Enhancing Rice Leaf Disease Detection Through Image Embedding and Machine Learning

Steffanie Emillia Dedy[1*], Imam Yuadi[2]
[1]Magister of Human Resource Development, Universitas Airlangga, Indonesia, [2]Department of Information and Library Science, Faculty of Social and Political Sciences, Universitas Airlangga, Indonesia
Email: steffanie.emillia.dedy-2025@pasca.unair.co.id

Rice leaf diseases significantly affect plant productivity, making timely and accurate detection important for agricultural management. This study aimed to evaluate whether image embeddings from pre-trained convolutional neural networks, combined with classical machine learning classifiers, can classify five rice leaf conditions (healthy, bacterial blight, brown spot, leaf blast, and rice hispa) using 517 images from a public Kaggle dataset. Features were extracted with Image Embedding in Orange Data Mining using three pre-trained architectures (VGG16, VGG19, and InceptionV3), and each embedding was paired with Support Vector Machine, Logistic Regression, and Neural Network classifiers, yielding nine embedding–classifier combinations evaluated through stratified 5-fold cross-validation using AUC, Classification Accuracy (CA), F1-score, Precision, Recall, and Matthews Correlation Coefficient (MCC). Among the nine combinations tested, InceptionV3 embedding combined with Logistic Regression achieved the best performance, with AUC 0.986, CA 0.911, F1-score 0.910, Precision 0.911, Recall 0.911, and MCC 0.886, exceeding the best VGG16-based (CA 0.880) and VGG19-based (CA 0.897) combinations tested with the same classifiers. Healthy leaves remained the most difficult class (15 of 37 images correctly classified in the InceptionV3 confusion matrix). These results indicate that integrating deep feature embedding with classical machine learning classifiers can achieve high classification performance on this public dataset. Because computational efficiency and performance under field conditions were not assessed, further validation on independent, field-acquired images is needed before the approach is applied to agricultural decision support.
Keywords: convolutional neural network, image embedding, InceptionV3, machine learning, rice leaf disease.

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

Corresponding Author:
Steffanie Emillia Dedy
Magister of Human Resource Development, Universitas Airlangga, Indonesia
steffanie.emillia.dedy-2025@pasca.unair.co.id

1. Introduction

Rice (*Oryza sativa*) plays a crucial role in global food security, particularly in Asian countries where it serves as a primary staple food. In Indonesia, rice has even greater strategic importance because it is not only the main daily staple for most of the population but also a national commodity closely associated with economic stability and food security [1]. Rice production also supports the livelihoods and socioeconomic welfare of a substantial proportion of agricultural communities. Therefore, maintaining sustainable rice production is essential to ensuring food availability and supporting more than 270 million Indonesians who depend on rice as their primary staple food [2], [3].

Despite its strategic importance, rice production remains vulnerable to various plant diseases that can reduce crop productivity and quality [4]. Leaf diseases, including bacterial blight, brown spot, leaf blast, and rice hispa, can cause considerable yield losses when they are not identified and treated at an early stage. In practice, disease identification still frequently depends on visual field observations conducted by farmers or agricultural experts. Although manual inspection can be effective, it is time-consuming, dependent on individual expertise, potentially subjective, and difficult to implement consistently across extensive agricultural areas. These limitations are particularly relevant in Indonesia, where maintaining rice

productivity is increasingly important in response to population growth and the corresponding demand for staple food [5], [6].

Recent developments in artificial intelligence have encouraged the application of deep learning (DL) and computer vision to plant disease detection and classification [7]. Image-based diagnosis offers a promising alternative because it is non-invasive and can potentially be integrated with increasingly accessible digital and mobile imaging technologies. Previous studies have demonstrated that convolutional neural networks (CNNs) can extract complex visual characteristics from plant images and achieve effective disease classification. In addition to end-to-end CNN training, image embedding using pre-trained CNN architectures provides another approach in which high-level visual features can be extracted without developing and training an entire deep-learning model from scratch [8], [9]. Such an approach may reduce computational requirements while retaining the representational capabilities of established CNN architectures.

Nevertheless, existing studies on rice leaf disease classification have predominantly emphasized the training or fine-tuning of a single deep-learning architecture, such as EfficientNet-B6 [11], optimized deep transfer learning models [5], or vision transformers [14], and have generally reported the performance of one end-to-end pipeline. These studies show that deep models can recognize rice leaf diseases, but they do not indicate whether features extracted from frozen pre-trained CNNs, without further training, are sufficient for this task. Studies that combine CNN-derived features with classical classifiers usually fix a single pairing, as in the hybrid CNN–SVM model for rice leaf disease classification [8]. A related comparison of VGG16, VGG19, and InceptionV3 features [18] evaluated only Logistic Regression and SVM, leaving the contribution of a non-linear Neural Network classifier unexamined. As a result, three questions remain unresolved for multiclass rice leaf classification: (1) which pre-trained embedding architecture provides the most discriminative features, (2) whether the best-performing classifier remains the same across embeddings or changes with the feature representation, and (3) which leaf conditions remain difficult to separate when performance is examined beyond overall accuracy, particularly when class sizes are unequal. Furthermore, approaches that require extensive model training or programming expertise may present practical barriers for researchers, educators, and agricultural practitioners without access to advanced computing infrastructure. Consequently, there remains a need to investigate whether feature representations extracted from established pre-trained CNN models can be effectively combined with conventional classification algorithms to provide accurate multiclass rice leaf condition classification through an accessible workflow.

Orange Data Mining provides a suitable environment for addressing this methodological need because it offers a visual and user-friendly workflow for implementing image embedding and machine-learning classification. Through its Image Embedding functionality, pre-trained architectures such as VGG16, VGG19, and InceptionV3 can be used to generate feature representations that can subsequently be classified using machine-learning algorithms [10]. Previous applications have demonstrated the potential of Orange as an accessible platform for constructing machine-learning workflows without requiring extensive programming [11]. However, the comparative effectiveness of different combinations of CNN-based image embeddings and classification algorithms for distinguishing multiple rice leaf conditions requires further investigation. In particular, determining which combination produces the most reliable classification performance is important for selecting the embedding–classifier pipeline on an empirical basis rather than by default.

Accordingly, this study addresses this gap by systematically comparing combinations of pre-trained CNN-based image embedding methods and machine-learning classifiers for multiclass rice leaf condition classification within the Orange Data Mining environment. Five categories are examined: healthy rice leaves,

bacterial blight, brown spot, leaf blast, and rice hispa. Image data obtained from a publicly available Kaggle dataset are transformed into feature embeddings and subsequently evaluated using three classification algorithms: Support Vector Machine (SVM), Logistic Regression, and Neural Network. Rather than focusing solely on the performance of a single architecture, the study evaluates alternative embedding–classifier combinations under a consistent experimental workflow to determine their comparative classification performance.

The three embedding architectures were selected because they represent contrasting design principles while being available within the same Orange Image Embedding environment. VGG16 and VGG19 belong to the same architecture family but differ in depth [13], allowing the study to examine whether additional convolutional layers yield more discriminative features for rice leaf lesions. InceptionV3, in contrast, extracts visual information at multiple receptive-field scales through parallel Inception modules [14], [15], which may be relevant because rice leaf diseases differ in lesion size, shape, texture, and distribution. The three classifiers were selected because they form decision boundaries in different ways: Logistic Regression is a linear probabilistic model [18], SVM separates classes by maximizing the margin between them [17], and the Neural Network (multilayer perceptron) can model non-linear relationships among features. Pairing every embedding with every classifier in a 3 × 3 design, under identical preprocessing, cross-validation, and evaluation metrics, makes it possible to attribute differences in performance to the embedding, the classifier, or their combination rather than to differences in experimental procedure. Reporting AUC, CA, F1-score, Precision, Recall, and MCC together with class-level confusion matrices further allows performance to be judged beyond overall accuracy.

