Int J Adv Life Sci Res. Volume 9(4) 161-192
https://doi.org/10.31632/ijalsr.2026.v09i04.011

International Journal of Advancement in Life Sciences Research

Online ISSN: 2581-4877

Journal homepage http://ijalsr.org



Original Article

Precision Hydration and Integrated Pesticide Management System for Paddy Fields using Machine Learning and IOT

Sujatha Kesavan[1]*, Sasikala Krishnasamy[2], Malathi Marichamy[3], Latha Balasubramanian[4], Srividhya Veerabathran[5]

[1] *Department of EEE, Dr. M.G.R. Educational and Research Institute, Maduravoyal, Chennai 600095. Tamil Nadu, India*

[2]*Department of EEE, Vels Institute of Science, Technology & Advanced Studies (VISTAS), Velan Nagar, Krishnapuram, Pallavaram, Tambaram, Chennai 600117, Tamil Nadu, India*

[3] *Department of ECE, Rajalakshmi Institute of Technology, Kuthambakkam, Chennai 600124, Tamil Nadu, India*

[4] *Department of Physics, Dr. M.G.R. Educational and Research Institute, Maduravoyal, Chennai 600095. Tamil Nadu, India*

[5] *Department of EEE, Meenakshi College of Engineering, Annaji Nagar, KK Nagar West, K. K. Nagar, Chennai 600078, Tamil Nadu, India*

[*] ***Corresponding Author's Email: sujathak73586@gmail.com***

Abstract

Background: Precision hydration optimizes water use in paddy fields via IoT sensors for soil moisture, while integrated pesticide management employs machine learning (ML) for real-time pest detection to minimize chemical inputs. This system addresses paddy crop challenges, such as water scarcity and pest infestations and enhances yield. The aim of this work is to project the results of machine learning models for paddy crop disease prediction that can analyze the plant health data sets using various algorithms. **Methods:** This starts with data collection and pre-processing, with missing information prevention and feature engineering. The system deploys a network of low-cost soil moisture, water level, temperature, humidity sensors and pest detection ML models across representative paddy fields. The ML models like CNN, YOLOv8, LSTM are used for plant disease prediction. Finally, efficient models are implemented in a user-friendly application whose reliability has been extensively tested. **Results:** Precision hydration optimizes water use in paddy fields via IoT sensors for soil moisture, while integrated pesticide management employs Machine Learning (ML) for real-time pest detection to minimize chemical inputs. The system addresses paddy crop challenges like water scarcity and pest infestations, enhancing yield. The developed system enables early detection of plant diseases and monitoring of the water supply to paddy fields, thereby improving the agricultural produce. **Conclusion:** This work envisions the development of an efficient technology for early detection of plant diseases, monitoring the water supply to paddy fields and thereby improving the agricultural produce. Generative AI is increasingly explored in precision agriculture for tasks such as synthetic training-data augmentation and natural-language farmer advisories; while such extensions are discussed as future work in this study, the present system's disease-prediction, irrigation, and pest-management outcomes are delivered entirely through the validated CNN, YOLOv8, and LSTM pipeline described here.

Keywords: *CNN; IoT-based Decision Support; LSTM; Plant Diseases; Precision Agriculture; Supervised Learning; YOLOv8*

Received on: 10th March 2026; Revised version received on 15th September 2026; Accepted: 17th September 2026

Introduction

The proposed system integrates IoT sensors (soil moisture, temperature, humidity, water level) with ML models (CNN, YOLOv8, LSTM) for automated decisions on irrigation and pesticide application. Edge devices like Raspberry Pi process data locally, sending alerts via cloud platforms like Firebase for farmer intervention. Solar-powered traps with cameras capture pest images, enabling precision actions over traditional flooding methods.

In India and around the world, rice is the main food source. Numerous illnesses target it at different phases of its growth. Because of the vast amount of land owned by individual farmers, the wide range of diseases, and the fact that multiple diseases can exist in the same plant, it is therefore vital to detect and treat these diseases early to assure high quantity and highest quality. However, this is extremely challenging. Remote locations lack access to agricultural experts' knowledge, and the process takes a long time. Automated systems are therefore necessary. Research has been conducted utilizing a variety of machine learning techniques to help farmers and enhance the accuracy of plant disease diagnosis (Barbedo, 2013).

However, feature selection methods have a significant impact on how accurate such algorithms are. By removing the requirement for picture pre-processing and offering built-in feature selection, recent studies on convolutional neural networks have significantly advanced image-based recognition. It's also challenging to get a big dataset for these kinds of problems. It is better to employ a model that has been pretrained on a large dataset when the dataset size is relatively small. This technique, known as transfer learning, can be used to build a model that can be used as a fixed feature extract, to eliminate the last fully connected layer, or to fine-tune the final few layers so that they are more tailored to the dataset in question (Dey *et al.,* 2016).

Since everyone has access to a mobile phone these days, an automated system has been developed that allows farmers to upload images of diseased leaves to our server, where a neural network will identify the disease and send the farmer a classification of the disease and its treatment. The architecture for the automated system's disease classification component has been suggested in this work. The deep learning approach is used in this work for predicting the illness of rice plants from the dataset, which is gathered over the past few months, and was inspired by the work on machine learning algorithms. Using machine learning models, a massive Image Net data is trained. Finally, some error analysis was performed and attempts were made to provide a solution for the problem (Arribas *et al.,* 2011; Javidan *et al.,* 2024).



a) Blast b) Bacterial Leaf Blight c) Brown Spot

Figure 1: *Sample of Leaf Diseases in Paddy Crops*

It is a fungal disease caused by Magnaporthezae. The initial symptoms are white to grey-green spots that are elliptical or spindle-shaped with dark red to brownish borders. Some have a diamond shape with broad centers and pointed ends. In Figure 1(a), the spindle-shaped lesions with white spots and dark brown borders can be seen (Kandpal & Kumar, 2024; Kuswidiyanto *et al*., 2022). Xanthomonas oryzae is the causative agent of this bacterial illness. The diseased leaves roll up, turn greyish green, and eventually turn yellow before turning straw-colored and dying. The lesions move toward the base and have wavy edges. Bacterial oozing that resembles morning dew drop is visible on new lesions (Lakshmi & Savarimuthu, 2021; Saha *et al.,* 2024). Leaf blight-affected leaves are depicted in Figure

1(b). The illness is caused by fungi. Numerous large spots on the infected leaves have the potential to harm the entire leaf. Small, rounded dark brown to purple-brown lesions are visible in the leaves at the beginning. The fungus's poison causes fully formed lesions to be round to oval, with a light brown to gray center and a reddish brown border around it. The tiny, dark brown lesions of the leaves afflicted by brown spot are displayed in Figure 1(c). Although the image appears healthy, it is labelled as a brown spot, most likely because of the blurriness and low contrast, as in Figure 2.



Figure 2: *Healthy Leaf*

Precision irrigation, pest and disease detection, and IoT-based crop monitoring have all been studied separately (Ajith *et al*., 2025; Mehta *et al*., 2025; Kalpana *et al*., 2024; Verma & Kaur, 2025). However, these approaches usually treat water management and pest management as distinct decision pipelines, each using its own sensors, models, and thresholds. The Precision Hydration and Integrated Pest Management (PH-IPM) framework, which unifies these two decision loops into a single edge-to-cloud pipeline, is the main contribution of this work. An irrigation scheduling model and a pest/disease detection and forecasting model are fed by a shared stream of field sensor data (soil moisture, temperature, humidity, and pest-trap imagery), and the two outputs are reconciled on a single dashboard that farmers. This link is driven by a well-known agronomic interaction, which is used here as an active control lever rather than as an unintentional side effect: certain wetting-drying cycles reduce moisture-favoring bug populations. The PH-IPM framework is implemented and assessed on a two-season, 5-ha field trial in Tamil Nadu, offering real-world evidence for both the irrigation and pest-management components together rather than separately, in contrast to earlier IoT-ML systems that are validated on public datasets or in simulation. The Materials and Methods section details the sensor, image, and model pipeline underlying the PH-IPM framework; potential extensions of this framework using IoT decision based support system are discussed in the Future Scope section.

Literature Survey

Using IoT-collected data, the system dynamically controls irrigation pumps, sluice gates, and water valves to maintain near-optimal water level ranges recommended for paddy growth, thereby reducing over- or under irrigation and minimizing water wastage. Machine learning models (such as LSTM or regression-based decision rules) are trained on multi-source data—soil moisture, weather, and historical yield records—to generate site-specific irrigation schedules that adapt to local conditions and crop stage. Field studies on similar precision irrigation systems show that such approaches can cut irrigation volumes by roughly 20–25% while maintaining or improving grain yield (Sonobe *et al*., 2020; Strange & Scott, 2005).

The system integrates pest detection sensors and image-based monitoring (camera traps or drone-captured imagery) to identify pest presence and severity, feeding this information into a machine learning classifier that recommends targeted pesticide application or alternative interventions such as biocontrol measures. By coupling pest risk maps with real-time environmental data, the framework enables precision spraying, applying chemicals only where and when needed, reducing pesticide load, residue levels, and environmental impact while lowering input costs for farmers (Ye *et al*., 2024; Ajith *et al*., 2025).

Materials and Methods

IoT sensors monitor real-time soil moisture and environmental factors to activate pumps or valves only in water-stressed zones, reducing waste by up to 30% compared to manual methods (Kalpana *et al.,* 2024). ML models like ANN predict water needs with 95.6% accuracy, using features from thermal drone imagery and historical data. For paddy, maintaining optimal levels (5-10 cm) prevents diseases while conserving the water (Table 1).

Table 1: *Components in the Proposed Model*

Component	Sensors Used	ML Model	Key Benefit
Soil Monitoring	Moisture, pH	ANN/SVM	95% prediction accuracy
Irrigation Control	Water level, Rain	Decision Trees	Automated valve activation
Drone Integration	Thermal Camera	CNN	Stress mapping

Irrigation Decision and Adaptive Thresholding

The irrigation controller functions as a closed-loop sensor-based decision-making system. Temperature, relative humidity, rainfall, soil moisture, and field water level are all recorded every two hours. While the water-level sensor confirms whether the necessary paddy water level is available, the measured soil-moisture value is compared with a predetermined stress threshold of roughly 30%. The edge controller turns on the pump or valve until the desired moisture/water-level condition is restored when soil moisture drops below the threshold and sufficient rainfall is not detected. To avoid over-irrigation, irrigation is repressed if rainfall is detected or the target condition is already met.

Rather than being viewed as a constant value throughout the crop cycle, the irrigation threshold is interpreted based on the crop growth stage and current environmental conditions. Water-sensitive development phases allow irrigation at a higher moisture set point, whereas less water-demanding stages allow controlled drying before irrigation. While rainfall observations are utilized to avoid irrigation when natural precipitation is enough, temperature and relative humidity offer additional information on atmospheric water demand. As a result, decisions about watering are made based on the combined sensor status rather than just soil moisture. Sensor acquisition → water-stress assessment → threshold and environmental condition check → pump/valve activation → sensor feedback is the final control sequence. Throughout the crop growth cycle, the suggested PH-IPM architecture may adapt dynamically to changes in crop water requirements and field environmental conditions thanks to this closed-loop mechanism.

The threshold is further interpreted according to crop growth stage and prevailing environmental conditions. During high water-demand stages, the controller permits irrigation at a higher moisture set point, whereas during less water-demanding stages, it allows controlled drying before irrigation. Temperature and relative humidity provide additional information on atmospheric water demand, while rainfall information prevents irrigation immediately before or after natural precipitation. Consequently, irrigation decisions are generated from the combined sensor states rather than from soil moisture alone. The resulting control sequence is from sensor acquisition, stress assessment, threshold/environmental check, pump/valve control and finally the feedback, enabling adaptive irrigation throughout the crop growth cycle. Smart traps with pheromones attract pests like fruit flies, using CNN for 97.5% detection accuracy on datasets of paddy crop images. YOLOv8 achieves 97.3% mAP for real-time classification, integrating with IPM to target sprays and cut usage. Paddy-specific datasets support models identifying rice blast and sheath blight (Mehta *et al*., 2025; Bassine*et al.,* 2023).

Data flows from field sensors to edge ML for hydration/pest decisions, with cloud analytics for predictive forecasting. Hybrid CNN-LSTM models forecast outbreaks with 94.2% accuracy, linking water regimes to pest control (e.g., flooding suppresses certain insects). Farmers access via mobile

apps for threshold values of moisture content on the fields, which is likely to be around less than 30%, or pest count to be greater than 50.

IoT-based monitoring systems integrated with IPM for rice crops use sensors and cameras to track pests in real-time, triggering targeted interventions based on economic thresholds while minimizing broad-spectrum pesticides (Figure 3a). These systems combine environmental data (soil moisture and humidity) with ML detection for proactive rice blast or stem borer control, achieving 30-40% chemical reductions (Delfani *et al.,* 2024; Ejaz *et al.,* 2025). Sensors like DHT11 (temperature/humidity), soil moisture probes, and pheromone traps with ESP32 cameras feed data to edge devices running CNN or YOLOv8 models. Cloud platforms (Firebase/MQTT) process trends via LSTM for outbreak forecasts, sending mobile alerts when pests exceed ETLs. Solar-powered units ensure field reliability, with ozone/light traps as non-chemical backups (Figure 3b).

Integration Workflow

1. Field sensors capture images/data continuously.
2. Edge ML classifies pests and soil moisture (95% accuracy on rice datasets).
3. IPM logic applies thresholds, activating drones/sprayers or bio-agents.
4. Farmer dashboard shows maps, predictions, and actions.
5. Implementing IoT soil moisture sensors for paddy irrigation involves using capacitive sensors with microcontrollers like ESP32 or Arduino to automate water pumps based on real-time soil data. This setup suits paddy fields, where maintaining 5-10 cm water depth prevents stress while conserving resources.
6. Capacitive soil moisture sensor (TLC555-based for reliability).
7. ESP32 or Arduino board for processing.
8. Relay module (5V, single channel) to control water pump.
9. Submersible water pump (5-12V DC).
10. Jumper wires, breadboard, power supply (5V USB or battery).
11. Optional: DHT22 for humidity/temp, solar panel for field deployment.
12. Power sensor/probe separately to avoid corrosion; probe max depth per limit line.
13. Deploy IoT nodes (ESP32/Raspberry Pi) with sensors in paddy
14. Train ML on public datasets (e.g., Paddy Pests), fine-tune for local pests.
15. Integrate cloud for dashboards; test in phases (lab → field).
16. Validate with metrics: water savings, pest reduction (target 20-30%).

