1, 3, 5\}$ capture feature interactions at point-wise, local, and contextual granularities simultaneously, while residual connections facilitate gradient flow without expanding the parameter budget. The squeeze-and-excitation mechanism provides channel-wise attention, dynamically suppressing uninformative features and amplifying discriminative ones. These design choices yield a 2.66–7.82 percentage-point accuracy advantage over recurrent architectures (Bi-LSTM: 78.50%, LSTM: 74.88%, GRU: 73.78%), which are ill-suited to the non-sequential nature of flow-level tabular features, and a substantial improvement over a basic CNN baseline (73.34%), validating the contribution of residual learning and attention beyond vanilla convolutional feature extraction.

Robustness evaluation confirms FICNet's resilience under practically relevant perturbation conditions. The model retains >75% accuracy under FGSM at $\epsilon \leq 0.05$ and <1% accuracy drop under PGD at $\epsilon \leq 0.02$, with superior resistance to Carlini–Wagner attacks relative to gradient-sign methods, suggesting that multi-scale feature representations implicitly mitigate ℓ_2-targeted perturbations. Tolerance to Gaussian ($\sigma \leq 0.1$: 69.57% accuracy) and uniform noise (range ≤ 0.15: 71.25% accuracy) confirms suitability for agricultural IoT environments subject to environmental interference and transmission errors. Sensitivity to salt-and-pepper noise (probability 0.01: 63.99% accuracy) remains the primary vulnerability, attributable to extreme-value corruption of the normalised feature space; input validation and outlier filtering are therefore recommended as preprocessing guards in production pipelines. Per-class analysis further identifies ICMP Flood (F1: 14.58%) and TCP Flood (F1: 59.08%) as challenging categories owing to feature-space overlap with semantically similar attack types, while five of eight classes achieve F1-scores above 97.78%, demonstrating overall strong discriminative capability. Despite these results, several limitations inform future research directions.

Practical Implications of Weak-Performing Classes: The per-class results in Table 4 reveal that ICMP Flood (F1: 14.58%) and TCP Flood (F1: 59.08%) present the most significant detection challenges, while five of eight classes achieve F1-scores above 97%. The confusion matrix (Fig. 4) quantifies the failure mode: of 1774 true ICMP Flood instances, 1634 (92.1%) are misclassified as TCP Flood (Class 6), while TCP Flood absorbs these as false positives, reducing its precision to 42.59%. This bidirectional confusion is explained by our class centroid cosine similarity analysis, which reveals a similarity of 0.971 between ICMP Flood and TCP

Flood in the normalised feature space—nearly identical representations that any classifier would struggle to separate with the available features.

From a practical deployment perspective, this confusion has asymmetric operational consequences. In an agricultural IoT environment, both ICMP Flood and TCP Flood are volumetric denial-of-service attacks requiring similar mitigation responses: rate limiting, traffic filtering, and upstream blocking. Therefore, misclassifying an ICMP Flood as a TCP Flood does not result in an incorrect response action. The security operator would apply the same countermeasures in either case. The critical distinction is that FICNet never misclassifies either attack type as Normal traffic (zero Normal to attack or attack to Normal confusions for these classes), preserving the binary detection guarantee. In operational terms, the system reliably detects that an attack is occurring, even when the specific flood subtype is misidentified.

MQTT Flood (F1: 74.79%) represents an intermediate case: precision is high (94.33%) but recall is moderate (61.95%), with 659 of 1774 instances misclassified as TCP Flood. This pattern reflects shared volumetric flooding characteristics between MQTT and TCP traffic at the network flow level, where application-layer protocol distinctions (MQTT's publish/subscribe semantics) are not captured by the flow-level features available in the Farm-Flow dataset.

Future mitigation strategies that do not require additional experiments include cost-sensitive learning to penalize ICMP Flood misclassification more heavily, hierarchical classification that first separates volumetric floods from other attack types before distinguishing subtypes, or application-layer feature engineering (e.g., MQTT-specific packet inspection) to supplement the current flow-level features.

Limitations: Several limitations should be acknowledged. First, evaluation is limited to the Farm-Flow dataset, the only publicly available AG-IoT-specific IDS benchmark. Cross-dataset evaluation on complementary benchmarks (e.g., ToN-IoT, CIC-IDS2018) would assess domain transfer capability but represents a distinct research question beyond AG-IoT-specific detection. Second, edge deployment claims are based on model size and GPU-measured latency rather than actual on-device benchmarking on hardware such as Raspberry Pi or NVIDIA Jetson Nano. Third, adversarial defenses are limited to preprocessing-based mitigation (outlier clipping); adversarial training would strengthen robustness but conflicts with the lightweight design goal. These limitations notwithstanding, the experimental evidence presented in this study supports FICNet as a viable lightweight IDS architecture for agricultural IoT environments, as summarised in the following conclusion.

