Available online at http://scik.org
Commun. Math. Biol. Neurosci. 2025, 2025:43
https://doi.org/10.28919/cmbn/9198
ISSN: 2052-2541

MACHINE LEARNING BASED CLASSIFICATION OF RICE LEAF DISEASES: A COMPARATIVE STUDY

RAFFAEL HIZQYA BAKHTIAR, YASI DANI, DANI SUANDI*

Computer Science Department, School of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

Abstract. Rice is a staple food for most of the world's population, making its sustainable production essential. However, undetected outbreaks of rice leaf diseases often result in reduced production due to crop failure. Accurate identification of rice leaf diseases is an important step in effective disease management. Machine learning (ML) algorithms can be an effective solution for early classification of rice leaf diseases based on available data. This study compares the performance of five ML algorithms, namely K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest, Decision Tree, and Naïve Bayes, in classifying rice leaf diseases. The evaluation uses metrics such as accuracy, precision, sensitivity, F1 score, and training and prediction time. In addition, the analysis also includes ROC-AUC values and statistical tests such as Friedman test and Nemenyi post hoc test to determine the best algorithm significantly. The results showed that Random Forest had the best performance among the five algorithms with a ROC-AUC of 0.92 and an accuracy of 0.94, however, it is noteworthy that this particular algorithm requires a longer duration for both training and prediction compared to its counterparts. Ultimately, this study serves to provide invaluable insights and guidance for the selection of the most optimal machine learning algorithm, thereby facilitating more efficient and sustainable practices in the classification of rice leaf diseases, which is crucial for the advancement of agricultural productivity and food security.

Keywords: machine learning; rice leaf diseases; performance metrics evaluation; statistical test.

2020 AMS Subject Classification: 68T01, 68T20, 92B20, 97P80.

*Corresponding author

E-mail address: dani.suandi@binus.ac.id

Received February 16, 2025

1. INTRODUCTION

Machine learning (ML) has revolutionized numerous industries, offering innovative solutions for early and accurate detection of issues, which in turn leads to better decision-making and improved efficiency across various domains [1]. Agriculture, a cornerstone of human civilization, has greatly benefited from advancements in artificial intelligence (AI) and ML techniques. These technologies are increasingly applied in crop management, particularly for early disease detection and prevention, ensuring improved yield and quality [2, 3]. By reducing dependency on harmful chemicals and improving resource utilization, ML promotes sustainable farming practices, contributing to healthier crops and higher yields [4].

Rice, as a staple food for more than half of the global population, is vital for food security. Its consumption is projected to grow by 0.9% annually over the next decade, with Asia contributing to 65% of this increase due to population growth [5] shown in Figure 1. However, rice production faces significant challenges from diseases like bacterial leaf blight, brown spot, and blast, which can severely impact both yield and quality [6]. Detecting these diseases at an early stage is crucial to safeguarding production and preventing economic losses for farmers [7]. Despite advancements, achieving high accuracy in disease detection remains challenging due to the diverse characteristics of rice diseases [8, 9].

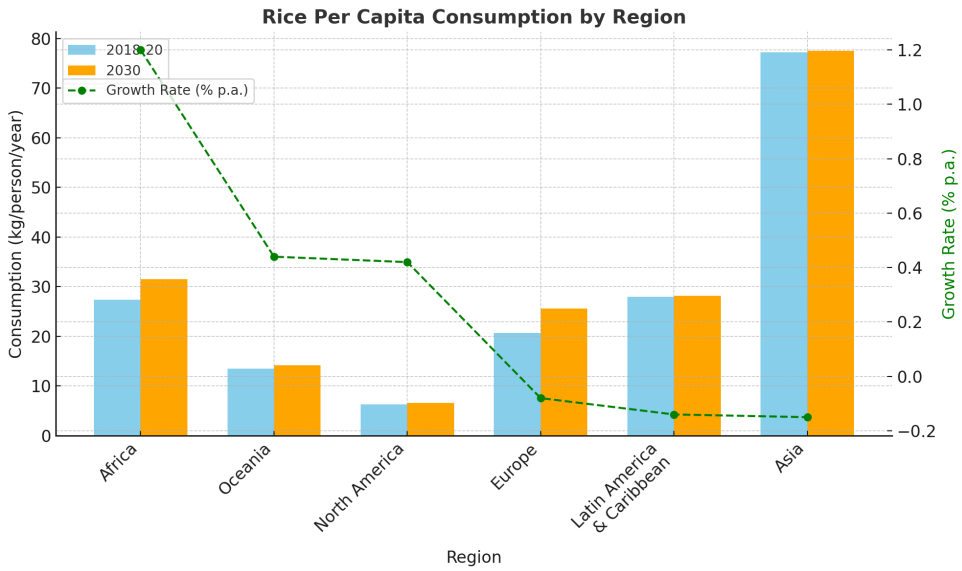

FIGURE 1. Rice per capita consumption by region.

However, amidst the optimism for increased rice production, a significant challenge persists in the form of highly susceptible range of diseases that can substantially impact both yield and quality of these crops. Due to limitations in human resources and expertise, farmers often struggle to effectively address crop diseases such as brown spot, bacterial leaf blast, false smut, and stem rot [10, 11]. This not only leads to incomplete disease coverage but also incurs high costs. In response to this challenge, researchers have increasingly turned to the advancements ML algorithms for the classification of rice leaf diseases. The algorithms here show promise in accurately identifying and categorizing different rice leaf diseases. Damaged leaves affect the plant's growth and grain quality, leading to poor yields.

It is clear that the demand for rice will continue to rise in the future due to increasing needs in various parts of the world. To prevent global food insecurity, it is crucial to ensure that the quality of rice production is consistently safeguarded. One effective way to avoid such issues is by protecting paddy leaves from diseases that could compromise the quality of the harvested rice. Image processing, when developed properly, can become an essential solution for detecting and providing early diagnoses of diseased paddy leaves. By identifying specific details about the disease, it enables farmers to take immediate and informed action to protect their crops. When diagnosing a disease, the most important task is to ensure the accuracy of the diagnosis by thoroughly analyzing all the symptoms. Identifying the exact cause of the paddy's illness is our core mission, which requires the machine to be highly precise in recognizing and segmenting the image. It must specifically focus on the areas of the leaf that indicate signs of sickness to deliver reliable results. This paper aims to implement an image processing system to enhance production quality using several method of preprocessing, feature extraction, and traditional machine learning algorithm. Hence, enabling farmers to achieve higher yields during harvest.

