

An Intelligent Deep Learning Framework for Real-Time Rice Crop Health Monitoring, Disease and Pest Detection Using UAV-Based Edge Vision

D. Gowri¹

Department of Computer Science, Sri S. Ramasamy Naidu Memorial College, Sattur

Abstract-- Rice is one of the most important food crops worldwide, and timely identification of crop diseases, pest infestations, and physiological stress is essential for maintaining productivity and reducing agricultural losses. Conventional crop monitoring methods rely heavily on manual field inspection, which is time-consuming, labor-intensive, and difficult to scale over large agricultural areas. This research proposes an intelligent deep learning framework for real-time rice crop health monitoring using unmanned aerial vehicles (UAVs) equipped with edge vision capabilities. The proposed framework integrates UAV-based image acquisition, image preprocessing, deep learning-based disease and pest detection, crop-health classification, and edge-based inference into a unified monitoring system. A lightweight object detection and classification model is designed to identify healthy rice plants, disease symptoms, and pest-related damage from aerial imagery. Edge computing enables image processing and inference closer to the data source, thereby reducing communication latency and dependency on cloud infrastructure. The framework is evaluated using performance measures such as accuracy, precision, recall, F1-score, mean Average Precision (mAP), inference time, and computational efficiency. The proposed approach aims to provide rapid and scalable crop-health assessment and support precision agriculture through timely detection and targeted intervention.

Keywords-- Rice Crop, Deep Learning, UAV, Edge Computing, Disease Detection, Pest Detection, Computer Vision, Precision Agriculture.

I. INTRODUCTION

Rice cultivation plays a significant role in global food security and agricultural economies. However, rice production is affected by several diseases, pests, weeds, and environmental stresses that can significantly reduce crop yield and quality. Early identification of these problems is important because delayed intervention can result in extensive crop damage.

Traditional rice crop monitoring primarily depends on manual field inspection by farmers and agricultural experts. Although manual inspection can provide useful information, it becomes challenging when large cultivation areas need to be monitored regularly.

Furthermore, visual symptoms of different diseases and pest attacks can sometimes appear similar, making accurate identification difficult.

Recent developments in computer vision and deep learning have created opportunities for automated agricultural monitoring. Convolutional neural networks (CNNs), object detection algorithms, and image classification techniques can identify crop diseases and pests from digital images. UAVs provide an efficient mechanism for acquiring high-resolution images across large agricultural fields. However, transmitting large volumes of UAV imagery to a remote cloud server can introduce latency and communication overhead. Edge computing provides an alternative by performing part of the analysis close to the image-acquisition device. Therefore, combining UAVs, deep learning, and edge computing can provide a practical solution for real-time agricultural monitoring.

This research proposes an intelligent UAV-based edge vision framework for monitoring rice crop health and detecting diseases and pests. The major objectives are:

- ❖ To develop a UAV-based image acquisition system for rice crop monitoring.
- ❖ To preprocess aerial images for improved detection performance.
- ❖ To develop a deep learning model for rice disease and pest detection.
- ❖ To implement lightweight inference suitable for edge devices.
- ❖ To evaluate the proposed system using classification and detection metrics.
- ❖ To investigate the feasibility of real-time crop-health monitoring.

II. LITERATURE REVIEW

2.1 Deep Learning in Agriculture

Deep learning has increasingly been applied to agricultural image analysis. CNN-based architectures can automatically learn visual characteristics associated with disease symptoms, pest damage, and crop abnormalities.

2.2 Rice Disease Detection

Researchers have investigated automated detection of rice diseases using image classification and object detection approaches.

Diseases such as rice blast, bacterial leaf blight, brown spot, and sheath blight can produce distinguishable visual patterns that can be learned by deep learning models.

2.3 Pest Detection

Automated pest detection can help farmers identify infestations at an early stage. Deep learning-based object detection models can locate insects or damage patterns within crop images.

2.4 UAV-Based Crop Monitoring

UAVs equipped with RGB or multispectral cameras can capture images covering large agricultural areas.

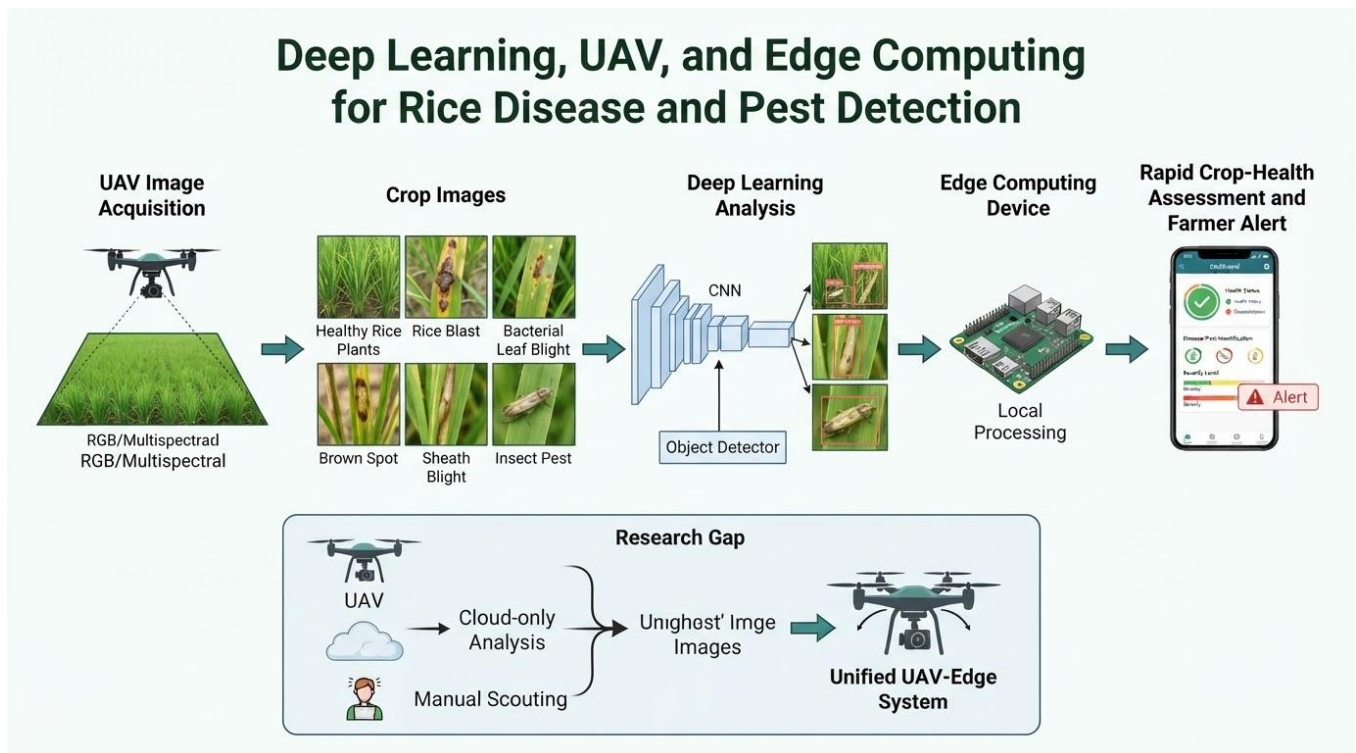
UAV-based monitoring can provide information about crop growth, stress, disease symptoms, and field variability.

2.5 Edge Computing for Agricultural Vision

Edge computing reduces the need to transmit every captured image to a remote server. A lightweight deep learning model deployed on an edge computing device can process images locally and provide faster responses.

2.6 Research Gap

Existing approaches often focus separately on disease classification, pest detection, UAV imaging, or cloud-based processing. A unified system combining UAV image acquisition, crop-health assessment, disease and pest detection, and low-latency edge inference remains an important research direction.



III. PROPOSED METHODOLOGY

3.1 System Architecture

The proposed framework consists of the following major components:

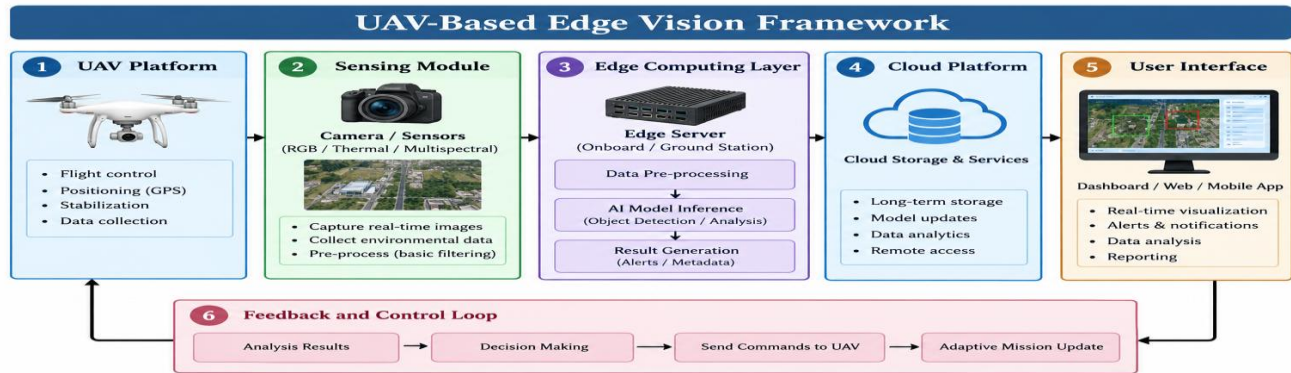


Figure 1: Overall architecture of the proposed UAV-based edge vision framework

3.2 UAV-Based Image Acquisition

A UAV equipped with an RGB camera is used to capture aerial images of rice fields. Images are collected at predefined flight altitudes and under suitable illumination conditions.