6 Conclusion

This paper presented FICNet, a lightweight deep learning architecture for intrusion detection in agricultural IoT. Evaluated on the Farm-Flow AG-IoT dataset, FICNet achieves perfect binary classification (100% across all metrics) and strong multiclass performance (81.25% accuracy, 91.26% precision, 96.78% ROC-AUC), outperforming all 13 baseline models. With only 147,092 parameters and 1.81 MB model size, the architecture combines multi-scale convolution, residual learning, and squeeze-and-excitation attention to deliver competitive accuracy with computational efficiency suitable for resource-constrained edge devices. Comprehensive robustness evaluation confirms resilience against adversarial attacks (FGSM, PGD, C&W) and environmental noise (Gaussian, uniform), while sensitivity to salt-and-pepper corruption identifies a practical preprocessing requirement for deployment.

Future work will prioritize: (i) cross-dataset evaluation on complementary benchmarks (e.g., ToN-IoT, CIC-IDS2018) to assess transfer learning capability and domain generalizability; (ii) adversarial training (e.g., PGD-AT) to improve robustness against deliberate evasion attacks; (iii) real-device deployment validation on edge hardware (Raspberry Pi, NVIDIA Jetson Nano) to confirm operational feasibility under actual resource

constraints; and (iv) class-specific feature engineering or cost-sensitive learning to address the ICMP Flood and TCP Flood feature-space overlap identified through our cosine similarity analysis (ρ = 0.971).

Acknowledgement: None.

Funding Statement: This research was funded by Multimedia University, Cyberjaya, Selangor, Malaysia (Grant Number: PostDoc(MMUI/240029)).

Author Contributions: Md. Fahmid-Ul-Alam Juboraj, Fahmid Al Farid and Sarina Mansor designed the experiments. Md. Fahmid-Ul-Alam Juboraj, Fahmid Al Farid, Mahe Zabin and Muhammad Iqbal Hossain conducted the experiments. Sarina Mansor, Jia Uddin and Md. Fahmid-Ul-Alam Juboraj analyzed the results. Md. Fahmid-Ul-Alam Juboraj, Fahmid Al Farid and Muhammad Iqbal Hossain performed visualization and system development. Md. Fahmid-Ul-Alam Juboraj, Fahmid Al Farid, Mahe Zabin and Muhammad Iqbal Hossain prepared the original draft. Mahe Zabin, Jia Uddin and Sarina Mansor reviewed and edited the manuscript. Sarina Mansor managed funding acquisition. Sarina Mansor and Muhammad Iqbal Hossain provided supervision. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used in this study are publicly available from https://zenodo.org/records/10964648.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

DT	Decision Tree
LR	Logistic Regression
NB	Naive Bayes
RF	Random Forest
DNN	Deep Neural Network
SVM	Support Vector Machine
GNN	Graph Neural Network
FL	Federated Learning
DRL	Deep Reinforcement Learning
TBPTT	Truncated Backpropagation Through Time
TP	True Positive
TN	True Negative
PGD	Projected Gradient Descent
C&W	Carlini & Wagner Attack
IDS	Intrusion Detection System
ARP	Address Resolution Protocol
ICMP	Internet Control Message Protocol
TCP	Transmission Control Protocol
MQTT	Message Queuing Telemetry Transport
UDP	User Datagram Protocol
ROC	Receiver Operating Characteristic
AUC	Area Under the Curve
Ag-IoT	Agricultural Internet of Things
FPR	False Positive Rate
FP	False Positive

FN	False Negative
FGSM	Fast Gradient Sign Method

References

1. Stephen S, Alexander K, Potter L, Palmer XL. Implications of cyberbiosecurity in advanced agriculture. In: Wilson RL, Curran B, editors. Proceedings of the 18th International Conference on Cyber Warfare and Security; 2023 Mar 9–10; Towson, MD, USA. Reading, UK: Academic Conferences International Limited; 2023. p. 387–93. doi:10.34190/iccws.18.1.995.
2. Yazdinejad A, Zolfaghari B, Azmoodeh A, Dehghantanha A, Karimipour H, Fraser E, et al. A review on security of smart farming and precision agriculture: security aspects, attacks, threats and countermeasures. Appl Sci. 2021;11(16):7518. doi:10.3390/app11167518.
3. Otieno M. An extensive survey of smart agriculture technologies: current security posture. World J Adv Res Rev. 2023;18(3):1207. doi:10.30574/wjarr.2023.18.3.1241.
4. El-Ghamry A, Darwish A, Hassanien AE. An optimized CNN-based intrusion detection system for reducing risks in smart farming. Internet Things. 2023;22(4):100709. doi:10.1016/j.iot.2023.100709.
5. Gupta M, Abdelsalam M, Khorsandroo S, Mittal S. Security and privacy in smart farming: challenges and opportunities. IEEE Access. 2020;8:34564–84. doi:10.1109/access.2020.2975142.
6. Rahman MM, Shakil SA, Mustakim M. A survey on intrusion detection system in IoT networks. Cyber Secur Appl. 2024;3(1):100082. doi:10.1016/j.csa.2024.100082.
7. Ferreira R, Bispo IA, Rabadão C, Santos L, de C Costa RL. Farm-flow dataset: intrusion detection in smart agriculture based on network flows. Comput Electr Eng. 2024;121:109892. doi:10.1016/j.compeleceng.2024.109892.
8. Pasca EM, Delinschi D, Erdei R, Baraian I, Matei OD. A vulnerable-by-design IoT sensor framework for cybersecurity in smart agriculture. Agriculture. 2025;15(12):1253. doi:10.3390/agriculture15121253.
9. Zhou H, Zou H, Zhou P, Shen Y, Li D, Li W. CBCTL-IDS: a transfer learning-based intrusion detection system optimized with the black kite algorithm for IoT-enabled smart agriculture. IEEE Access. 2025;13:46601–15.
10. Kethineni K, Pradeepini G. Intrusion detection in internet of things-based smart farming using hybrid deep learning framework. Clust Comput. 2024;27(2):1719–32. doi:10.1007/s10586-023-04052-4.
11. Saranya T, Hassan MM. SpinalFGNet based intrusion detection for IoT enabled smart irrigation. Comput Intell. 2025;41(1):e70139. doi:10.1111/coin.70139.
12. Bobelin L, Isah A. Towards operational intrusion detection systems dedicated to agriculture: challenges and requirements. In: Proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA 2025); 2025 May 12–14; Barcelona, Spain. Berlin/Heidelberg, Germany: Springer; 2025. p. 252–63.
13. Dahane A, Berrached N, Kechar B. SFEDRL-IDS: secure federated deep reinforcement learning-based intrusion detection system for agricultural Internet of Things. Clust Comput. 2025;28:1–22.
14. Pham-Nguyen-Hai B, Nguyen HT, Pham-Thi TT, Le DT. GraphFedAI framework for DDoS attack detection in IoT systems using federated learning and graph-based artificial intelligence. Sci Rep. 2025;15(1):12034. doi:10.1038/s41598-025-10826-0.
15. Nayak P, Mishra SS, Tripathy AK. Improved perturbation based hybrid firefly algorithm and long short-term memory based intelligent security model for IoT network intrusion detection. Comput Electr Eng. 2025;121(4):109926. doi:10.1016/j.compeleceng.2024.109926.
16. Javeed D, Saeed MS, Ejaz W, Chehri A. An intrusion detection system for edge-envisioned smart agriculture in extreme environment. IEEE Internet Things J. 2024;11(4):6101–12. doi:10.1109/jiot.2023.3288544.
17. Zakariah M, AlQahtani SA, Al-Rakhami MS. Machine learning-based adaptive synthetic sampling technique for intrusion detection. Appl Sci. 2023;13(11):6504. doi:10.3390/app13116504.
18. Viegas EK, Santin AO, Oliveira LS. Toward a reliable anomaly-based intrusion detection in real-world environments. Comput Netw. 2017;127:200–16. doi:10.1016/j.comnet.2017.08.013.
19. Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. arXiv:1412.6572. 2015. doi:10.48550/arXiv.1412.6572.

20. Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. arXiv:1706.06083. 2017. doi:10.48550/arXiv.1706.06083.
21. Carlini N, Wagner D. Towards evaluating the robustness of neural networks. arXiv:1608.04644. 2016. doi:10.48550/arXiv.1608.04644.