Recent years have seen a surge in research focusing on ML-based approaches for rice disease detection. Traditional techniques such as support vector machines (SVM), K-nearest neighbor (KNN), and Decision Tree have demonstrated considerable success in classifying diseased leaves based on image data [12, 13, 14]. Phadikar et al. (2013) compared Naïve Bayes and SVM classifier for finding better accuracy in detection [8]. The images used were preprocessed

into a grayscale image and Otsu's image segmentation in order to prevent misclassification due to the shadow effect and color distortion. Result of this comparison favored the Bayes classifier with the highest success rate at 79.5%. Pinki et al. (2017) proposed a method using SVM classifier with preprocessed images with image segmentation using K-means clustering technique [9]. Later after classified, the model will give suggestion of the related pesticides or fertilizers. Result in this study presented an outcome of a high overall accuracy at 92.06%. Joshi et al. (2016) applied MDC and KNN classifier to detect 4 types of diseases. Preprocessed using skel operation to acquire a morphological cropped infected area of the images [12]. Each classifier result in ranging accuracy according to the 4 diseases types. For MDC the accuracy ranges from 87.87% to 90.62% and for KNN it ranges from 84.61%-90.62%. Prajapati et al. (2017) identified rice leaf diseases using SVM learning approach [15]. The study preprocessed images with converting from RGB images to HSV images, disease clustering with K-means clustering using thresholding. The result presented accuracy in 93%. Ahmed et al. (2019) developed a ML-based system to detect rice leaf diseases like bacterial blight, brown spot, and leaf smut [7]. They utilized models such as SVM, Decision Tree, and KNN, employing features like color, texture, and shape for classification. A unique aspect of their work is the combination of multiple traditional ML models with handcrafted feature extraction techniques, achieving over 90% accuracy while maintaining computational efficiency. Hasan et al. (2019) conducted a performance analysis of various machine learning models for rice disease detection, focusing on diseases such as bacterial blight and brown spot [16]. They evaluated models like SVM, Random Forest, and KNN, comparing their accuracy and efficiency. A distinctive feature of their work is the thorough evaluation and comparison of multiple traditional machine learning models to determine the optimal approach for rice disease detection. In their research they succeeded to achieve 87.5% accuracy. Katuwawala et al. (2023) proposed a system for detecting and classifying rice plant diseases using image processing techniques using KNN and SVM [17]. Their approach involved preprocessing, feature extraction, and classification of diseases like bacterial blight and blast. A unique aspect of their work is the integration of image processing for enhanced feature extraction, leading to improved accuracy in disease classification. This leads to an accuracy of 94.89% for KNN and 94.04% for SVM.

Advanced methods such as Random Forest and ensemble learning techniques have shown robust performance in multiclass disease classification tasks. Studies by Saminathan et al. (2023) highlighted the effectiveness of Random Forest classifiers in detecting multiple diseases, leveraging features like color and texture for improved accuracy. Similarly, Saha et al. (2021) proposed a novel feature extraction method using intensity moments, which was combined with Random Forest and yielded a classification accuracy of 91.47% [6]. These advancements underscore the potential of integrating traditional ML methods with innovative feature engineering to enhance classification performance.

The application of deep learning (DL) further enhances the capabilities of rice disease detection systems [18, 19, 20]. Convolutional neural networks (CNNs), such as DenseNet and ResNet, have been successfully used to classify diseases with high precision and scalability [21]. For instance, Arinichev et al. (2021) demonstrated the superior performance of DenseNet in identifying rice diseases, achieving an accuracy of 95.6% [22]. The integration of DL with image processing enables real-time, automated detection, offering scalable solutions for large-scale agricultural deployments [23, 24, 25].

This paper is organized into four main sections. It begins by introducing the background, the need for the study, and the contributions of the research. Section II provides the materials and methodology for detecting rice leaf diseases starting from image data pre-processing, how to extract color features, ML methods used to classify rice leaf diseases, and then how to evaluate the performance of these models. Section III presents and discusses the results obtained. Finally, the last section concludes the research presented in this paper.

2. MATERIALS AND METHODOLOGIES

The methodology for rice leaf disease classification is structured into four primary steps, as illustrated in Figure 2. The first step involves data preprocessing, which standardizes image formats and minimizes variability in the input data. Next, color features are extracted from the images to identify unique characteristics that differentiate diseased leaves from healthy ones. In the third step, a machine learning model is applied to classify the images based on the extracted features. Finally, the model's performance is evaluated to ensure its accuracy, effectiveness, and reliability in detecting diseases within the images.

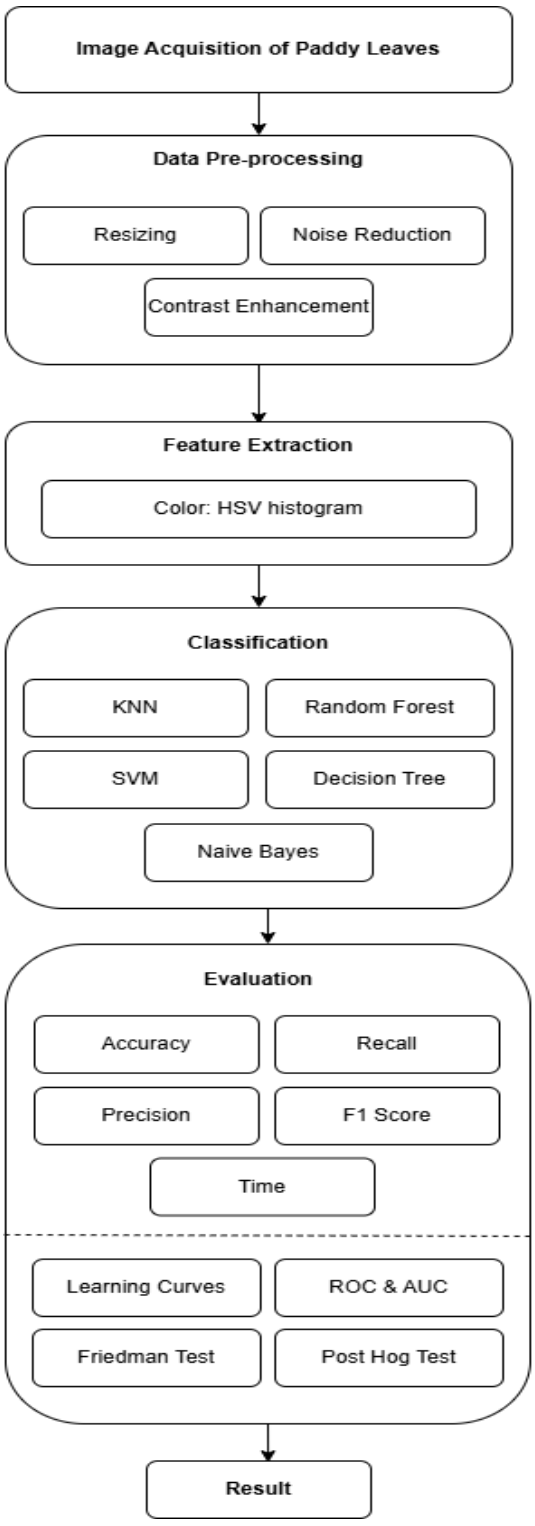

FIGURE 2. Overview of the steps of the proposed model.

2.1. Image Acquisition. The rice leaf image dataset used in this study was compiled from two publicly available Kaggle repositories: Paddy Leaf Disease [26] and Rice Leaf Diseases [27], combined with existing images from the authors' repository. To enhance analytical precision, high-resolution images were divided into smaller patches, enabling granular analysis of disease-specific features. The dataset was partitioned into training and testing subsets, with stratification to maintain proportional representation of disease classes. The final distribution of samples across categories is summarized in Table 1.

TABLE 1. Dataset of images (I) number of original images, (II) number of images used for training, (III) number of images used for testing.

Leaf Diseases	I	II	III
Bacterial Blight	369	296	73
Blast	262	210	52
Brownspot	226	181	45
Leaf Smut	274	220	54
Total	1131	907	224

2.2. Data Preprocessing. Raw datasets often contain missing values, redundant information, or implausible data, which can compromise the integrity of analytical outcomes. To address these challenges, data preprocessing is employed to cleanse and refine datasets, thereby enhancing their reliability. This study focuses on preprocessing techniques for image data, specifically resizing, background removal, and noise reduction.

First, images are resized to a standardized resolution while preserving their aspect ratio, ensuring uniform dimensions for computational consistency. Subsequent noise reduction is achieved via Gaussian blur, a spatial averaging technique that prioritizes proximal pixels to suppress high-frequency noise while retaining critical structural features.

Finally, contrast limited adaptive histogram wqualization (CLAHE) is applied to enhance image contrast. Unlike conventional histogram equalization, CLAHE operates on localized image regions and imposes contrast constraints to prevent noise amplification. This method improves visibility in underexposed or overexposed areas, yielding balanced and interpretable

images. Collectively, these steps optimize data quality, facilitating robust feature extraction and analysis.

2.3. Feature Extraction. In the feature extraction process, this study employs the HSV histogram as the sole primary feature for dataset analysis, selected for its effectiveness in class distinction, computational efficiency, and robustness to lighting variations. The HSV model consists of three components: hue (H), which identifies dominant colors for object differentiation; saturation (S), which quantifies color intensity to distinguish vivid from dull regions; and value (V), which represents brightness, providing insights into overall illumination. To enhance classification performance and stability, image features undergo min-max normalization before being processed by the classifier. The image conversion process is illustrated in Figure 3.

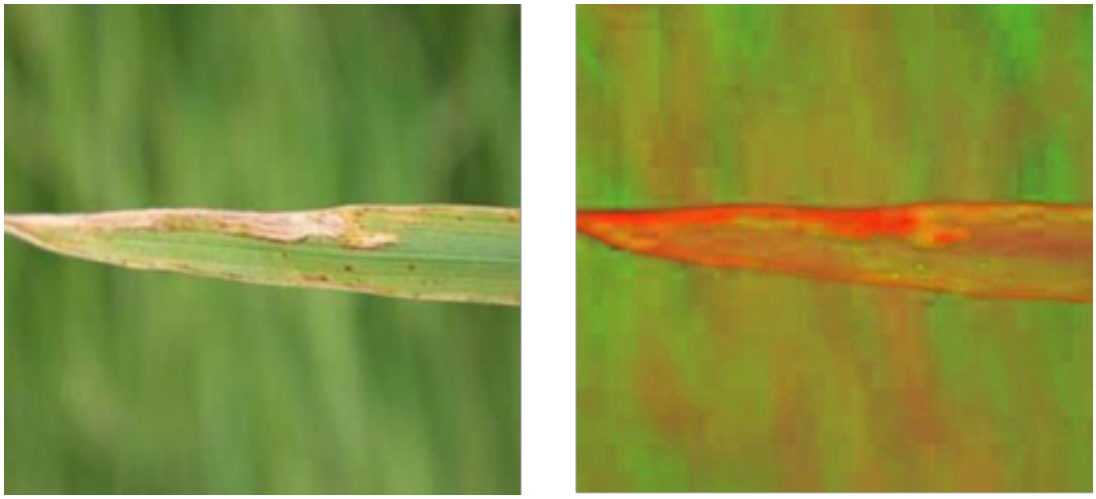

FIGURE 3. RGB to HSV color channel.

2.4. Classification using Machine Learning Algorithms. The subsequent phase involves predictive analysis using extracted features from the dataset. Multiple supervised machine learning algorithms: KNN, SVM, Naïve Bayes, Random Forest, and Decision Tree are employed to classify rice leaf diseases based on training data. These models will predict whether a leaf exhibits a specific disease and will later be evaluated and compared based on prediction accuracy.

KNN, a non-parametric algorithm suitable for classification or regression tasks [28], operates by calculating the Euclidean distance between a query sample and all training instances in the

feature space. The algorithm identifies the 'k' nearest neighbors and assigns the disease class via major voting. the predicted class $\hat{y}$, derived directly from this process, is expressed in (1). As an instance-based learner, KNN requires no explicit training, relying instead on proximity-based inference for classification.

$$\hat{y} = \arg\max_{c} \sum_{i \in N_k(x)} \delta(y_i, c), \tag{1}$$

where $N_k(x)$ is set of KNN of x, y_i is class label of the i-th neighbor, c is candidate class, $\delta(y_i, c)$ is indicator function that equals 1 if $y_i = c$, and 0 otherwise.

SVM is a supervised learning algorithm commonly used for classification tasks. As a non-probabilistic binary linear classifier, SVM categorizes data points into one of two classes by evaluating their position relative to a decision boundary [29, 30]. For linearly separable data, SVM identifies an optimal hyperplane that maximizes the margin—the distance between the closest training samples (termed support vectors) of each class. This hyperplane is defined by the linear equation:

$$w \cdot x + b = 0, \tag{2}$$

where w is the weight vector (perpendicular to the hyperplane), x is the feature vector of the new data point, b is the bias term. If the result of $w \cdot x + b$ is positive, the data point is classified as belonging to one class, and if the result is negative, the data point is classified as belonging to the other class.

The Naïve Bayes model works on the paddy disease dataset by calculating the likelihood of each feature for all disease classes using the training data [31]. The Bayes' Theorem assumes that the features are independent of each other and multiplies the probabilities of each feature given the classes. The class with the highest probability is then chosen as the final prediction. This makes Naïve Bayes a quick and straightforward method for classification, especially when dealing with lots of features, as it relies on probabilistic reasoning while maintaining computational simplicity. The Naïve Bayes classifier is based on Bayes' theorem and can be expressed as:

$$P(C_k \mid x_1, x_2, \ldots, x_n) = \frac{P(C_k) \cdot P(x_1, x_2, \ldots, x_n \mid C_k)}{P(x_1, x_2, \ldots, x_n)}. \tag{3}$$

Under the assumption of conditional independence of features given the class, this simplifies to:

$$P(C_k \mid x_1, x_2, \ldots, x_n) \propto P(C_k) \prod_{i=1}^{n} P(x_i \mid C_k), \tag{4}$$

where $P(C_k)$ is prior probability of class C_k, $P(x_1, x_2, \ldots, x_n \mid C_k)$ is likelihood of the features given the class, $P(x_1, x_2, \ldots, x_n)$ is evidence or marginal likelihood (often omitted since it is constant for all classes), $P(C_k \mid x_1, x_2, \ldots, x_n)$ is posterior probability of class C_k given the features. To classify a new instance, compute $P(C_k \mid x_1, x_2, \ldots, x_n)$ for each class C_k, and select the class with the highest posterior probability.

A Decision Tree classifies data by splitting images into smaller subsets based on the most relevant features. By selecting the best suitable attribute at the root, this algorithm partitions the dataset into subsets. The primary goal of these partitions is to unmix the dataset, making it easier to classify [32]. The goal is to reduce the mixture of classes in each subset, making it easier to classify the data. In this implementation, the algorithm uses Gini impurity as the criterion to evaluate the quality of each split, ensuring that the most informative features are chosen at each node. This criterion allows the model to fully explore the data and capture patterns without constraints.

The Gini impurity used is calculated using the following formula:

$$Gini(t) = 1 - \sum_{i=1}^{C} p_i^2, \tag{5}$$

where $Gini(t)$ is the Gini impurity for a given node t, C is the number of classes in the target variable, p_i is the probability (or proportion) of the class i at the node.

The Random Forest model uses the paddy disease dataset by training multiple Decision Tree, each on a random subset of the data and features (a process known as bagging) [33]. Each tree produces a classification output for a new sample. The final output is determined by aggregating the predictions from all trees, typically using majority voting. This ensemble approach reduces overfitting and improves the model's accuracy and robustness in handling complex datasets.

For classification, the Random Forest predicts the class label $\hat{y}$ as the mode (most frequent class) of the predictions from individual Decision Tree:

$$\hat{y} = \text{mode}\left\{h_t(x) \mid t = 1, 2, \ldots, T\right\}, \tag{6}$$

where $h_t(x)$ is prediction of the t-th Decision Tree, T is total number of trees in the forest, mode is function that selects the most frequent class label.

2.5. Evaluation of Machine Learning Model. The classifiers in this study were compared based on performance metrics such as accuracy, sensitivity, and precision. The evaluation will be using confusion matrix and area under curve (AUC).

Using the confusion matrix is needed in evaluating how well the model is performing. It provides a comprehensive summary of the model's predictions compared to the actual class labels in the dataset. The matrix is organized into a table which displays four categories, true positive (TP), true negative (TN), false positive (FP), and false negative (FN). The performance measurement using this matrix will calculate metrics such as accuracy, precision, sensitivity, and F1-score. With this information, important metrics like accuracy, precision, sensitivity, and the F1-score can be calculated. These metrics will assess the model performance, highlighting the strengths and pointing out areas that might need improvement or fine-tuning. The confusion matrix measurements are expressed below [34].

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{7}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$F1 - Score = \frac{2 \times Sensitivity \times precision}{Sensitivity + Precision}. \tag{10}$$

In this research, ROC are employed to visualize the trade-off between a classifier's true positive rate (TPR, sensitivity) and false positive rate (FPR, 1 - precision) across varying classification thresholds. This graphical representation enables a nuanced evaluation of a model's ability to discriminate between positive and negative classes. To complement this analysis, the area under the ROC curve (AUC) is calculated as a scalar performance metric. The AUC quantifies

the classifier's overall diagnostic capability, where a value of 1 denotes perfect separation of classes, while 0.5 reflects random guessing.

To understand more of how well the model worked, we also use learning curves as it provides a critical diagnostic tool for assessing ML model performance as training data scales. These curves plot the relationship between training dataset size and model accuracy (or other performance metrics) for both training and validation sets. By analyzing trends in these curves, researchers can diagnose issues such as underfitting (high bias) or overfitting (high variance), informing adjustments to model complexity or data strategies.

2.6. Statistical Comparison of Models. The Friedman test, a non-parametric statistical method, was employed to determine whether significant differences exist among the performance of multiple models evaluated on the same dataset [35]. The test ranks each model's performance across dataset instances (or cross-validation folds) and analyzes the collective rankings. For this study, it was applied to compare the five ML models, generating two key metrics: a test statistic and a p-value. The test statistic quantifies variability in the models' relative rankings, while the p-value estimates the probability of observing such variability under the null hypothesis (H_0), which assumes no performance differences between models. The alternative hypothesis (H_1) posits that at least one model differs significantly.

A significance threshold of $\alpha = 0.05$ was adopted: if the p-value < 0.05, H_0 is rejected, indicating statistically significant disparities; otherwise, insufficient evidence exists to conclude differences. When the Friedman test detects significant differences, post-hoc analysis using the Nemenyi test [36] is conducted to identify pairwise discrepancies between models. This test adjusts for multiple comparisons to mitigate Type I errors, providing adjusted p-values for each model pair. For instance, a p-value < 0.05 implies a statistically significant difference. Results are visualized through critical difference diagrams, which highlight significant pairwise comparisons, offering granular insights into model performance hierarchies.

3. Result and Discussion

This section presents the comprehensive result of each performance comparison of five traditional ML models: Decision Tree, Random Forest, SVM, Naïve Bayes, and KNN for the classification of rice leaf diseases. The evaluation will cover for multiple dimensions, which include confusion matrix, performance metrics (accuracy, precision, sensitivity, and F1-score), training and prediction times, ROC curves, learning curves, and statistical analyses using the Friedman test and post-hoc comparisons.

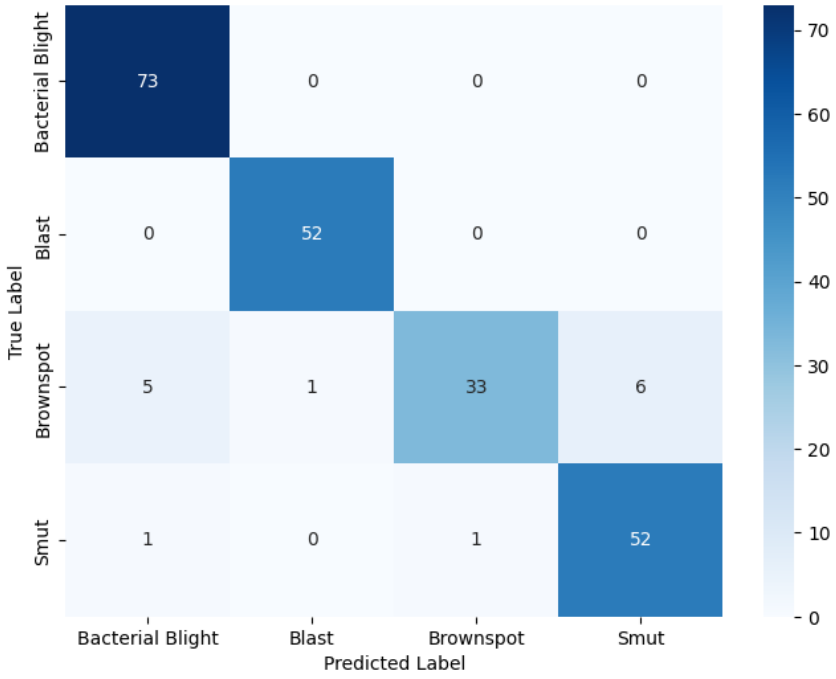

FIGURE 4. Confusion matrix of Random Forest.

Figure 4 shows the confusion matrix of one of the evaluated algorithms, Random Forest. From the results of our research it shows that the Decision Tree achieved 53 true positives for bacterial blight, indicating moderate classification accuracy. The SVM improved upon this result, recording 67 true positives for the same category. The KNN model demonstrated stronger performance, achieving the highest true positives for bacterial blight (70) and smut (51), suggesting superior discriminative capability for these classes. In contrast, the Naïve Bayes model exhibited the weakest performance, with frequent misclassifications evidenced by elevated false

positives across multiple categories. The Random Forest model outperformed all others, attaining 73 true positives for bacterial blight and 52 for smut, while maintaining balanced classification accuracy across all disease types.

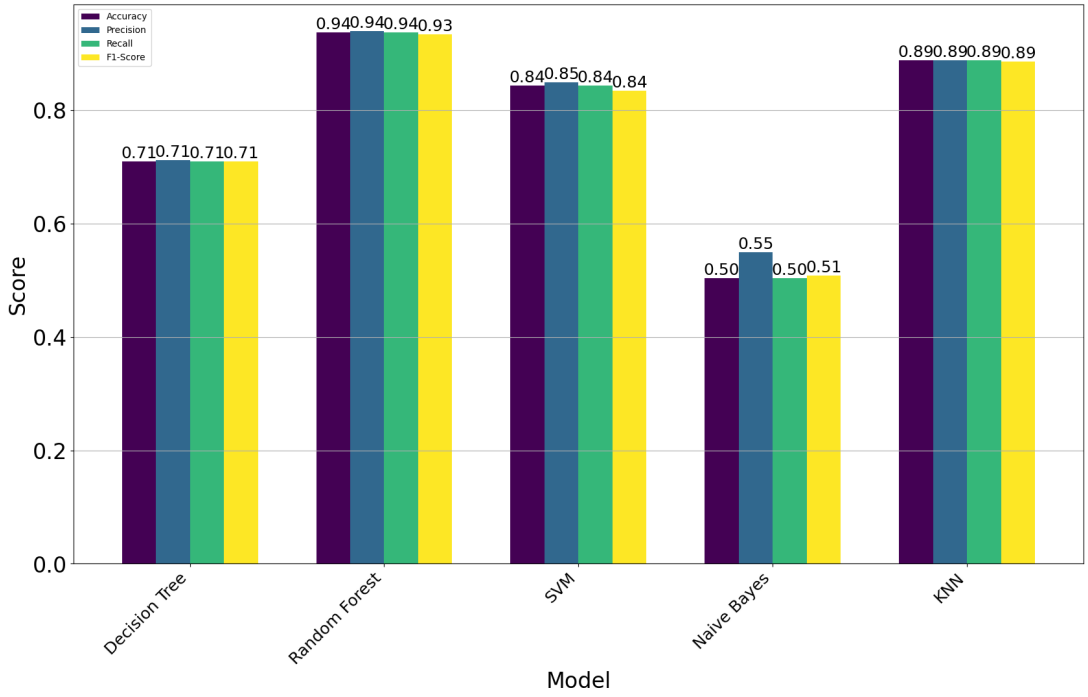

FIGURE 5. Performance metrics comparison.

Figure 5 presents a bar chart comparing the performance of the five models across four evaluation metrics: accuracy, precision, sensitivity, and F1-score. The Decision Tree model achieved a consistent score of 0.71 across all metrics. The SVM model recorded accuracy and precision scores of 0.77, while its sensitivity and F1-score were 0.76. The KNN model demonstrated strong performance, achieving a uniform score of 0.89 across all metrics. In contrast, the Naïve Bayes model exhibited the lowest performance, with an accuracy of 0.55, precision and sensitivity at 0.50, and an F1-score of 0.51. Among all models, the Random Forest model achieved the highest performance, with scores of approximately 0.94 across all metrics.

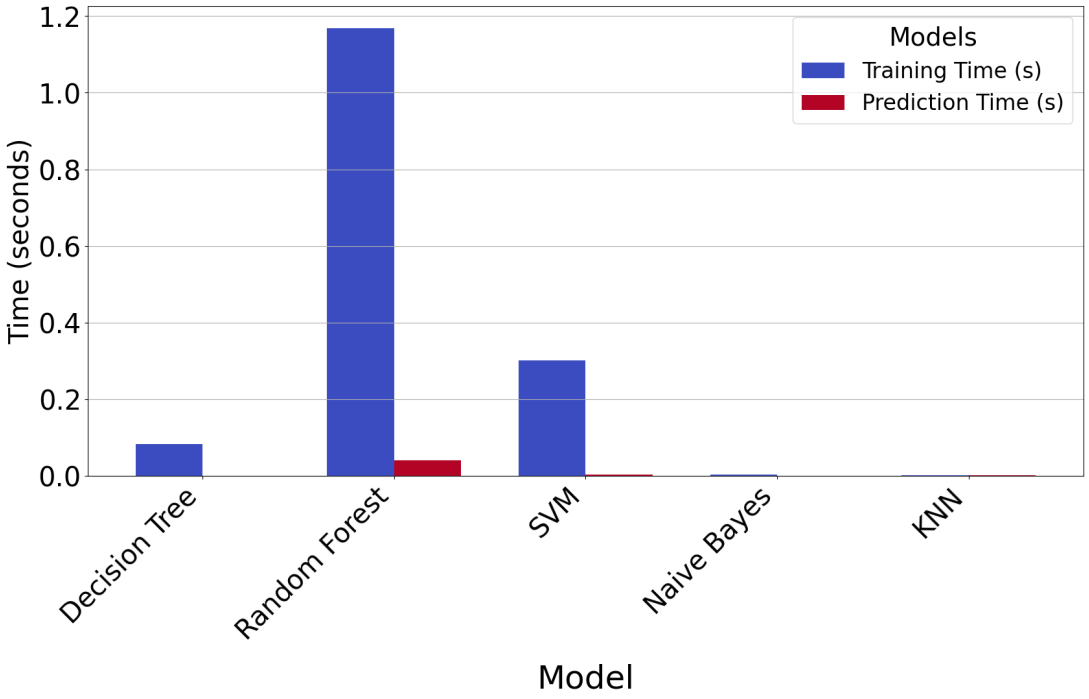

FIGURE 6. Training and prediction time comparison.

Figure 6 illustrates a comparison of the training and prediction times of five machine learning models. The Random Forest model exhibited the longest training time, significantly exceeding that of the other models. This aligns with its ensemble structure, as it involves training multiple Decision Trees and aggregating their outputs. The SVM model required a moderate amount of training time, reflecting the computational complexity involved in optimizing the hyperplane in high-dimensional spaces. In contrast, Decision Tree, Naïve Bayes, and KNN demonstrated relatively short training times, with Naïve Bayes being the fastest due to its straightforward probabilistic approach and the absence of iterative optimization. Prediction times were generally low across all models, with no notable variations. While KNN had the shortest training time overall, it exhibited slightly higher prediction times compared to other models. This is attributed to its instance-based approach, where predictions rely on searching through the dataset for the nearest neighbors. Meanwhile, Naïve Bayes and Decision Tree achieved the fastest prediction times, making them well-suited for real-time applications.

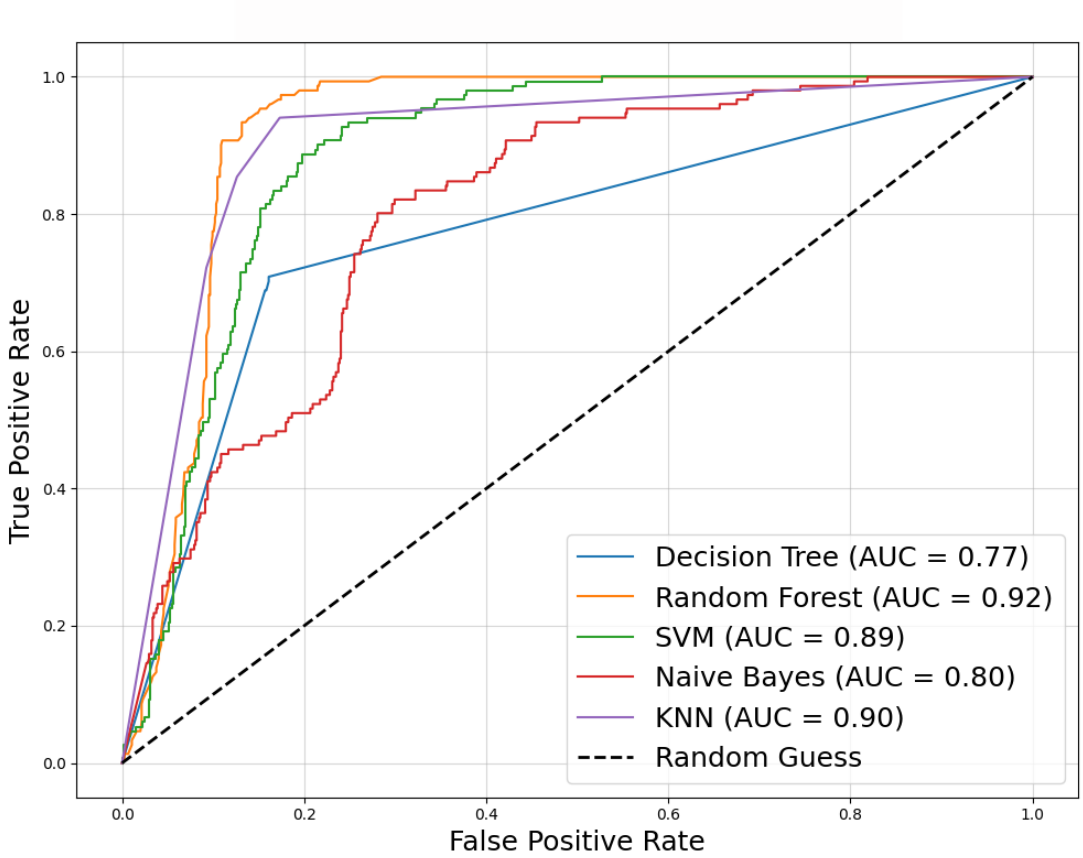

FIGURE 7. ROC curve comparison across models.

Figure 7 presents the ROC curves for the five machine learning models, along with their respective AUC values. The AUC score is a key indicator of a model's ability to differentiate between classes. The Naïve Bayes model attained an AUC of 0.80, signifying moderate performance, though its assumption of feature independence limits its effectiveness in handling complex decision boundaries. The SVM model achieved an AUC of 0.87, placing it slightly below KNN and Random Forest. KNN demonstrated strong classification capability with an AUC score of 0.90. Conversely, the Decision Tree model recorded the lowest AUC at 0.77, indicating a weaker ability to distinguish between disease classes. Among all models, Random Forest exhibited the highest performance, attaining an AUC of 0.92, highlighting its superior classification ability with a high true positive rate and a low false positive rate.

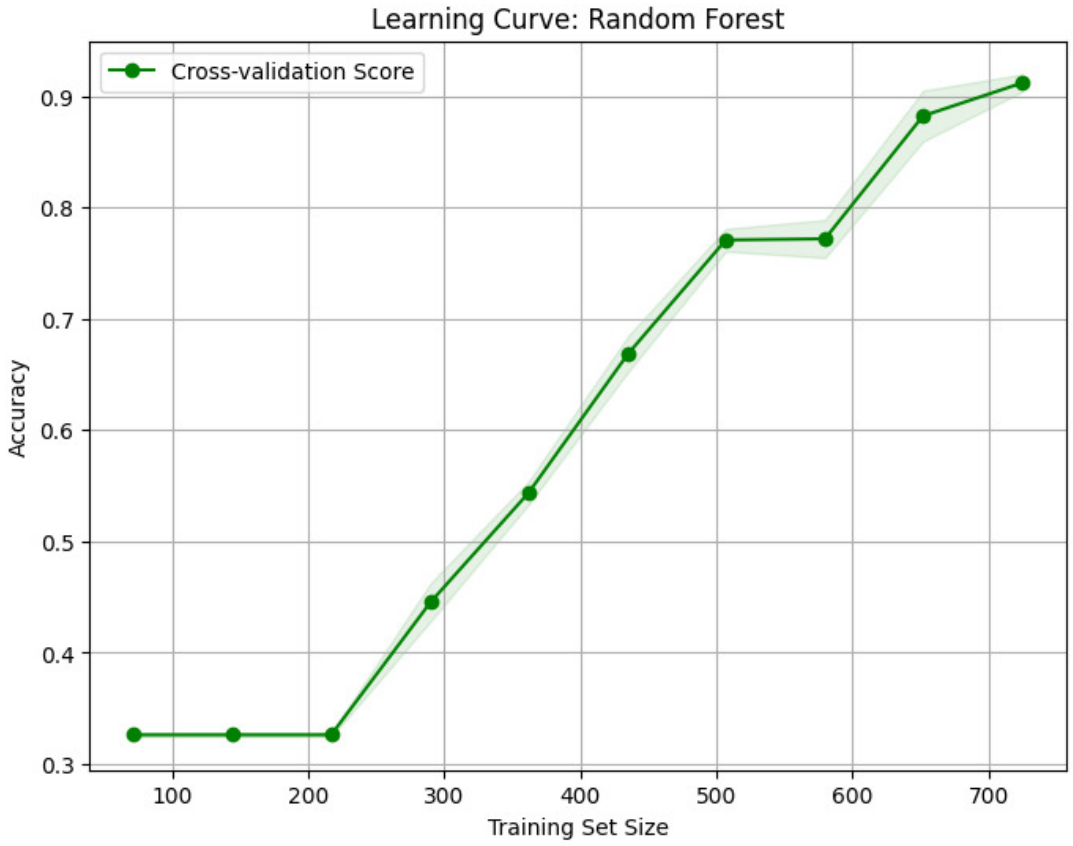

FIGURE 8. Random Forest learning curve.

Figure 8 illustrates the learning curve for the Random Forest model, showing the relationship between training set size and model accuracy. The x-axis represents the number of training samples, while the y-axis indicates accuracy based on cross-validation scores. Initially, with smaller training set sizes, accuracy remains low, fluctuating around 0.3. However, as the training set size increases beyond 200 samples, accuracy begins to rise steadily, demonstrating the model's ability to learn more effectively with additional data. Around 500 samples, the accuracy reaches approximately 0.75, followed by a gradual improvement beyond 0.9 as more training samples are incorporated. The shaded region around the curve represents variability in the cross-validation scores, which narrows as the dataset grows, indicating increased model stability. Overall, this learning curve suggests that Random Forest benefits significantly from larger training datasets, leading to improved performance and generalization.

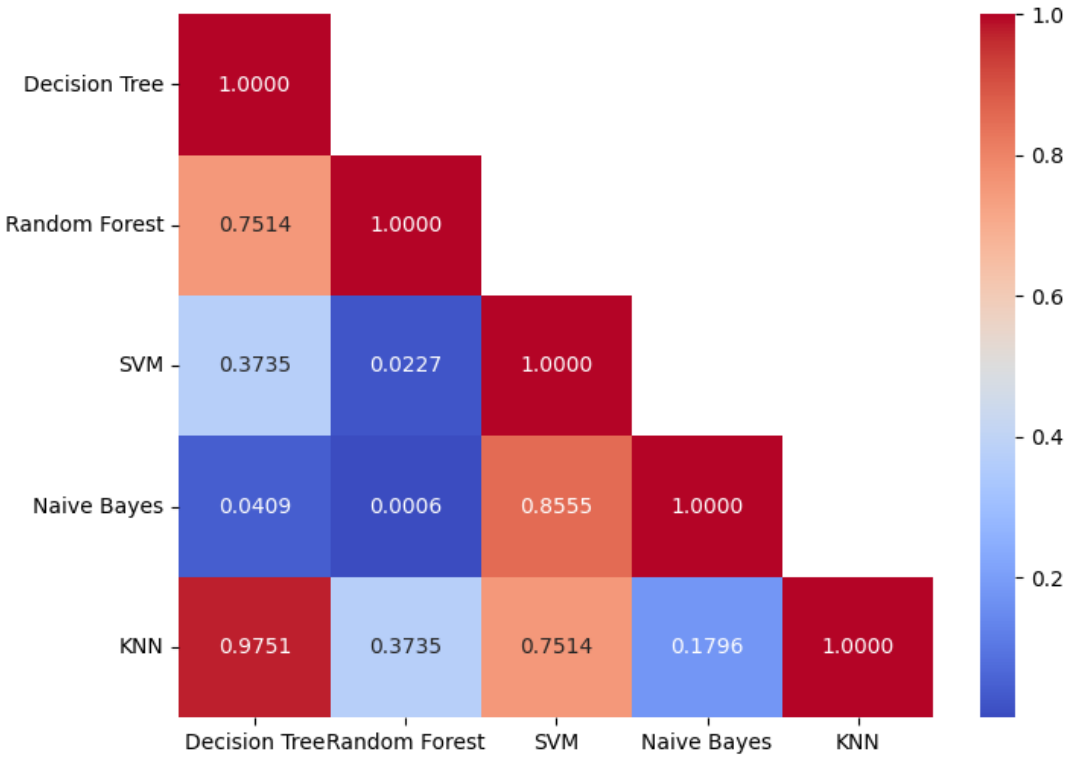

FIGURE 9. Post-hoc Nemenyi test results.

To further evaluate model performance, statistical tests were conducted using the Friedman and Nemenyi tests. The Friedman test identified significant differences among the classifiers ($\chi^2 = 19.36,\ p = 0.0007$) indicating that the models did not perform equally. The post-hoc Nemenyi test (Figure 9) provided insights into pairwise differences between classifiers. Specifically, the Random Forest model demonstrated a statistically significant advantage over Naïve Bayes ($p = 0.0006$) and SVM ($p = 0.0227$), confirming its superior performance. However, no significant difference was observed between Decision Tree and KNN ($p = 0.9751$), suggesting that these models had similar effectiveness in this classification task. These findings reinforce that Random Forest is the most discriminative model, while Decision Tree and KNN exhibit comparable classification capabilities.

4. CONCLUSION

This study highlights the effectiveness of machine learning in rice disease classification. Among the five models evaluated, Random Forest consistently outperformed the others, demonstrating superior accuracy and robustness in distinguishing rice leaf diseases. The model's high classification performance can be attributed to its ensemble nature, which effectively reduces overfitting and enhances generalization. The statistical significance of these results was confirmed by rigorous analysis, including the Friedman and Nemenyi post hoc tests, which identified Random Forest as the most effective model for this task. The model significantly outperformed Naïve Bayes and SVM, while Decision Tree and KNN showed comparable performance. These results highlight the importance of selecting a well-suited algorithm for disease classification, as different models vary in their ability to handle complex decision boundaries and dataset characteristics. Furthermore, the learning curve analysis showed that increasing the training data improves model performance, with Random Forest showing a steady increase in accuracy as more samples are included. This suggests that larger datasets can further improve classification reliability, making data collection and augmentation critical for future improvements. In practical applications, the efficiency of a model also plays a key role. While Random Forest provided the best classification results, Naïve Bayes and Decision Tree showed the shortest prediction times, making them suitable for real-time disease detection systems where rapid decision making is required. These findings provide valuable guidance for deploying ML-based solutions in precision agriculture, where timely and accurate identification of rice diseases can help reduce crop losses and optimize disease management strategies. Future research could explore deep learning approaches, hybrid models, or advanced feature engineering techniques to further improve classification accuracy. In addition, expanding the dataset to include different environmental conditions and disease variations would strengthen model generalization and ensure its effectiveness in different rice-growing regions. Ultimately, this study demonstrates the potential of machine learning in modern agriculture and provides a foundation for the development of intelligent, automated plant disease monitoring systems.

DATA AVAILABILITY STATEMENT

The Python source code utilized in this study is publicly accessible. The code is accessible via GitHub at https://github.com/maevlava/paddy-leaf-comparison, and may be downloaded free of charge. The repository contains comprehensive instructions for executing the code, along with any requisite dependencies and datasets utilized within the program.

CREDIT AUTHORSHIP CONTRIBUTION STATEMENT

Raffael Hizqya Bakhtiar: Writing – original draft, Investigation, Formal analysis, Visualization. Yasi Dani: Writing – review & editing, Formal analysis. Dani Suandi: Writing – review & editing, Methodology, Formal analysis, Conceptualization, Supervision.

ACKNOWLEDGMENTS

This work is supported by Bina Nusantara University as a part of Bina Nusantara University's BINUS International Research - Basic entitled *Investigation and Formulation of Optimal Strategies for the Control of the Spread of Infectious Diseases in the Agricultural Sector* with contract number: 096/VRRTT/VII/2024 and contract date 02 Juli 2024.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

REFERENCES

[1] C. Janiesch, P. Zschech, K. Heinrich, Machine Learning and Deep Learning, Electron. Mark. 31 (2021), 685–695. https://doi.org/10.1007/s12525-021-00475-2.

[2] S. Aggarwal, M. Suchithra, N. Chandramouli, et al. Rice Disease Detection Using Artificial Intelligence and Machine Learning Techniques to Improvise Agro-Business, Sci. Program. 2022 (2022), 1757888. https://doi.org/10.1155/2022/1757888.

[3] G.K.V.L. Udayananda, C. Shyalika, P.P.N.V. Kumara, Rice Plant Disease Diagnosing Using Machine Learning Techniques: A Comprehensive Review, SN Appl. Sci. 4 (2022), 311. https://doi.org/10.1007/s42452-022-05194-7.

[4] N. Parven, M. Rashiduzzaman, N. Sultana, et al. Detection and Recognition of Paddy Plant Leaf Diseases Using Machine Learning Technique, Int. J. Innov. Technol. Explor. Eng. 9 (2020), 634–638. https://doi.org/10.35940/ijitee.E2509.039520.

[5] J.W. Glauber, A. Mamun, Global Rice Market: Current Outlook and Future Prospects, IFPRI Discussion Papers 2310, International Food Policy Research Institute (IFPRI), (2024).

[6] S. Saha, Sk.Md.M. Ahsan, Rice Disease Detection Using Intensity Moments and Random Forest, in: 2021 International Conference on Information and Communication Technology for Sustainable Development, IEEE, Dhaka, Bangladesh, 2021: pp. 166–170. https://doi.org/10.1109/ICICT4SD50815.2021.9396986.

[7] K. Ahmed, T.R. Shahidi, S.Md. Irfanul Alam, S. Momen, Rice Leaf Disease Detection Using Machine Learning Techniques, in: 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), IEEE, Dhaka, Bangladesh, 2019: pp. 1–5. https://doi.org/10.1109/STI47673.2019.9068096.

[8] S. Phadikar, J. Sil, A.K. Das, Rice Diseases Classification Using Feature Selection and Rule Generation Techniques, Comput. Electron. Agric. 90 (2013), 76–85. https://doi.org/10.1016/j.compag.2012.11.001.

[9] F.T. Pinki, N. Khatun, S.M.M. Islam, Content Based Paddy Leaf Disease Recognition and Remedy Prediction Using Support Vector Machine, in: 2017 20th International Conference of Computer and Information Technology (ICCIT), IEEE, Dhaka, 2017: pp. 1–5. https://doi.org/10.1109/ICCITECHN.2017.8281764.

[10] A. Stephen, A. Punitha, A. Chandrasekar, Designing Self Attention-Based ResNet Architecture for Rice Leaf Disease Classification, Neural Comput. Appl. 35 (2023), 6737–6751. https://doi.org/10.1007/s00521-022-07793-2.

[11] M.K. Bag, B.M. Bashyal, P.K. Kaman, et al. Rice: Blast, Brown Spot and False Smut, in: Diseases of Nationally Important Field Crops, Today and Tomorrows Printers and Publishers, New Delhi, (2021).

[12] A.A. Joshi, B.D. Jadhav, Monitoring and Controlling Rice Diseases Using Image Processing Techniques, in: 2016 International Conference on Computing, Analytics and Security Trends (CAST), IEEE, Pune, India, 2016: pp. 471–476. https://doi.org/10.1109/CAST.2016.7915015.

[13] B. Sowmiya, K. Saminathan, M.C. Devi, A Comprehensive Review on Diagnosis and Classification of Paddy Leaf Diseases Using Advanced Computer Vision Technologies, ICTACT J. Image Video Process. 13 (2023), 2973–2986. https://doi.org/10.21917/ijivp.2023.0424.

[14] Y. Dani, M. Artanta Ginting, Comparison of Iris Dataset Classification with Gaussian Naïve Bayes and Decision Tree Algorithms, Int. J. Electr. Comput. Eng. 14 (2024), 1959. https://doi.org/10.11591/ijece.v14i2.pp1959-1968.

[15] H.B. Prajapati, J.P. Shah, V.K. Dabhi, Detection and Classification of Rice Plant Diseases, Intell. Decis. Technol. 11 (2017), 357–373. https://doi.org/10.3233/IDT-170301.

[16] Md.M. Hasan, A.F.M.S. Uddin, M.R. Akhond, Md.J. Uddin, Md.A. Hossain, Md.A. Hossain, Machine Learning and Image Processing Techniques for Rice Disease Detection: A Critical Analysis, Int. J. Plant Biol. 14 (2023), 1190–1207. https://doi.org/10.3390/ijpb14040087.

[17] N. Katuwawala, K. Kumarasinghe, R. Rajapaksha, D. Rathnayaka, Riceguardnet: Custom Cnns for Precise Bacterial and Fungal Infection Classification, in: 2023 8th International Conference on Information Technology Research (ICITR), IEEE, 2023, pp. 1–6.

[18] K. Kiratiratanapruk, P. Temniranrat, A. Kitvimonrat, et al. Using Deep Learning Techniques to Detect Rice Diseases from Images of Rice Fields, in: H. Fujita, P. Fournier-Viger, M. Ali, J. Sasaki (Eds.), Trends in Artificial Intelligence Theory and Applications. Artificial Intelligence Practices, Springer, Cham, 2020: pp. 225–237. https://doi.org/10.1007/978-3-030-55789-8_20.

[19] R. Deng, M. Tao, H. Xing, et al. Automatic Diagnosis of Rice Diseases Using Deep Learning, Front. Plant Sci. 12 (2021), 701038. https://doi.org/10.3389/fpls.2021.701038.

[20] Y. Lu, S. Yi, N. Zeng, et al. Identification of Rice Diseases Using Deep Convolutional Neural Networks, Neurocomputing 267 (2017), 378–384. https://doi.org/10.1016/j.neucom.2017.06.023.

[21] J.C. Bortty, G.S. Chakraborty, T. Ahmed, et al. Deep Learning-Based Image Classification Framework for Rice Leaf Disease Detection: A Systematic Implementation Approach with Comparative Analysis, in: 2024 8th International Conference on Electronics, Communication and Aerospace Technology (ICECA), IEEE, Coimbatore, India, 2024: pp. 1428–1434. https://doi.org/10.1109/ICECA63461.2024.10800955.

[22] I.V. Arinichev, S.V. Polyanskikh, G.V. Volkova, I.V. Arinicheva, Rice Fungal Diseases Recognition Using Modern Computer Vision Techniques, Int. J. Fuzzy Logic Intell. Syst. 21 (2021), 1–11. https://doi.org/10.5391/IJFIS.2021.21.1.1.

[23] D. Wang, W. Cao, F. Zhang, et al. A Review of Deep Learning in Multiscale Agricultural Sensing, Remote Sensing 14 (2022), 559. https://doi.org/10.3390/rs14030559.

[24] V.G. Dhanya, A. Subeesh, N.L. Kushwaha, et al. Deep Learning Based Computer Vision Approaches for Smart Agricultural Applications, Artif. Intell. Agric. 6 (2022), 211–229. https://doi.org/10.1016/j.aiia.2022.09.007.

[25] A. Morchid, M. Marhoun, R. El Alami, B. Boukili, Intelligent Detection for Sustainable Agriculture: A Review of IoT-Based Embedded Systems, Cloud Platforms, DL, and ML for Plant Disease Detection, Multimedia Tools Appl. 83 (2024), 70961–71000. https://doi.org/10.1007/s11042-024-18392-9.

[26] M.I. Kabir, Paddy Leaf Disease, Kaggle, (2023). https://www.kaggle.com/datasets/iashiqul/paddy-leaf-disease.

[27] Vbookshelf, Rice Leaf Diseases Dataset, Kaggle, (2023). https://www.kaggle.com/datasets/vbookshelf/rice-leaf-diseases.

[28] F. Sabry, K Nearest Neighbor Algorithm: Fundamentals and Applications, One Billion Knowledgeable, 2023.

[29] D.A. Pisner, D.M. Schnyer, Support Vector Machine, in: Machine Learning, Elsevier, 2020: pp. 101–121. https://doi.org/10.1016/B978-0-12-815739-8.00006-7.

[30] V. