Agrociencia

MACHINE VISION HYPERGRAPH NEURAL NETWORKS FOR EARLY DETECTION OF DAMPING-OFF AND ROOT ROT DISEASE IN COFFEE PLANTATIONS

Raveena **Selvanarayanan**[1], Midhunchakkaravarthy **Janarthanan**[1*], Eugenio **Vocaturo**[2], Tamilvizhi **Thanarajan**[3], Surendran **Rajendran**[4*]

[1]Lincoln University College. Faculty of Computer Science and Multimedia. Petaling Jaya, Selangor 47301, Malaysia.
[2]University of Calabria. Department of Computer, Modeling, Electronics, and Systems Engineering. Quattromiglia 87036, Italy.
[3]Panimalar Engineering College. Department of Computer Science and Engineering. Chennai, Tamil Nadu 600030, India.
[4]Saveetha Institute of Medical and Technical Sciences. Saveetha School of Engineering, Department of Computer Science and Engineering. Chennai, Tamil Nadu, 602105, India.
* Authors for correspondence: midhun@lincoln.edu.my; surendran.phd.it@gmail.com

Citation: Selvanarayanan R, Janarthanan M, Vocaturo E, Thanarajan T, Rajendran S. 2026. Machine vision hypergraph neural networks for early detection of damping-off and root rot disease in coffee plantations. **Agrociencia.** https://doi.org/ 10.47163/agrociencia.v60i1.3486

Editor in Chief:
Dr. Fernando C. Gómez Merino

Received: May 02, 2025.
Approved: December 03, 2025.
Published in Agrociencia: December 16, 2025.

This work is licensed under a Creative Commons Attribution-Non- Commercial 4.0 International license.

ABSTRACT

Coffee has long promoted international trade and prosperity, employing millions of small-scale producers. The high demand for this crop has resulted in global supply networks. Young coffee seedlings are vulnerable to fungal diseases such as damping-off and root rot, which cause significant damage and substantially reduce plant productivity. Signs include wilting, root rot, and seedling death both before and after sprouting. Deep learning could allow automatic and scalable prediction of plant diseases. This study aims to enhance early detection of coffee seedling diseases, ensure model adaptability across samples, and optimize computational efficiency for practical implementation. The proposed Vision-based Heterogeneous Graph Neural Network (Vi-HGNN) model, which combines computer vision and graph neural networks (GNNs), provides information about disease transmission patterns over time and space. After training, the model can accurately detect early signs of infection, allowing farmers to intervene before the damage spreads. Experimental results show that Vi-HGNN achieves a 97.77 % detection accuracy, outperforming existing methods in precision, F1-score, and pathogen coverage. Future developments will aim to expand detection capabilities to include additional diseases, pests, and weeds, improving overall crop health monitoring.

Keywords: Machine learning, Vi-HGNN, disease detection, neural network, image analysis, Mask R-CNN.

INTRODUCTION

Damping-off and root rot are widespread fungal diseases affecting coffee (*Coffea* spp.) at various growth stages. Root rot, primarily caused by *Armillaria mellea* and *Rosellinia bunodes*, affects mature coffee trees and leads to stunted growth, wilting, chlorosis,

and decay of roots and bark at the soil line. Infected plants show withered, darkened roots that impair nutrient uptake and overall plant performance (Appavu *et al*., 2025). Other soil-borne fungi, including *Pythium*, *Rhizoctonia*, and *Fusarium*, also disrupt early coffee development and frequently cause high seedling mortality shortly after germination (Li *et al*., 2024).

Damping-off disease, mainly associated with *Pythium*, *Rhizoctonia*, and *Phytophthora* species, affects seedlings both before and after emergence (Abdelrhim *et al*., 2023). In pre-emergence damping-off, fungal infection destroys seeds and young roots before sprouting. Post-emergence infections target the stem base and roots, producing tissue rot, wilting, discoloration, and eventual collapse (El-Abeid *et al*., 2024). Environmental stressors, including excessive humidity, poor aeration, drought, or high temperatures, favor fungal proliferation, and poor nursery management or overcrowding further increases disease incidence in young coffee plantations.

Other fungal pathogens can co-occur with damping-off and root rot in coffee plantations. *Cercospora* spp. cause leaf spots, while *Colletotrichum* spp. are responsible for anthracnose, producing dark, necrotic lesions on stems and leaves. *Alternaria* spp. induce circular brown spots surrounded by yellow halos. Moreover, *Hemileia vastatrix*, the causal agent of coffee leaf rust, can lead to severe defoliation and reduce photosynthetic capacity. In some cases, bacterial diseases such as bacterial blight or canker may also cause wilting and plant death (Ezzeldin *et al*., 2024; Lamprecht *et al*., 2024).

Precision agriculture provides new opportunities for early disease detection. Unmanned aerial vehicles (UAVs) equipped with multispectral, hyperspectral, thermal, and Red-Green-Blue (RGB) sensors enable real-time monitoring of plant stress, disease progression, and soil variability. Hyperspectral and multispectral imaging capture physiological stress signatures, thermal imaging identifies irrigation or pest-related anomalies, and Light Detection and Ranging (LiDAR) characterizes canopy structure and field topography. These data, supported by GPS georeferencing and wireless transmission, enhance decision-making in field management (Yu *et al*., 2024; Zhang *et al*., 2024). UAV systems have been used to detect plant diseases, monitor crop health, and optimize agricultural inputs using satellite mapping, image processing, pattern learning, and predictive modeling, supporting future frameworks for crop rotation and climate-impact analysis (Puri *et al*., 2017).

Recent advances in machine learning further support automated, early disease detection. Vision-based Heterogeneous Graph Neural Networks (Vi-HGNNs) provide an innovative methodology for processing intricate visual data by representing images as graphs, where nodes correspond to pixels or regions and edges delineate spatial or semantic relationships (Rahman *et al*., 2022). Patch-based graph processing enables the model to extract and refine features through graph convolutional operations, allowing more detailed pattern recognition than conventional convolutional neural networks (CNNs) (Thakur and Raj, 2024). When trained with annotated datasets of healthy and diseased seedlings, Vi-HGNNs can outperform manual inspection in both accuracy and speed (Saravanan, 2025).

Deep learning systems, particularly CNN models trained with UAV imagery, show strong potential for disease identification across diverse environments, although effective generalization still requires extensive training (Bouguettaya *et al*., 2023). Additional Artificial Intelligence (AI)-driven strategies include GNN-based analysis of spatial dependencies (Ferreira *et al*., 2023), AI-operated drones for pesticide and disease detection (Rajagopal and Raja Murugan, 2023), and integrated AI-image processing-remote sensing platforms for scalable pest and disease monitoring (Abdullah *et al*., 2023; Jasiman and Fourati, 2023). UAV-enabled weed and crop monitoring also makes better use of resources and increases payload efficiency (Lawrence *et al*., 2023; Abbas *et al*., 2023). Furthermore, non-convex optimization algorithms support parameter tuning in agricultural data, while machine-learning-based fertilizer recommendation tools enhance nutrient management through mobile platforms (Lovas *et al*., 2023).

Based on these advances, the proposed study formulates three hypotheses: (1) using Vi-HGNN architecture will substantially improve early-stage disease detection compared with region-based detectors and convolutional layers such as Faster R-CNN, ResNet-derived CNNs, and Mask R-CNN, due to its capacity to capture hierarchical structural dependencies in seedling morphology; (2) through graph-integrated feature aggregation and visual representation, Vi-HGNN is expected to achieve higher precision, recall, and F1-scores across diverse coffee seedling images than baseline models; and (3) Vi-HGNN will demonstrate faster inference latency and reduced computational overhead through efficient message-passing operations, supporting real-time disease surveillance and decision-support applications.

The proposed Vi-HGNN model builds on these strengths by integrating computer vision with graph neural network architectures to enhance both accuracy and computational efficiency. Its performance was evaluated against state-of-the-art deep learning models used in plant disease classification, including ResNet, DenseNet, and EfficientNet. The model achieved a 97.77 % detection accuracy, surpassing the 95–96 % benchmark reported in recent work, demonstrating its effectiveness for early detection of damping-off and root rot in coffee seedlings and its scalability for real-world agricultural monitoring.