This study makes three main contributions. First, it provides a controlled comparison of nine embedding–classifier combinations for five-class rice leaf condition classification on a public dataset, using multiple complementary metrics rather than accuracy alone. Second, it examines whether the best-performing classifier depends on the embedding architecture, which indicates whether the embedding and classifier should be selected jointly as one pipeline rather than independently. Third, through class-level confusion-matrix analysis, it identifies which leaf conditions remain difficult to distinguish, providing direction for future data collection and model improvement. In addition, the workflow is implemented in an accessible low-code environment, offering a reproducible baseline that does not require developing a computationally intensive end-to-end deep-learning model. Because the evaluation is limited to a public dataset and does not assess computational efficiency or field performance, the findings are intended as a methodological foundation for the further development and validation of image-based rice disease identification systems that may support broader efforts toward accessible digital agriculture and sustainable rice production in Indonesia.

2. Method

This study employed a quantitative experimental approach based on image analysis to classify rice leaf conditions using pre-trained convolutional neural network (CNN) feature extraction and machine-learning classifiers. The experimental workflow consisted of five main stages: (1) dataset collection and preparation, (2) image preprocessing, (3) feature extraction using pre-trained CNN architectures, (4) classification using three machine-learning algorithms, and (5) model evaluation. All experiments were conducted in Orange Data Mining using the same dataset and evaluation procedure to ensure that the performance of each embedding–classifier combination could be compared under consistent experimental conditions. The overall research procedure is presented in Figure 1.

 ISSN 3025-2792

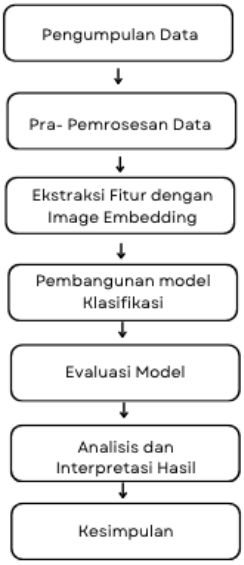


Figure 1. Research Methodology

a. Data Collection

The dataset used in this study consisted of 517 rice leaf images obtained from the publicly available Rice Leaf Diseases Dataset on Kaggle. The dataset is available through the Rice Leaf Diseases Dataset on Kaggle. The images represent five classes of rice leaf conditions: healthy, bacterial blight, brown spot, leaf blast, and rice hispa.

The class labels provided in the dataset were retained as the target variables for classification. All 517 images available for the five selected categories were included in the experiment; therefore, no additional sampling procedure was applied. The same dataset was used for every experimental combination to ensure that differences in classification performance were attributable to the feature extraction and classification methods rather than differences in input data.

b. Data Preprocessing

Before feature extraction, the images were prepared to establish a consistent input format. Each image was resized to 256 × 256 pixels while retaining its corresponding class label. No additional manual feature extraction was performed because visual features were automatically generated using pre-trained CNN architectures during the image embedding stage.

After resizing, the images and their corresponding labels were imported into Orange Data Mining. The five target labels were defined as healthy rice leaves, bacterial blight, brown spot, leaf blast, and rice hispa. The prepared image data were subsequently connected to the Image Embedding widget to generate numerical feature representations.

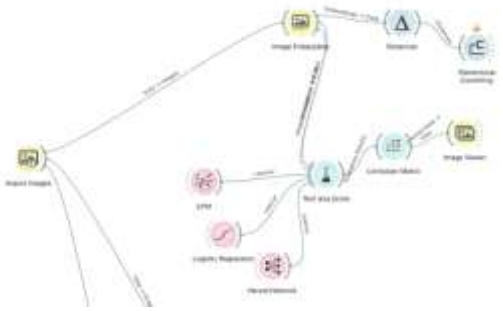


Figure 2. Data processing in Orange

 ISSN 3025-2792

This preprocessing procedure ensured that the same image dataset and image dimensions were used for all feature extraction models. Consequently, VGG16, VGG19, and InceptionV3 were evaluated using consistent input data.

c. Feature Extraction with Image Embedding

Feature extraction was performed using the Image Embedding functionality in Orange Data Mining. Instead of training a CNN from scratch, this study employed pre-trained CNN architectures as feature extractors. Each input image was transformed into a numerical feature vector representing visual characteristics learned by the pre-trained networks. These feature vectors subsequently served as predictor variables for the classification algorithms.

Three pre-trained CNN architectures were evaluated: VGG16, VGG19, and InceptionV3. VGG16 is a deep CNN architecture commonly employed for image feature extraction and has demonstrated high classification performance in previous image-based applications [12]. VGG19 extends the VGG architecture and contains 19 weighted layers, including 16 convolutional layers and three fully connected layers, enabling the extraction of hierarchical visual representations [13].

InceptionV3 was included because its architecture is designed to capture visual patterns at different scales while maintaining computational efficiency [14]. Its Inception modules employ parallel operations that enable the network to extract multiple forms of image information at different receptive-field scales [15]. Previous research has demonstrated the effectiveness of InceptionV3 for image classification in several domains, including agricultural and medical applications [16].

The three architectures were applied independently to the same set of 517 images. Thus, three feature datasets were produced, corresponding to VGG16, VGG19, and InceptionV3 embeddings. Each resulting feature dataset was subsequently evaluated using the same three classification algorithms. This procedure generated nine embedding–classifier combinations for comparative evaluation.

d. Classification Models

Three classification algorithms were applied to the extracted image features: Support Vector Machine (SVM), Logistic Regression, and Neural Network. Each classifier received the numerical feature vectors generated by VGG16, VGG19, and InceptionV3 rather than the original image pixels. Accordingly, the experimental design consisted of the following nine combinations: VGG16–SVM, VGG16–Logistic Regression, VGG16–Neural Network, VGG19–SVM, VGG19–Logistic Regression, VGG19–Neural Network, InceptionV3–SVM, InceptionV3–Logistic Regression, and InceptionV3–Neural Network.

a. *Support Vector Machine (SVM)*
   SVM constructs a decision boundary by identifying a hyperplane that maximizes the margin between classes in the feature space. In this study, the input feature space was generated from each of the three CNN embedding architectures. The use of CNN-derived features with SVM has previously demonstrated effective image classification performance because CNNs provide high-level visual representations that can subsequently be separated by a discriminative classifier [17]. SVM was therefore evaluated to determine how effectively it could distinguish the five rice leaf classes based on each embedding representation.
b. *Logistic Regression*
   Logistic Regression was employed as a probabilistic linear classification method. Although originally formulated for binary outcomes, it can be extended to multiclass classification and has the advantages of computational simplicity and interpretability [18]. In this experiment, Logistic Regression classified

 ISSN 3025-2792

the five rice leaf conditions based on feature vectors generated independently by VGG16, VGG19, and InceptionV3.

c. *Neural Network*
The Neural Network classifier was used to capture potentially non-linear relationships among the CNN-derived image features. The classifier was based on a multilayer perceptron (MLP), in which the embedded features were provided as input variables and the five rice leaf conditions constituted the target classes. As with SVM and Logistic Regression, the Neural Network was evaluated separately using features generated by each of the three embedding architectures.