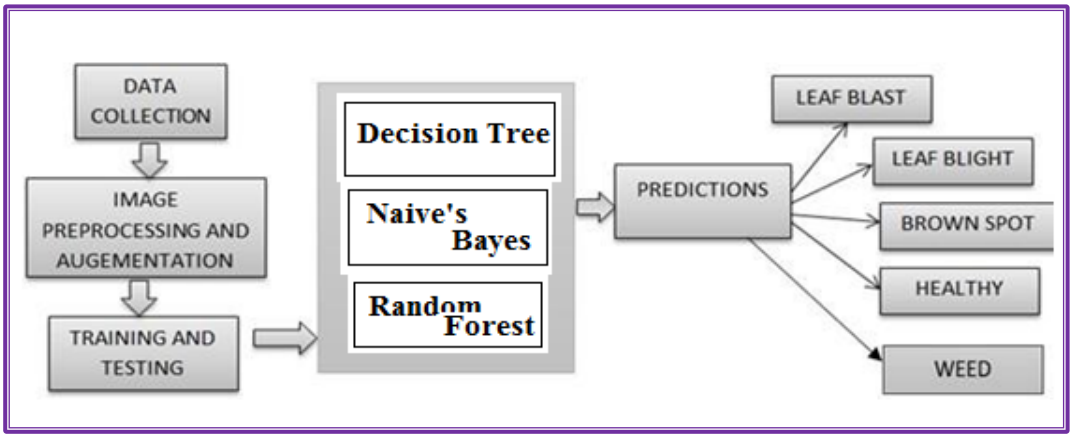


Figure 3a: *Block Diagram for Paddy Crop Disease Identification*

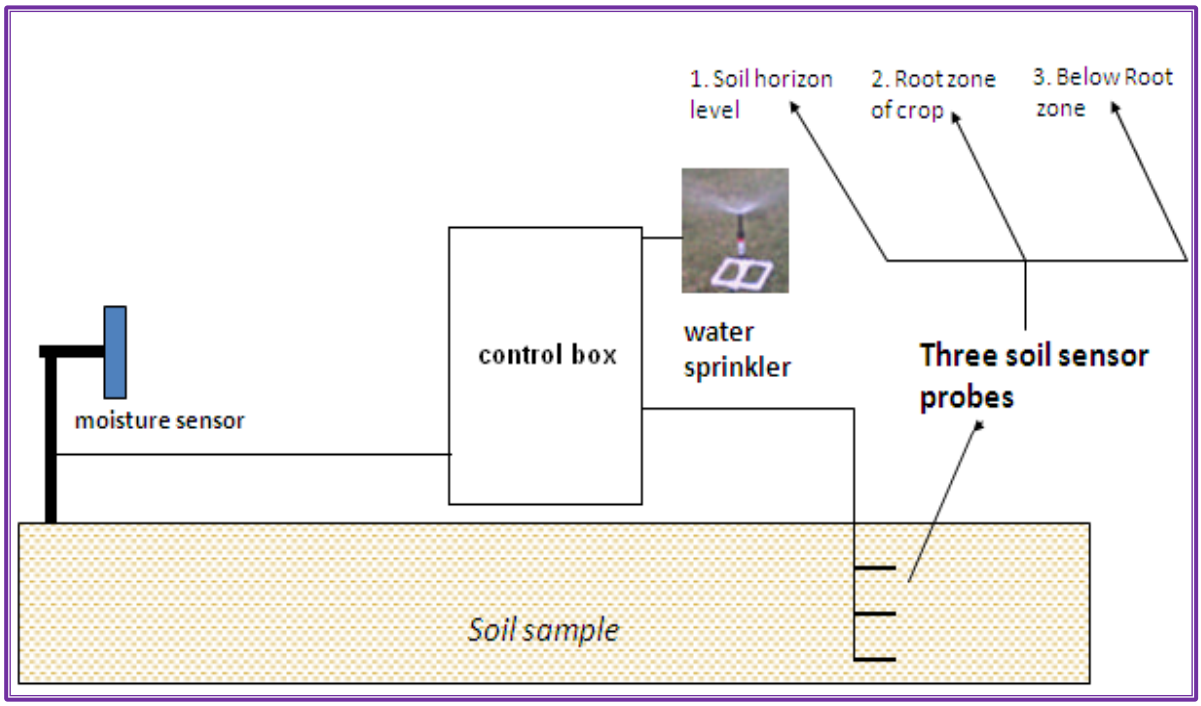


Figure 3b: *Block Diagram for Precision Hydration of Paddy Fields*

Data Acquisition: Using solar-powered pheromone camera traps and drone thermal cameras placed throughout representative field zones, ESP32/Raspberry Pi edge nodes gather data on soil moisture, temperature, humidity, water-level sensor readings, and photos.

Pre-processing: Before sensor streams and images are sent to any model, they are subjected to feature engineering and missing-value handling. Frame-level detection (CNN/YOLOv8 stage): For pest/leaf-disease classification, individual captured images are sent to CNN when a single static verdict is adequate (e.g., lab-style trap datasets) and to YOLOv8 when real-time localization of multiple infected regions is needed directly on the edge device (Raspberry Pi) using field-captured images.

Temporal Forecasting (CNN-LSTM stage): To forecast outbreak trajectories rather than just current-frame status, the sequence of CNN/YOLOv8 outputs across time is input into the CNN-LSTM, along with environmental sensor time series data.

Environmental variability is addressed through temporal feature extraction, including rate of change, rolling statistics, and time-of-day/season encoding. Thus, irrigation and pest-risk predictions are based on temporal trends rather than isolated sensor measurements.

Decision/IPM Logic Layer: To activate irrigation valves, targeted sprayers, or bio-control measures, thresholds (such as pest count > 50, moisture < 30%) are applied to the combined CNN/YOLOv8/LSTM outputs.

Cloud and Alerting: Classification maps, forecasts, and suggested actions are displayed on the farmer-facing mobile app/dashboard, while Firebase/MQTT aggregates data.The schematic of the flow is pictorially represented in Figure 3(c).

The PH-IPM system integrates sensing, edge-based image analysis, temporal forecasting, irrigation decision logic, pest/IPM decision assistance, and cloud-based visualization. Each of these elements' contributions is examined separately as well as in light of the entire decision loop. Wetting-drying circumstances can be considered as an additional pest-management lever since irrigation and pest-management decisions are integrated, and the same environmental sensor stream serves as the foundation for both hydration and pest-risk decisions. System-level proof is provided by the two-season, 5-ha field validation, where the integrated framework reportedly reduced irrigation water use

by 25.3%, increased yield by 10.21%, and reduced pesticide use by 40%. Rather than being ascribed to a single machine learning model, these results are understood as the outputs of the entire system.

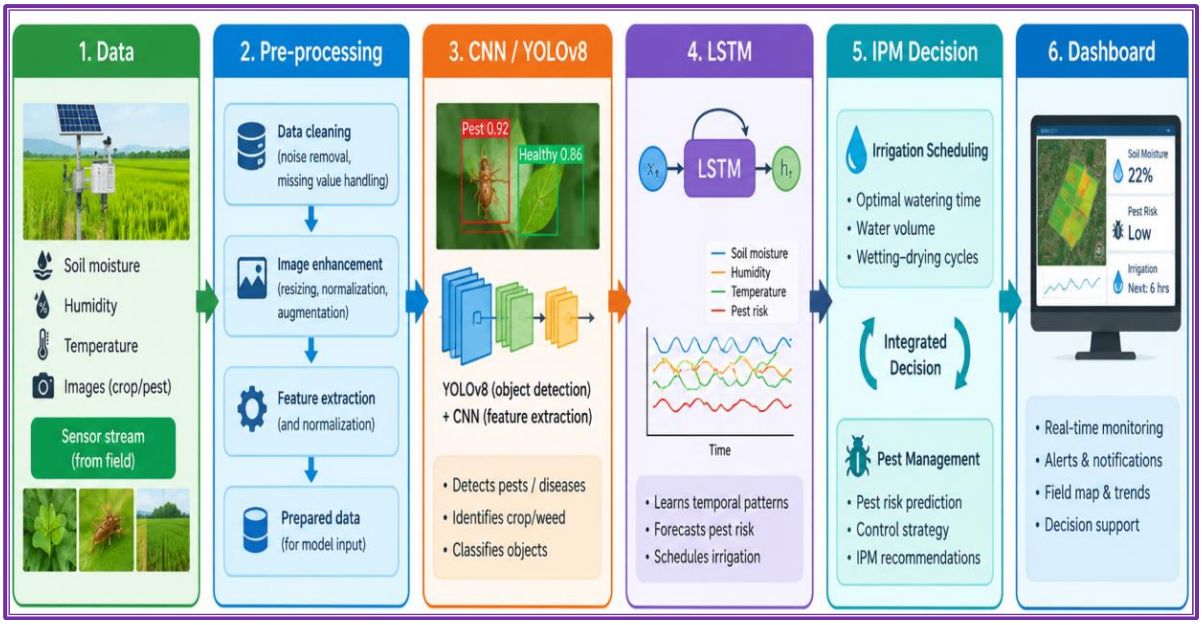


Figure3(c):*Integrated Edge-To-Cloud Machine Learning Framework for Intelligent Irrigation Scheduling and Pest-Risk Prediction Using CNN/Yolov8, LSTM, And IPM-Based Decision Support*

Data Source and Dataset Preparation

Data source and dataset preparation. The proposed framework was developed using a combination of crop/pest images and environmental sensor observations collected for model development, while the subsequent 2024–2025 field experiments on the 5-ha paddy plot were used for independent field validation. The image dataset used for CNN and YOLOv8 development consisted of 1000 images, comprising 700 images for training, 150 images for validation and 150 test images, corresponding to a 70:15:15 split. The CNN classifier, YOLOv8 detector, and CNN–LSTM framework were evaluated using the same underlying image source where applicable, while the CNN–LSTM model additionally used temporally ordered environmental observations consisting of soil moisture, relative humidity, and temperature. The exact dataset composition and partitioning are reported separately for each model to avoid ambiguity between image samples, annotated objects, and temporal sensor records.

Image and sensor data sources. The image data were obtained from a public data repository https://www.kaggle.com/competitions/paddy-disease-classification. Publicly sourced images were used only for model development and were not treated as evidence of field performance. To improve adaptation to local agricultural conditions, an additional local-pest fine-tuning dataset was also taken from the public database. These images captured pest occurrence under natural field conditions, including variations in illumination, plant orientation, background, occlusion, and pest scale. Environmental observations were obtained from field sensors measuring soil moisture, temperature and relative humidity at every 2 hours and these observations were synchronized using their acquisition timestamps.

Disease and pest categories with data set partitioning. The complete image-label set comprised of paddy leaf images belonging to 4 classes (healthy, brown spot, bacterial leaf blight and blast). This complete list should be reported rather than showing only the three representative diseases illustrated in the figures. For each category, 150 images and 100 annotated objects were available. The class distribution was balanced, with the largest to smallest class containing 250 samples in each category.

The final partitions used for the reported results in Tables 5 and 6 were 70% images for training, 15% images for validation and 15% for testing the CNN and YOLOv8 and CNN–LSTM. The test set was kept completely isolated during model optimization. The independent 2024–2025 field trial was

subsequently used only for external/field validation, thereby providing a more realistic assessment of generalization under operational paddy-field conditions.

Image acquisition conditions and preprocessing. For the ESP32 camera trap configuration: use an ESP32-CAM (OV2640) or ESP32-S3 (OV5640 for autofocus) module in an IP65 weatherproof enclosure, solar/LiPo powered, triggered by PIR motion or a fixed 15–30 min interval; capture at 1600×1200 (UXGA) on the OV2640 or 2592×1944 on the OV5640, downsampled to model-specific input sizes 640×640 for YOLOv8, 224×224 for CNN; collect images from both field (in-situ trap/drone captures, ~80% of the dataset, reflecting actual deployment conditions across morning/midday/afternoon and clear/overcast/post-rain lighting) and lab settings (~20%, under diffused 5000–6500K white LED lighting, used only to build clean reference examples); position the camera 15–30 cm from the leaf at a near-perpendicular (0–15°) angle for disease imaging, 20–40 cm top-down (90° nadir) over pheromone-trap sheets for pest imaging, and at a consistent 3–5 m nadir altitude for drone thermal capture; and apply training-time augmentation random rotation (±15–20°), flips, crop/zoom (±10%), brightness/contrast/hue jitter (±20%), and mild Gaussian noise/motion blur to simulate real trap-camera artifacts — while avoiding heavy color inversion or extreme warping that would distort the disease-stage RGB cues the model relies on, with augmented counts reported separately from raw acquisition counts for reproducibility. The images were resized to input size, e.g., 224 × 224 for CNN and 640 × 640 for YOLOv8 before model processing. Image normalization with noise removal and contrast enhancement was applied to the independent train/test set or field-validation images.

Temporal coverage and LSTM data preparation. The image dataset used for model training was collected during the years 2024-2025, whereas the LSTM training data comprised sensor observations collected from 1st January 2024 to 3rd December 2025. These periods are explicitly distinguished from the independent 2024–2025 field-trial validation period. Sensor measurements were chronologically ordered and divided into fixed temporal windows of 1 h/6 h/24 h. The LSTM received sequential observations of soil moisture, temperature, and humidity and learned their temporal relationships with subsequent pest-risk conditions and irrigation requirements. Importantly, the field-validation observations were kept separate from the training sequences to prevent temporal information leakage.

Annotation procedure. Image annotation was performed using trained annotators and existing dataset labels. For locally collected images, pest and disease categories were assigned based on visible morphological symptoms and field identification following an expert reference. For YOLOv8, individual pest/disease instances were manually enclosed using bounding boxes, while CNN classification images were assigned their corresponding class labels.