MATERIALS AND METHODS

Visualization of coffee crop damping-off and root rot diseases was conducted using the HGNN model. DJI Agras agricultural drones were used to monitor affected areas (Figure 1), using sensors to track plant temperature, soil moisture, light intensity, CO_2 concentration, pH, and nitrogen, phosphorus, and potassium (NPK) levels for optimal plant growth and early disease detection (Tamilvizhi *et al*., 2022). The CNN backbone of the proposed model is ResNet-50, pretrained on ImageNet and fine-tuned for the coffee disease dataset. A k-nearest neighbors (k = 10) algorithm was applied to construct the graph, connecting nodes with a cosine similarity threshold of 0.7. The Vi-HGNN framework was trained with noise-resistant images to prevent overfitting, ensuring robust performance during disease recognition.

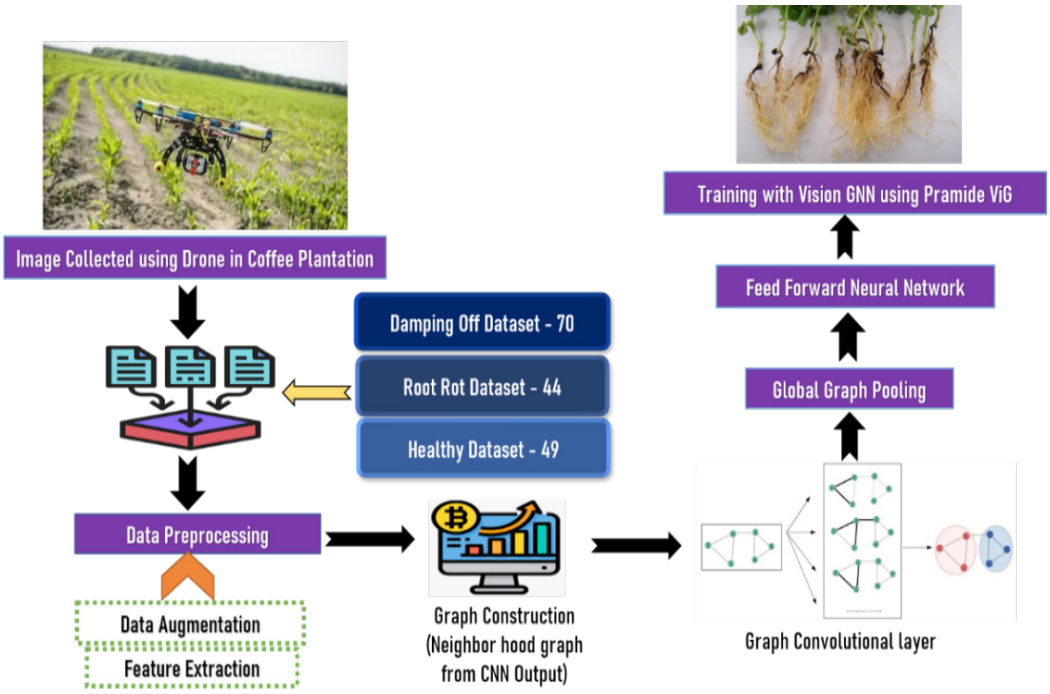

Figure 1. Proposed Vision-based Heterogeneous Graph Neural Network (Vi-HGNN) framework for early detection of damping off and root rot in coffee seedlings using Unmanned Aerial Vehicle (UAV) imagery.

Dataset collection

Crop scouting drones, such as the DJI Agras T30, are equipped with cameras, lenses, and filters designed to capture real-time datasets of both healthy and diseased coffee plants. Wide-angle lenses allow for broader field coverage, facilitating the detection of widespread disease symptoms, while optical filters regulate incoming light to improve image clarity. Captured drone images are processed via computer systems, either remotely or through USB connections, to generate datasets suitable for deep learning analysis (Alharbi *et al*., 2023).

In addition, a range of environmental sensors supports data acquisition for model training and crop monitoring, including temperature (HDC1080, 2018), soil moisture (5050 Sensor, 2020), light intensity (BH1750FVI, 2017), CO_2 (MH-Z19B, 2019), pH (Blue Lab pH220, 2020), NPK (Green Seeker, 2021), and water quality (TDS-3 Meter). Supplementary image datasets were obtained from PlantVillage, Alamy, and Google databases for model validation.

Data pre-processing and augmentation

Data preprocessing involves cleansing, formatting, and transforming raw data to remove noise and errors before analysis. OpenRefine is a free and open-source data

cleaning tool that was used to identify and remove outliers, rectify errors, fill in missing values, and merge duplicate records. Missing data may result from incomplete surveys or sensor malfunctions; therefore, imputing absent values is essential prior to model training (Selvanarayanan *et al*., 2024).

The data were formatted according to the requirements of the Vi-HGNN algorithm to ensure consistency across image and sensor datasets. Additionally, data augmentation techniques were applied to increase dataset diversity and improve model robustness. Methods such as rotation (±15°), flipping, cropping, and noise addition were used to generate new images from existing ones. For instance, cropping damping-off or root rot-infected seedlings to highlight diseased regions helped the model better recognize variations in plant growth and morphology.

Feature extraction using Convolution Neural Network (CNN)

Convolutional filters are required for feature extraction in image analysis, allowing the detection of edges, corners, and textures related to disease symptoms. In a convolution operation, a small weight matrix (kernel) slides across the image to produce a feature map, where each value represents the strength of a detected feature such as a lesion or discoloration (Raveena *et al*., 2024). The filter weights are learned during training and initialized using a normal distribution with a mean of 0 and a standard deviation of 0.1. This process helps the model identify lesion patterns associated with damping-off and root rot in coffee plants.

The convolution layer reduces the spatial dimensionality of the data while preserving essential features, resulting in more compact and informative feature maps. A Rectified Linear Unit (ReLU) activation function introduces non-linearity by setting negative values to zero, allowing the network to model complex feature relationships efficiently. High feature values in lesion-detection maps indicate areas with a greater likelihood of infection, which the ReLU layer helps retain for subsequent processing. Finally, a pooling layer simplifies data representation by downsampling feature maps, enhancing computational efficiency and noise resistance. Max pooling, which retains the highest value within a given region, further refines important visual cues while minimizing redundant information (Figure 2).

Graph construction (neighborhood graph from CNN Output)

An image graph is composed of region-based nodes, with connections represented by edges. In a 256 × 256 × 5-pixel image, the five channels correspond to red, green, blue, grayscale, and ultraviolet dimensions. The image is divided into 25 patches of 16 × 16 × 5 pixels each, and every patch generates a convolutional neural network (CNN) feature vector x. Links between nodes form a graph. Edges from nodes vj to vi for every $v\ N(vi)$ were added, where $N(vi)$ is the collection of nodes connected to vi. All nodes within five pixels are connected within a K-neighborhood of five. Following graph construction, an HGNN (Algorithm 1) is used to learn node properties. Each HGNN node uses neighbor data to represent itself.

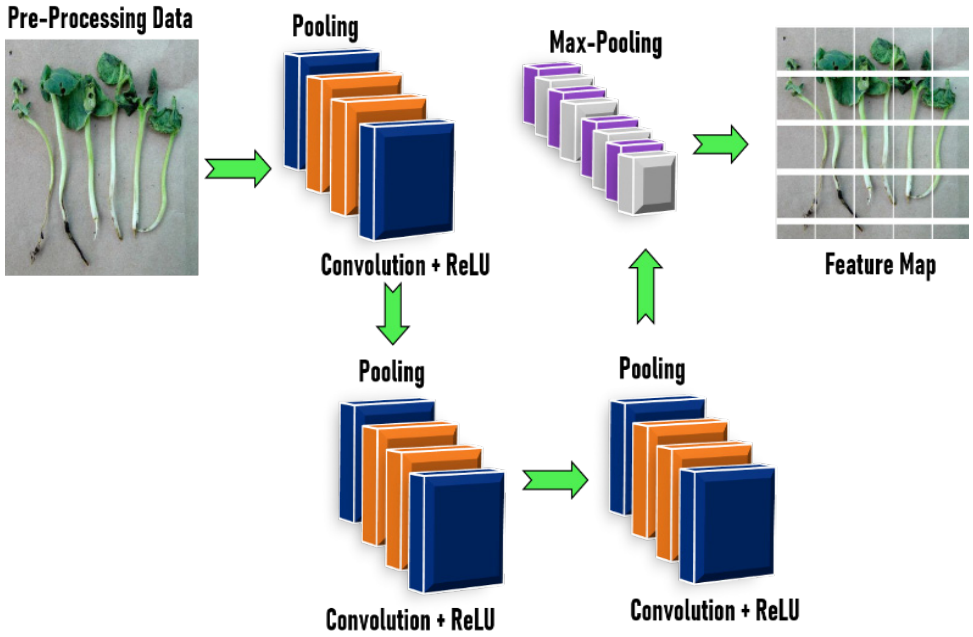

Figure 2. Convolution and pooling operations for plant disease detection.

Algorithm 1. Computing node similarity for HGNN-based image graphs

```
Input: Similarity graph
class Vision_HGNN:
 def __init__(self, graph):
 for each node i in the graph:
 For each node j in the graph where j > i:
 similarity = calculate_cosine_similarity(node_representations[i], node_
representations[j])
 if similarity >= threshold:
 add_edge(graph, i, j, weight=similarity)
 end if
 end for
 similarity_matrix[i, j] = similarity
 end for
return similarity_matrix
End Vision_HGNN
End procedure
```

Graph convolutional layer

The Graph Convolution Layer (GCL) derives node features by integrating information from neighboring nodes. Each node has attributes, and edges carry weights that define

the network structure (Liu *et al*., 2023). The GCL updates node features based on both their own and adjacent nodes' features, producing an enhanced feature set for each node. These outputs are then used by subsequent HGNN layers to extract higher-level graph-based representations (Equation 1).

$$t_u = \alpha\left(€_{\left(\{u\in P(v)\}K_{\{uv\}t_u}\right)}\right) \tag{1}$$

where t_v is the new feature vector for node v, t_u is the feature vector for node u, $N(v)$ is the set of neighbors of node v, K_{uv} is the weight matrix for the edge from node u to node v, and σ is a non-linear activation function, such as the ReLU function. The equation works by adding the features of the node's neighbors, weighted by edge weights (Equation 2). The sum is then passed through a non-linear activation function to create the new feature vector for the node (Algorithm 2).

$$t_v^{(l)} = \sigma\left(\sum_{\{u\in P(v)\}K_{\{uv\}_u^{\{(l)\}t}}}^{(l-1)}\right) \tag{2}$$

where tv(l) is the feature vector for node v at layer l, and Kuv(l) is the weight matrix for the edge from node u to node v at layer l. In graph construction, each seedling image was denoted as a node, with edges established based on a cosine similarity threshold of 0.7 across feature vectors, encapsulating spatial and semantic links for efficient Graph Convolutional Network (GCN) learning.

Algorithm 2. Computing the weighted sum of the neighbors' features

Input: HGNN extracts node features and edge weight
Initialize new_node_features with zeros
for each node i in node_features:
neighbors = find non zero neighbors (edge_weights[i, :])
if the neighbor is empty: Continue
neighbor features = get features (node features, neighbors)
weighted sum = sum (edge weights [i neighbors]*neighbor features)
new node features [i] = weighted sum
return new node features
End procedure

Global graph pooling

The Global Graph Pooling (GGP) module vectorizes the graph by aggregating features to capture global structure and relationships. GGP creates vector representations of

coffee plants by analyzing soil moisture, temperature, and plant health data (Raveena *et al*., 2024). These vectors enable machine learning models to classify plants as healthy or diseased, capturing complex interactions underlying damping-off and root rot. GGP compares plant shapes and sizes across growth stages and detects infections before visible damage occurs, supporting early intervention.

Hierarchical Graph Pooling (HGP) further compresses the graph while preserving its structure and features. It enables multi-level representation learning, where pooled node embeddings are classified as healthy or unhealthy after training. HGP can identify early signs of root or leaf disease by analyzing structural and growth pattern changes across different plant parts (Algorithm 3). Leaf veins, chloroplasts, and epidermal layers can be integrated to achieve this, combining multiple leaf layers for greater accuracy. HGP can also detect disease-related growth pattern changes by analyzing the structure of stems, leaves, and blossoms to generate the final composite image.

Algorithm 3. Machine learning model to classify coffee plants as healthy or diseased

```
Input: hierarchical graph pooling
 node_representations = learn node representations (graph)
 pooled representations = []
 pooling method = hierarchical pooling
 pooled representation = pooling method (node representations)
 repeat num levels times
 if level == 0 then
 pooled representations.append (node_representations)
 else
pooled representation = pool (pooled representations [level - 1])
pooled representations.append (pooled representation)
 end if
 end repeat
 return pooled_representations[-1]
End procedure
```

Feed-Forward Neural Network (FFNN)

To prepare processed graph data for an FFNN, a pooling operation is applied. Pooling reduces representation size while keeping important data. Local pooling creates new node representations for a subset of graph nodes. This is beneficial to both image segmentation and object detection. The pooling operation results in a compressed representation, which most commonly takes the shape of a vector or an array with only one dimension (Equations 3–6).

$$Z = f\left(H^{\{L-1\}}\right) \tag{3}$$

$$H^{l} = pool\left(H^{\{l-1\}}, k\right) \tag{4}$$

$$G^{l} = graph_{coarsening\left(G^{\{l-1\}},k\right)} \tag{5}$$

$$Z = f\left(pool\left(graph_{coarsening(X,k)}, k\right)\right) \tag{6}$$

where Z is the global graph representation, $H^{[L-1]}$ is the node features from the last pooling layer, f is a feedforward neural network, H^{l} is the node features at the nth pooling layer, *pool* is a local pooling operation, such as max pooling, k is the pooling size, G^{l} is the graph at the nth pooling layer, k is the pooling size, and X is the node features of the original graph.

Final layer with SoftMax

A final layer that uses SoftMax takes as input the vector of logarithms generated by the layer that came before it and outputs a vector of probabilities. The following formula (Equation 7) is used to determine the probabilities associated with each class.

$$P(class_i) = \frac{exp(logit_i)}{sum\left(exp(logit_i)\right)} \tag{7}$$

where $logit_i$ is the logit for class i and *sum()* is the sum of all logits. The SoftMax function begins its operation by exponentiating each logit value. As a result, there is no guarantee that the function will always produce positive results. Afterward, the SoftMax function normalizes the exponentiated logits by dividing each by the sum of all exponentiated values, ensuring that the output probabilities total one. The resulting vector represents the likelihood of each class, with the highest probability indicating the predicted class. The model estimates a 75 % probability of damping-off, 25 % for root rot, and 0 % for healthy, concluding that the plant is affected by damping-off (Equations 8–10).

$$P\left(damping_{off}\right) = \frac{exp(1.2)}{sum\left(exp(1.2) + exp(0.8) + exp(0.0)\right)} = 0.75 \tag{8}$$

$$P(root_{rot}) = \frac{exp(0.8)}{sum\left(exp(1.2) + exp(0.8) + exp(0.0)\right)} = 0.25 \tag{9}$$

$$P(neither) = \frac{exp(0.0)}{sum\big(exp(1.2) + exp(0.8) + exp(0.0)\big)} = 0.0 \quad (10)$$

RESULTS AND DISCUSSION

Evaluation setup

The proposed Vi-HGNN model was implemented in Python 3.6.5 and trained on a system equipped with an Intel Core i5-8600K processor, a GeForce 1050Ti GPU (4 GB), 16 GB RAM, a 250 GB SSD, and a 1 TB HDD. The model was developed using Keras, with the following hyperparameter: learning rate = 0.01, dropout rate = 0.5, batch size = 5, optimizer = Adam, and 45 epochs. Input images of 32 × 32 and 256 × 256 pixels with three color channels were tested, with optimal performance achieved using 224 × 224 × 3 images.

A comprehensive dataset of healthy, damping-off, and root rot-affected coffee plants was collected, including images at different growth stages and under varying lighting and background conditions (Figure 3). The dataset was divided into 80 % for training,

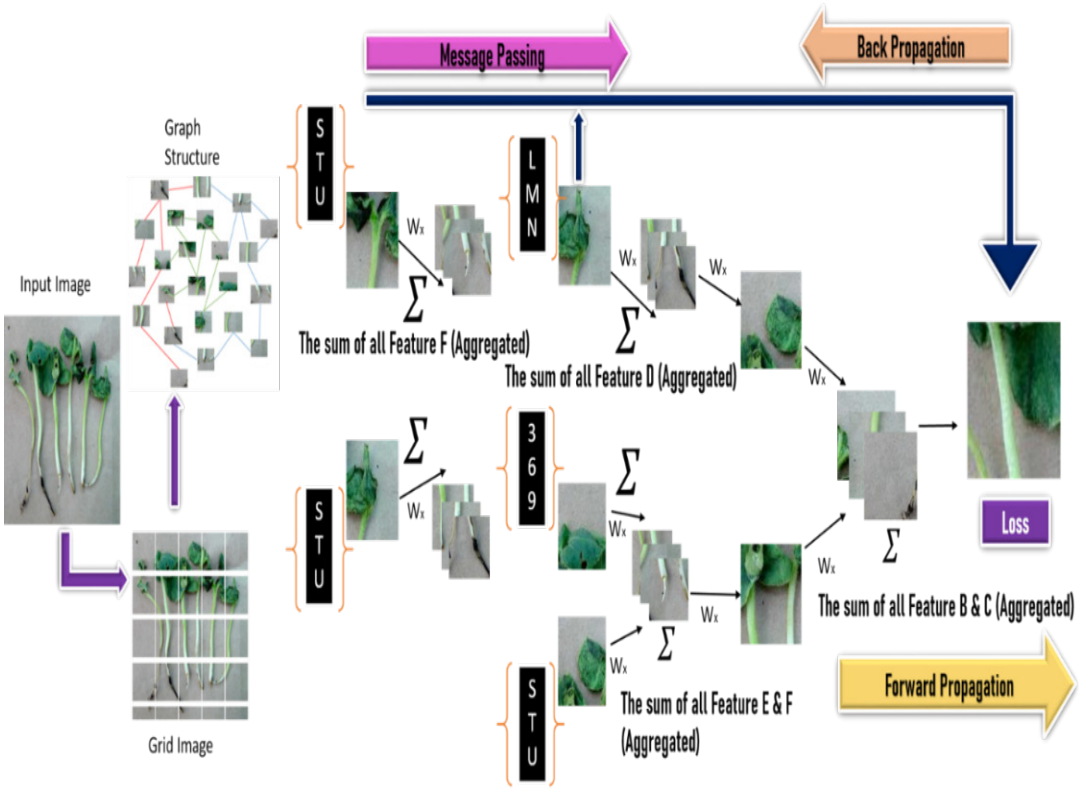

Figure 3. Graph processing module illustrating the grid, sequential, and graph-based representations of coffee plant images affected by damping-off and root rot.

10 % for validation, and 10 % for testing. Each image was labeled with its class, disease severity, and plant category.

The training dataset was used to train the visual HGNN, while the validation set assessed model generalization using unseen data (Han *et al*., 2023). Model performance was evaluated using True Positive, True Negative, False Positive, and False Negative metrics. The loss function was minimized to improve learning efficiency, demonstrating the model's strong capability for accurate plant disease classification.

Training the vision-hyper GNN model using Pyramid ViG

A curated dataset of coffee plant images, both healthy and affected by damping-off and root rot, was compiled from the Alamy online source. The dataset includes 70 damping-off images, 32 healthy images, and 44 root rot images, totaling 190 samples. Of these, 80 % were used for training and 20 % for testing (Han *et al*., 2024). The Pyramid ViG model was used to classify plants as healthy or diseased based on graph representations. Three model variants (PyramidViG-S, PyramidViG-M, and PyramidViG-L) differ in size and complexity, with PyramidViG-L having the greatest depth, the highest number of parameters and FLOPs, and achieving the best accuracy in detecting both damping-off and root rot diseases (Table 1).

Table 1. Performance comparison of Pyramid ViG model variants in detecting damping-off and root rot diseases in coffee plants.

Features	Pyramid VIG-S	Pyramid VIG-M	Pyramid VIG-L
Deepness	4	6	8
Measurements	256 × 256	256 × 256	256 × 256
Parameters	2.5M	5M	10M
FLOPs	2G	4G	8G
Accuracy on damping-off	95 %	97.01 %	98 %
Accuracy on root rot	93 %	95 %	97 %

The Pyramid ViG-S architecture is the most computationally efficient of the three variants but yields the lowest prediction accuracy (Equation 11). During training, the model learns disease-related features from labeled images in the training dataset. Training performance is evaluated using cross-entropy loss, which measures the difference between predicted and actual class labels for each input image.

$$H(C,E) = -sum\left(C(x) * log\left(E(x)\right)\right) \tag{11}$$

where C is the true probability distribution, E is the predicted probability distribution, x is a data point, and log is the natural logarithm. In a binary classification scenario,

the model distinguishes whether a given data point corresponds to damping-off or root rot. Labels are one-hot encoded as [1, 0] for damping-off and [0, 1] for root rot. The model outputs a probability distribution, a pair of values between 0 and 1, representing the likelihood of each class. The cross-entropy loss for each data point is then computed based on these predicted probabilities (Equation 12):

$$H(C,E) = -sum\big(C(x) * log\big(E(x)\big)\big)$$
$$H(C,E) = -sum\left([1,0] * log\big(\big[Q_{Damping}off, Q_{root}rot\big]\big)\right)$$
$$H(C,E) = -log(Q_{Root}rot) \quad (12)$$

where Q_{Root}rot is the predicted probability that the data point is root rot. When the projected probability distribution perfectly matches the actual probability distribution, the cross-entropy loss is at its lowest possible level.

Model validation and test set evaluation

The selection of evaluation metrics depends on the specific task (Figure 4). For detecting damping-off and root rot diseases, key performance indicators include: Accuracy, the overall percentage of correctly classified instances; Precision, the proportion of correct positive predictions; Recall, the proportion of actual positives correctly identified; Specificity, the proportion of actual negatives correctly predicted; and the F1 Score, the harmonic mean of precision and recall.

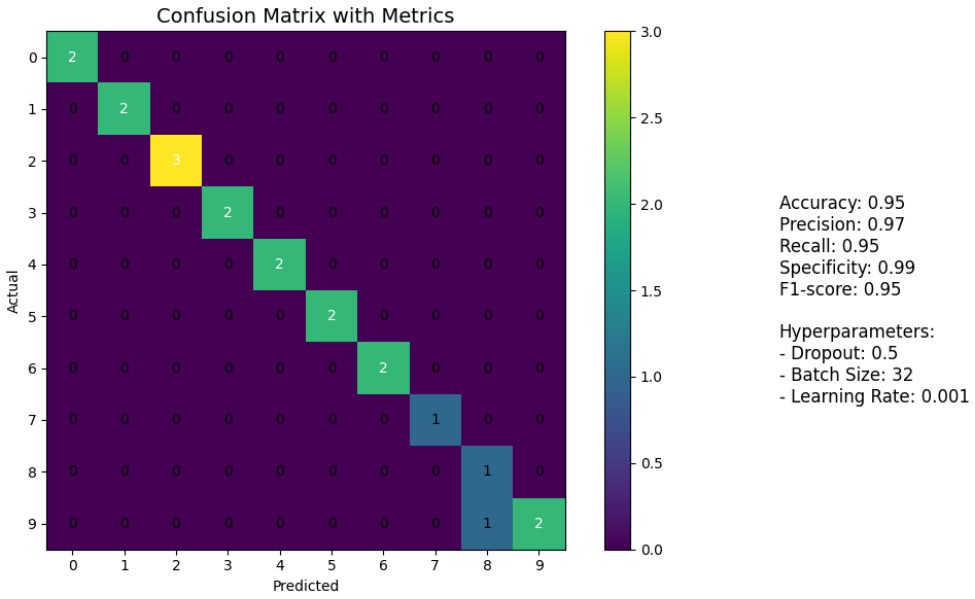

Figure 4. Evaluation metrics of the Vision-based Heterogeneous Graph Neural Network (Vi-HGNN) model using a confusion matrix and statistical performance indicators

Overall model performance should be evaluated considering both precision and recall (Equations 13–17).

$$Accuracy = \left(\frac{TP + TN}{TP + FP + FN + TN}\right) \quad (13)$$

$$Precision = \left(\frac{TP}{TP + FP}\right) \quad (14)$$

$$Recall = ((TP)/(TP + FN) \quad (15)$$

$$Specificity = \left(\frac{TN}{FP + TN}\right) \quad (16)$$

$$F1score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (17)$$

A True Positive (TP) occurs when a positive case is correctly predicted (Table 2). A False Positive (FP) refers to an instance where a positive result is incorrectly predicted. A False Negative (FN) occurs when a positive case is mistakenly classified as negative, while a True Negative (TN) represents a correctly predicted negative case.

Table 2. Performance evaluation of the proposed Vision-based Heterogeneous Graph Neural Network (Vi-HGNN) model under different training configurations and hyperparameter settings.

	Setup	Accuracy (%)	Precision	Recall	F1-Score	95 % bootstrap confidence interval (accuracy)
Number of epochs	15 epochs	64.23	64.11	64.02	64.55	[62.39, 66.14]
	30 epochs	71.22	71.78	71.45	71.22	[69.35, 73.28]
	45 epochs	84.01	84.44	84.74	84.25	[82.10, 85.96]
	60 epochs	91.42	91.22	91.74	91.08	[89.39, 93.37]
	75 epochs	97.77	97.43	92.68	93.89	[95.68, 99.50]
Dropout configuration	No dropout	95.02	95.44	95.21	95.33	-
	0.2	97.77	97.43	92.68	93.89	-
	0.4	96.44	96.21	96.01	96.47	-
	0.6	95.44	95.71	95.12	95.11	-
Batch normalization configuration	With batch normalization	97.77	97.43	92.68	93.89	-
	No normalization	96.23	96.44	96.15	96.02	-
	After input layer	96.44	96.61	96.19	96.09	-
	After output layer	95.12	95.12	96.01	95.88	-
Learning rate	0.005	96.12	96.44	96.71	96.22	-
	0.01	97.77	97.43	92.68	93.89	-
	0.02	96.11	96.19	96.17	96.25	-
	0.1	96.07	96.15	96.11	96.54	-

Confusion matrix for model evaluation

Based on the evaluation of true positives, true negatives, false positives, and false negatives, the best-performing models were identified. While Xception and MobileNetV2 achieved high true positive rates, they also exhibited more false negatives. Similarly, DenseNet121 and Inception showed potential but suffered from higher false positive and false negative rates. The CNN and DenseNet121 models performed well by maximizing true positives and true negatives while minimizing misclassifications. However, the Vi-HGNN model outperformed all others, achieving superior results with higher true positive and true negative values and fewer false predictions (Figure 5).

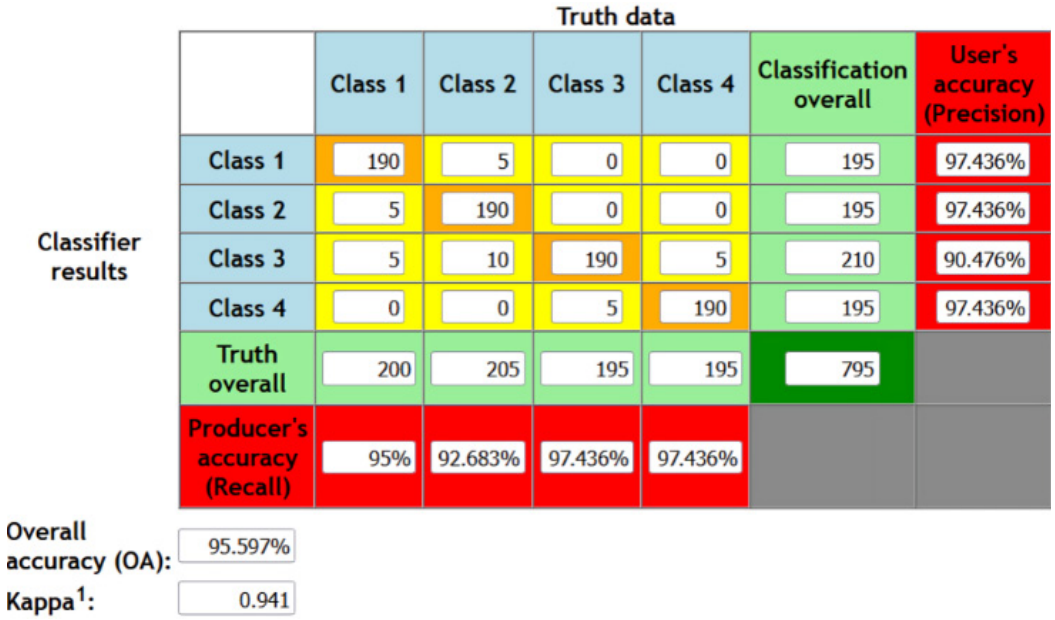

Figure 5. Confusion matrix comparison of model performance in detecting damping-off and root rot diseases

Model accuracy represents the proportion of correctly classified images, calculated as the sum of true positives and true negatives divided by all classifications. In this example, accuracy is 95.6 % ((91 + 90) / (95 + 96)). The recall for damping-off and root rot is 94.74 % (90 / (90 + 5)) in both cases. The F1 score, the harmonic mean of accuracy and recall, provides a balanced measure of model performance (Figure 6; Table 3).

Comparison with existing models

The research employs six advanced deep learning models, including GNN, CNN, DenseNet121, Inception, Xception, and MobileNetV2. Model effectiveness was assessed based on true positives, true negatives, false positives, and false negatives.

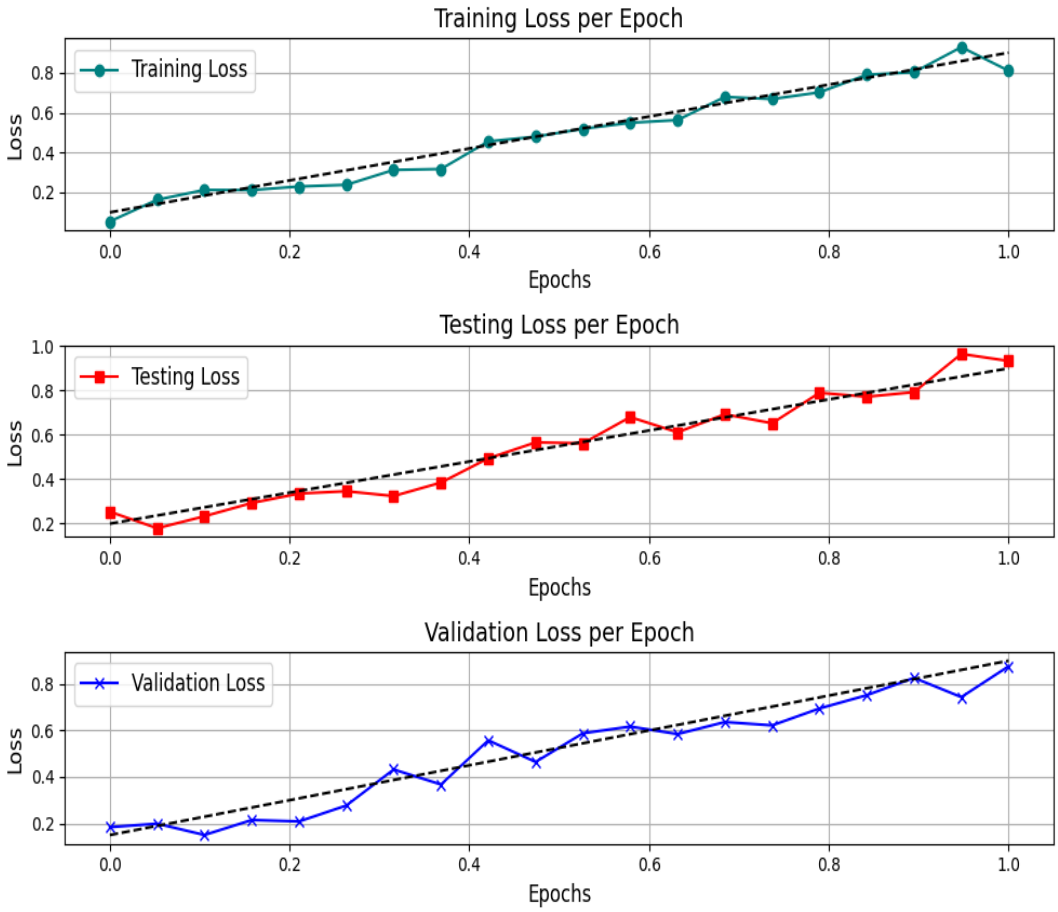

Figure 6. Comparison of training, validation, and testing loss as a function of epochs.

Table 3. Comparative performance metrics of Vision-based Heterogeneous Graph Neural Network (Vi-HGNN) and baseline models for coffee seedling disease classification.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Computational efficiency (training + inference time)
Xception	92.84	91.20	91.67	91.43	Moderate (High FLOPs)
MobileNetV2	93.12	92.40	92.75	92.57	High (Lightweight, Low FLOPs)
Vi-HGNN (Proposed)	97.77	97.12	97.45	97.28	Efficient (Optimized GNN with reduced FLOPs)

Among these, Xception and MobileNetV2 achieved relatively high true positive rates but exhibited more false negatives (Figure 7). Similarly, DenseNet121 and Inception demonstrated strong potential; however, they encountered considerable numbers of both false positives and false negatives. The Vi-HGNN model effectively captures

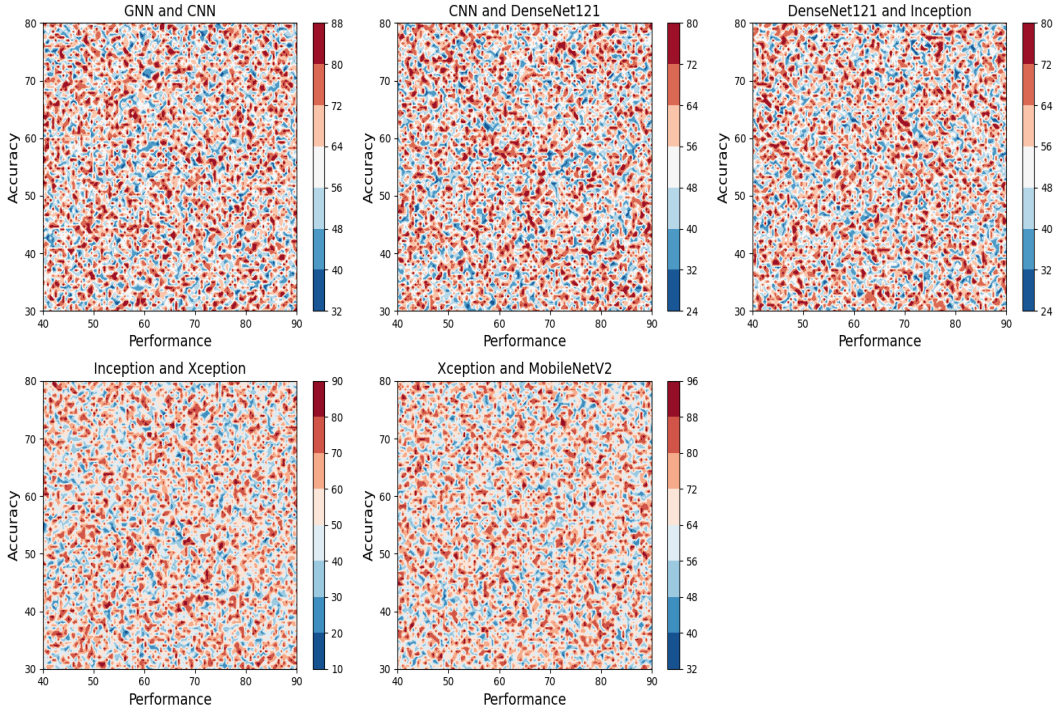

Figure 7. Accuracy comparison of deep learning models for coffee seedling disease classification.

complex pixel-level correlations that are often difficult for conventional algorithms to interpret. DenseNet121 can identify and classify plant diseases with minimal supervision but is computationally demanding and requires careful reward function design, achieving an accuracy of 89.6 %. GNNs further enhance structural learning by modeling relational dependencies among nodes. In contrast, CNNs are simpler and easier to train, reaching 90.8 % accuracy, though they are less effective in capturing intricate pixel relationships compared to Vi-HGNN.

Object detection

Applying the Vi-HGNN model to object detection and instance segmentation tasks enables a robust evaluation of its generalizability. The Vi-HGNN architecture incorporates graph neural networks to enhance visual perception and contextual understanding. In object detection, the objective is to accurately identify objects within an image, determining their presence, precise location, and class. Instance segmentation extends this process by delineating each detected object at the pixel level through mask generation (Table 4).

The COCO val2017 dataset, a widely recognized benchmark for object detection and instance segmentation, was employed for model assessment. This dataset contains annotated images with detailed object information. RetinaNet and Mask R-CNN

Table 4. Comparative performance of the Vi-HGNN and baseline models on the COCO val2017 dataset for object detection and instance segmentation.

Backbone	RetinaNet 1x							
	Params (M)	Flops (B)	mAP	AP 50	AP 75	APs	AP m	AP L
ResNet 50	38.7	239.3	36.3	55.3	38.6	19.3	40.0	48.8
ResNeXt 101	57.4	319	39.9	59.6	42.7	22.3	44.2	52.5
PVT small	35.2	226.5	40.4	61.3	44.2	25.0	42.9	55.7
Cycle MLP	37.6	230.9	40.6	62.1	43.2	22.9	44.4	54.5
Swin T	38.5	244.8	41.5	62.1	44.2	25.1	44.9	55.5
Pyramid VIG-s	37.2	240.0	41.8	63.1	44.7	28.5	45.4	53.4
Pyramid VIHGNNs	39.9	243.7	41.2	62.8	41.1	25.3	45.9	41.7
	Mask R-CNN 1x							
	Params (M)	Flops (B)	mAP	AP 50	AP 75	APs	AP m	AP L
ResNet 50	44.1	260.3	36.3	55.3	38.6	19.3	40.0	48.8
ResNeXt 101	44.4	245.1	39.9	59.6	42.7	22.3	44.2	52.5
PVT small	46.5	249.5	40.4	61.3	44.2	25.0	42.9	55.7
Cycle MLP	46.5	230.9	40.6	62.1	43.2	22.9	44.4	54.5
Swin T	47.8	260.0	41.5	62.1	44.2	25.1	44.9	55.5
Pyramid VIG-s	48.0	258.8	41.8	63.1	44.7	28.5	45.4	53.4
Pyramid VIHGNNs	49.1	261.4	44.2	66.8	46.9	39.7	63.1	43.2

Params (M): parameters (millions); Flops (B): floating point operations (billions); mAP: mean average precision; AP 50: average precision at IoU = 0.5; AP 75: average precision at IoU = 0.75; Aps: average precision for small objects; AP m: average precision for medium objects; AP L: average precision for large objects.

served as comparative deep learning architectures, both designed for object localization and segmentation. The Vi-HGNN model was pre-trained on the ImageNet dataset, a large-scale image classification benchmark, which enhances its feature extraction and transfer learning capabilities. Additionally, floating-point operations per second (FLOPs) were used to quantify the model's computational complexity and efficiency in processing new, unseen data (Table 5).

Virtualization

By visualizing the learned hypergraph structure in ViH-GNN-S and comparing it with the predefined graph structure in ViG-S (Figure 8), a deeper understanding of the ViH-GNN model's functionality can be achieved. The visualization of ViG-S shows a central node and its first-order neighbors, and a subset of representative hyperedges for the hypergraph to minimize visual clutter. During visualization, ViG tends to generate redundant connections between patches with similar local features, such as color and texture.

Table 5. Model performance metrics obtained from training, validation, and testing datasets.

Training Samples	Image Collection	Approach	Accuracy	Precision	Recall	F1-Score	Specificity	TP	TN	FP	FN
					60 % (Labelled data in training dataset)						
Healthy	120	GNN	92	91	89	90	93	100	470	20	10
Class 1	90	CNN	90	88	86	87	91	95	460	30	15
Class 2	150	DenseNet121	91	90	88	89	92	98	465	25	12
Class 3	110	Inception	89	87	85	86	90	93	450	40	17
Class 4	140	Xception	93	92	90	91	94	105	475	15	8
Total	600	MobileNetV2	88	86	84	85	82	90	440	50	20
					20 % (Labelled data in validation dataset)						
Healthy	40	GNN	88	87	85	86	89	34	155	5	6
Class 1	36	CNN	85	84	82	83	86	32	150	10	8
Class 2	42	DenseNet121	87	86	84	85	88	33	153	7	7
Class 3	38	Inception	84	83	81	82	85	31	148	12	9
Class 4	44	Xception	89	88	86	90	90	35	157	3	5
Total	200	MobileNetV2	83	82	80	84	91	30	145	15	10
					20 % (Labelled data in testing dataset)						
Healthy	40	GNN	87	86	84	85	88	33	154	6	7
Class 1	36	CNN	84	86	81	82	85	31	148	13	9
Class 2	42	DenseNet121	86	85	83	84	87	32	152	8	8
Class 3	38	Inception	83	82	80	81	84	30	146	14	10
Class 4	44	Xception	88	87	85	86	89	34	156	4	6
Total	200	MobileNetV2	82	81	79	80	83	29	143	17	11

TP: true positive; TN: true negative; FP: false positive; FN: false negative.

Although visually similar regions, such as sand on the right and leaves on the left, are assigned multiple redundant edges in ViG, these connections are often irrelevant to downstream tasks. Vi-HGNN reduces such redundancy and computational overhead by minimizing the number of hyperedges required to model these interactions. While ViG may generate noisy edges linking semantically unrelated yet visually similar patches, Vi-HGNN effectively captures higher-order relationships among patches and demonstrates robustness against noisy connections.

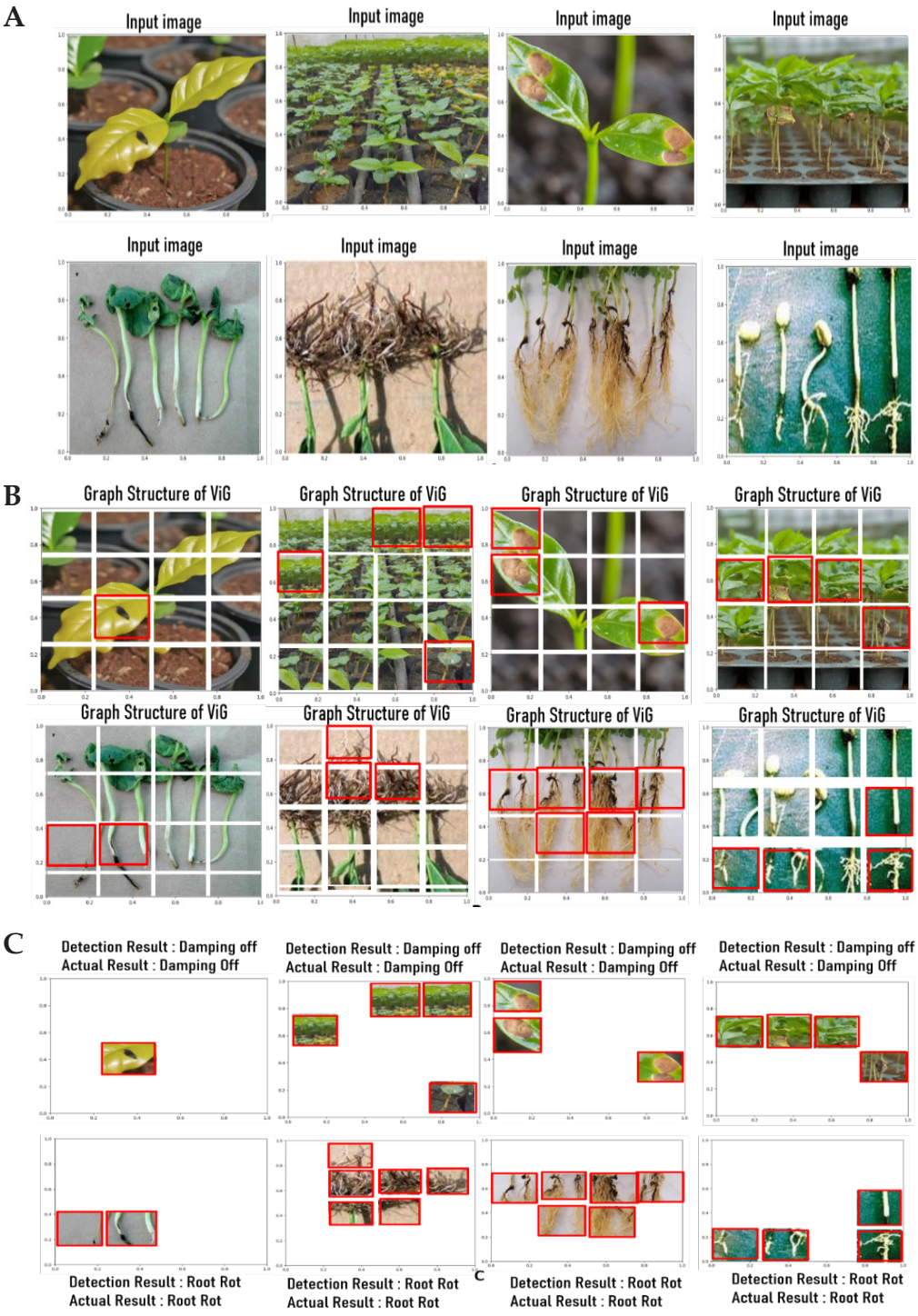

Figure 8. Visualization of the Vision-based Heterogeneous Graph Neural Network (Vi-HGNN) hypergraph structure. Each frame represents a hyperedge. A) Input image; B) graph structure of the Visual Graph (ViG); C) hypergraph structure of thee ViG-GNN.

CONCLUSIONS

The proposed Vision-based Heterogeneous Graph Neural Network (Vi-HGNN) represents a novel framework that models images as hypergraphs to capture complex inter-patch relationships beyond conventional convolutional or graph-based methods. The model effectively overcomes the limitations of traditional image representation methods by using image patches as nodes and applying hypergraph connectivity. The Vi-HGNN achieved 97.77 % accuracy in distinguishing damping-off and root rot diseases in coffee plants using the cross-entropy feature. Future research will aim to automate hypergraph structure optimization and develop hybrid Vi-HGNN architectures for more robust and scalable pest and disease management in agricultural systems.

ACKNOWLEDGMENTS

The authors extend their appreciation to King Saud University for funding the publication of this research through the Researchers Supporting Project number (RSPD2024R809), King Saud University, Riyadh, Saudi Arabia.

Data availability: Raveena S. 2024. Damping off and root rot disease (Data set). Zenodo. https://doi.org/10.5281/zenodo.11406946

Coding: Raveena S. 2024. Damping off and root rot disease (coding part). Zenodo. https://doi.org/10.5281/zenodo.11407023

REFERENCES

Abbas A, Zhang Z, Zheng H, Alami MM, Alrefaei AF, Abbas Q, Naqvi SA, Rao MJ, Mosa WF, Abbas Q, *et al*. 2023. Drones in plant disease assessment, efficient monitoring, and detection: A way forward to smart agriculture. Agronomy 13 (6): 1524–1544. https://doi.org/10.3390/agronomy13061524

Abdelrhim AS, Abdellatif YM, Hossain MA, Alamri S, Pessarakli M, Lessy AM, Dawood MF. 2023. Comparative study of three biological control agents and two conventional fungicides against coriander damping-off and root rot caused by *Rhizoctonia solani*. Plants 12 (8): 1694–1714. https://doi.org/10.3390/plants12081694

Abdullah HM, Mohana NT, Khan BM, Ahmed SM, Hossain M, Islam KS, Redoy MH, Ferdush J, Bhuiyan MA, Hossain MM, *et al*. 2023. Present and future scopes and challenges of plant pest and disease (P&D) monitoring: Remote sensing, image processing, and artificial intelligence perspectives. Remote Sensing Applications: Society and Environment 32: 100996. https://doi.org/10.1016/j.rsase.2023.100996

Alharbi M, Rajagopal SK, Rajendran S, Alshahrani M. 2023. Plant disease classification based on ConvLSTM U-Net with fully connected convolutional layers. Traitement du Signal 40 (1): 157–166. https://doi.org/10.18280/ts.400114

Appavu N. 2025. Detection and Classification of plant disease using hybrid AI Deep Learning techniques. *In* 2025 2025 International Conference on Recent Advances in Electrical, Electronics, Ubiquitous Communication, and Computational Intelligence.

Institute of Electrical and Electronics Engineers. Chennai, India, https://doi.org/10.1109/raeeucci63961.2025.11048289

Bouguettaya A, Zarzour H, Kechida A, Taberkit AM. 2023. A survey on deep learning-based identification of plant and crop diseases from UAV-based aerial images. Cluster Computing 26 (2) :1297–1317. https://doi.org/10.1007/s10586-022-03627-x

El-Abeid, Sozan E, Mosa MA, El-Tabakh MAM, Saleh AM, El-Khateeb MA, Haridy MSA. 2024. Antifungal activity of copper oxide nanoparticles derived from *Zizyphus spina* leaf extract against *Fusarium* root rot disease in tomato plants. Journal of Nanobiotechnology 22 (1). https://doi.org/10.1186/s12951-023-02281-8

Ezzeldin I, Ahmad AA, Abdo S, Bakr MA, Khalil MA, Abdallah Y, Ogunyemi SO. 2024. Suppression of root rot fungal diseases in common beans (*Phaseolus vulgaris* L.) through the application of biologically synthesized silver nanoparticles. Nanomaterials 14 (8): 710–725. https://doi.org/10.3390/nano14080710

Ferreira LVM, Leite RA, de Carvalho F, Andrade JFC, de Medeiros FHV, Moreira FMS. 2023. Rhizobacteria control damping-off and promote the growth of lima beans with and without co-inoculation with *Rhizobium tropici* CIAT899. Archives of Microbiology 205 (5). https://doi.org/10.1007/s00203-023-03555-3

Han B, Lu Z, Zhang J, Almodfer R, Wang Z, Sun W, Dong L. 2024. Rep-ViG-Apple: A CNN-GCN hybrid model for apple detection in complex orchard environments. Agronomy 14 (8): 1733. https://doi.org/10.3390/agronomy14081733

Han Y, Wang P, Kundu S, Ding Y, Wang Z. 2023. Vision HGNN: An image is more than a graph of nodes. *In* 2023 IEEE/CVF International Conference on Computer Vision. Institute of Electrical and Electronics Engineers. Paris, France, pp: 19878–19888. https://doi.org/10.1109/iccv51070.2023.01820

Jasiman F, Fourati LC. 2023. Agriculture 4.0 from IoT, artificial intelligence, drone, and blockchain perspectives. *In* 2023 15th International Conference on Developments in eSystems Engineering. Institute of Electrical and Electronics Engineers. Baghdad, Iraq, pp: 262–267. https://doi.org/10.1109/dese58274.2023.10099927

Lamprecht SC, Phasoana TJ, van Wyk W, Spies CFJ. 2024. First report of *Pythium* defense causing damping-off and root rot of soybean in South Africa. Journal of Plant Pathology 106 (2): 771. https://doi.org/10.1007/s42161-024-01592-5

Lawrence ID, Vijayakumar R, Agnishwar J. 2024. Dynamic application of unmanned aerial vehicles for analyzing the growth of crops and weeds for precision agriculture. *In* Gupta RK, Jain A, Wang J, Bharti SK, Patel S. (eds.), Artificial Intelligence Tools and Technologies for Smart Farming and Agriculture Practices. IGI Global Scientific Publishing: Hershey, PA, USA, pp: 115–132. https://doi.org/10.4018/978-1-6684-8516-3.ch007

Li J, Ai M, Hou J, Zhu P, Cui X, Yang Q. 2024. Plant pathogen interaction with root rot of *Panax notoginseng* as a model: Insight into pathogen pathogenesis, plant defense response, and biological control. Molecular Plant Pathology 25 (2): 13–27. https://doi.org/10.1111/mpp.13427

Liu W, Yan C, Li R, Chen G, Wang X, Wen Y, Zhang C, Wang X, Xu Y, Wang Y. 2023. *VqMAPK3*/*VqMAPK6*, *VqWRKY33*, and *VqNSTS3* constitute a regulatory node in enhancing resistance to powdery mildew in grapevine. Horticulture Research 10 (7). https://doi.org/10.1093/hr/uhad116

Lovas A, Lytras I, Rasonyi M, Sabanis S. 2023. Taming neural networks with Tusla: Nonconvex learning via adaptive stochastic gradient Langevin algorithms SIAM. Journal on Mathematics of Data Science 5 (2): 323–345. https://doi.org/10.1137/22m1514283

Puri V, Nayyar A, Raja L. 2017. Agriculture drones: A modern breakthrough in agriculture. Journal of Statistics and Management Systems 20 (4): 507–518. https://doi.org/10.1080/09720510.2017.1395171

Rahman MZ, Talukder MMR, Liberia MG, Rahman MA, Islam MR. 2022. Management of damping-off disease in bottle gourd seedlings under floating agriculture. Bangladesh Journal of Plant Pathology 38 (2): 9–14.

Rajagopal MK, Bala Murugan MS. 2023. Artificial Intelligence-based drone for early disease detection and precision pesticide management in cashew farming. arXiv. https://doi.org/10.48550/arXiv.2303.08556

Raveena S, Surendran R, Khalaf AOI, Hamam H. 2024. Empowering coffee farming using counterfactual recommendation based RNN driven IoT integrated soil quality command system. Scientific Reports 14 (1): 6269–6280. https://doi.org/10.1038/s41598-024-56954-x

Saravanan MS. 2025. Smart plant disease management system integrating temporal data and deep learning techniques. *In* 2025 2025 6th International Conference on Data Intelligence and Cognitive Informatics. Institute of Electrical and Electronics Engineers. Tirunelveli, India, pp: 771–776. https://doi.org/10.1109/icdici66477.2025.11134823

Selvanarayanan R, Rajendran S, Alotaibi Y. 2024. Early detection of *Colletotrichum kahawae* disease in coffee cherry based on computer vision techniques. Computer Modeling in Engineering and Sciences 139 (1): 759–782. https://doi.org/10.32604/cmes.2023.044084

Tamilvizhi T, Surendran R, Anbazhagan K, Rajkumar K. 2022. Quantum behaved particle swarm optimization-based deep transfer learning model for sugarcane leaf disease detection and classification. Mathematical Problems in Engineering 3: 524–533. https://doi.org/10.1155/2022/3452413

Thakur S, Raj H. 2024. Synergistic strategies for sustainable crop protection: Harnessing soil solarization and biofumigants to combat damping-off pathogens in solanaceous vegetable crops. Journal of Plant Diseases and Protection 2: 1–10 https://doi.org/10.21203/rs.3.rs-3238787/v1

Yu C, Lv J, Xu H. 2024. Plant growth-promoting fungi and rhizobacteria control *Fusarium* damping-off in Mason pine seedlings by impacting rhizosphere microbes and altering plant physiological pathways. Plant and Soil 499 (1–2): 503–519. https://doi.org/10.1007/s11104-024-06475-3

Zhang B, Aidong C, Sun M, Zheng W, Zhang H, Fu Z, Zhao S, Zhang Z, Wang L, Zhang H, *et al*. 2024. Effect of five seed-coating agents on the germination rate and the evaluation of control effect on the damping-off disease of sugar beet. Sugar Tech 26 (6): 1746–1750. https://doi.org/10.1007/s12355-024-01360-w