Unless otherwise modified in the experimental configuration, the classifiers were implemented through Orange Data Mining using the parameter settings applied consistently throughout the experiments. Keeping the experimental procedure consistent allowed direct comparison among the nine embedding–classifier combinations.

e. Model Evaluation

The experimental procedure was designed to compare the influence of both feature extraction architecture and classification algorithm. First, the standardized images were processed separately using VGG16, VGG19, and InceptionV3. Second, the feature vectors produced by each architecture were connected independently to SVM, Logistic Regression, and Neural Network classifiers. This resulted in nine experimental configurations.

Model performance was evaluated through the stratified 5-fold cross-validation procedure available in Orange Data Mining. During cross-validation, the dataset was repeatedly divided into training and validation subsets so that each model was trained on one portion of the data and evaluated on previously unseen observations. Specifically, the 517 images were divided into five folds of 103–104 images while preserving the class proportions, so that each fold contained 24 images of each disease class and seven or eight healthy images. In each of the five iterations, four folds (413–414 images) were used for training and the remaining fold was used for validation; consequently, every image was used for validation exactly once. The same validation procedure was applied to all nine model combinations to maintain comparability and reduce the possibility that performance differences resulted from inconsistent evaluation procedures.

f. Evaluation Metrics

Six metrics were used to evaluate classification performance: Area Under the Curve (AUC), Classification Accuracy (CA), Precision, Recall, F1-score, and Matthews Correlation Coefficient (MCC). Multiple metrics were considered because classification accuracy alone may not provide a complete representation of model performance, particularly when class distributions are unequal. Classification Accuracy represents the proportion of correctly classified observations and is expressed as:

$$CA = (TP + TN) / (TP + TN + FP + FN)$$

where *TP* denotes true positives, *TN* true negatives, *FP* false positives, and *FN* false negatives.
Precision represents the proportion of correctly predicted positive observations among all observations predicted as positive:

$$Precision = TP / (TP + FP)$$

Recall measures the proportion of actual positive observations that are correctly identified:

$$Recall = TP / (TP + FN)$$

The F1-score provides a balanced measure of Precision and Recall and is calculated as their harmonic mean:

$$F1 = 2 \times (Precision \times Recall) / (Precision + Recall)$$

 ISSN 3025-2792

AUC evaluates the ability of a classifier to discriminate among classes based on the Receiver Operating Characteristic (ROC) curve. A higher AUC indicates stronger discriminative capability, with values approaching 1 representing better classification performance [19].
MCC was additionally employed because it considers all components of the confusion matrix and can provide a balanced assessment even when class distributions are unequal. It is expressed as:

$$MCC = (TP \times TN - FP \times FN) / \sqrt{[(TP + FP)(TP + FN)(TN + FP)(TN + FN)]}$$

MCC ranges from −1 to +1, where +1 indicates perfect prediction, 0 indicates performance equivalent to random prediction, and −1 represents complete disagreement between predictions and actual classes.

The evaluation scores generated through cross-validation were compared across the nine embedding–classifier combinations. The combination producing the strongest overall performance across AUC, CA, F1-score, Precision, Recall, and MCC was subsequently identified as the most effective configuration for classifying the five rice leaf conditions examined in this study.

3. Results And Discussion

This study evaluated three classification algorithms—Support Vector Machine (SVM), Logistic Regression, and Neural Network—using image features extracted from three pre-trained CNN architectures: VGG16, VGG19, and InceptionV3. The classification task involved five rice leaf conditions: bacterial blight, brown spot, healthy leaves, leaf blast, and rice hispa. Figures 3–7 present representative images of the five classes used in the classification process.



Figure 3. Bacterial blight



Figure 4. Brown spot



Figure 5. Healthy Leaf



Figure 6. Leaf Blast

 ISSN 3025-2792

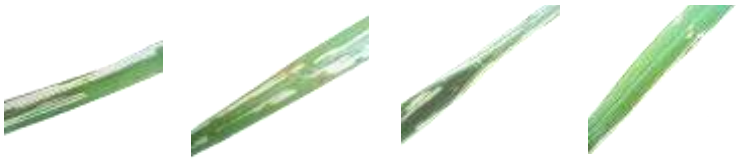


Figure 7. Rice Hispa

Computation and Aggregation of Multiclass Evaluation Metrics

Because the task involves $K = 5$ classes, the binary expressions given in Section 2.6 cannot be applied directly to the whole confusion matrix. In this study, they are applied to each class separately in a one-vs-rest manner and then aggregated into a single score, following the procedure implemented in the Test and Score widget of Orange Data Mining [26], [27]. Let C denote the 5 × 5 confusion matrix pooled over the five stratified cross-validation folds, where Cij is the number of images whose actual class is i and whose predicted class is j; N = 517 is the total number of images; n_k is the number of images that actually belong to class k (row total, or class support); and m_k is the number of images predicted as class k (column total). For each class k, the one-vs-rest counts are:

$$\mathrm{TP}_k = C_{kk},\ \ \mathrm{FP}_k = m_k - C_{kk},\ \ \mathrm{FN}_k = n_k - C_{kk},\ \ \mathrm{TN}_k = N - \mathrm{TP}_k - \mathrm{FP}_k - \mathrm{FN}_k \quad (1)$$

Classification Accuracy is the proportion of all images lying on the main diagonal of C. This expression replaces the binary form (TP + TN)/(TP + TN + FP + FN), which is defined only for a single positive class:

$$\mathrm{CA} = \frac{\sum_{k=1}^{K} C_{kk}}{N} \quad (2)$$

Per-class Precision, Recall, and F1-score are obtained from the one-vs-rest counts of Equation (1):

$$\mathrm{Precision}_k = \frac{\mathrm{TP}_k}{\mathrm{TP}_k+\mathrm{FP}_k},\ \ \mathrm{Recall}_k = \frac{\mathrm{TP}_k}{\mathrm{TP}_k+\mathrm{FN}_k},\ \ \mathrm{F1}_k = \frac{2\cdot\mathrm{Precision}_k\cdot\mathrm{Recall}_k}{\mathrm{Precision}_k+\mathrm{Recall}_k} \quad (3)$$

The single values reported in Table 1 are support-weighted averages of these per-class values (the "average over classes" setting of Orange), in which each class contributes in proportion to its size [27]:

$$M_{\mathrm{w}} = \sum_{k=1}^{K} \frac{n_k}{N} M_k\,,\ \ M \in \{\mathrm{Precision}, \mathrm{Recall}, \mathrm{F1}\} \quad (4)$$

Two consequences of Equation (4) are important for interpreting the results. First, the weighted Recall is mathematically identical to CA, because $\sum(n_k/N)(C_{kk}/n_k) = \sum C_{kk}/N$. This explains why Recall and CA are equal for every combination in Table 1, and the two columns should therefore not be read as independent evidence. Second, the weighted F1-score is the weighted mean of the per-class F1-scores, not the harmonic mean of the weighted Precision and Recall, so it can be lower than both when per-class performance is uneven (e.g., 0.872 versus 0.886 and 0.878 for InceptionV3–SVM). Because the healthy class contains only 37 of 517 images, it receives a weight of only 0.072 in Equation (4). Macro-averaged values, which give every class equal weight, are therefore also reported in Sections 3.7 and 3.8:

$$M_{\mathrm{macro}} = \frac{1}{K}\sum_{k=1}^{K} M_k \quad (5)$$

For MCC, the binary formula in Section 2.6 is replaced by its multiclass (K-category) generalization [28], which uses the full confusion matrix instead of a single positive class:

$$\mathrm{MCC} = \frac{c\cdot N - \sum_{k=1}^{K} m_k\cdot n_k}{\sqrt{\left(N^2 - \sum_{k=1}^{K} m_k{}^2\right)\left(N^2 - \sum_{k=1}^{K} n_k{}^2\right)}},\ \ c = \sum_{k=1}^{K} C_{kk} \quad (6)$$

Equation (6) reduces to the binary MCC when $K = 2$. For example, applying it to the InceptionV3 confusion matrix in Table 4 ($c = 454$, $N = 517$) gives MCC = 174,006 / 205,839 ≈ 0.845, which reproduces the value exported by Orange. Finally, AUC is computed for each class as a one-vs-rest ROC curve based on the predicted probability of that class [19], and the K class-wise values are combined as:

$$\mathrm{AUC} = \sum_{k=1}^{K} w_k \mathrm{AUC}_k,\ \ w_k = \frac{n_k(N-n_k)}{\sum_{j=1}^{K} n_j(N-n_j)} \quad (7)$$

With this weighting, each of the four classes with 120 images receives a weight of 0.229, whereas the healthy class receives 0.085. In Orange, AUC is calculated within each validation fold and then averaged across folds, whereas CA, Precision, Recall, F1-score, and MCC are calculated from the predictions pooled over all folds, so that every image is predicted exactly once by a model that did not see it during training. The correctness of this description was verified by recomputing the metrics from the confusion matrices in Tables 2–4: the weighted Precision, Recall, F1-score, and multiclass MCC obtained with Equations (2)–(6) match the values exported by Orange to three decimal places.

Overall Comparison of the Nine Embedding–Classifier Combinations

Table 1 summarizes all six evaluation metrics for the nine embedding–classifier combinations in a single table, together with the number of correctly classified images and the 95% Wilson confidence interval (CI) of CA [29]. The combinations are ranked by MCC because, unlike weighted Recall, MCC uses every cell of the confusion matrix.

Table 1. Classification performance of the nine embedding–classifier combinations (cross-validation, $N = 517$)

Embedding	Classifier	AUC	CA	Precision	Recall	F1-score	MCC	Correct (of 517)	95% CI of CA	Rank
VGG16	SVM	0.972	0.855	0.853	0.855	0.851	0.814	442	0.822–0.883	9
	Logistic Regression	0.976	0.880	0.880	0.880	0.879	0.846	455	0.849–0.905	5
	Neural Network	0.978	0.878	0.877	0.878	0.876	0.844	454	0.847–0.904	7
VGG19	SVM	0.977	0.882	0.882	0.882	0.876	0.849	456	0.851–0.907	4
	Logistic Regression	0.972	0.859	0.855	0.859	0.857	0.819	444	0.826–0.886	8
	Neural Network	0.979	0.897	0.895	0.897	0.895	0.868	464	0.868–0.921	2
InceptionV3	SVM	0.985	0.878	0.886	0.878	0.872	0.845	454	0.847–0.904	6
	Logistic Regression	**0.986**	**0.911**	**0.911**	**0.911**	**0.910**	**0.886**	**471**	**0.883–0.933**	**1**
	Neural Network	0.984	0.896	0.894	0.896	0.895	0.866	463	0.866–0.919	3

Note: Precision, Recall, and F1-score are support-weighted averages over the five classes (Eq. 4); AUC is the weighted one-vs-rest average (Eq. 7); MCC is the multiclass coefficient (Eq. 6). The 95% CI of CA is the Wilson interval computed from the number of correctly classified images. Rank is based on MCC. Bold indicates the best-performing combination, which achieved the highest value for every metric.

Across the nine combinations, CA ranged from 0.855 (VGG16–SVM; 442 correct) to 0.911 (InceptionV3–Logistic Regression; 471 correct), a difference of 29 images, and MCC ranged from 0.814 to 0.886. The ranking by CA and the ranking by MCC were identical, indicating that the ordering of the pipelines is not an artifact of a single metric. AUC varied much less (0.972–0.986) and exceeded 0.97 for every combination. This shows that all pipelines assigned a high predicted probability to the correct class even when the final class decision was wrong; AUC is therefore less useful than CA or MCC for discriminating among the

pipelines. Nevertheless, AUC reveals a consistent embedding effect: all three InceptionV3 combinations (0.984–0.986) achieved a higher AUC than any VGG16 or VGG19 combination (≤ 0.979).

Variation in the cross-validation results must also be considered when interpreting the ranking. Because stratified 5-fold cross-validation was used, each validation fold contained only 103–104 images, including seven or eight healthy images, so fold-level estimates, particularly for the healthy class, are expected to fluctuate considerably. The scores in Table 1 are calculated from pooled out-of-fold predictions, and the Orange output used in this study does not report fold-to-fold standard deviations; the Wilson CI is therefore used to express the sampling uncertainty of CA for a dataset of 517 images. The CI of the best combination, InceptionV3–Logistic Regression (0.883–0.933), overlaps with the CIs of the second- and third-ranked combinations, VGG19–Neural Network (0.868–0.921) and InceptionV3–Neural Network (0.866–0.919), and its advantage over the second-ranked combination corresponds to only seven images (1.4 percentage points). Consequently, InceptionV3–Logistic Regression should be described as the best-performing combination observed in this experiment rather than as significantly superior to all alternatives; confirming its superiority would require repeated cross-validation or a paired test on per-image predictions. In contrast, its margin over the two lowest-ranked combinations, VGG19–Logistic Regression (0.826–0.886) and VGG16–SVM (0.822–0.883), is larger and more robust.

Classification Performance Using VGG16

The results obtained using VGG16 show that all three classifiers were able to classify the rice leaf images with relatively high performance (Table 1). Logistic Regression achieved a Classification Accuracy (CA) of 0.880, while Neural Network achieved a CA of 0.878, F1-score of 0.876, and Recall of 0.878. SVM produced the lowest CA among the three classifiers at 0.855, which was also the lowest value among all nine combinations (MCC 0.814). Logistic Regression also achieved an MCC of 0.846.

These results indicate that VGG16-generated features contain sufficient discriminative information for separating the five rice leaf conditions. However, the relatively small difference between Logistic Regression and Neural Network (one image; 455 versus 454 correct) suggests that neither linear nor non-linear classification alone determines performance. Instead, classification effectiveness is also influenced by the quality of the visual representations generated during the embedding stage.

The VGG16 results therefore establish an important baseline for comparison with the deeper VGG19 architecture and the multi-scale feature extraction mechanism of InceptionV3. Although VGG16 provided good classification results, subsequent experiments demonstrate that changing the embedding architecture can further improve classification performance.

Classification Performance Using VGG19

VGG19 produced higher peak classification performance than VGG16 (Table 1). The Neural Network achieved a CA of 0.897, F1-score of 0.895, Precision of 0.895, Recall of 0.897, and MCC of 0.868. SVM achieved a CA of 0.882 and MCC of 0.849, whereas Logistic Regression recorded a CA of 0.859.

The performance pattern differs from that obtained with VGG16. With VGG19 features, Neural Network clearly produced the highest reported CA, F1-score, Precision, Recall, and MCC among the three classifiers. The increase in peak CA from 0.880 with VGG16 to 0.897 with VGG19 indicates that the choice of embedding architecture affects the discriminative quality of the extracted features.

VGG19 contains a deeper feature extraction structure than VGG16 [13]. The results suggest that the representation produced by VGG19 can be effectively processed by a Neural Network classifier for this dataset. However, the results also demonstrate that a more complex feature extractor does not automatically improve every classifier. Logistic Regression, for example, achieved a lower CA with VGG19

(0.859) than with VGG16 (0.880), corresponding to 11 additional errors. Therefore, classification performance depends on the interaction between the embedding architecture and the classifier rather than on embedding depth alone.

Classification Performance Using InceptionV3

Among the tested configurations, the highest overall classification results were obtained when InceptionV3 features were combined with Logistic Regression (Table 1). This combination achieved an AUC of 0.986, CA of 0.911, F1-score of 0.910, Precision of 0.911, Recall of 0.911, and MCC of 0.886. Neural Network also performed strongly with a CA of 0.896 and MCC of 0.866, while SVM achieved a CA of 0.878.

The CA of 0.911 obtained by InceptionV3–Logistic Regression was higher than the highest reported CA obtained using VGG16 (0.880) and VGG19 (0.897). The MCC value of 0.886 further indicates strong agreement between predicted and actual classifications. Thus, the performance improvement is not reflected only in accuracy but is also supported by another evaluation measure that accounts for all cells of the confusion matrix. However, as discussed in Section 3.2, the margin over VGG19–Neural Network and InceptionV3–Neural Network (seven and eight images, respectively) is small relative to the sampling uncertainty of the cross-validation estimate.

A possible explanation for this result is related to the characteristics of the InceptionV3 architecture. InceptionV3 is designed to capture image information at multiple scales through its Inception modules [14], [15]. Such representations may be advantageous when disease categories are distinguished by variations in lesion shape, texture, distribution, and other visual characteristics. In this experiment, the resulting feature representation was effectively separated by Logistic Regression, indicating that the extracted InceptionV3 features provided a feature space suitable for distinguishing the five classes.

An important finding is that the most effective classifier changed according to the embedding architecture. Neural Network produced the highest reported CA with VGG19, whereas Logistic Regression produced the highest CA with VGG16 and InceptionV3. This finding demonstrates that classifier selection should not be considered independently from the feature extraction method. Instead, the embedding and classifier should be evaluated as a combined classification pipeline.

Comparative Analysis of Embedding–Classifier Combinations

Comparison of the three embedding approaches in Table 1 demonstrates that the best-performing configuration was InceptionV3 combined with Logistic Regression, which achieved a CA of 0.911 and MCC of 0.886. The best reported configuration using VGG19 achieved a CA of 0.897 and MCC of 0.868 with Neural Network, while the highest reported VGG16 accuracy was 0.880 using Logistic Regression.

Averaged over the three classifiers, the mean CA was 0.871 for VGG16, 0.879 for VGG19, and 0.895 for InceptionV3, and the mean MCC was 0.835, 0.845, and 0.866, respectively. InceptionV3 therefore provided the most discriminative features on average, not only in its best pairing. Averaged over the three embeddings, the mean CA was 0.872 for SVM, 0.883 for Logistic Regression, and 0.890 for Neural Network. The classifiers also differed in their sensitivity to the feature representation. The CA of the Neural Network varied within a narrow range (0.878–0.897; range 0.019), that of SVM varied by 0.027 (0.855–0.882), and that of Logistic Regression varied by 0.052 (0.859–0.911), from the second-lowest to the highest value in Table 1. Thus, Logistic Regression produced the best single result but was also the least stable classifier across embeddings, whereas the Neural Network was the most consistent choice when the embedding is not fixed in advance.

These results provide an important answer to the research problem because they demonstrate that effective rice leaf classification does not necessarily require training a CNN architecture from scratch. Pre-

trained CNNs can instead be employed to generate visual feature representations that are subsequently processed by conventional classification algorithms. Among the combinations examined in this study, InceptionV3 features provided the strongest performance when paired with Logistic Regression.

The findings also address the methodological gap identified in this study by demonstrating the importance of evaluating multiple embedding–classifier combinations within the same experimental workflow. The results show that the relative performance of the classifiers changes across embedding architectures. For example, Logistic Regression performed strongly with InceptionV3 but produced a lower CA with VGG19, while Neural Network performed particularly well with VGG19. Consequently, evaluating only one CNN embedding or one classifier could provide an incomplete assessment of the classification pipeline.

Confusion Matrix Analysis

In addition to aggregate evaluation metrics, confusion matrices were examined to determine how predictions were distributed across individual rice leaf classes. This analysis is important because similar overall accuracy values can conceal substantial differences in model performance for particular classes.

Tables 2–4 present the confusion matrices as editable tables, together with the class-level Recall (row-wise) and Precision (column-wise). Each matrix contains all 517 out-of-fold predictions. To identify the classifier that produced each matrix, the metrics were recomputed from the matrix using Equations (2)–(6). The VGG16 matrix (454 correct; CA 0.878, weighted Precision 0.877, F1-score 0.876, MCC 0.844) reproduces VGG16–Neural Network; the VGG19 matrix (456 correct; CA 0.882, Precision 0.882, F1-score 0.876, MCC 0.849) reproduces VGG19–SVM; and the InceptionV3 matrix (454 correct; CA 0.878, Precision 0.886, F1-score 0.872, MCC 0.845) reproduces InceptionV3–SVM in Table 1. The class-level analysis below therefore describes these three combinations. For comparison, the best-performing InceptionV3–Logistic Regression combination misclassified 46 images (471 correct), 17 fewer than InceptionV3–SVM.

VGG16 Confusion Matrix (Neural Network)

Table 2. Confusion matrix of the VGG16 embedding with the Neural Network classifier

Actual \ Predicted	Bacterial blight	Brown spot	Healthy	Leaf blast	Rice hispa	Total (n_k)	Recall
Bacterial blight	**101**	0	0	17	2	120	0.842
Brown spot	0	**115**	3	1	1	120	0.958
Healthy	0	4	**22**	1	10	37	0.595
Leaf blast	10	1	0	**107**	2	120	0.892
Rice hispa	3	1	4	3	**109**	120	0.908
Total predicted (m_k)	114	121	29	129	124	517	
Precision	0.886	0.950	0.759	0.829	0.879		

Note: Rows are actual classes and columns are predicted classes; bold diagonal cells are correctly classified images. Recall = diagonal / row total; Precision = diagonal / column total. Values are pooled over all cross-validation folds (N = 517).

As shown in Table 2, the VGG16 confusion matrix indicates that most observations were correctly classified. Bacterial blight was correctly predicted in 101 of 120 images, while brown spot achieved 115 correct predictions out of 120 images. Leaf blast was correctly identified in 107 of 120 images, and rice hispa was correctly classified in 109 of 120 images.

Healthy leaves showed the lowest number of correct classifications, with 22 of 37 images correctly identified (Recall 0.595). The remaining healthy images were primarily confused with rice hispa (10 images) and brown spot (4 images). Previous studies have also indicated that visual similarity among leaf conditions

 ISSN 3025-2792

can complicate image-based classification [20], [21]. Similarly, 17 bacterial blight images were classified as leaf blast, and 10 leaf blast images were classified as bacterial blight, indicating that distinguishing visually related disease symptoms remains challenging. Together, these two directions accounted for 27 of the 63 errors (42.9%) made by VGG16–Neural Network, making bacterial blight–leaf blast the dominant confusion pair for this embedding.

The high number of correct predictions for brown spot and rice hispa suggests that the VGG16 feature representation captured visual information that differentiated these categories relatively well. In contrast, the errors involving healthy leaves and certain disease categories indicate that class-specific visual separability remains uneven.