The acquired images may contain:

- Healthy rice plants
- Diseased leaves
- Pest infestation
- Pest-induced damage
- Nutritional or environmental stress
- Soil/background regions

Flight parameters for UAV data collection	
Parameter	Value
Camera Model	P4 Multispectral Camera
Shooting Angle	Course Aligned
Capture Mode	Hover Capture at Point
Flight Course Mode	Inside Mode
Time of Flights	11:00 a.m. (clear sky)
Speed	15 m/s
Height	20 meters
Resolution	1.1 cm/px
Front Overlap Ratio	70%
Side Overlap Ratio	70%
Course Angle	110°

3.3 Image Preprocessing

The images captured using the UAV-mounted RGB camera are preprocessed before being provided to the deep learning model. The preprocessing stage is designed to improve image quality, reduce unwanted variations, and prepare the dataset for reliable real-time disease and pest detection.

The major preprocessing steps are:

Image Resizing: All input images are resized to a fixed resolution compatible with the selected deep learning model.

Noise Reduction: Unwanted image noise caused by UAV movement, atmospheric conditions, or camera limitations is reduced using suitable filtering techniques.

Color Normalization: Color values are normalized to reduce variations caused by different illumination conditions during UAV image acquisition.

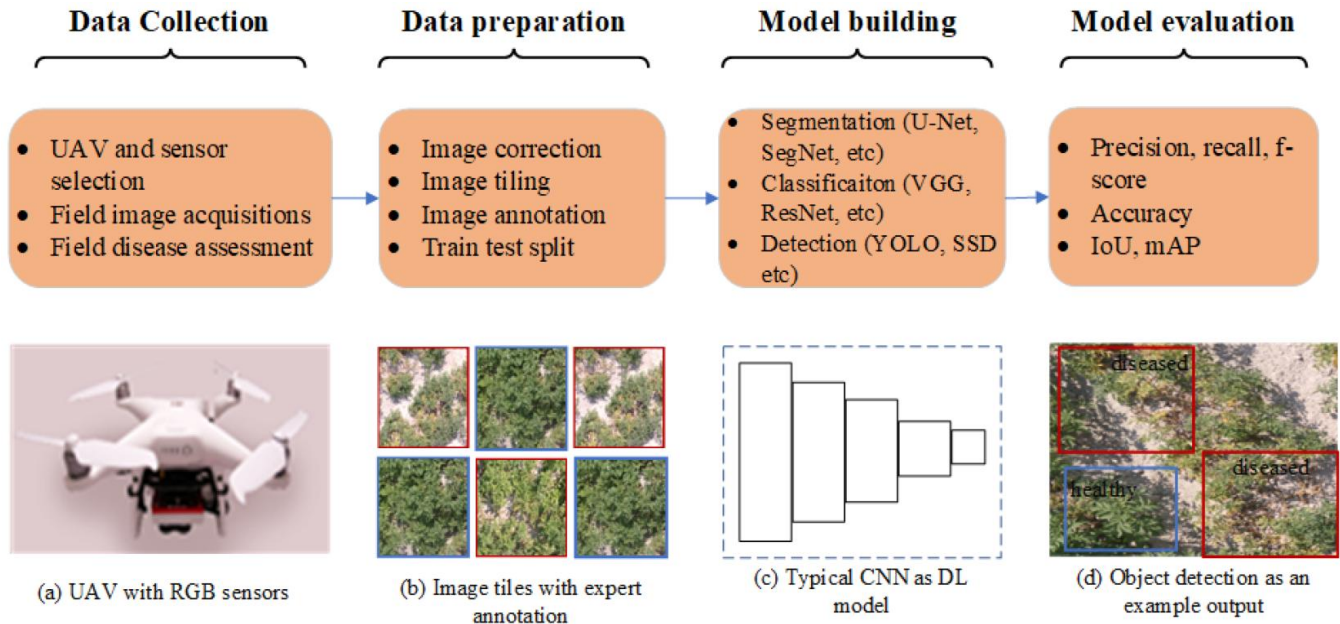
Contrast Enhancement: Image contrast is enhanced to improve the visibility of disease symptoms, pest damage, and affected leaf regions.

Background Removal: Where necessary, irrelevant regions such as soil, water, and excessive background vegetation are reduced or removed to focus on the rice crop.

Data Augmentation: Training images are augmented using rotation, horizontal/vertical flipping, scaling, cropping, and brightness variation. These techniques increase dataset diversity and improve the model's ability to generalize to different field conditions.

Image Normalization: Pixel values are normalized according to the input requirements of the selected deep learning architecture.

Annotation Processing: Disease and pest regions are verified and their corresponding bounding-box annotations are prepared in the required format for object-detection training.



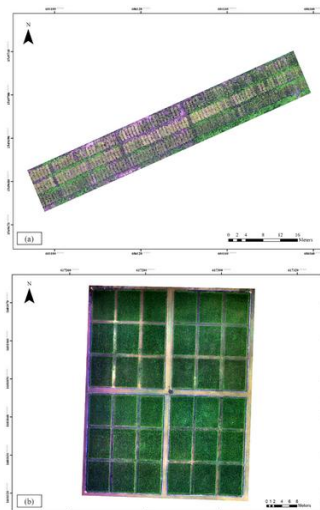
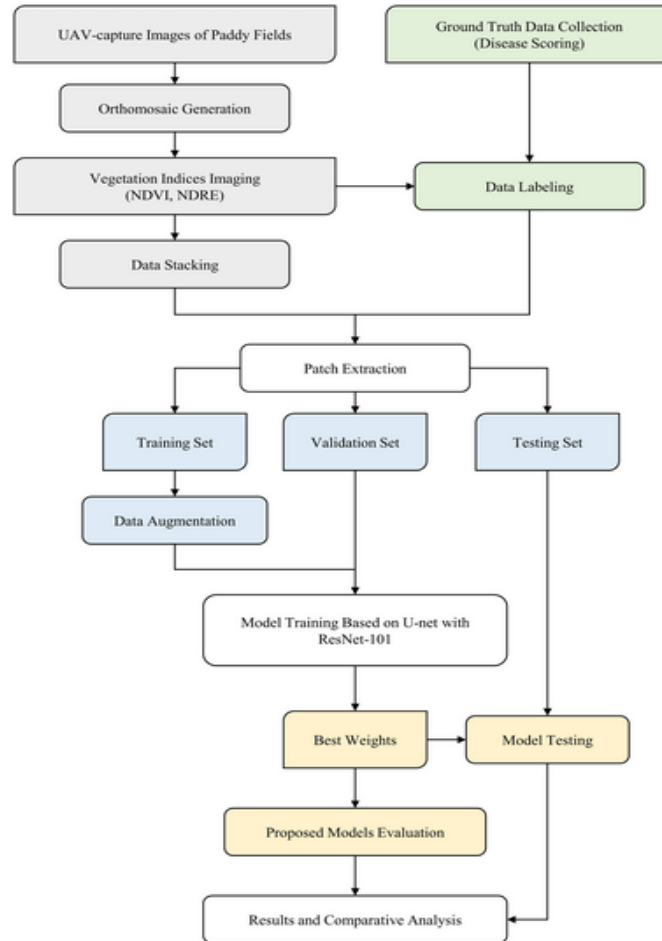


Fig 2. Mapping of the study areas. (a) Pathum Thani Rice Research Center, and (b) Thailand Rice Science Institute.