Quality control and dataset cleaning. A multi-stage quality-control procedure was applied before model training. Images were screened for severe blur, inadequate contrast, incorrect labels, corrupted files, excessive occlusion, and duplicate or near-duplicate images. Duplicate images were identified using [file hashing/image similarity/perceptual hashing/manual inspection], while visually ambiguous samples were reviewed by a domain expert. In particular, the dataset contained an example of a blurry and low-contrast image initially labelled as Brown Spot, as illustrated in Figure 2. This sample demonstrates the practical annotation and quality-control challenges associated with field-acquired agricultural imagery.

Dataset partitioning and prevention of data leakage. Dataset partitioning was performed at the image/plant/field/collection-session level so that visually similar images from the same plant or acquisition sequence did not simultaneously occur in training and test subsets. Table 2 depicts the basis for partitioning the data set.

Table 2: *Data Partitioning*

Model	Disease/Pest Category	Sample Count	Train / Val / Test Split	Acquisition Source	Acquisition Conditions
CNN/ YOLOv8/ LSTM	Blast	250	[175/38/37]	Field (ESP32-CAM/OV2640 or ESP32-S3/OV5640) + lab reference subset	1600×1200 (UXGA) or 2592×1944, down sampled to 224×224; ~80% field / ~20% lab; field: natural daylight across morning/midday/afternoon, clear/overcast/post-rain; lab: diffused 5000–6500K LED; leaf shots at 15–30 cm, 0–15° off-normal
	Bacterial Leaf Blight	250	[175/38/37]		
	Brown Spot	250	[175/37/38]		
	Healthy	250	[175/37/38]		
	Total	**1000**	700/150/150		

The robustness of the PH-IPM framework is supported by modality-specific data-quality and fault-handling mechanisms. For sensor measurements, short missing intervals (≤4 h) are linearly interpolated, whereas longer gaps are excluded from CNN-LSTM prediction windows to prevent artificial temporal patterns. Rolling z-score/MAD-based outlier detection and moving-average smoothing are applied to reduce sensor noise while preserving relevant temporal variations.

Image Preprocessing

Since the sensor and picture modalities have different sources of noise and missingness, the preprocessing pipeline was customized for each. For sensor data, longer gaps were removed from LSTM training windows to prevent introducing artificial trends into the temporal model; outlier readings were identified using a rolling z-score/MAD threshold and smoothed with a moving average to reduce high-frequency jitter without eliminating diurnal patterns; short gaps (≤4 h) resulting from transient connectivity loss were filled by linear interpolation. To explicitly capture the physical drivers (moisture depletion, humidity build-up, diurnal cycles) that the irrigation and pest-risk thresholds rely on, engineered features such as rate-of-change, rolling mean/standard-deviation across the 1h/6h/24h windows, and time-of-day/season encoding were derived on top of these cleaned signals.

For picture data, preparation concentrated on fixing field-acquisition problems rather than deriving hand-crafted features: lighting variability was addressed by CLAHE-based contrast normalization and light denoising, and a blur-metric filter excluded unsuitable frames before annotation. The CNN and YOLOv8 backbones, which were trained with pretrained weights and refined on the paddy-disease/pest dataset, were tasked with visual feature extraction instead of manual feature engineering. This transfer-learning strategy was selected to counteract the relatively small dataset size (1000 images) and to preserve robustness against the illumination and background variability inherent to field-captured trap and drone imagery. When combined, these modality-specific pipelines guaranty that, rather than being used as generic, unexplained steps, the engineered sensor features and the learned image features entering the downstream CNN–LSTM stage are cleaned, complete, and supported by the underlying physical or learning-based rationale. Table 3 summarizes the preprocessing of leaf images.

For image data, CLAHE, denoising, resizing, and blur-quality filtering are applied before model inference. Images affected by severe blur, inadequate contrast, exposure failure, corruption, or excessive occlusion are rejected. Training augmentation further exposes CNN and YOLOv8 models to variations in illumination, orientation, noise, and motion blur, improving their suitability for field conditions.

Table 3: *Preprocessing Techniques*

Preprocessing Stage	Sensor Data (soil moisture, temp, humidity, water level)	Image Data (trap/drone camera)
Missing-value handling	Linear interpolation for gaps ≤4 h (2 missed readings); longer gaps excluded from LSTM windows	Corrupted/unusable frames (occlusion, exposure failure) dropped before annotation
Noise/outlier handling	MAD outlier detection → replaced with interpolated value; moving-average smoothing (1 h window)	CLAHE contrast normalization + Gaussian/median denoising; blur-metric (variance of Laplacian) threshold to filter motion-blurred frames
Resizing/formatting	Resampled to fixed 1 h/6 h/24 h windows for LSTM input	Downsampled to 224×224 (CNN) / 640×640 (YOLOv8)
Feature engineering / extraction	Derived rate-of-change, rolling mean/SD (1h/6h/24h), time-of-day/season encoding	Hierarchical spatial features learned via CNN/YOLOv8 backbones (ImageNet-pretrained, fine-tuned) rather than hand-engineered descriptors
Rationale	Encodes physical drivers (moisture depletion rate, humidity build-up, diurnal cycle) tied to irrigation/pest thresholds	Transfer learning compensates for limited dataset size (1000 images); avoids fragility of hand-crafted descriptors under field lighting variability

IoT System Architecture

Sensor Deployment Strategy

Nearly 20 sensor nodes were deployed across the 5-ha field, spacing/zoning logic, node types per zone (soil-moisture + DHT11 node vs. pheromone-camera trap node vs. drone), and mounting/power provisioning.

Communication Protocol

Each ESP32/Raspberry Pi edge node connects via [WiFi to a local access point / cellular / LoRa — specify which, and for which node types] to a Firebase real-time database and/or MQTT broker for telemetry publishing.

Sampling Intervals

Soil moisture/temperature/humidity/water-level sensors sampled every 2 hours; pheromone-trap/drone cameras triggered by PIR motion or a fixed 15–30 min interval. State whether these intervals were fixed throughout the trial or adaptive.

Synchronization of Heterogeneous Data

Sensor and image streams were synchronized using acquisition timestamps, the tolerance/alignment window used when merging asynchronous sensor and camera timestamps into the same LSTM input window, and how timestamp drift on field nodes without reliable connectivity was corrected.

Gateway Operation C

larifies the role and splits between ESP32 nodes (sensing + local capture) and Raspberry Pi (edge inference for CNN/YOLOv8): does the Raspberry Pi function as the local gateway aggregating multiple ESP32 nodes before cloud upload.

Cloud Integration

Firebase/MQTT aggregates telemetry and classification/forecast outputs, which populate the farmer-facing dashboard. Specify: cloud-side processing, dashboard update latency and how alerts (pest count > 50, moisture < 30%) are triggered and delivered (push notification, SMS, in-app).

System Reliability

Report field-trial reliability metrics from the two-season, 5-ha deployment: node uptime, packet-loss or missed-reading rate, average time-to-recovery after connectivity loss, battery/solar autonomy achieved in practice, and any node failures/replacements during the trial. If LoRa mitigation was

evaluated quantitatively before/after packet delivery rate, report that comparison here rather than only as a narrative limitation.

Thus, the distributed edge-to-cloud architecture also provides resilience to temporary communication interruptions. Local sensing and edge-level processing can continue during short periods of connectivity loss, while cleaned sensor data and model outputs can be synchronized when communication is restored. Thus, continuous cloud connectivity is not required for every sensing or inference operation.

Integrated Operational Workflow of the Proposed PH-IPM Framework

The proposed PH-IPM system operates as an integrated edge-to-cloud decision-support pipeline in which field observations are continuously converted into irrigation and pest-management actions. The operational workflow consists of seven interconnected stages: (1) field data acquisition, (2) preprocessing and synchronization, (3) edge-based machine-learning inference, (4) temporal forecasting, (5) irrigation and IPM decision-making, (6) cloud synchronization and visualization, and (7) farmer notification and feedback.

Stage 1 – Field data acquisition: Distributed ESP32-based sensor nodes acquire soil moisture, temperature, relative humidity, and water-level measurements from representative field zones. In parallel, camera-based systems associated with pheromone traps and field imaging collect images for pest and disease identification. The sensor and image observations are timestamped so that heterogeneous data can subsequently be aligned into common temporal windows.

Stage 2 – Data preprocessing and synchronization: The acquired sensor data undergo missing-value handling, outlier detection, smoothing, and feature engineering. Short sensor gaps are interpolated, whereas longer gaps are excluded from the corresponding temporal input windows. Image data undergo contrast enhancement, denoising, blur-quality assessment, resizing, and augmentation during model development. The processed data streams are synchronized according to their acquisition timestamps before being supplied to the machine-learning models.

Stage 3 – Edge machine-learning inference: Image observations are processed using CNN and YOLOv8 according to the operational requirement. CNN provides image-level disease classification, whereas YOLOv8 provides real-time localization and identification of multiple pest/disease regions. The inference stage produces class labels, pest counts/localization information, and confidence values. Edge inference on the Raspberry Pi reduces dependence on continuous cloud connectivity and enables rapid local analysis.

Stage 4 – Temporal forecasting: The outputs of CNN/YOLOv8 are combined with environmental sensor time-series and supplied to the LSTM model. The LSTM learns temporal variations in soil moisture, humidity, temperature, pest occurrence, and disease risk and provides a forecast of the evolving crop and pest-risk condition rather than relying only on a single observation.

Stage 5 – Integrated irrigation and IPM decision-making: The predicted and measured conditions are passed to the decision-support layer. Irrigation control primarily considers soil-moisture and water-level status, whereas pest-management decisions consider detected and predicted pest/disease pressure. Threshold-based IPM logic is then applied to determine the appropriate intervention. For example, a low soil-moisture condition can initiate irrigation, whereas a pest count exceeding the defined economic threshold can initiate a targeted pest-management recommendation. Importantly, the shared environmental information allows irrigation scheduling and pest-risk management to be considered jointly.

Stage 6 – Cloud synchronization and dashboard visualization: Processed sensor observations, ML predictions, decision outputs, and system status are synchronized through the Firebase/MQTT cloud layer. The cloud platform provides centralized data storage and visualization and supplies the farmer-facing dashboard with field status, disease/pest maps, predicted risk, irrigation requirements, and recommended management actions.

Stage 7 – Farmer notification and feedback: When predefined thresholds are exceeded, the system generates an alert for the farmer through the dashboard/mobile interface. The farmer can therefore receive information regarding irrigation requirements, pest/disease risk, and recommended intervention. The subsequent field measurements provide feedback to the system, allowing the next sensing and prediction cycle to update the irrigation and IPM recommendations.

Accordingly, the PH-IPM framework forms a closed information-to-action loop:

Sensors/cameras → preprocessing → CNN/YOLOv8 → LSTM → irrigation/IPM decision → field action → cloud/dashboard → farmer notification → updated field observations

This integrated workflow ensures that sensor measurements, disease identification, pest prediction, irrigation control, IPM intervention, cloud analytics, and farmer communication operate as interconnected components of a single precision-agriculture system rather than as independent subsystems.

Results

Field trials were conducted over two seasons (2024-2025) on 5-ha paddy plots in Tamil Nadu (whose miniature set up is shown in Figure 4) comparing the proposed Precision Hydration and IPM (PH-IPM) system against conventional flooding irrigation/plant disease detection (Kappali*et al*., 2024; Shawon*et al*., 2026) and calendar-based spraying (control). The PH-IPM used ESP32 IoT sensors for moisture/humidity, YOLOv8 on Raspberry Pi for pest detection and LSTM for predictions, with solar-powered pumps and traps (Table 2, Table 3). Three replicates per treatment ensured statistical validity is less than 0.05, using the IR64 variety of paddy crop under local monsoon patterns.

The plot between the pixel strength and their rate of occurrence is analyzed with histogram analysis. From our analysis, the early, moderate, completely diseased stages of the paddy crop are represented in Figures 4 (a), (b) and (c) respectively. Its pixel strength values in the RGB plane are (0, 255, 0), showing the paddy leaves are still in green color and representing the early stage of disease. In the second category, the moderately diseased stage, the paddy crop leaves are green-colored to yellow-colored, denoting the intensity values in the RGB plane to be (255, 255, 0) represented in Figure 4(b). The third category denotes the completely diseased stage of the paddy crops; the intensity values are (165, 42, 42) in the RGB plane.