VGG19 Confusion Matrix (SVM)

Table 3. Confusion matrix of the VGG19 embedding with the SVM classifier

Actual \ Predicted	Bacterial blight	Brown spot	Healthy	Leaf blast	Rice hispa	Total (n_k)	Recall
Bacterial blight	**112**	0	0	3	5	120	0.933
Brown spot	1	**117**	0	2	0	120	0.975
Healthy	2	4	**16**	1	14	37	0.432
Leaf blast	14	3	0	**101**	2	120	0.842
Rice hispa	2	1	3	4	**110**	120	0.917
Total predicted (m_k)	131	125	19	111	131	517	
Precision	0.855	0.936	0.842	0.910	0.840		

Note: Rows are actual classes and columns are predicted classes; bold diagonal cells are correctly classified images. Recall = diagonal / row total; Precision = diagonal / column total. Values are pooled over all cross-validation folds (N = 517).

Table 3 shows that the VGG19 confusion matrix also shows a predominance of correct predictions (456 of 517). Bacterial blight was correctly identified in 112 of 120 images (Recall 0.933), with 3 images misclassified as leaf blast and 5 as rice hispa. Brown spot recorded 117 correct classifications out of 120 images (Recall 0.975), the highest class-level Recall in the three matrices. Leaf blast was correctly recognized in 101 of 120 images, and 14 of its 19 errors were predictions of bacterial blight. Rice hispa achieved 110 correct predictions out of 120 images.

Healthy leaves again represented the most difficult category, with only 16 correct classifications out of 37 images (Recall 0.432). Of the 21 misclassified healthy images, 14 were classified as rice hispa, 4 as brown spot, 2 as bacterial blight, and 1 as leaf blast. This pattern is consistent with the challenge of separating categories when image characteristics overlap [22], [23]. Previous studies have similarly identified visual similarity among plant leaf conditions as a factor affecting classification performance [24], [25].

The confusion pattern between bacterial blight and leaf blast is particularly noteworthy because errors occurred in both directions. This suggests that some images from these categories contain overlapping feature representations. In VGG19–SVM, however, the confusion was strongly asymmetric (14 leaf blast images predicted as bacterial blight versus 3 in the opposite direction), indicating that this classifier tended to assign ambiguous elongated lesions to bacterial blight, which also lowered the Precision of bacterial blight to 0.855. Therefore, although VGG19 improved overall model performance, certain class-level classification difficulties remained.

InceptionV3 Confusion Matrix (SVM)

Table 4. Confusion matrix of the InceptionV3 embedding with the SVM classifier

Actual \ Predicted	Bacterial blight	Brown spot	Healthy	Leaf blast	Rice hispa	Total (n_k)	Recall
Bacterial blight	**101**	0	0	9	10	120	0.842
Brown spot	2	**115**	0	3	0	120	0.958
Healthy	0	3	**15**	1	18	37	0.405
Leaf blast	7	4	0	**105**	4	120	0.875
Rice hispa	0	0	1	1	**118**	120	0.983
Total predicted (m_k)	110	122	16	119	150	517	
Precision	0.918	0.943	0.938	0.882	0.787		

Note: Rows are actual classes and columns are predicted classes; bold diagonal cells are correctly classified images. Recall = diagonal / row total; Precision = diagonal / column total. Values are pooled over all cross-validation folds (N = 517).

The InceptionV3 confusion matrix (Table 4) shows strong classification performance for most disease categories. Bacterial blight was correctly classified in 101 of 120 images, while some images were classified as leaf blast (9 images) and rice hispa (10 images). Brown spot achieved 115 correct predictions out of 120 images. Leaf blast was correctly classified in 105 of 120 images, with some observations confused with bacterial blight (7 images) and brown spot (4 images).

Rice hispa showed particularly strong class-level performance, with 118 of 120 images correctly classified (Recall 0.983). However, rice hispa also absorbed many errors from other classes, receiving 150 predictions for only 120 actual images, so its Precision fell to 0.787, the lowest class-level Precision in the three matrices. In contrast, healthy leaves remained the most difficult category, with only 15 of 37 images correctly identified and 18 images (82% of the healthy-leaf errors) classified as rice hispa. Thus, even with the embedding that produced the highest overall performance, classification performance was not equally distributed across all classes.

This result provides an important qualification to the aggregate performance metrics. The CA of 0.911 and MCC of 0.886 obtained with Logistic Regression demonstrate strong overall classification performance, but the confusion matrix of the same embedding indicates that classification errors remain concentrated in particular categories. Therefore, the higher overall performance of InceptionV3 should be interpreted together with its class-level behavior rather than solely from accuracy.

Classes with the Most Misclassification

Table 5. Class-level F1-score and the effect of the averaging method in the three confusion matrices

Class / measure	n (%)	VGG16 – Neural Network	VGG19 – SVM	InceptionV3 – SVM
F1-score: Bacterial blight	120 (23.2)	0.863	0.892	0.878
F1-score: Brown spot	120 (23.2)	0.954	0.955	0.950
F1-score: Healthy	37 (7.2)	0.667	0.571	0.566
F1-score: Leaf blast	120 (23.2)	0.859	0.874	0.879
F1-score: Rice hispa	120 (23.2)	0.893	0.876	0.874
Weighted F1-score (as in Table 1)	517 (100)	0.876	0.876	0.872
Macro F1-score (Eq. 5)		0.847	0.834	0.829

Class / measure	n (%)	VGG16 – Neural Network	VGG19 – SVM	InceptionV3 – SVM
CA (= weighted Recall)		0.878	0.882	0.878
Balanced accuracy (macro Recall)		0.839	0.820	0.813
Errors made on healthy leaves / all errors		15 / 63 (23.8%)	21 / 61 (34.4%)	22 / 63 (34.9%)

Note: n = number of images in the class (percentage of N = 517). Weighted values follow Eq. (4) and macro values follow Eq. (5). Balanced accuracy is the unweighted mean of the five class-level Recall values. Bold marks the minority (healthy) class.

Table 5 compares the class-level F1-scores across the three matrices, and three consistent patterns emerge. First, healthy leaves were the most frequently misclassified class in every matrix, with a Recall of only 0.595, 0.432, and 0.405 and F1-scores of 0.566–0.667, compared with 0.859–0.955 for the four disease classes. Although healthy leaves represent only 7.2% of the dataset, they accounted for 23.8%–34.9% of all errors. In contrast, their Precision was relatively high (0.759, 0.842, and 0.938) because the classifiers predicted the healthy class only rarely (29, 19, and 16 predictions for 37 actual images). This combination of high Precision and low Recall indicates that the classifiers were conservative toward the minority class and assigned uncertain healthy images to larger classes, most often rice hispa (10, 14, and 18 images). A plausible visual explanation is that rice hispa damage often appears as narrow pale streaks on an otherwise green leaf, so mildly damaged rice hispa leaves and healthy leaves share most of their image area [22].

Second, bacterial blight and leaf blast formed the most frequent confusion pair among the disease classes (27, 17, and 16 errors in Tables 2, 3, and 4, respectively). Both diseases produce elongated lesions that can appear similar when the image is reduced to 256 × 256 pixels, which is consistent with the visual-similarity problem reported in previous studies [22], [23]. Third, brown spot was the most reliably recognized class in all three matrices (F1-score 0.950–0.955), suggesting that its relatively small, rounded brown lesions are well captured by all three embeddings. These patterns indicate that further improvement should focus primarily on the healthy–rice hispa and bacterial blight–leaf blast boundaries rather than on overall accuracy.

Effect of Class Imbalance

The dataset is moderately imbalanced: four classes contain 120 images each (23.2%), whereas the healthy class contains 37 images (7.2%), giving an imbalance ratio of 3.2:1. No resampling or class weighting was applied, so the classifiers were trained on this distribution in every fold. The imbalance affects the results in two ways. First, as shown by Equations (4) and (7), the healthy class contributes only 7.2% of the weight in the weighted Precision, Recall, and F1-score and 8.5% of the weight in AUC; the aggregate scores in Table 1 therefore mainly reflect performance on the four disease classes. Table 5 shows that giving each class equal weight lowers the F1-score from 0.876 to 0.847 for VGG16–Neural Network, from 0.876 to 0.834 for VGG19–SVM, and from 0.872 to 0.829 for InceptionV3–SVM, and that the balanced accuracy (0.813–0.839) is 0.04–0.07 lower than the corresponding CA (0.878–0.882). Second, the minority class was predicted less often than it occurred, which produced the low healthy-leaf Recall described in Section 3.7.

MCC, which accounts for all cells of the confusion matrix and is less affected by class proportions than CA or F1-score [30], provides a more conservative summary (0.844–0.849 for these three combinations, compared with CA values of 0.878–0.882). For this reason, MCC and macro-averaged metrics should be considered together with CA when comparing the pipelines. Future work should collect additional healthy-

leaf images or apply data augmentation, oversampling, or class-weighted training, and should report macro-averaged metrics ; although the stratified 5-fold design used here ensured that every validation fold contained healthy images, each fold contained only seven or eight of them, so repeated stratified cross-validation is needed to obtain stable estimates for the minority class.

Overall Discussion

Overall, the experimental findings demonstrate that the selection of the image embedding architecture substantially affects the subsequent performance of the classification algorithms. VGG16 provided a useful baseline, VGG19 produced improved peak performance when combined with Neural Network, and InceptionV3 generated the highest overall evaluation results when combined with Logistic Regression. The InceptionV3–Logistic Regression configuration, with an AUC of 0.986, CA of 0.911, F1-score of 0.910, Precision of 0.911, Recall of 0.911, and MCC of 0.886, was the strongest configuration among those evaluated, and InceptionV3 also produced the highest mean CA and MCC across the three classifiers.

At the same time, the results demonstrate that no classifier consistently produced the highest performance across all embedding architectures. This reinforces the importance of evaluating feature extraction and classification as an integrated pipeline. The findings therefore extend the use of pre-trained CNNs beyond standalone deep-learning classification by demonstrating that CNN-derived image embeddings can be effectively combined with conventional machine-learning classifiers.

The confusion matrix analysis further reveals that strong aggregate performance does not eliminate class-specific classification difficulties. Brown spot and rice hispa generally showed high numbers of correct predictions, whereas healthy leaves remained comparatively difficult to distinguish, being the most misclassified class in all three confusion matrices and most often confused with rice hispa. Misclassification was also observed between bacterial blight and leaf blast. These results suggest that visual similarity among categories remains an important challenge in image-based rice leaf classification [20]–[25]. Because the healthy class is also the smallest class, the weighted metrics in Table 1 partly mask this weakness, as shown by the lower macro-averaged F1-score and balanced accuracy in Table 5.

Taken together, the findings address the research gap by providing a direct comparison of multiple pre-trained CNN embeddings and classification algorithms within the same analytical workflow. Rather than assuming that a deeper embedding architecture or more complex classifier will automatically produce better results, this study demonstrates that performance depends on the specific embedding–classifier combination. Within the configurations and dataset evaluated in this study, InceptionV3 combined with Logistic Regression produced the highest overall evaluation results, while the confusion matrix analysis identifies class-specific limitations that should be considered when interpreting the model's performance. Because the confidence intervals of the three highest-ranked combinations overlap, this selection should be confirmed using repeated cross-validation and an independent, field-acquired test set.

4. Conclusion

This study evaluated the performance of three pre-trained CNN architectures, VGG16, VGG19, and InceptionV3, as image embedding methods combined with Support Vector Machine, Logistic Regression, and Neural Network for classifying five rice leaf conditions in a public Kaggle dataset of 517 images. The results demonstrate that classification performance varies according to the combination of embedding architecture and classification algorithm. Among the nine configurations evaluated, InceptionV3 combined with Logistic Regression achieved the highest overall performance, with an AUC of 0.986, Classification Accuracy of 0.911, F1-score of 0.910, Precision of 0.911, Recall of 0.911, and MCC of 0.886, although the 95% confidence interval of its accuracy (0.883–0.933) overlapped with those of the next two best

combinations. Within this dataset, these findings indicate that features extracted from pre-trained CNNs, without end-to-end training, can be combined with a conventional machine-learning classifier to achieve high multiclass classification performance. They do not, however, establish the approach as a practical solution for the diagnosis or early detection of rice diseases in the field.

The study contributes a comparative evaluation of multiple CNN embedding–classifier combinations within the same analytical workflow. The findings also show that the best-performing classifier depends on the feature representation used, as Neural Network performed particularly well with VGG19, whereas Logistic Regression achieved the highest overall performance when combined with InceptionV3. However, the confusion matrix analysis indicates that classification performance was not uniform across all classes, with healthy leaves remaining more difficult to classify than several disease categories (Recall of 0.405–0.595 in the confusion matrices examined), most often being misclassified as rice hispa.

Several limitations should be considered when interpreting these results. First, the experiment used a single public dataset with a limited number of images (517), and the healthy class contained only 37 images, so the reported metrics, particularly those for healthy leaves, are sensitive to a small number of predictions. Second, the images were taken from an existing online collection and the models were not validated under field conditions, where variation in illumination, background, camera device, leaf orientation, growth stage, and the co-occurrence of several diseases may reduce performance. Third, performance was estimated only by stratified 5-fold cross-validation on the same dataset, without an independent external test set; the dataset does not distinguish disease stages, so the ability to detect early symptoms was not evaluated; and computational efficiency was not assessed. Consequently, the results should be regarded as evidence of feasibility on this dataset rather than as proof of a field-ready diagnostic or early-detection system.

Future research should test the embedding–classifier pipelines on larger and more diverse datasets, particularly images acquired directly from rice fields under real conditions in different regions, seasons, and growth stages and with different devices, and should evaluate the models on independent field-acquired test sets. Such studies should also address class imbalance through additional data collection or class-weighted training, report fold-level variation using repeated cross-validation, include labelled images of early-stage symptoms, and assess computational cost on the mobile or low-resource devices for which the approach is intended before it is considered for agricultural decision support.

5. References

[1] C. Yan *et al.*, “Rice Leaf Disease Image Enhancement Based on Improved CycleGAN,” *Smart Agriculture*, vol. 6, no. 6, pp. 96–108, Nov. 2024, doi: 10.12133/J.SMARTAG.SA202407019.

[2] M. E. Rana, V. A. Hameed, I. K. Y. Eng, H. K. Tripathy, and S. Mallik, “Harnessing Artificial Intelligence for Sustainable Rice Leaf Disease Classification,” *Front. Plant Sci.*, vol. 16, Sep. 2025, doi: 10.3389/fpls.2025.1594329.

[3] P. Preethi, R. Swathika, S. Kaliraj, R. Premkumar, and J. Yogapriya, “Deep Learning–Based Enhanced Optimization for Automated Rice Plant Disease Detection and Classification,” *Food Energy Secur.*, vol. 13, no. 5, Sep. 2024, doi: 10.1002/fes3.70001.

[4] T.-H. Nguyen, T.-N. Nguyen, and B.-V. Ngo, “A VGG-19 Model with Transfer Learning and Image Segmentation for Classification of Tomato Leaf Disease,” *AgriEngineering*, vol. 4, no. 4, pp. 871–887, Oct. 2022, doi: 10.3390/agriengineering4040056.

[5] U. Shakeel *et al.*, “Enhanced Rice Leaf Disease Classification Via Contour-Driven Segmentation and Optimized Deep Transfer Learning Architectures,” *PLoS One*, vol. 21, no. 5, p. e0348290, May 2026, doi: 10.1371/journal.pone.0348290.

[6] Z. Zhang, Q. Gao, L. Liu, and Y. He, "A High-Quality Rice Leaf Disease Image Data Augmentation Method Based on a Dual GAN," *IEEE Access*, vol. 11, pp. 21176–21191, 2023, doi: 10.1109/ACCESS.2023.3251098.

[7] J. Liu and X. Wang, "Early Recognition of Tomato Gray Leaf Spot Disease Based on Mobilenetv2-Yolov3 Model," *Plant Methods*, vol. 16, no. 1, p. 83, Dec. 2020, doi: 10.1186/s13007-020-00624-2.

[8] A. T. Adiana, J. Jumadi, and E. Nurlatifah, "Klasifikasi Penyakit Pada Daun Padi Menggunakan Model Hybrid CNN-SVM," *Smatika Jurnal*, vol. 15, no. 02, pp. 258–267, Dec. 2025, doi: 10.32664/smatika.v15i02.1548.

[9] S. Sheila, I. P. Sari, A. B. Saputra, M. K. Anwar, and F. R. Pujianto, "Deteksi Penyakit Pada Daun Padi Berbasis Pengolahan Citra Menggunakan Metode Convolutional Neural Network (CNN)," *Multietnics*, vol. 9, no. 1, pp. 27–34, Apr. 2023, doi: 10.32722/multinetics.v9i1.5255.

[10] Y. Istikorini *et al.*, "Penyuluhan Hama dan Penyakit pada Tanaman Padi dan Hortikultura di Desa Cihamerang, Sukabumi," *Jurnal Pusat Inovasi Masyarakat*, vol. 7, no. 1, pp. 102–115, May 2025, doi: 10.29244/jpim.7.1.102-115.

[11] A. C. Milano, "Klasifikasi Penyakit Daun Padi Menggunakan Model Deep Learning Efficientnet-B6," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 1, Jan. 2024, doi: 10.23960/jitet.v12i1.3855.

[12] N. A. Afiah, S. Syahrullah, R. Ardiansyah, R. Laila, and R. Pohontu, "Application of VGG16 Architecture in Wood Type Classification Using Convolutional Neural Network," *Jurnal Teknik Informatika (Jutif)*, vol. 6, no. 1, pp. 335–344, Feb. 2025, doi: 10.52436/1.jutif.2025.6.1.3874.

[13] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, International Conference on Learning Representations, ICLR, Sep. 2014. Accessed: Sep. 16, 2026. [Online]. Available: https://arxiv.org/pdf/1409.1556

[14] R. K. Rachman, D. R. I. M. Setiadi, A. Susanto, K. Nugroho, and H. M. M. Islam, "Enhanced Vision Transformer and Transfer Learning Approach to Improve Rice Disease Recognition," *Journal of Computing Theories and Applications*, vol. 1, no. 4, pp. 446–460, Apr. 2024, doi: 10.62411/jcta.10459.

[15] W. Yu, P. Zhou, S. Yan, and X. Wang, "InceptionNeXt: When Inception Meets ConvNeXt," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun. 2024, pp. 5672–5683. doi: 10.1109/CVPR52733.2024.00542.

[16] M. Dutta, D. Gupta, Y. Gulzar, M. S. Mir, C. W. Onn, and A. B. Soomro, "Leveraging Inception V3 for Precise Early and Late Blight Disease Classification in Potato Crops," *Traitement du Signal*, vol. 41, no. 2, pp. 705–715, Apr. 2024, doi: 10.18280/ts.410213.

[17] Y. Zhang, J.-P. Allem, J. B. Unger, and T. B. Cruz, "Automated Identification of Hookahs (Waterpipes) on Instagram: An Application in Feature Extraction Using Convolutional Neural Network and Support Vector Machine Classification," *J. Med. Internet Res.*, vol. 20, no. 11, p. e10513, Nov. 2018, doi: 10.2196/10513.

[18] W. Habibi and I. Yuadi, "Evaluating Logistic Regression and SVM for Image Analysis Using VGG-16, VGG-19, and Inception V3 Features," *Jurnal Ilmiah Teknologi dan Rekayasa*, vol. 30, no. 2, pp. 1–10, 2025, Accessed: Sep. 16, 2026. [Online]. Available: https://ejournal.gunadarma.ac.id/archive/index.php/tekno/article/view/14056

[19] T. Fawcett, "An Introduction to ROC Analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006, doi: 10.1016/j.patrec.2005.10.010.

[20] R. R. Burhanuddin, "Klasifikasi Penyakit Padi Melalui Citra Daun Menggunakan Metode Naive Bayes," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4012.

[21] J. J. Hidayat, C. Setyowati, and A. P. Werdana, "Analisis Penyakit pada Daun Padi Menggunakan VGG-16 Transfer Learning dan Teknik Segmentasi K-Mean," *Jurnal Media Infotama*, vol. 21, no. 1, pp. 98–104, Apr. 2025, doi: 10.37676/jmi.v21i1.7313.

[22] A. Walascha *et al.*, "Review Artikel: Inventarisasi Jenis Penyakit yang Menyerang Daun Tanaman Padi (Oryza sativa L.)," *Prosiding Seminar Nasional Biologi*, vol. 1, no. 2, pp. 471–478, 2021, doi: 10.24036/PROSEMNASBIO/VOL1/150.

[23] T. Febriyanto and S. Syofian, "Implementasi Deep Learning Menggunakan Vision Transformer untuk Klasifikasi Penyakit Daun Padi," *Journal TIFDA (Technology Information and Data Analytic)*, vol. 1, no. 2, pp. 34–39, Dec. 2024, doi: 10.70491/tifda.v1i2.47.

[24] P. Novantara, R. L. Firmansyah, and M. Arismawati, "Deteksi Hama Penyakit Daun Padi dengan Menggunakan Teknik Optimasi Deep Lesarning Convolutional Neural Network," *bit-Tech*, vol. 7, no. 3, pp. 975–983, Apr. 2025, doi: 10.32877/bt.v7i3.2284.

[25] R. Agustin, M. Nurlailli, K. P. Yuanda, and B. A. Sudamto, "Deteksi Penyakit Daun Padi Menggunakan MobileNetV2: Pendekatan Deep Learning untuk Meningkatkan Ketahanan Produksi Pangan," in *Prosiding Semnas Inotek (Seminar Nasional Inovasi Teknologi)*, Jul. 2025, pp. 1294–1303. doi: 10.29407/TEVJ0K39.

[26] J. Demšar et al., "Orange: Data Mining Toolbox in Python," *J. Mach. Learn. Res.*, vol. 14, pp. 2349–2353, 2013.

[27] M. Sokolova and G. Lapalme, "A Systematic Analysis of Performance Measures for Classification Tasks," *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, Jul. 2009, doi: 10.1016/j.ipm.2009.03.002.

[28] J. Gorodkin, "Comparing Two K-Category Assignments by a K-Category Correlation Coefficient," *Comput. Biol. Chem.*, vol. 28, no. 5–6, pp. 367–374, Dec. 2004, doi: 10.1016/j.compbiolchem.2004.09.006.

[29] E. B. Wilson, "Probable Inference, the Law of Succession, and Statistical Inference," *J. Am. Stat. Assoc.*, vol. 22, no. 158, pp. 209–212, Jun. 1927, doi: 10.1080/01621459.1927.10502953.

[30] D. Chicco and G. Jurman, "The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, Jan. 2020, doi: 10.1186/s12864-019-6413-7.