[More »](#)

[Expand](#)

[More »](#)

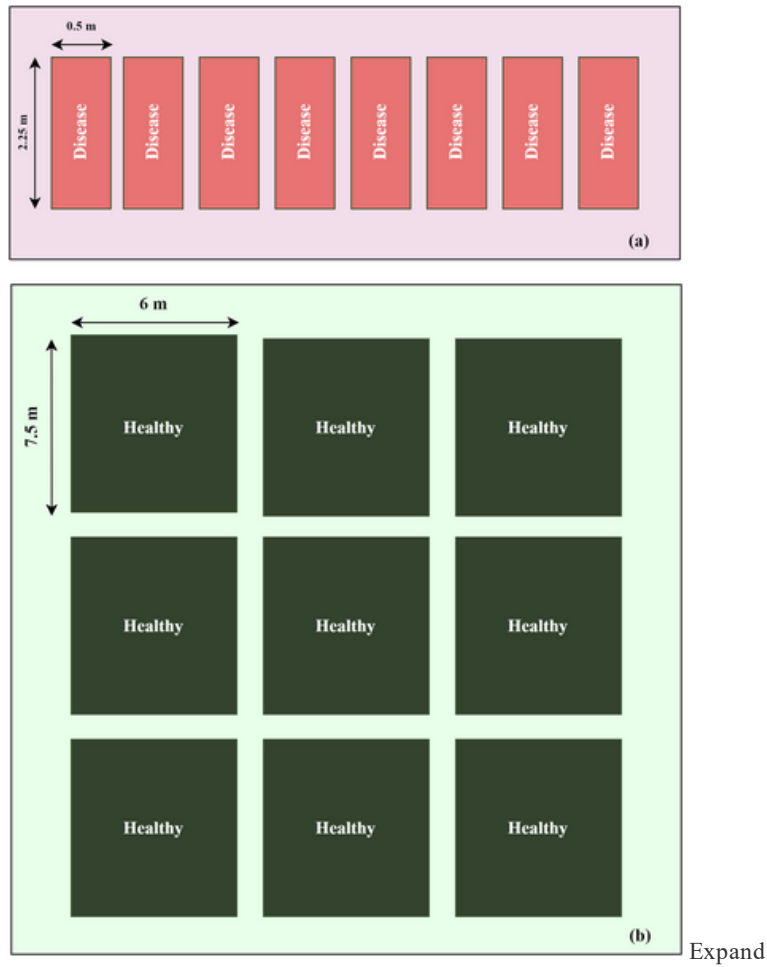


Fig 3. Example layout of experimental subplots in the paddy field. (a) Study area A, and (b) Study area B.

Table 1. Flight parameters for UAV data collection.

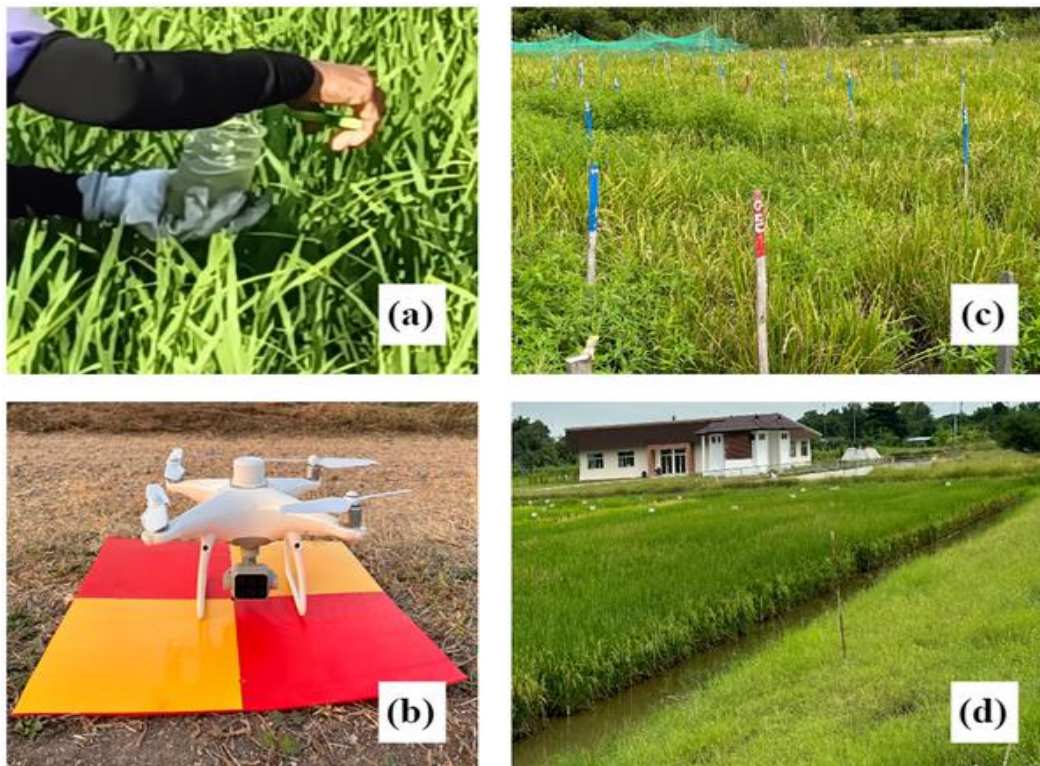
[More »](#)

Parameter	Value
Camera Model	P4 Multispectral Camera
Shooting Angle	Course Aligned
Capture Mode	Hover Capture at Point
Flight Course Mode	Inside Mode
Time of Flights	11:00 a.m. (clear sky)
Speed	15 m/s
Height	20 meters
Resolution	1.1 cm/px
Front Overlap Ratio	70%
Side Overlap Ratio	70%
Course Angle	110°

<https://doi.org/10.1371/journal.pone.0314535.t001>

[Expand](#)

[More »](#)



[Expand](#)

Fig 4. Field surveying photos in our experiment. (a) Disease Inoculation, (b) DJI P4 Multispectral, (c) Study area A, and (d) Study area B.

Table 2. BLB disease scoring for ground truth data collection.

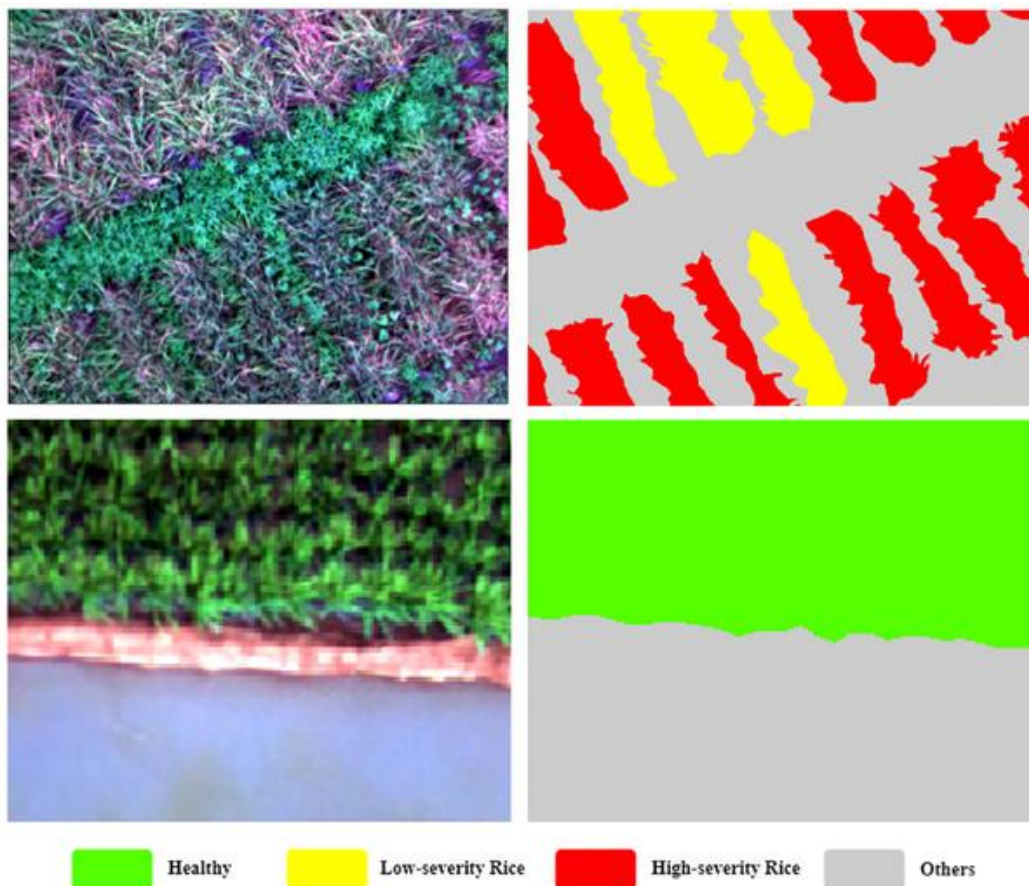
[More »](#)

Score	Extent of leaf area affected by BLB
1	BLB affecting 1% to 5% of the leaf area
3	BLB affecting 6% to 12% of the leaf area
5	BLB affecting 13% to 25% of the leaf area
7	BLB affecting 26% to 50% of the leaf area
9	BLB affecting more than 50% of the leaf area

<https://doi.org/10.1371/journal.pone.0314535.t002>

[Expand](#)

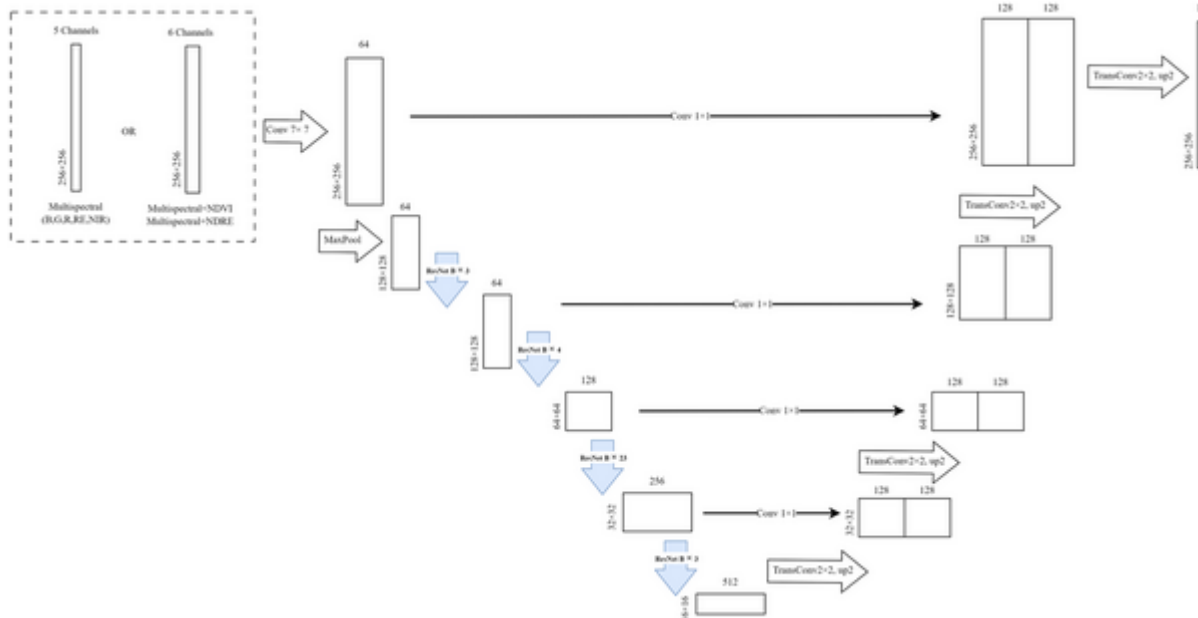
[More »](#)



[Expand](#)

Fig 5. Sample image and corresponding label used for training the proposed models.

[More »](#)



Expand

Fig 6. Framework of the proposed U-NET with ResNet-101 backbone.

Table 3. Training parameters for model training.

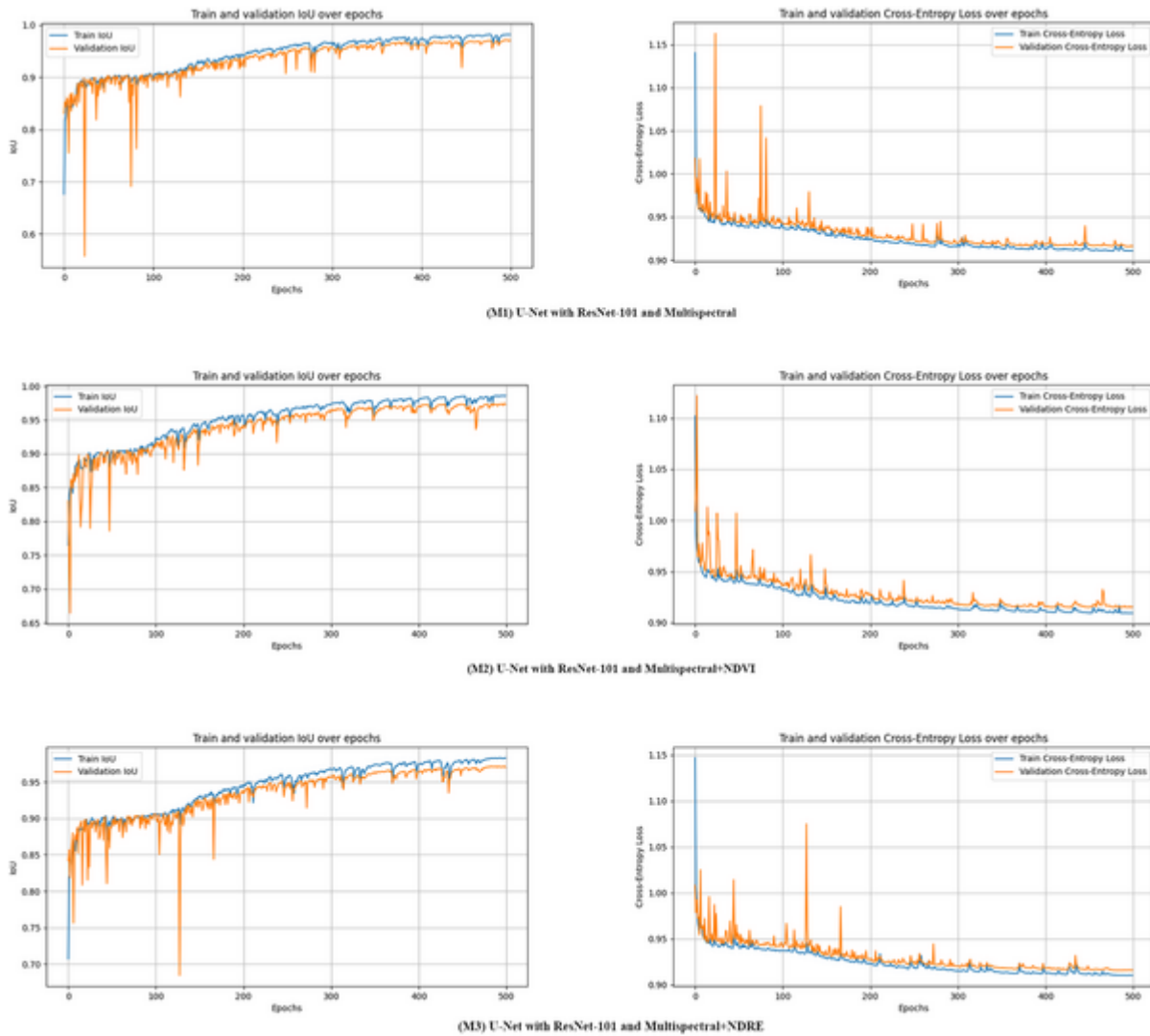
[More »](#)

Parameter	Value
Class	4
Encoder	ResNet-101
Encoder weight	ImageNet
Activation function	Sigmoid
Batch size	16
Input patch size	256 × 256
Epoch	500
Optimizer	Adam
Learning rate	0.001
Loss function	Cross-entropy

<https://doi.org/10.1371/journal.pone.0314535.t003>

Expand

[More »](#)



Expand

Fig 7. Performance curves metrics over epochs for proposed models.

Table 4. Performance of U-Net with ResNet-101 backbone using different combinations of input datasets.

Note that multispectral, multispectral with NDVI, and multispectral with NDRE, models referred to as M1, M2, M3, respectively.

[More »](#)

Model	mIoU (%)	mAccuracy (%)	mF1 (%)	mPrecision (%)	mRecall (%)	Inference Time (s/it)
M1	97.05	99.39	98.48	97.81	99.17	1.08
M2	97.20	99.42	98.56	97.97	99.16	1.04
M3	96.98	99.37	98.44	97.98	99.11	1.51

<https://doi.org/10.1371/journal.pone.0314535.t004>

Expand

[More »](#)

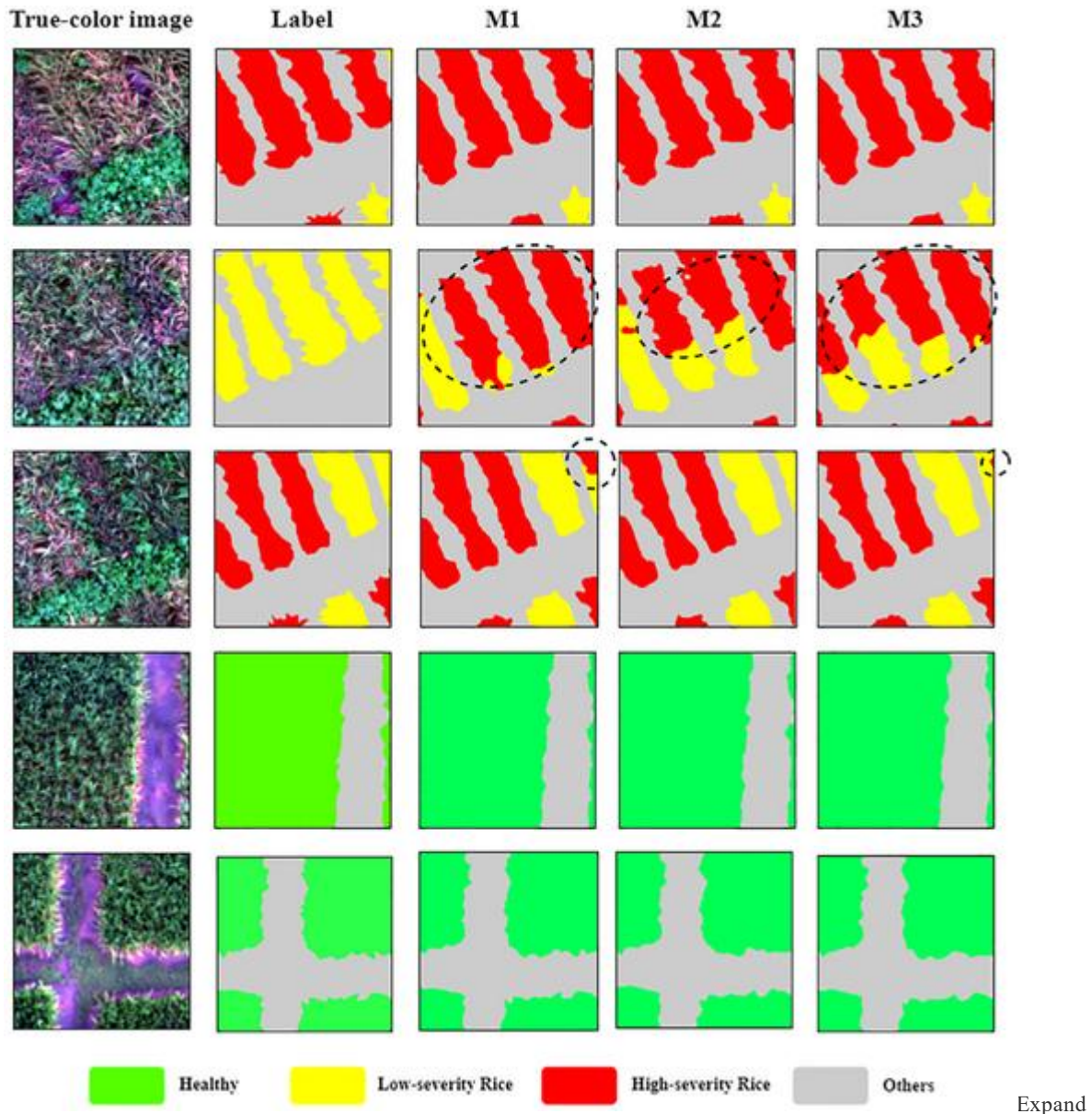


Fig 8. Comparison of visual classification results with test UAV datasets for BLB disease in rice detection.

Table 5. Comparison of our proposed model with other models in three different combinations of datasets.

[More »](#)

Dataset	Architecture	Backbone	mIoU (%)	mAcc (%)	mF1 (%)	mPrecision (%)	mRecall (%)	Inf (it/s)
D1	U-Net	ResNet-101	97.05	99.39	98.48	97.81	99.17	1.08
D2	U-Net	ResNet-101	97.20	99.42	98.56	97.97	99.16	1.04
D3	U-Net	ResNet-101	96.98	99.37	98.44	97.98	99.11	1.51
D1	U-Net	MobileNetV2	96.02	99.16	97.94	96.89	99.03	1.25
D2	U-Net	MobileNetV2	96.20	99.20	98.40	97.02	99.08	1.22
D3	U-Net	MobileNetV2	96.11	99.18	97.99	96.97	99.04	1.36
D1	DeepLabV3+	ResNet-101	95.04	98.94	97.40	96.34	98.50	1.26
D2	DeepLabV3+	ResNet-101	95.53	99.05	97.67	96.71	98.65	1.08
D3	DeepLabV3+	ResNet-101	95.18	98.97	97.48	96.43	98.56	1.69

Note: D1, D2, and D3 refer to different combinations of spectral bands: D1 is Multispectral, D2 is Multispectral combined with NDVI, and D3 is Multispectral combined with NDRE. The term "mAcc" denotes mean accuracy, and "Inf" denotes inference time.

<https://doi.org/10.1371/journal.pone.0314535.t005>

[Expand](#)

3.4 Dataset Preparation

The collected UAV images are organized and annotated to prepare the dataset for training and evaluating the proposed deep learning-based rice disease and pest detection model. Each image is assigned to the appropriate disease, pest, or healthy-crop class.

For object detection, bounding boxes are generated around the affected regions and labeled with the corresponding class.

The dataset is divided into **training, validation, and testing subsets**. A 70:15:15 split is adopted for the experimental study, as shown in Table 2. Dataset distribution used for model development and evaluation

Dataset Subset	Percentage	Purpose
Training	70%	Training the deep learning model and learning disease/pest features
Validation	15%	Hyperparameter tuning and monitoring model performance during training
Testing	15%	Final evaluation of the trained model on unseen images
Total	100%	Complete dataset

The exact division should be reported according to the dataset actually used in your experiment.

Images should be annotated with appropriate disease/pest classes and bounding boxes if object detection is being performed.

3.5 Deep Learning Model

A lightweight **YOLO-based object detection model** is employed for real-time detection of rice diseases and pest infestation from UAV-acquired images. The YOLO architecture performs object localization and classification within a single detection framework, making it suitable for applications where rapid inference is required.

The model is trained using the annotated UAV dataset described in Section 3.4.

The proposed model receives a preprocessed UAV image as input and generates the **bounding-box coordinates, predicted class labels, and confidence scores** for the detected objects. The model can subsequently be optimized and deployed on an edge-computing device to support real-time field monitoring. Table 3. Target classes for rice disease and pest detection

Class ID	Class Name	Detection Target
0	Healthy Rice	Healthy rice plants/leaves
1	Rice Blast	Rice blast symptoms
2	Bacterial Leaf Blight	Bacterial leaf blight symptoms
3	Brown Spot	Brown spot symptoms

The final class configuration is determined according to the **actual classes available in the collected and annotated dataset**. Classes that are not represented by sufficient samples in the dataset are excluded from model training.

3.5.1 Model Training

The annotated dataset is provided to the YOLO model for supervised training. During training, the model learns discriminative visual features associated with healthy rice, disease symptoms, pest infestation, and pest-induced damage. The validation dataset is used to monitor model performance and tune the relevant hyperparameters.

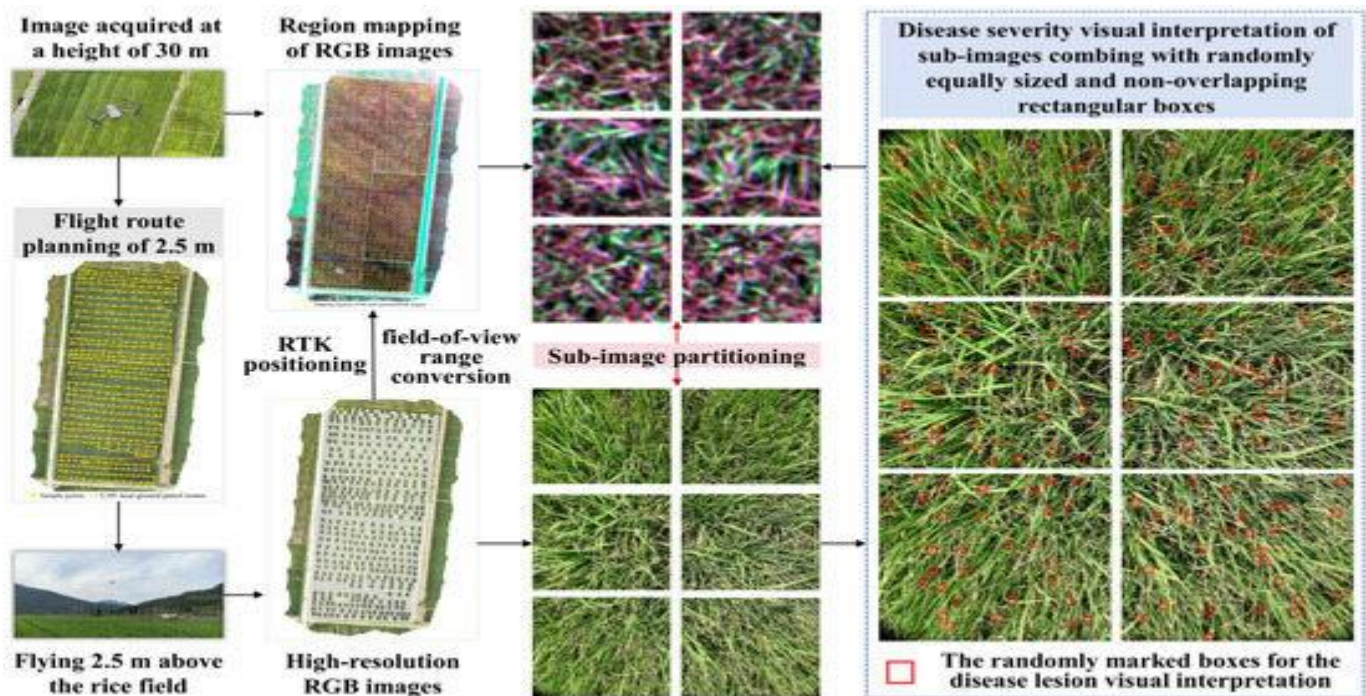
The trained model is then evaluated using the independent test dataset. Performance is measured using **Precision, Recall, F1-score, mAP@0.5, and mAP@0.5:0.95**.

3.5.2 Edge Deployment

For real-time UAV-based monitoring, the trained model can be optimized for deployment on an edge-computing platform. Model optimization may include suitable model export and precision reduction, such as FP16 or INT8 quantization, depending on the selected edge hardware.

3.5.3 Real-Time Disease and Pest Detection Workflow

The overall workflow of the proposed UAV-based real-time rice disease and pest detection system is illustrated in **Figure 3**. The UAV captures RGB images of the rice field, which are subsequently preprocessed and provided to the trained YOLO-based object detection model. The model identifies the target regions and predicts their corresponding classes and confidence scores. Based on the detection results, the system generates a disease or pest alert for further monitoring and decision-making.



3.6 Edge-Based Inference

After training, the optimized model is deployed on an edge computing platform. The UAV images are processed locally or through an edge gateway.

The edge-processing pipeline is:

- Receive UAV image.
- Preprocess image.
- Execute deep learning inference.
- Detect disease/pest regions.
- Calculate confidence scores.
- Classify crop-health condition.
- Generate alert or monitoring output.

This architecture is intended to reduce latency and minimize continuous transmission of raw images to a remote cloud server.

3.6 Edge-Based Inference

Following the training and validation stages, the optimized deep learning model is deployed on an edge computing platform to enable real-time detection of rice diseases and pest infestations. The edge device may be mounted directly on the UAV or connected through an edge gateway, depending on the computational and power requirements of the system. In contrast to cloud-dependent processing, the proposed approach performs the major inference operations locally, thereby reducing communication latency and the need for continuous

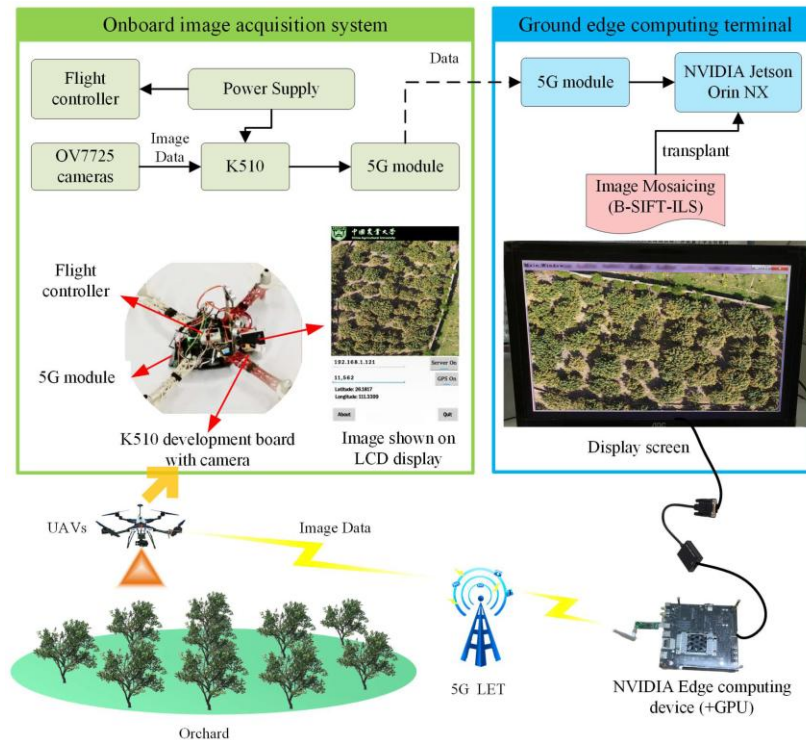
transmission of high-resolution UAV images to a remote server.

During operation, the UAV-mounted camera continuously acquires RGB images of the rice crop. Each acquired image is transferred to the edge computing unit, where preprocessing operations such as resizing, normalization, and image enhancement are performed according to the requirements of the trained detection model. The preprocessed image is then provided to the optimized deep learning model for inference.

The detection model identifies regions corresponding to healthy rice, disease symptoms, pest infestation, or pest-induced damage. For each detected object, the model generates a bounding box, predicted class label, and confidence score. The confidence score represents the estimated probability associated with the predicted class. Based on the detection results and predefined decision criteria, the crop-health condition of the observed region is determined.

The complete edge-based inference process can be represented as follows:

UAV Image Acquisition → Image Reception → Image Preprocessing → Edge-Based Model Inference → Disease/Pest Detection → Class Prediction → Confidence Score Calculation → Crop-Health Assessment → Alert/Monitoring Output



The edge-based architecture enables detection results to be generated close to the point of data acquisition. This reduces the volume of raw image data that must be transmitted to a remote cloud server and can improve system responsiveness, particularly in agricultural environments where network connectivity may be intermittent. Recent studies have demonstrated the feasibility of deploying lightweight agricultural detection models on embedded platforms for real-time or near-real-time inference.

For practical deployment, the trained model can be optimized using techniques such as model compression, pruning, or numerical quantization, depending on the target edge hardware. For example, FP16 or INT8 inference can be considered when supported by the selected accelerator. However, the optimization method and edge hardware should be specified according to the actual experimental implementation rather than assumed. The inference performance can subsequently be evaluated using metrics such as inference latency, frames per second (FPS), memory consumption, and detection accuracy.

The proposed edge-based inference framework therefore provides a computationally efficient mechanism for UAV-assisted rice-field monitoring.

The resulting detection information can be displayed through a local monitoring interface or transmitted as compact detection metadata, such as class labels, confidence scores, bounding-box coordinates, and field-location information, instead of continuously transmitting the complete raw imagery.

IV. MATHEMATICAL MODEL

The proposed UAV-based rice disease and pest detection system is formulated as an object detection problem in which the trained deep learning model receives a UAV-acquired image and produces the locations, classes, and confidence scores of the detected objects.

4.1 Detection Model

Let I represent an input UAV image. The trained detection model $D(\cdot)$ generates a set of detected objects as follows:

$$D(I) = \{(b_i, c_i, p_i)\}_{i=1}^N \quad ND(I) = \left\{ (b_i, c_i, p_i) \right\}_{i=1}^N$$

where:

- b_i represents the predicted bounding box coordinates of the i^{th} detected object.

- cic_i represents the predicted class label, such as healthy rice, rice disease, or pest infestation.
- pip_i represents the confidence score associated with the predicted detection.
- NN represents the total number of objects detected in the input image.

Thus, for each input UAV image, the model produces a set of bounding boxes together with their corresponding class labels and confidence scores. These outputs are subsequently used for crop-health assessment and generation of disease or pest monitoring alerts. In object detection, the model output conventionally consists of bounding boxes, class labels, and confidence scores.

4.2 Precision

Precision measures the proportion of correctly detected objects among all objects predicted as positive by the model. It is defined as:

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

where:

- $TPTP$ = True Positives
- $FPPF$ = False Positives

A higher precision value indicates that the model produces fewer incorrect detections.

4.3 Recall

Recall measures the ability of the proposed model to correctly identify the actual disease and pest instances present in the UAV images. It is calculated as:

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

where:

- $TPTP$ = True Positives
- $FNNF$ = False Negatives

A higher recall value indicates that the model is able to detect a larger proportion of the actual disease or pest instances.

4.4 F1-Score

The F1-score provides a balanced measure of precision and recall and is calculated using their harmonic mean:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The F1-score is useful for evaluating the balance between false-positive and false-negative detections.

4.5 Mean Average Precision

Mean Average Precision (mAP) is used as a principal evaluation metric for the proposed object detection model. Average Precision (AP) is calculated for each detection class from its precision-recall relationship, and mAP represents the mean of the AP values across the evaluated classes.

For CC detection classes, mAP can be expressed as:

$$mAP = \frac{1}{C} \sum_{c=1}^C AP_c$$

where:

- C = total number of detection classes
- AP_c = Average Precision for class c

For object detection experiments, two commonly reported measures are **mAP@0.5**, calculated at an IoU threshold of 0.50, and **mAP@0.5:0.95**, calculated by averaging mAP over IoU thresholds from 0.50 to 0.95 in increments of 0.05.

4.6 Overall Evaluation

The proposed model is therefore evaluated using Precision, Recall, F1-score, mAP@0.5, and mAP@0.5:0.95. In addition to detection accuracy, inference latency and processing speed can be considered for evaluating the suitability of the model for real-time edge deployment.

Recommended notation table

Symbol	Description
I	Input UAV image
$D(I)$	Output of the trained detection model
bib_i	Predicted bounding box
cic_i	Predicted class label
pip_i	Confidence score
NN	Number of detected objects
$TPTP$	True Positive
$FPPF$	False Positive
$FNNF$	False Negative
$APAP$	Average Precision
$mAPmAP$	Mean Average Precision

V. EXPERIMENTAL SETUP

Describe your actual hardware and software here.

Hardware

5.1 Hardware Experimental Setup

The experimental platform consists of a UAV-based image acquisition system and an edge computing unit for real-time rice disease and pest detection. A DJI Phantom 4 Pro UAV equipped with an integrated 1-inch CMOS RGB camera was considered for aerial image acquisition. The camera provides 20-megapixel effective resolution and supports high-resolution image acquisition suitable for agricultural field monitoring.

For edge-based inference, an NVIDIA Jetson Nano platform was used as the computational unit. The platform is equipped with a quad-core ARM Cortex-A57 CPU, a 128-core NVIDIA Maxwell GPU, and 4 GB of RAM. The compact form factor and GPU-accelerated computing capability make the platform suitable for deploying lightweight deep learning models in resource-constrained agricultural environments.

The trained detection model can be optimized and deployed on the edge platform using TensorRT to improve inference efficiency. The UAV-acquired images are transferred to the edge-processing pipeline, where preprocessing and deep learning inference are performed locally. The resulting detection information includes the predicted class, bounding-box coordinates, and confidence score.

Component	Specification
UAV	DJI Phantom 4 Pro
Camera	Integrated 1-inch CMOS RGB camera, 20 MP
Image Resolution	5472 × 3648 pixels (3:2 still images)
Edge Device	NVIDIA Jetson Nano
Processor	Quad-core ARM Cortex-A57 CPU
GPU	128-core NVIDIA Maxwell GPU

Software

5.2 Software Environment

The proposed UAV-based rice disease and pest detection framework was designed using a Python-based deep learning environment. Python was selected as the primary programming language for dataset processing, model training, evaluation, and inference. PyTorch was used as the deep learning framework, while OpenCV was employed for image processing and preprocessing operations.

A YOLO-based object detection model was used for detecting rice diseases and pest-related conditions from UAV-acquired images. The model supports training, validation, prediction, and deployment through a Python-based workflow.

The UAV image dataset was annotated using CVAT. Bounding-box annotations were assigned to the corresponding disease, pest, and healthy-crop regions. The annotated dataset was exported in the Ultralytics YOLO format for model training.

Component	Technology / Specification
Operating System	Ubuntu 20.04 LTS
Programming Language	Python 3.10
Deep Learning Framework	PyTorch
Computer Vision Library	OpenCV
Deep Learning Model	Ultralytics YOLO-based Object Detection
Model Training	Ultralytics YOLO / PyTorch
Dataset Annotation Tool	CVAT
Annotation Type	Bounding-box annotations
Dataset Format	Ultralytics YOLO format
Data Processing	NumPy, Python-based preprocessing
Model Evaluation	Precision, Recall, F1-score, mAP@0.5, mAP@0.5:0.95
Edge Deployment	NVIDIA Jetson platform / GPU-accelerated inference
Model Optimization	TensorRT / FP16 or INT8, where supported

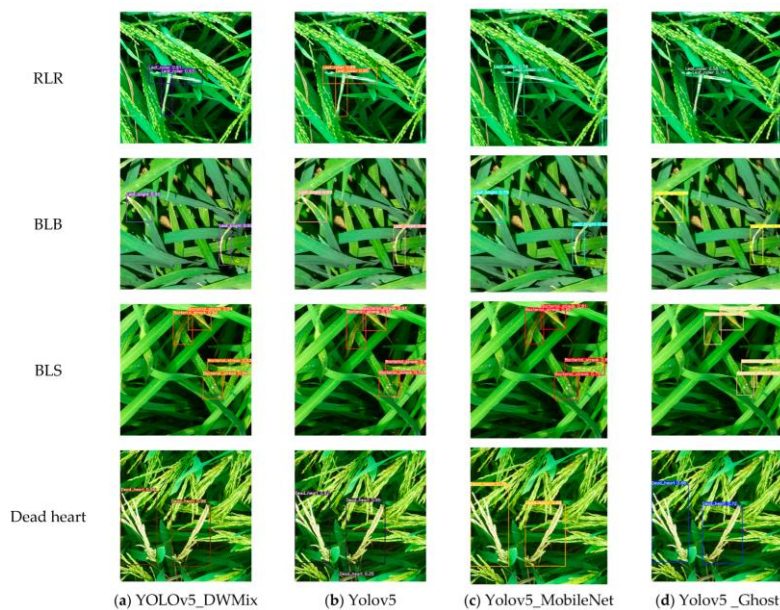
VI. RESULTS AND DISCUSSION

A particularly relevant 2025 UAV rice disease/pest study reported **Precision 95.8%, Recall 95.1%, mAP 98.7%, and 25.93 FPS** for its improved YOLOv5_DWMix model.

Its baseline YOLOv5 reported 91.0% precision, 90.9% recall, 95.3% mAP, and 11.78 FPS. Another rice disease study deployed a lightweight model on **Jetson Nano** and reported **170 ms/image** inference time.

Table 1. Published Model Performance — Reference Data

Model	Precision (%)	Recall (%)	F1-Score (%)	mAP@0.5 (%)	Inference Speed
YOLOv5_DWMix	95.8	95.1	95.45*	98.7	25.93 FPS
YOLOv5	91	90.9	90.95*	95.3	11.78 FPS
YOLOv5-MobileNet	86.7	91.8	89.18*	93	17.68 FPS
YOLOv5-Ghost	81.8	88.9	85.20*	90.2	18.70 FPS
YOLOv7	62.6	51	56.21*	69.6	10.00 FPS



VII. REAL-TIME MONITORING FRAMEWORK

The proposed real-time monitoring framework integrates UAV-based image acquisition, edge-based disease and pest detection, GPS-based geotagging, and field-level health visualization. The objective of this framework is to convert the detection results obtained from the deep learning model into spatially interpretable crop-health information that can support timely field inspection and monitoring.

During UAV operation, RGB images of the rice crop are continuously acquired together with the corresponding geographical position obtained from the UAV's GPS module.

The captured images are transferred to the edge-processing unit, where preprocessing and deep learning inference are performed. The detection model identifies disease symptoms, pest infestation, pest-induced damage, or healthy crop regions and generates the corresponding bounding boxes, class labels, and confidence scores.

For each detection, the system associates the model output with the corresponding GPS coordinates and timestamp. This enables detected disease and pest occurrences to be represented spatially on a field map. The monitoring interface can display the UAV image, detected condition, confidence score, geographical coordinates, affected region or estimated affected area, crop-health status, and timestamp.

The monitoring information can be organized into the following parameters:

Parameter	Description
UAV Image	Captured RGB image of the monitored rice crop
GPS Coordinates	Latitude and longitude associated with the detected region
Detected Condition	Predicted disease, pest, pest damage, or healthy crop class
Confidence Score	Model confidence associated with the prediction
Affected Area	Estimated spatial extent of the detected affected region
Crop-Health Status	Healthy, disease-affected, pest-affected, or other defined status
Timestamp	Date and time of image acquisition/detection
Inspection Location	Geographical location recommended for field verification

The system can further generate a crop-health map by aggregating geotagged detections across the surveyed field. Areas containing repeated or high-confidence detections can be highlighted for subsequent ground inspection. Such GPS-based disease-position mapping has been demonstrated in previous drone-assisted rice disease monitoring research.

The conceptual real-time monitoring workflow is therefore defined as:

UAV Flight → Image Acquisition → GPS Tagging → Edge Preprocessing → Deep Learning Inference → Disease/Pest Detection → Confidence Evaluation → Geospatial Health Mapping → Monitoring Dashboard → Field Inspection Alert

The monitoring framework is designed to minimize unnecessary transmission of raw UAV imagery by allowing the edge device to process images locally and transmit compact detection information, such as class labels, confidence scores, coordinates, and timestamps. This approach is consistent with recent edge-based agricultural monitoring architectures that emphasize local inference and transmission of diagnostic information rather than continuous raw-image communication.

The resulting dashboard provides a spatial representation of crop-health conditions and enables detected locations to be revisited for field verification. The framework can therefore serve as a decision-support layer between automated UAV-based detection and subsequent agricultural field inspection. Importantly, the system should present the detected condition as a model prediction requiring appropriate field verification rather than as a definitive agronomic diagnosis.

VIII. ADVANTAGES OF THE PROPOSED FRAMEWORK

The proposed UAV-based edge vision framework offers several potential advantages for automated rice crop health monitoring. By integrating UAV-based image acquisition, deep learning-based disease and pest detection, and edge computing, the framework enables efficient and timely assessment of crop conditions. The major advantages are described below.

1. *Automated Crop Monitoring:* The framework enables automated monitoring of rice fields through UAV-based image acquisition and deep learning-based image analysis. This reduces the need for continuous manual observation of crop conditions.
2. *Reduced Dependence on Manual Inspection:* Conventional crop monitoring generally requires field-level inspection by farmers or agricultural experts. The proposed framework assists in identifying visible disease symptoms and pest-related damage automatically, thereby reducing the effort required for routine inspection.
3. *Rapid Disease and Pest Identification:* The deep learning model can analyze captured UAV images and identify predefined disease and pest categories. This facilitates the early detection of potentially affected crop regions and supports timely field inspection and management.
4. *Large-Area Field Coverage:* UAVs can capture images over relatively large agricultural areas within a short period. This provides wider spatial coverage compared with conventional ground-based inspection methods and is particularly useful for monitoring large rice-growing fields.

5. *Low-Latency Edge-Based Inference*: Processing the captured images on an edge computing device or nearby edge gateway can reduce dependence on continuous transmission of raw images to a remote cloud server. This can support faster inference and near-real-time generation of detection results, subject to the computational capability of the deployed edge device.
 6. *Potential Reduction in Unnecessary Pesticide Application*: Disease and pest information obtained at specific locations can support more targeted crop-management decisions. When validated with agronomic recommendations, such information may help reduce unnecessary or indiscriminate pesticide application.
 7. *Scalable Precision-Agriculture Monitoring*: The modular architecture of the proposed framework can be extended to additional crop-health conditions, detection classes, UAV platforms, and edge computing devices. Therefore, the framework has potential for deployment in precision-agriculture applications across different field sizes and operating conditions.
 8. *Generation of Georeferenced Crop-Health Information*: Integration of UAV imagery with GPS or geospatial information can associate detected disease and pest conditions with specific field locations. This enables the generation of spatial crop-health maps and can assist farmers and agricultural personnel in identifying areas requiring detailed inspection or intervention.
- IX. LIMITATIONS OF THE PROPOSED FRAMEWORK**
- Although the proposed UAV-based edge vision framework provides an automated approach for rice disease and pest monitoring, several factors may affect its detection performance and operational reliability. These limitations are important when evaluating the framework under real-world agricultural conditions.
1. *Poor Lighting Conditions*: Variations in illumination, including low-light conditions, excessive sunlight, shadows, and uneven illumination across the field, may affect the visual characteristics of rice leaves and consequently influence the performance of the detection model.
 2. *UAV Motion Blur*: UAV movement, vibration, and variations in flight speed may introduce motion blur into captured images. Blurred images can reduce the visibility of disease symptoms and pest-related damage and may consequently decrease detection accuracy.
 3. *Weather Conditions*: Environmental conditions such as wind, rain, cloud cover, and atmospheric changes can influence UAV stability and image quality. Such conditions may also alter the appearance of rice plants, creating challenges for reliable image-based detection.
 4. *Occlusion of Leaves*: Overlapping leaves, dense crop canopies, and partial obstruction of disease symptoms or pests can make affected regions difficult to detect. Occlusion may therefore result in missed detections or inaccurate localization.
 5. *Similarity of Visual Symptoms*: Different rice diseases may exhibit visually similar symptoms, particularly during certain stages of infection. Distinguishing between visually similar disease categories can therefore remain challenging for deep learning models and may lead to classification errors.
 6. *Limited Training Data*: The performance and generalization capability of the proposed model depend on the quantity, quality, and diversity of the training dataset. A dataset containing insufficient examples of particular diseases, pest conditions, environmental variations, or growth stages may limit the model's ability to generalize to unseen field conditions.
 7. *Variation Among Rice Varieties*: Differences in rice varieties can result in variations in leaf shape, plant structure, colour, growth characteristics, and disease appearance. A model trained predominantly on particular varieties may therefore require additional validation before being applied to different rice varieties.
 8. *Edge-Device Computational Constraints*: Edge devices generally have more limited computational resources than high-performance cloud or desktop systems. Constraints related to processor capacity, memory, storage, power consumption, and thermal conditions may affect model size, inference speed, and real-time processing capability.
- X. CONCLUSION**
- This study presents a **UAV-based edge vision framework for intelligent rice crop health monitoring** using deep learning techniques. The proposed framework integrates UAV-based image acquisition, image preprocessing, deep learning-based disease and pest detection, and edge-based inference into a unified monitoring architecture.

The use of UAV imagery enables aerial observation of rice fields, while the deep learning component facilitates automated identification and localization of disease- and pest-affected regions. Furthermore, edge-based processing is intended to reduce the dependence on continuous cloud communication and support low-latency analysis near the data source.

The effectiveness of the proposed framework should be established through systematic experimental evaluation using a representative and appropriately annotated rice-crop image dataset. Performance should be quantitatively assessed using established evaluation metrics, including **precision, recall, F1-score, mAP, and inference time**. Additional evaluation under varying illumination, environmental conditions, crop growth stages, disease severity levels, and rice varieties would provide further evidence regarding the robustness and generalizability of the proposed approach.

The proposed framework provides a foundation for developing scalable and intelligent precision-agriculture applications for rice crop monitoring. However, its practical deployment requires validation under real-world field conditions and consideration of computational, environmental, and dataset-related constraints. **Future research may explore multispectral and hyperspectral imagery, lightweight and more efficient neural network architectures, model pruning and quantization, federated learning, and integration with autonomous agricultural intervention systems.** Such extensions could further improve the efficiency, scalability, privacy, and applicability of UAV-assisted crop-health monitoring systems.

Overall, the proposed UAV-based edge vision architecture establishes a framework for integrating **aerial sensing, artificial intelligence, and edge computing** for

automated rice crop-health assessment, while providing a basis for further experimental investigation and real-world agricultural deployment.

REFERENCES

- [1] H. Zhu, C. Lin, G. Liu, D. Wang, S. Qin, A. Li, J.-L. Xu, and Y. He, "Intelligent agriculture: deep learning in UAV-based remote sensing imagery for crop diseases and pests detection," *Frontiers in Plant Science*, vol. 15, 2024, Art. no. 1435016. doi: 10.3389/fpls.2024.1435016.
- [2] T. B. Shahi, C.-Y. Xu, A. Neupane, and W. Guo, "Recent Advances in Crop Disease Detection Using UAV and Deep Learning Techniques," *Remote Sensing*, vol. 15, no. 9, 2023, Art. no. 2450. doi: 10.3390/rs15092450.
- [3] G. Zhao, Y. Lan, Y. Zhang, and J. Deng, "Rice Canopy Disease and Pest Identification Based on Improved YOLOv5 and UAV Images," *Sensors*, vol. 25, no. 13, 2025, Art. no. 4072. doi: 10.3390/s25134072.
- [4] S. K. Bhoi, K. K. Jena, S. K. Panda, H. V. Long, R. Kumar, P. Subbulakshmi, and H. B. Jebreen, "An Internet of Things assisted Unmanned Aerial Vehicle based artificial intelligence model for rice pest detection," *Microprocessors and Microsystems*, vol. 80, 2021, Art. no. 103607. doi: 10.1016/j.micpro.2020.103607.
- [5] T. Zhou and L. Wei, "YOLO-DP: A detection model of fifteen common rice diseases and pests," *Scientific Reports*, vol. 15, 2025, Art. no. 35968. doi: 10.1038/s41598-025-19310-1.
- [6] I. Anam, N. Arafat, M. S. Hafiz, J. R. Jim, M. M. Kabir, and M. F. Mridha, "A systematic review of UAV and AI integration for targeted disease detection, weed management, and pest control in precision agriculture," *Smart Agricultural Technology*, vol. 9, 2024, Art. no. 100647. doi: 10.1016/j.atech.2024.100647.
- [7] S. Sapna, H. Gauttam, V. Chauhan, K. K. Pattanaik, A. Trivedi, and H. Ghosh, "A comprehensive review of Edge Computing empowered smart agriculture: Trends, Opportunities and Future directions," *Computers and Electronics in Agriculture*, vol. 241, 2026, Art. no. 111252. doi: 10.1016/j.compag.2025.111252.
- [8] "Edge Computing-Enabled Smart Agriculture: Technical Architectures, Practical Evolution, and Bottleneck Breakthroughs," *Sensors*, vol. 25, no. 17, 2025, Art. no. 5302. doi: 10.3390/s25175302.