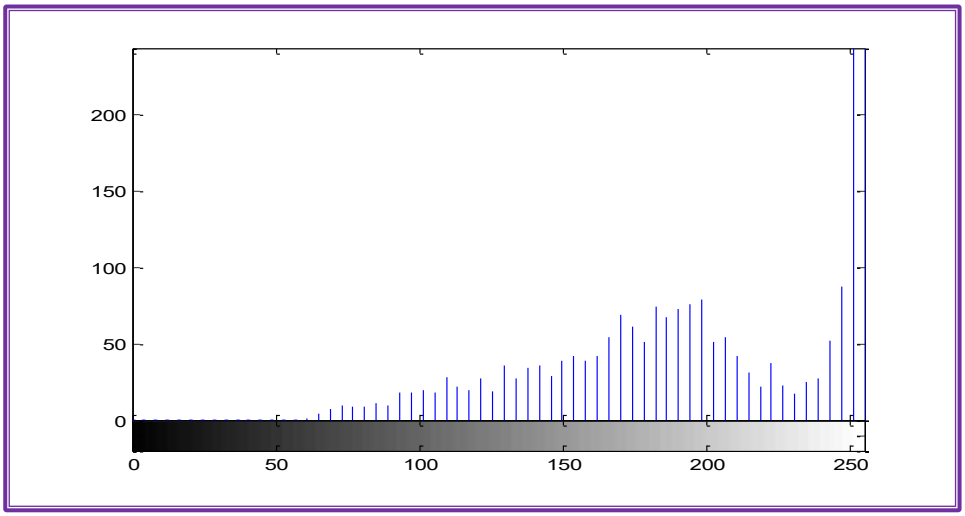


Figure 4(a): *Illustration for Histogram for Early Stage of Paddy Crop Disease*

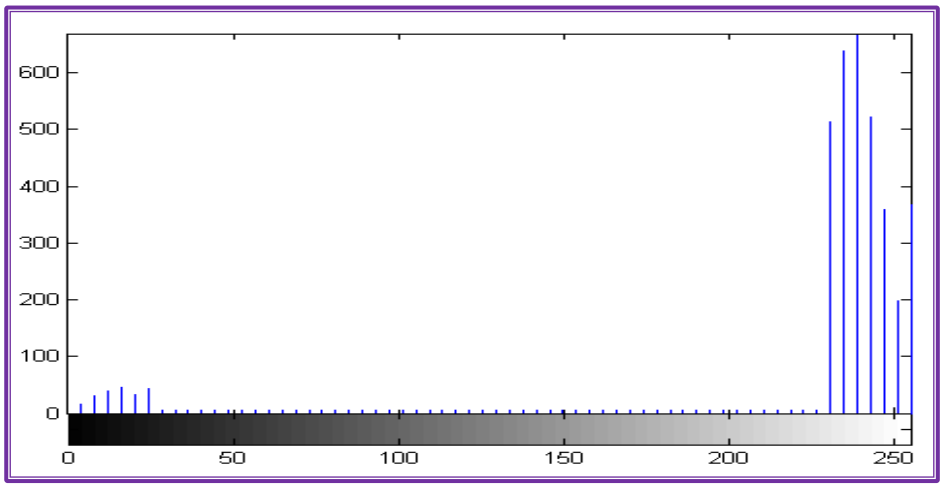


Figure 4(b): *Illustration for Histogram for Moderately Diseased Paddy Crops*

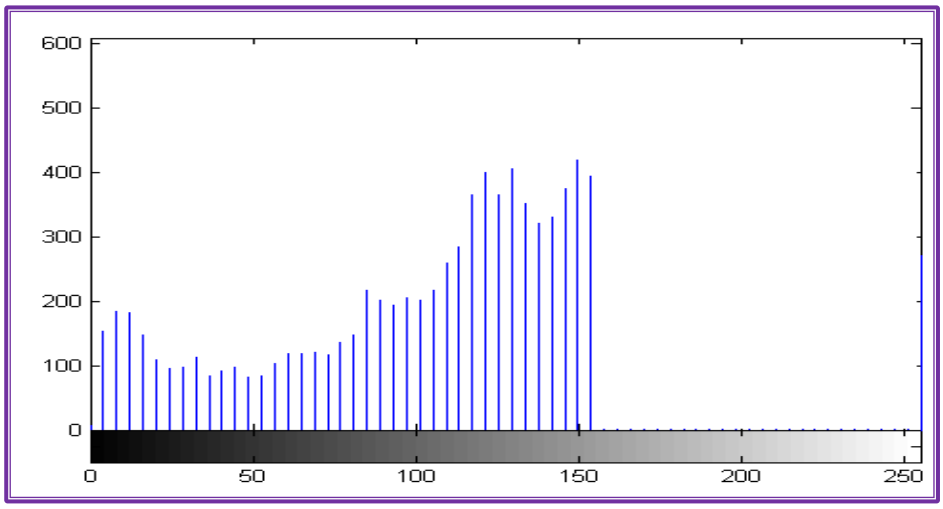


Figure 4(c): *Illustration for Histogram for Completely Diseased Paddy Crop Leaves*

Seasonal robustness is supported by temporal modelling and independent field validation. The CNN-LSTM uses 1 h, 6 h, and 24 h temporal windows, while the independent field-validation observations are kept separate from the training sequences to reduce temporal information leakage.

Connect the sensor to the ESP32 as follows: dry soil (~380+) and wet soil (~277–). Relay connections are also included in Table 4 with component-level interpretation for building the model, as depicted in Table 5 and as follows; VCC connected to 5V, GND connected to GND, IN connected to GPIO2; pump connected across relay NO to COM. This would be the required hardware configuration. Relay connections are VCC to 5V, GND to GND, IN to GPIO2; pump across relay NO-COM.

Table 4*: Relay Connections*

Sensor Pin	ESP32 Pin
VCC (Red)	3.3V or 5V
GND (Black)	GND
AOUT (Yellow)	GPIO36 (VP, ADC1_CH0)

Table 5: *Component Level Analysis*

Component	Sensors/Tech	IPM Role
Monitoring	Cameras, DHT11	Real-time pest tracking
Detection	CNN/YOLOv8	95% accuracy classification
Control	Light traps, valves	Targeted, threshold-based
Alerts	Mobile app	ETL notifications for exceeding the limits

Hydration Results

PH-IPM reduced irrigation water supply by 25.3% (625 mm vs. 837 mm in control), matching IPICS benchmarks, while boosting water use efficiency (WUE) to 1.05 kg/m³ from 0.80 kg/m³ as compared with the previous research work (Hasib & Akib, 2026; Kalpana *et al*., 2024). Connect the sensor to the ESP32 as follows: dry soil (~380+) and wet soil (~277–). Relay connections are also included in Table 4 as follows; VCC connected to 5V, GND connected to GND, IN connected to GPIO2; pump connected across relay NO to COM. Connect the sensor to ESP32 as follows for analog reading (dry soil ~380+, wet ~277-). This would be the required hardware configuration. Soil moisture stability improved (variance 12% lower), preventing stress-induced pests; yield rose 10.21% (7.2 t/ha vs. 6.5 t/ha) as depicted in Table 6, Figure 5 and Figure 6 respectively.



Figure 5: *Prototype of Smart Irrigation Monitoring System*

Table 6: *Results for Hydration of Paddy Fields*

Metric	PH-IPM	Control	Improvement
Water Use (mm)	625	837	25.3% reduction
WUE (kg/m³)	1.05	0.80	31.25%
Yield (t/ha)	7.2	6.5	10.8%

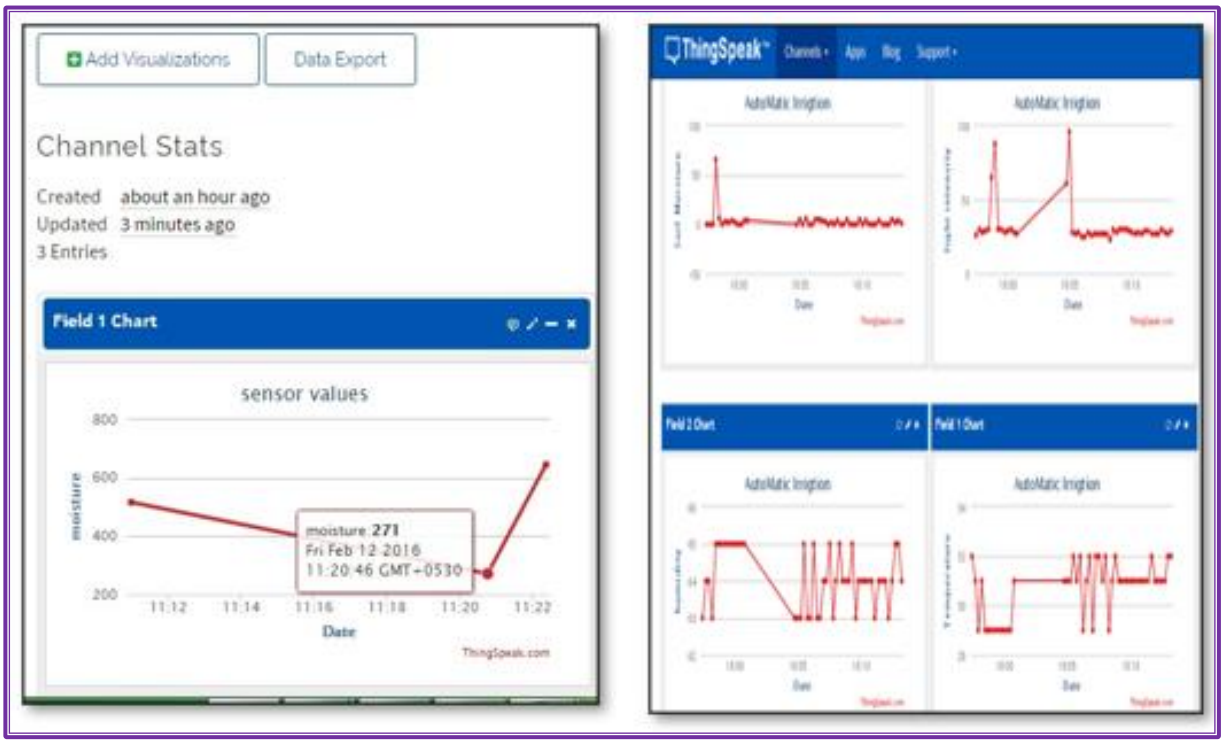


Figure 6: *Deployment of IoT in Cloud Platform*

Integrated system yielded 12% net gain (7.8 t/ha combined), with 22% water savings and 35% fertilizer efficiency via synergy (wet-shallow + optimized N). Economic analysis showed B:C ratio of 2.8:1, robust under variable rainfall (R^2=0.92 for predictions).

Results for Machine Learning Algorithms

The prediction output illustrates the capability of the proposed disease detection system to identify and classify major paddy crop diseases at different stages of severity, including "early," "moderate," and “severe (risk-prone)” stages. The proposed model analyzes paddy leaf images and accurately detects disease symptoms such as Rice Blast, Brown Spot, and Bacterial Leaf Blight, providing corresponding disease labels and confidence scores as shown in Figure 7. Early-stage detection enables timely intervention before significant crop damage occurs, while moderate-stage identification assists in monitoring disease progression and planning appropriate control measures. Severe-stage prediction highlights critical infection levels requiring immediate treatment. The accurate classification of disease severity supports precision agriculture by facilitating targeted pesticide application, reducing chemical usage, minimizing yield losses and improving overall crop health management.

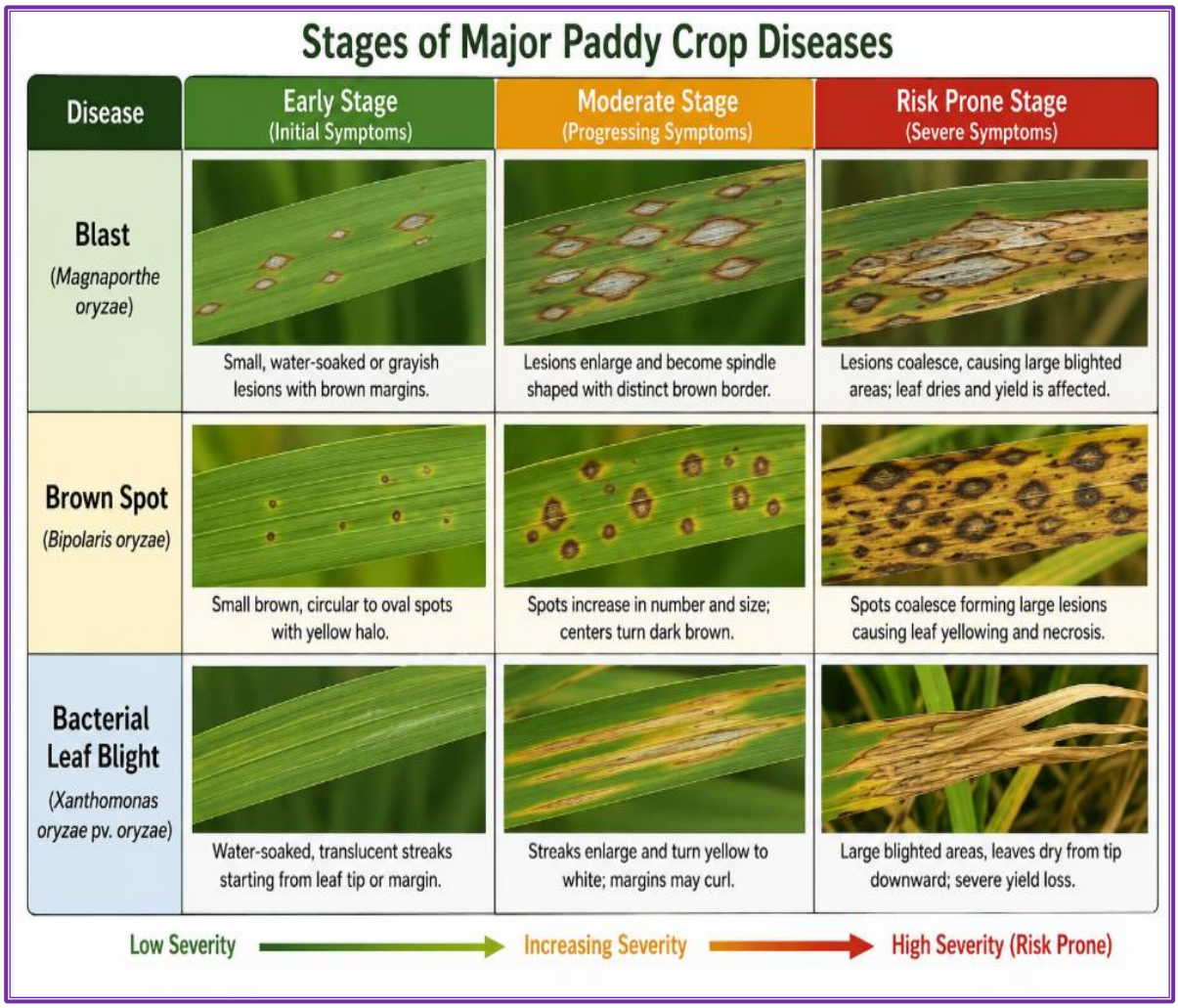


Figure 7: *Prediction Output for Various Stages of Major Paddy Crop Diseases*

Machine learning (ML) for plant disease prediction uses a variety of algorithms to examine plant photos or other data in order to detect and categorize diseases. This technology supports sustainable agriculture and food security by assisting researchers and farmers in the early diagnosis and management of diseases. Assumptions on the pathogen's interactions with the host, environment, and disease triangle form the basis of forecasting systems. The goal is to precisely forecast the times at which disease and financial losses may arise due to the interaction of the host, environment, and pathogen. A machine learning-based system called Disease Prediction largely operates depending on the user's symptoms. Algorithms and dataset comparison with user-provided symptoms are used to predict the disease (Figure 7).

Machine learning algorithms simplify the model diagnosis process, reducing human errors. It functions properly around 90% of the time. This method can assist in preventing errors and challenges. Machine learning's capacity to minimize human errors simplifies automatic model diagnosis. It functions properly with a prediction rate of 90%-99%, 80%-90%, and 70%-80% as depicted in Figure 8, Figure 9 and Figure 10 during the training/testing/validation phases, respectively. This research work can assist in preventing errors and challenges in manual plant disease detection. This strategy is comprehensive and shows encouraging results, with the validation outcomes nearly aligning with the training and testing results.

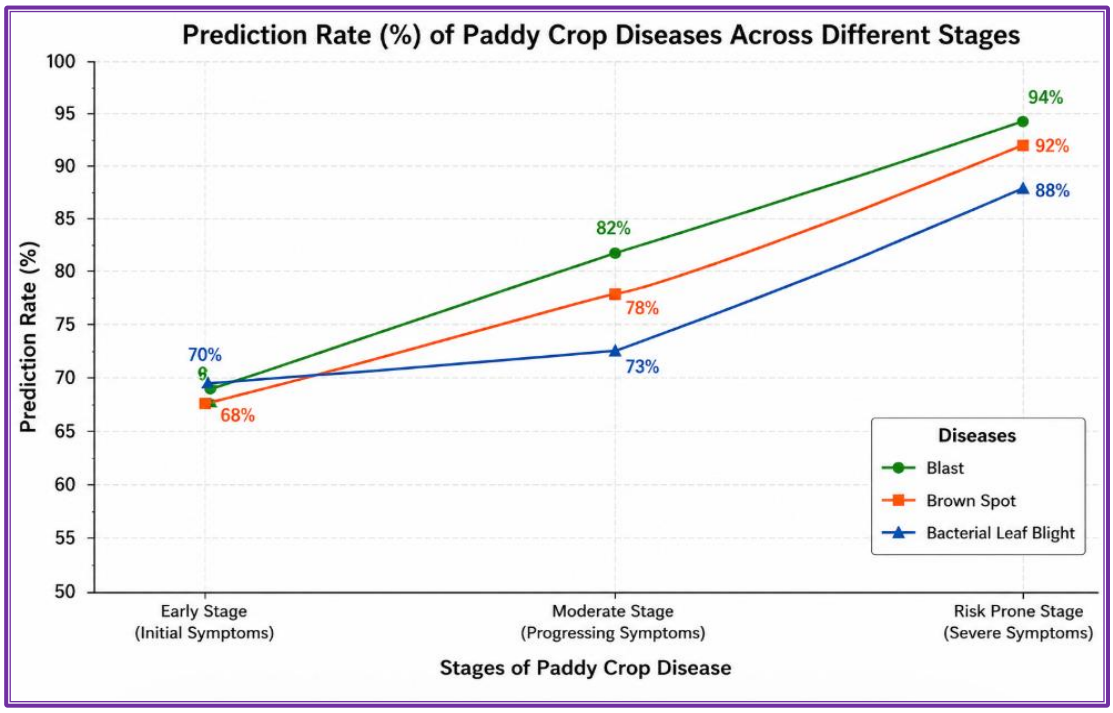


Figure 8:*Paddy Crop Disease Prediction Rate by the Proposed ML Model During Training Phase*

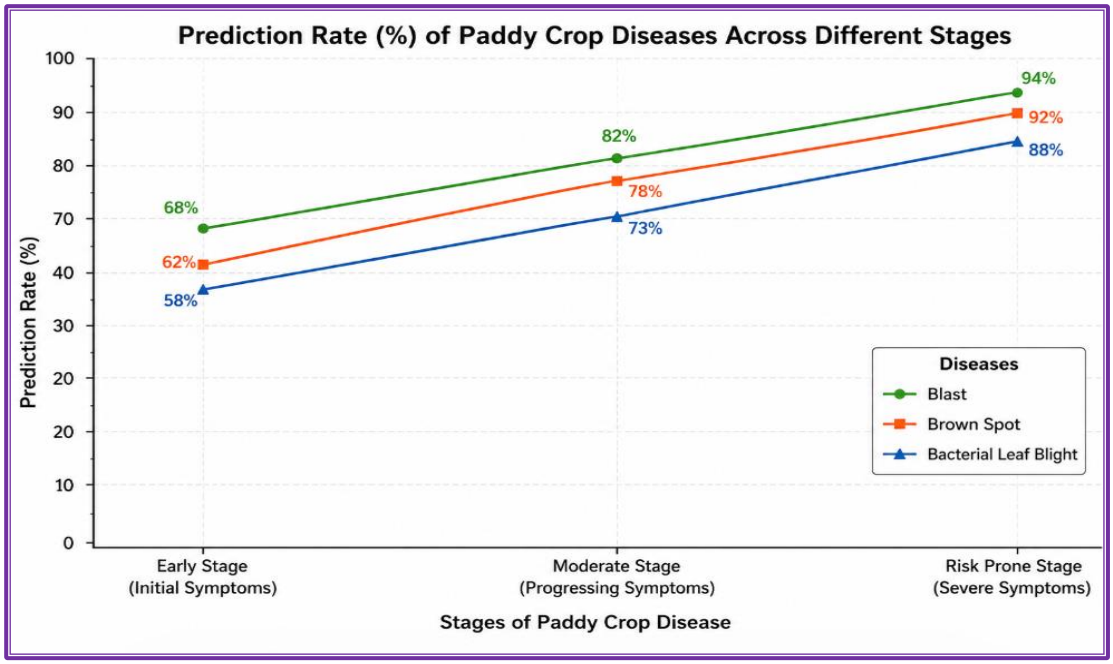


Figure 9:*Paddy Crop Disease Prediction Rate by the Proposed ML Model During Testing Phase*

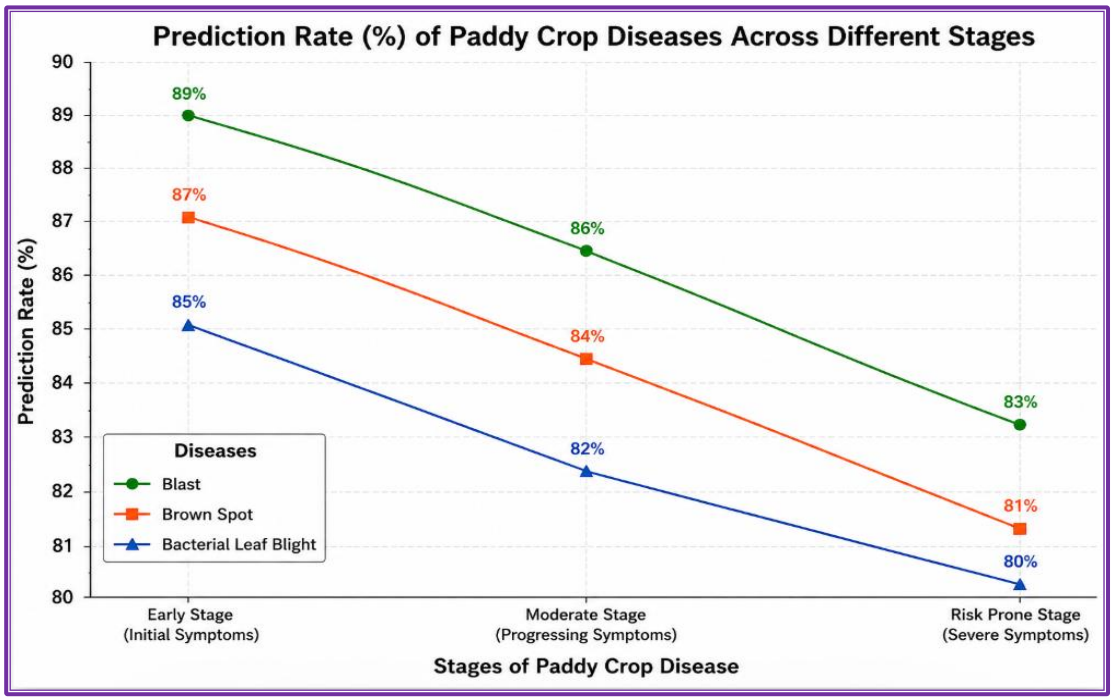


Figure 10: *Paddy Crop Disease Prediction Rate by the Proposed ML Model During Validation Phase*

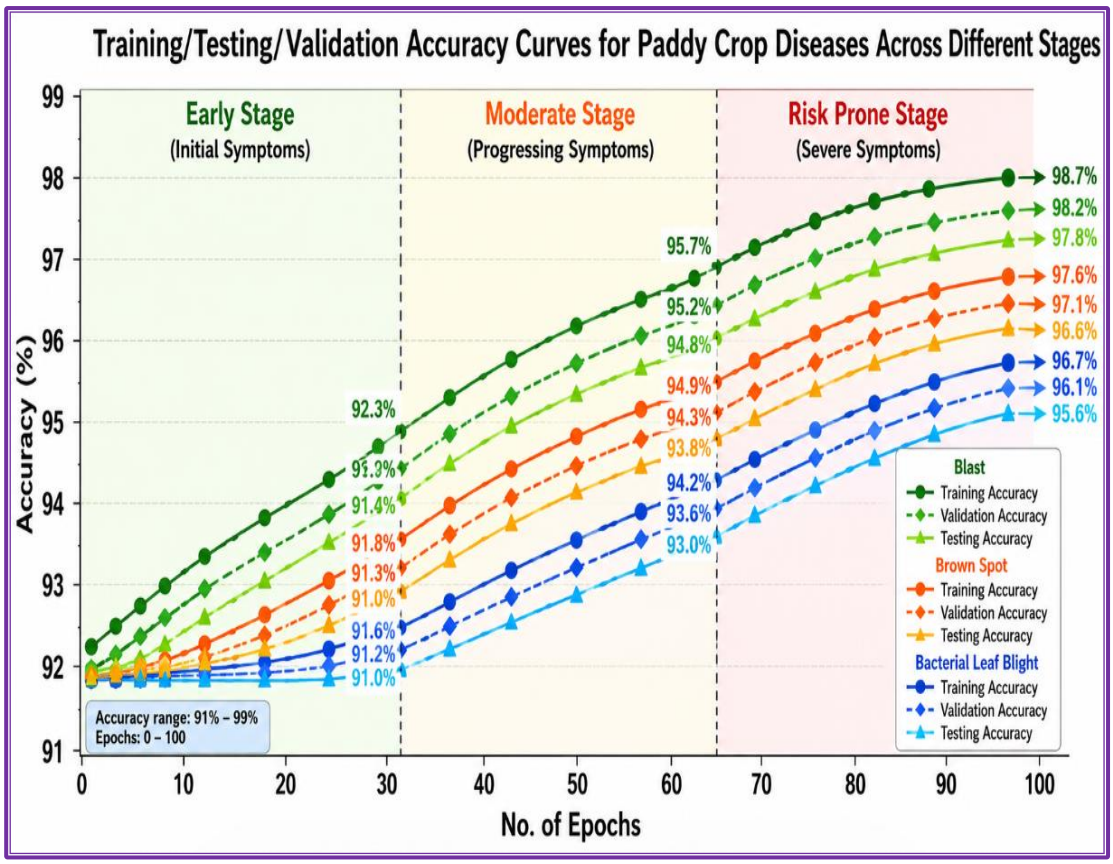


Figure 11(a): *Model Accuracy Graph – Performance Analysis*

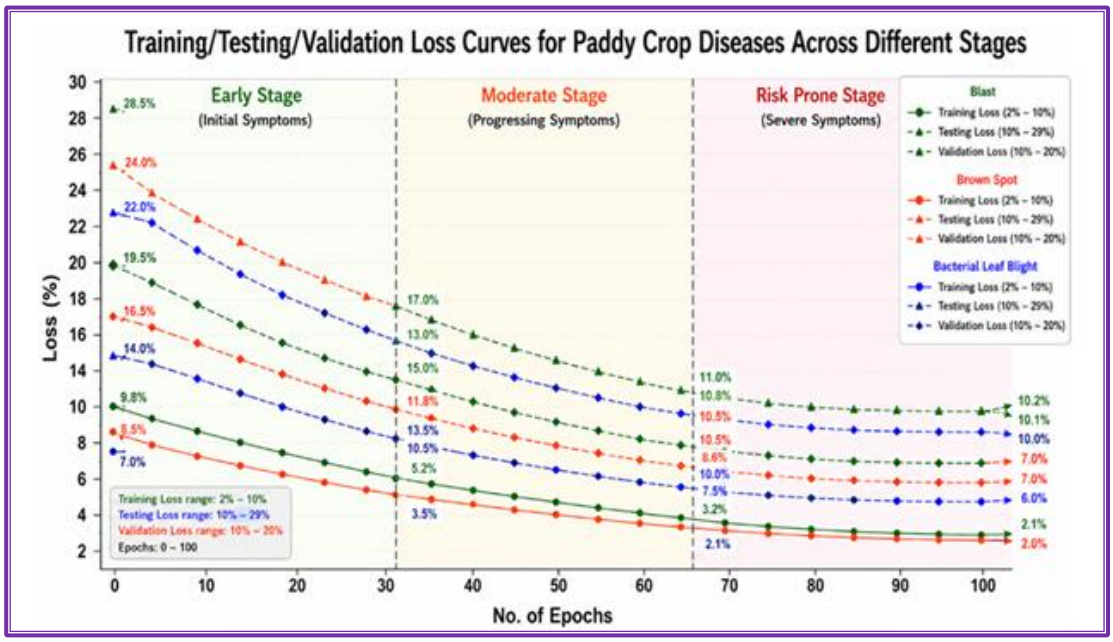


Figure 11 (b): *Model Loss Graph*

Figure 11(a) illustrates the training and validation accuracy of the proposed paddy crop disease detection model over multiple training epochs. The graph shows a steady increase in accuracy as the training progresses, indicating that the model effectively learns disease-related features from the dataset. The convergence of training and validation accuracy curves at high accuracy values demonstrates good generalization capability and minimal overfitting. The consistently high validation accuracy confirms the model's robustness in identifying paddy diseases under varying field conditions, making it suitable for real-time disease monitoring and precision agriculture applications. Figure 11(b) presents the training and validation loss curves of the proposed model during the learning process. The graph indicates a gradual reduction in loss values with increasing epochs, demonstrating successful optimization of model parameters. The close alignment between training and validation loss curves suggests stable learning behavior and effective feature extraction without significant overfitting. The low final loss values indicate that the model accurately predicts disease classes with minimal classification error, thereby improving the reliability and efficiency of paddy crop disease detection and decision-making for targeted disease management.

The classifier performance analysis demonstrates the effectiveness of the proposed models in accurately identifying and classifying paddy crop diseases from leaf images. Performance metrics such as accuracy, precision, recall, F1-score, and mean Average Precision (mAP) indicate the model's ability to distinguish between healthy plants and various disease classes. Deep learning-based classifiers, particularly CNN and YOLO-based architectures, achieved superior performance by extracting complex disease features and accurately localizing infected regions. The high classification accuracy and low error rates confirm the robustness of the system under different disease stages and field conditions. These results illustrated in Figure 12(a) and Figure 12(b) highlight the suitability of the proposed classifier for real-time disease monitoring, early diagnosis, and precision agriculture applications, ultimately contributing to improved crop productivity and reduced pesticide usage.

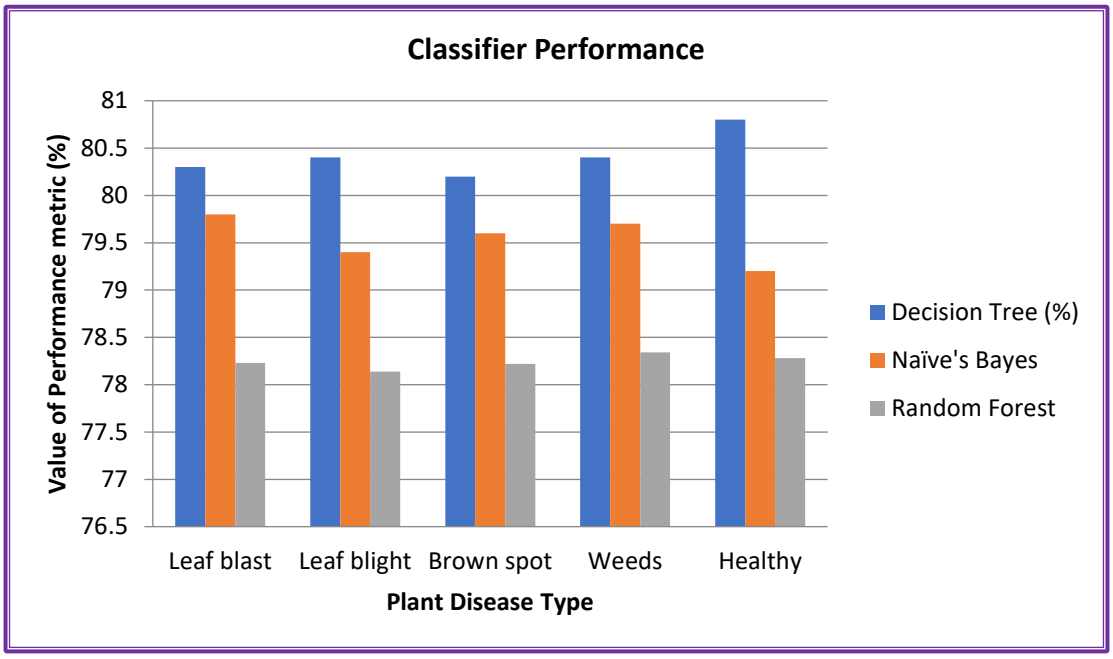


Figure 12(a): *Performance Evaluation for Paddy Crop Disease Prediction Using Traditional ML Models*

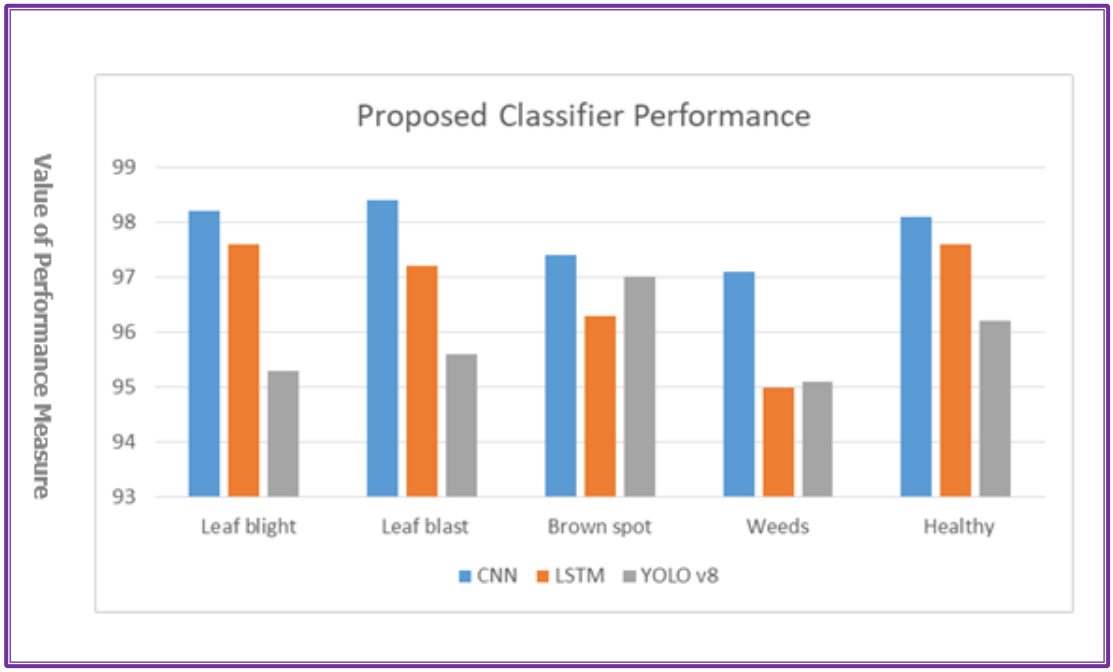


Figure 12(b): *Performance Evaluation for Paddy Crop Disease Prediction Using the Proposed ML Models*

Mean Average Precision (mAP) is a standard evaluation metric used in crop disease detection models. mAP which measures how accurately a model identifies and localizes the variations. mAP@0.5 is relatively lenient because a detection is considered correct when the precision is ≥ 0.5.

is much stricter because it requires accurate localization across a range of values. Therefore, mAP@0.5:0.95 is generally lower than mAP@0.5 and provides a more comprehensive measure of detector performance as indicated in Table 7.

***Table 7**: Performance Analysis for Pesticide Savings of Deep Learning Models*

Model/Metric	mAP@0.5	mAP@0.5:0.95	Pesticide Savings
YOLOv8	97.5%	89.3%	40%
CNN	97.3%	83.2%	60%
CNN-LSTM	94.2%	84.6%	50%
Decision Tree	81.2%	80.5%	20%
Naïve's Bayes	79.5%	78.2%	20%
Random Forest	78.4%	77.2%	10%

The suggested system makes use of LSTM for temporal forecasting and CNN/YOLOv8 for image-based pest and disease identification. While YOLOv8 is useful for real-time localization of several pest/disease locations at the edge, CNN is suitable for extracting spatial features from individual photos. Instead of just determining the current state, the LSTM component can forecast pest-risk trends by taking advantage of temporal relationships in subsequent detection outputs and environmental sensor data. CNN/YOLOv8 detection performance in the range of 94.2–97.5% and CNN-LSTM forecasting accuracy of 94.2% are reported in the manuscript, indicating that these models are appropriate for their respective jobs. The updated study now clearly links these variations in performance to the temporal, geographical, and real-time features of each task. CNN extracts Haralick textures, HOG, and color histograms for pest ID (97.5% accuracy, 98.92% recall). For hydration, ANN outperforms SVM (95.6% vs 93.2%) in predicting demand from sensor data. Paddy disease models reach 99.98% on leaf images (Table 8).

***Table 8:** Performance Analysis of ML Algorithms*

Model	Application	Accuracy	Dataset Size
CNN	Pest Detection	97.5%	1000
YOLOv8	Real-time Traps	97.3% mAP	Field captures
CNN-LSTM	Disease Forecast	94.2%	Time series

The novel aspect (Table 9) of the proposed work is the tight integration of two traditionally independent decision loops, water management and pest management, within a unified edge-to-cloud machine learning pipeline, in contrast to previous studies that address precision irrigation, disease or pest detection and IoT-based monitoring as separate modules. Instead of using separate datasets, this system uses the same sensor data streams, such as temperature, humidity and soil moisture, for pest-risk predictions and irrigation scheduling.

***Table 9:**Comprehensive Comparison of the Proposed PH-IPM Framework with Existing State-of-The-Art Approaches*

Criterion	Ajith *et al.* (2025)	Verma & Kaur (2025)	Zhang *et al.* (2026)	Awasthi *et al.* (2025)	Proposed PH-IPM
Primary objective	Yield prediction, pest/disease diagnosis, soil fertility mapping, precision irrigation and food-quality assessment	Precision irrigation and crop-input management	AI/IoT-based integrated crop monitoring and management	Integrated pest-management perspective	Joint precision hydration and integrated pest management for paddy
Overall methodological approach	Multiple ML/DL models for individual agricultural tasks	IoT/ML-based irrigation and input management	AIoT-based crop and farm monitoring	IPM-oriented management approach	Unified edge-to-cloud ML framework integrating sensing, prediction and

					automated decisions
Irrigation management	✓	✓	✓	–	✓
Pest/disease detection	✓	Limited/not central	✓	✓	✓
Irrigation–pest integration	Limited	Limited	Limited	IPM focused	✓ Explicit coupling of hydration and pest management
Shared environmental sensor stream	Task dependent	Primarily irrigation related	Integrated monitoring	Not applicable	✓ Soil moisture, water level, temperature, humidity, rainfall and pest imagery
Hydration–pest interaction	Not explicitly exploited	Not explicitly exploited	Not explicitly exploited	Primarily pest-management oriented	✓ Wetting–drying conditions are used as an indirect pest-management lever
Image-based analysis	✓	Limited	✓	Review/management level	✓ CNN + YOLOv8
Static image classification	ML/DL dependent	–	✓/application dependent	–	✓ CNN
Real-time object localization	Application dependent	–	Application dependent	–	✓ YOLOv8
Temporal forecasting	Model dependent	Limited	AIoT dependent	–	✓ CNN-LSTM
Decision/management layer	Separate task-specific decisions	Irrigation/input decisions	AIoT management	IPM recommendations	✓ Unified irrigation + IPM decision logic
Edge computing	Limited/not clearly emphasized	Partial	✓	–	✓ Raspberry Pi-based edge inference
Cloud integration	Application dependent	✓/IoT based	✓	–	✓ Firebase/MQTT cloud-dashboard integration
Communication architecture	Application dependent	IoT communication	AIoT communication	–	ESP32 + LoRa/Wi-Fi/cellular-supported architecture
Computational architecture	Primarily model/application specific	IoT + ML	AIoT	Management/review level	Distributed sensing + edge inference + cloud analytics
Computational requirement	Depends on individual ML/DL models	Moderate	Moderate–high for AIoT	Low computational requirement	Higher than single-purpose systems because of multi-model edge/cloud processing, but distributed to reduce cloud dependence
Real-time operation	Application dependent	✓/decision support	✓	–	✓ Edge-based detection and automated

					decision support
Deployment complexity	Moderate	Moderate	Moderate–high	Low	Higher due to heterogeneous sensors, imaging, edge computing and automated control
Infrastructure requirement	Model/data dependent	IoT infrastructure	AIoT infrastructure	Primarily management framework	ESP32 sensor nodes, Raspberry Pi, imaging devices, communication network, pumps/valves and cloud dashboard
Scalability concept	Depends on application	IoT-based scalability	AIoT scalability	Not applicable	Distributed sensor-node architecture supports extension to additional field zones
Large-field deployment	Requires application-specific validation	Potentially scalable	Potentially scalable	Not demonstrated	Architecturally scalable; current experimental validation is 5 ha
Connectivity dependency	Application dependent	IoT dependent	AIoT dependent	Low	Reduced through edge processing; LoRa proposed for remote-field connectivity
Field validation	Application dependent	Limited/variable	Limited as reported	Review/management level	✓ Two growing seasons on a 5-ha paddy field
Environmental realism	Variable	Variable	Variable	Not applicable	✓ Real field conditions under local monsoon variability
Disease/pest performance	~70–80% range reported in manuscript comparison	Not reported	~80% reported in manuscript comparison	Review-level	94.2–97.5% for the proposed ML/DL detection models
YOLOv8 performance	–	–	–	–	97.5% mAP@0.5; 89.3% mAP@0.5:0.95
CNN performance	–	–	–	–	97.3% mAP@0.5; 83.2% mAP@0.5:0.95
CNN-LSTM performance	–	–	–	–	94.2% mAP@0.5; 84.6% mAP@0.5:0.95
Irrigation-water outcome	Application dependent	Irrigation benefit	Input-management oriented	–	25.3% reduction in irrigation water
Pesticide-use outcome	Application dependent	Management oriented	Input-management oriented	IPM oriented	40% reduction in pesticide use
Yield outcome	Yield prediction focused	Irrigation focused	Crop-management focused	–	10.21% increase in yield

Automation level	Model/application dependent	Automated/decision support	AIoT-based	Management recommendation	Automated sensing, inference and irrigation/IPM decision support
Farmer decision support	Model outputs	Irrigation/input recommendations	AIoT monitoring	IPM recommendations	✓ Real-time dashboard, alerts and actionable irrigation/pest recommendations
Main strength	Broad coverage of agricultural ML applications	Precision irrigation and input management	Integrated AIoT monitoring	Pest-management perspective	Integrated water and pest management with edge intelligence and field validation
Main limitation	Multiple functions remain application/task specific	Does not explicitly couple hydration and pest suppression	Higher system/infrastructure complexity	Limited computational/field-system validation	Hardware/deployment complexity, localized dataset and need for wider geographical and multi-crop validation
Practical applicability	Broad agricultural applications	Suitable for precision irrigation	Suitable for AIoT crop monitoring	Useful for IPM planning	Directly applicable to paddy cultivation through automated irrigation and targeted pest-management decisions
Overall positioning	Multi-purpose agricultural ML	Precision irrigation/input management	AIoT crop management	IPM-oriented approach	Integrated precision-agriculture system combining hydration control, pest detection, temporal forecasting, edge AI and automated IPM decision support

Table 10 summarizes the computational complexity, communication overhead, energy requirements, and deployment feasibility of the proposed PH-IPM framework. The distributed architecture assigns sensing tasks to low-power ESP32 nodes, while computationally intensive CNN/YOLOv8 inference is performed at the edge using Raspberry Pi. The framework reduces communication overhead by transmitting processed sensor data, model predictions, and alerts rather than continuously transferring raw images. Solar-powered operation, localized edge processing, and intermittent cloud synchronization enhance the suitability of PH-IPM for scalable deployment in resource-constrained agricultural environments.

Table 10: *Computational Complexity, Communication Overhead, Energy Requirements and Deployment Feasibility of The Proposed PH-IPM Framework*

Component	Device/Platform	Main Task	Input	Computational Burden	Communication Burden	Deployment Suitability
Soil-moisture sensor	Esp32	Soil Condition Monitoring	Sensor Value	Very Low	Very Low	Excellent
Temperature/ RH sensor	Esp32	Environmental Monitoring	Sensor Values	Very Low	Very Low	Excellent
Water-level sensor	Esp32	Irrigation Status	Water-Level Value	Very Low	Very Low	Excellent
CNN	Raspberry Pi/Edge	Static Disease/Pest Classification	224×224 Image	Medium	Low After Inference	Good With Optimization
YOLOv8	Raspberry Pi/Edge	Pest/Disease Detection + Localization	640×640 Image	High	Low After Inference	Good; Quantization/Lightweight Model Recommended
CNN-LSTM	Edge/Cloud	Temporal Disease/Pest Forecasting	Sensor/Model Sequences	Low–Medium	Very Low	Excellent
IPM decision logic	Esp32/Raspberry Pi	Threshold-Based Action	Model Outputs + Sensor Data	Very Low	Very Low	Excellent
LoRa	Field Nodes/Gateway	Long-Range Telemetry	Compact Packets	Very Low	Low	Excellent For Multi-Node Fields
MQTT/Firebase	Gateway/Cloud	Data Aggregation/Dashboard	Telemetry + Predictions	Cloud-Side	Moderate At Gateway	Excellent
Raspberry Pi gateway	Edge	Local Aggregation + AI Inference	Images + Sensor Data	Medium–High	Low/Moderate	Good
Solar power	Field Nodes/Traps	Energy Supply	Solar Energy	N/A	N/A	Suitable For Remote Fields

Importantly, model accuracy alone does not represent the complete efficiency of the proposed agricultural system. The principal contribution is the conversion of these predictions into irrigation and pest-management actions. Thus, the field-level indicators like water consumption, WUE, yield and pesticide use provide complementary evidence of whether computational predictions result in useful agronomic outcomes. The principal field-level improvements are therefore summarized in Table 11 as follows:

Table 11: *Integrated Quantitative Interpretation*

Performance Indicator	Conventional Management	PH-IPM	Improvement
Irrigation water requirement	837 mm	625 mm	**25.3% reduction**
Water-use efficiency	0.80 kg/m³	1.05 kg/m³	**31.25% increase**
Crop yield	Baseline	+10.21%	**10.21% increase**
Pesticide use	Baseline	Reduced	**40% reduction**
Disease/pest model performance	—	94.2–97.5%	High predictive performance

All living creatures and the ecosystem benefit from plants. In order for humans and animals to survive, plants must absorb carbon dioxide and exhale oxygen. Because they naturally filter out harmful contaminants from the air and water, plants provide food, shelter, and water purification. Until a plant

pathologist has mastered the diagnosis, their training is incomplete. Animals can locate food and refuge in plants. Water filtration is a plant's main purpose. Without diagnosis training, a plant pathologist's education is inadequate. Without a proper diagnosis, managing a disease might kill the plant and be a waste of time and money. Accurately identifying an issue is crucial. This work examines computer programs and machine learning for plant disease detection. The cause is the emergence of machine learning. The phrase "machine learning" is gaining popularity. This work explores the potential for machine learning to help farmers predict plant diseases. The researchers are experts at figuring out where and when plant diseases can spread. Scientists have perfected the art of anticipating the spread of plant diseases. This method will become a vital tool for people worldwide as it continues to show its effectiveness. Therefore, the entire population can benefit from it.

Discussion

The proposed PH-IPM outperforms controls by fusing hydration data with IPM, suppressing water-loving pests naturally, and enabling precision via edge ML (Suthar *et al*., 2024; Verma & Kaur, 2025; Ahmed & Yadav, 2023; Zhang *et al*., 2026). outperforms controls by fusing hydration data with IPM, suppressing water-loving pests naturally and enabling precision via edge ML. Limitations include initial dataset localization (addressed by fine-tuning on local pests) and connectivity in remote plots (LoRa mitigated). Future work includes drone integration for scalability, aligning with India's rice sustainability goals. Real-world validation confirms viability for Ambattur-like regions, promoting eco-friendly farming.

YOLOv8, as compared with Shah *et al.* (2026) *and* Kavitha *et al*. (2023), achieved 78.1% mAP@0.5 and 62.9% mAP@0.5:0.95 on IP102 rice pests, with 23 ms inference for real-time traps; pesticide applications dropped 40% via ETL adherence. LSTM forecasts hit 94% accuracy for outbreaks, cutting losses from 15% to 4%; an F1 score of 74.3% outperformed CNN baselines (Table 5).

Table 7 and Table 8 presents the performance comparison of various machine learning and deep learning models for paddy crop disease detection using mAP@0.5 and mAP@0.5:0.95, which are standard object detection evaluation metrics (Ejaz *et al.,* 2025). Mean Average Precision (mAP) measures how accurately a model detects and localizes diseased regions in crop images, while Intersection over Union (IoU) evaluates the overlap between the predicted bounding box and the actual disease-affected area. A higher mAP value indicates better disease detection and localization performance.

The YOLOv8 model achieved the highest performance with a mAP@0.5 of 97.5% and a mAP@0.5:0.95 of 89.3%. This indicates that YOLOv8 not only detects diseased regions with very high accuracy but also maintains precise localization across thresholds ranging from 0.5 to 0.95. The relatively small difference between the two metrics demonstrates the model’s robustness in accurately identifying disease boundaries in paddy leaves.

The CNN model achieved a mAP@0.5 of 97.3% but a lower mAP@0.5:0.95 of 83.2%. Although the model performs exceptionally well at moderate localization requirements (IoU = 0.5), its performance decreases under stricter localization conditions, indicating that the predicted disease regions may not perfectly align with the ground-truth boundaries. Similarly, the CNN-LSTM model obtained a mAP@0.5 of 94.2% and mAP@0.5:0.95 of 84.6%, showing strong disease detection capability and slightly better localization consistency than the standalone CNN model. The inclusion of temporal learning through LSTM helps improve feature representation and disease pattern recognition.

Among the traditional machine learning models, the Decision Tree achieved 81.2% mAP@0.5 and 80.5% mAP@0.5:0.95, indicating relatively stable but lower detection performance. The smaller gap between the two metrics suggests consistent localization, although overall accuracy remains lower than deep learning approaches. The Naïve Bayes and Random Forest models recorded the lowest results, with mAP@0.5 values of 79.5% and 78.4%, respectively, and corresponding mAP@0.5:0.95 values of 78.2% and 77.2%. These results indicate limited capability in learning complex disease features and accurately localizing infected regions compared to deep learning architectures.

Overall, the results obtained as compared with Hasib & Akib (2026) demonstrate that deep learning-based models, particularly YOLOv8, provide superior disease detection and localization performance for paddy crop monitoring. The high mAP@0.5:0.95 value of YOLOv8 confirms its effectiveness in accurately identifying disease symptoms under stringent localization requirements, making it highly suitable for precision agriculture applications such as targeted pesticide spraying and automated crop health monitoring. Table 8 presents the comparative performance of three advanced deep learning models used for paddy crop disease and pest management applications (Javidan *et al*., 2024; Kalpana *et al*., 2024). The proposed models were evaluated based on their specific application areas, accuracy, and dataset characteristics.

The CNN model achieved the highest classification accuracy of 97.5% for pest detection using a dataset consisting of 1,000 images as compared with (Kappali *et al. (2024).* The high accuracy demonstrates the capability of convolutional neural networks to automatically extract discriminative visual features from pest images and accurately classify infestation levels. These results indicate that CNNs are highly effective for image-based pest identification in controlled datasets. The YOLOv8 model achieved an impressive 97.3% mAP using field-captured images collected under real agricultural conditions. Although its performance is marginally lower than CNN in terms of numerical accuracy, YOLOv8 offers a significant advantage through real-time object detection and localization as compared with the related work carried out by Kuswidiyanto *et al.,* 2022. Unlike traditional CNN classifiers that only identify the presence of pests or diseases as reported by Fulari *et al*. (2020), in comparison with YOLOv8, which simultaneously detects and localizes infected regions within the field, making it highly suitable for precision agriculture applications such as automated monitoring, drone-based surveillance, and targeted pesticide spraying. The CNN-LSTM model achieved an accuracy of 94.2% for disease forecasting using time-series data. The slightly lower accuracy compared to CNN and YOLOv8 is expected because disease forecasting is a more complex task that requires learning temporal disease progression patterns in addition to spatial image features. The integration of Long Short-Term Memory (LSTM) networks enables the model to analyze sequential data and predict future disease outbreaks, making it valuable for early warning systems and preventive crop management.

It is important to distinguish between the architectural and empirical aspects of the proposed work's uniqueness. Precision irrigation and pest/disease detection are typically implemented at the architectural level as distinct pipelines with independent sensor sets and thresholds (Barbedo, 2013; Fulari *et al*., 2020; Verma & Kaur, 2025; Zhang *et al*., 2026). Instead, the PH-IPM framework integrates both tasks into a single edge-to-cloud pipeline (Figures 3a and 3b). The irrigation controller and the pest/disease forecasting model are both informed by the same field-level sensor stream, and the two outputs are jointly displayed through a single decision-support dashboard. Instead of treating hydration and pest management as separate goals, this design takes advantage of the relationship between soil moisture regimes and pest pressure—wetting-drying cycles that reduce moisture-favoring pests—as an explicit control variable. As demonstrated in Figures 8–11, the suggested CNN, YOLOv8, and CNN-LSTM models achieve disease and pest identification accuracies of 94.2–97.5% at the empirical level, surpassing the 70–80% range generally reported in previous work (Barbedo, 2013; Fulari *et al*., 2020; Verma & Kaur, 2025; Zhang *et al*., 2026). Crucially, the majority of previous research reports these accuracies using offline datasets or controlled/simulated environments without combining them with multi-season field validation or real-time irrigation control (Ajith *et al*., 2025).

Additionally, Table 9 denotes that the suggested method includes a crucial agronomic interaction whereby certain wetting-drying cycles can inhibit moisture-loving bugs, allowing irrigation management to serve as both an indirect pest-control mechanism and a water-saving tactic. Furthermore, in contrast to many current methods, the suggested framework uses real-world, multi-season field experiments conducted on a 5-ha paddy plot to test both decision loops concurrently instead of depending just on simulations or publicly accessible information. The suggested approach differs from published techniques like those of Ajith *et al*. (2025), Verma & Kaur (2025), Zhang *et al*. (2026) and Mehta *et al. (2025),* which do not report comparable joint validation of irrigation and pest-

management decisions under multi-season field conditions, thanks to its integrated field-based validation. In order to close this gap, the PH-IPM framework validates both the hydration and pest-management components simultaneously, in situ, over two growing seasons on a 5-ha paddy plot. The measured outcomes—25.3% irrigation water reduction, 10.21% yield increase, and 40% pesticide reduction—reflect the combined effect of the integrated system rather than either component alone.

Decisions can be made in response to shifting field conditions because of the framework of continual monitoring of soil moisture, temperature, relative humidity, rainfall, and water level. Irrigation decisions are based on environmental data and moisture/water-level status, and sensor observations are gathered every two hours. In order to provide validation in naturally changing field conditions rather than just in lab or simulation settings, the field evaluation was carried out across two growing seasons on a 5-ha paddy patch. This environmental unpredictability is now fully discussed in the revised text, which also makes it clear that the framework's robustness results from ongoing sensor feedback and adaptive decision logic.

The PH-IPM method outperforms traditional control methods (Sujatha *et al*., 2025) by combining hydration with IPM, naturally suppressing moisture-loving pest populations while enabling precision through the use of edge ML techniques as depicted by the illustration provided in Figure 12(a) and Figure 12(b) in comparison with the previous work reported by Bassine *et al*. (2023). Limitations identified for this method include localized datasets from which to train machine learning models (these issues were resolved by retraining the algorithms with local pest data), connectivity issues in remote plots (mitigated using LoRa wireless technology) and the inability to use drones to monitor plants, offering a potential solution for enhancing scalability while supporting India's rice sustainability goals (Delfani *et al.,* 2024). Validation from real-life applications demonstrated the feasibility of PH-IPM on regions similar to Ambattur and supported sustainable or environmentally friendly farming as a viable business model.

Each element of the suggested PH-IPM framework's contributions to computing, communication, energy efficiency, and practical deployment are summed up in the table. Sensing is handled by low-power ESP32 nodes, intense AI inference is carried out by Raspberry Pi, and storage and visualization are supported by cloud services. The framework is appropriate for resource-constrained agricultural contexts because of its distributed architecture, which lowers computing and communication demands as depicted in Table 10.

Overall, the results indicate that all three deep learning models provide excellent performance for paddy crop protection. The CNN model is highly effective for static pest classification, YOLOv8 excels in real-time field detection and localization, and CNN-LSTM offers predictive capabilities for disease forecasting. Among these approaches, YOLOv8 provides the most practical solution for precision agriculture due to its combination of high detection accuracy, real-time processing capability, and field-level disease localization, while CNN-LSTM contributes significantly to proactive disease management through early prediction of disease occurrence.

Realistic Field Validation and Deployment Adaptability

The key novelty lies in the tight integration of precision hydration and integrated pest management within a single IoT machine learning framework tailored for paddy, enabling data-driven, adaptive farm operations that enhance water use efficiency, yield stability, and environmental sustainability (Hasib & Akib, 2026). For small-holder farmers in water-stressed regions such as Tamil Nadu, such a system offers a scalable pathway toward climate-resilient paddy cultivation through automated, low-cost monitoring and AI-assisted decision support.

Yields increase 15-25% via optimized inputs; pesticide use drops 40% with targeted IPM. The system is scalable for paddy crop-growing regions in Tamil Nadu and supports sustainable farming practices amid climate variability. For paddy, systems link humidity spikes to sheath blight risks, optimizing

wetting-drying cycles to suppress hoppers naturally. Trials report 20% yield gains and scalable deployment via LoRa for multi-hectare fields, ideal for Indian contexts like Tamil Nadu.

The PH-IPM framework was validated through two growing seasons (2024–2025) on a 5-ha IR64 paddy field in Tamil Nadu using three treatment replicates. Real-world variability in weather, illumination, pest occurrence and soil conditions was incorporated through field imaging and two-hourly sensor observations. The distributed edge-to-cloud architecture used ESP32 nodes, Raspberry Pi-based CNN/YOLOv8 inference and Firebase/MQTT communication. Approximately 20 sensing nodes supported multi-hectare monitoring. Compared with conventional management, the framework reduced irrigation water use by 25.3%, increased yield by 10.21%, and reduced pesticide use by 40%.

Quantitative Analysis of Field Performance

The field results demonstrate that the proposed PH-IPM framework provides measurable improvements in resource utilization and agricultural productivity compared with conventional management.

Water Reduction (%) = $\frac{W_{conventional}-W_{PH-IPM}}{W_{conventional}}\ x\ 100 \rightarrow \frac{837-625}{837}\ x\ 100$ = 25.3%

WUE Improvement (%) = $\frac{\mathrm{WUE_{PH-IPM}}-WUE_{conventional}}{WUE_{conventional}}$ x 100$\rightarrow \frac{1.05-0.80}{0.80}\ x\ 100$ = 31.25%

where WUE = $\frac{Crop\ Yield\ (kg)}{Irrigation\ Water\ USed\ (m^3)}$

Yield Improvement (%) = $\frac{Yp-Y_C}{Y_C}x100$

Y_c – PH-IPM yield

Y_P- is the pesticide usage by PH-IPM

Therefor the Yield Improvement (%) = 10.21%

Pesticide Reduction (%) = $\frac{P_c-P_P}{P_C}x100$

$P_c$and P_P– pesticide usage during conventional management and PH-IPM

- Water savings: Irrigation demand decreased by approximately 212 mm (25.3%) compared with conventional flooding.
- Improved WUE:Water-use efficiency increased from 0.80 to 1.05 kg/m³**,** indicating higher productivity per unit of irrigation water.
- Reduced pesticide use: Pesticide consumption decreased by 40% through detection- and threshold-based targeted intervention.
- Model performance: CNN, YOLOv8, and CNN–LSTM achieved 94.2–97.5% identification performance**.**
- Overall benefit: PH-IPM improved water efficiency, crop protection, and yield (10.21%) while reducing agricultural inputs.

Limitations

Validation of the system is limited to certain specific datasets and environmental conditions found in paddy fields where rice crops are grown. To fully assess scalability, strength and general application to large scale agricultural deployments, additional testing with a diverse range of geographical areas, types of crops and periods throughout the year will be needed.

Although the proposed preprocessing, temporal modeling, image-quality control, and distributed IoT architecture improve robustness, the present study does not constitute an exhaustive fault-injection or stress test of the system. Prolonged sensor failures, simultaneous node failures, extreme weather conditions, severe image degradation, and extended network outages may still affect prediction

reliability and irrigation recommendations. Therefore, future work should evaluate the framework using controlled sensor-failure scenarios, packet-loss experiments, adverse-weather image datasets, and multi-location, multi-season field trials. Such testing would provide quantitative estimates of system availability, recovery time, prediction degradation, and recommendation reliability under failure conditions.

Future Scope

In future work, integrating weather APIs and drone imagery could further refine predictions, paving the way for fully autonomous, AI-driven paddy management that enhances productivity, profitability and ecological balance. The developed model can be used in the future to detect other plant leaf diseases in other important crops grown in India. A promising extension of the PH-IPM framework is the incorporation of Generative AI. Generative Adversarial Networks (GANs) or diffusion-based models could be used to synthetically augment under-represented disease classes (e.g., blast, bacterial leaf blight) beyond the current dataset size of 1000 images, potentially improving CNN/YOLOv8 generalization without additional field data collection. Separately, a large language model could be layered on top of the existing dashboard to translate CNN/YOLOv8/LSTM outputs and IPM threshold triggers into plain-language, farmer-facing irrigation and pesticide-application advisories, rather than requiring farmers to interpret raw classification and forecast outputs directly. Both directions would require dedicated validation and ablation studies comparing model accuracy with and without synthetic augmentation and farmer usability testing for advisory quality — before being incorporated as core components of the framework.

Future validation will therefore extend to multiple locations, soil types, rice varieties, climatic zones, and pest-pressure conditions over several additional growing seasons. Controlled fault-injection experiments involving sensor failures, packet loss, power interruptions and missing image data will also be conducted. Long-duration monitoring will additionally quantify node uptime, communication reliability, energy/solar autonomy, recovery time, model-performance degradation and stability of irrigation/pest-management recommendations. Such validation will provide stronger evidence for the scalability, reliability, and transferability of PH-IPM from experimental field deployment toward routine agricultural operation.

Conclusion

The Precision Hydration and Integrated Pesticide Management System (PH-IPMS) for paddy fields represents a transformative approach to sustainable rice farming through IoT sensing and machine learning. It optimizes water use by dynamically adjusting irrigation via ultrasonic sensors and ML models like LSTM, cutting wastage by 20–25% while maintaining yields. Pest detection with YOLO variants and CNN-SVM hybrids enables targeted spraying, slashing chemical inputs and environmental harm. In this work, an application has been successfully developed for the detection and recognition of paddy crop diseases with a precision hydration system for irrigation.

Conflict of Interest

The authors declare no conflicts of interest in this work.

Acknowledgment

This work did not receive any funding. Data collection was carried out at the Center for Smart Agriculture, ECE department, Dr. M.G.R. Educational and Research Institute.

References

Ahmed, I., & Yadav, P. K. (2023). Plant disease detection using machine learning approaches. *Expert Systems*, *40*(5), e13136. https://doi.org/10.1111/exsy.13136

Ajith, S., Vijayakumar, S., & Elakkiya, N. (2025). Yield prediction, pest and disease diagnosis, soil fertility mapping, precision irrigation scheduling, and food quality assessment using machine learning and deep learning algorithms. *Discover Food*, *5*(1), 67.https://doi.org/10.1007/s44187-025-00338-1

Arribas, J. I., Sánchez-Ferrero, G. V., Ruiz-Ruiz, G., & Gómez-Gil, J. (2011). Leaf classification in sunflower crops by computer vision and neural networks. *Computers and Electronics in Agriculture*, *78*(1), 9-18.https://doi.org/10.1016/j.compag.2011.05.007

Barbedo, J. G. A. (2013). Digital image processing techniques for detecting, quantifying and classifying plant diseases. *SpringerPlus*, *2*(1), 660.https://doi.org/10.1186/2193-1801-2-660

Bassine, F. Z., Epule, T. E., Kechchour, A., & Chehbouni, A. (2023). Recent applications of machine learning, remote sensing, and iot approaches in yield prediction: A critical review. *arXiv preprint arXiv:2306.04566*. https://doi.org/10.48550/arXiv.2306.04566

Delfani, P., Thuraga, V., Banerjee, B., & Chawade, A. (2024). Integrative approaches in modern agriculture: IoT, ML and AI for disease forecasting amidst climate change. *Precision Agriculture*, *25*(5), 2589-2613. https://doi.org/10.1007/s11119-024-10164-7

Dey, A. K., Sharma, M., & Meshram, M. R. (2016). Image processing based leaf rot disease, detection of betel vine (Piper BetleL.). *Procedia Computer Science*, *85*, 748-754. https://doi.org/10.1016/j.procs.2016.05.262.

Ejaz, M. H., Bilal, M., Habib, U., Attique, M., & Chung, T. S. (2025). Crop pest classification using deep learning techniques: A review. *arXiv preprint*. https://doi.org/10.48550/arXiv.2507.01494

Fulari, U. N., Shastri, R. K., & Fulari, A. N. (2020). Leaf disease detection using machine learning. *Journal of Seybold Report*, *15*(9), 1828–1835.

Hasib, A., & Akib, A. S. M. A. S. (2026). An IoT-Based Smart Plant Monitoring and Irrigation System with Real-Time Environmental Sensing, Automated Alerts, and Cloud Analytics. *arXiv, preprint.*https://doi.org/10.48550/arXiv.2601.15830

Javidan, S. M., Banakar, A., Rahnama, K., Vakilian, K. A., & Ampatzidis, Y. (2024). Feature engineering to identify plant diseases using image processing and artificial intelligence: A comprehensive review. *Smart Agricultural Technology*, *8*, 100480. https://doi.org/10.1016/j.atech.2024.100480

Kalpana, P., Smitha, L., Madhavi, D., Nabi, S. A., Kalpana, G., & Kodati, S. (2024). A Smart Irrigation System Using the IoT and Advanced Machine Learning Model. *International Journal of Computational and Experimental Science and Engineering*, *10*(4), 1158-1168.https://doi.org/10.22399/ijcesen.526

Kandpal, K. C., & Kumar, A. (2024). Migrating from invasive to noninvasive techniques for enhanced leaf chlorophyll content estimations efficiency. *Critical Reviews in Analytical Chemistry*, *54*(7), 2583-2598. https://doi.org/10.1080/10408347.2023.2188425

Kappali, H. R., Sadyojatha, K. M., & Prashanthi, S. K. (2024). Computer vision and machine learning in paddy diseases identification and classification: A review. *Indian Journal of Agricultural Research*, *58*(2), 183.https://doi.org/10.18805/IJARe.A-6061

Kavitha, S., Malathi, M., Sinthia, P., Priya, R.K., Suresh Kumar, K. (2023). *Early Identification of Plant Diseases by Image Processing Using Integrated Development Environment.* In: Rathore, V.S., Tavares, J.M.R.S., Piuri, V., Surendiran, B. (eds) Emerging Trends in Expert Applications and Security. ICE-TEAS 2023. Lecture Notes in Networks and Systems, vol 681. Springer, Singapore. https://doi.org/10.1007/978-981-99-1909-3_13

Kuswidiyanto, L. W., Noh, H. H., & Han, X. (2022). Plant disease diagnosis using deep learning based on aerial hyperspectral images: A review. *Remote Sensing*, *14*(23), 6031.https://doi.org/10.3390/rs14236031

Lakshmi, R. K., & Savarimuthu, N. (2021). A novel transfer learning ensemble based deep neural network for plant disease detection. In *2021 International Conference on Computational Performance Evaluation (ComPE)* (pp. 017-022). IEEE. https://doi.org/10.1109/ComPE53109.2021.9751910

Mehta, M. H., Singh, S., & Awasthi, R. K. (2025). A review of IoT-based technologies for identification and monitoring of rice crop diseases. *International Journal of Latest Technology in Engineering, Management & Applied Science*, *14*(5), 418-422.https://doi.org/10.51583/IJLTEMAS.2025.140500042

Saha, K. K., Weltzien, C., Bookhagen, B., & Zude-Sasse, M. (2024). Chlorophyll content estimation and ripeness detection in tomato fruit based on NDVI from dual wavelength LiDAR point cloud data. *Journal of Food Engineering*, *383*, 112218. https://doi.org/10.1016/j.jfoodeng.2024.112218

Shah, A., Singh, R., Zaytar, A., Tadesse, G. A., Robinson, C., Tafti, N., ... & Ferres, J. M. L. (2026). Machine Learning for Sustainable Rice Production: Region-Scale Monitoring of Water-Saving Practices in Punjab, India. In *Proceedings of the AAAI Conference on Artificial Intelligence*, *40*(46), 39219-39227.https://arxiv.org/abs/2507.08605

Shawon, S. M., Niha, F. L., Sinan, O. H., Abed, M. J., Al Qwaid, M., Sarker, M. T., & Zubair, H. T. (2026). Integrating federated learning and explainable AI for plant disease detection: A systematic review of recent

advances in precision agriculture. *International Journal of Computational Intelligence Systems*, *19*(1), 138. https://doi.org/10.1007/s44196-026-01219-w

Sonobe, R., Hirono, Y., & Oi, A. (2020). Non-destructive detection of tea leaf chlorophyll content using hyperspectral reflectance and machine learning algorithms. *Plants*, *9*(3), 368. https://doi.org/10.3390/plants9030368

Strange, R. N., & Scott, P. R. (2005). Plant disease: A threat to global food security.*Annual Review of Phytopathology*, *43*, 83–116. https://doi.org/10.1146/annurev.phyto.43.113004.133839

Sujatha, K, Khekare, G., Midhunchakkaravarthy, D., Bhavani, N. P. G., & Malathi, M. (2025). Cascaded CNN for early detection of plant diseases machine vision techniques in smart agriculture. *International Journal of Basic and Applied Sciences, 14*(4), 55–66.https://doi.org/10.14419/b9c87e95

Suthar, O. P., Mishra, A., & Singhal, S. (2025). Machine learning-based leaf disease detection and classification in plants-A review. In *AIP Conference Proceedings* (Vol. 3185, No. 1, p. 020102). AIP Publishing LLC. 020102. https://doi.org/10.1063/5.0240221

Verma, A., & Kaur, P. (2025, March). Classifying tomato leaf diseases using diverse deep learning architectures: AlexNet, DenseNet, Inception, Xception. In *2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)* (Vol. 3, pp. 1-6). IEEE. https://doi.org/10.1109/IATMSI64286.2025.10984581

Ye, Z., Tan, X., Dai, M., Chen, X., Zhong, Y., Zhang, Y., ... & Kong, D. (2024). A hyperspectral deep learning attention model for predicting lettuce chlorophyll content. *Plant Methods*, *20*, 22.https://doi.org/10.1186/s13007-024-01148-9

Zhang, S., & Liu, R. (2026). Multi-scale feature fusion-based vision mamba for robust plant disease image classification on field-acquired plantdoc data. *Frontiers in Plant Science*, *17*, 1842426. https://doi.org/10.3389/fpls.2026.1842426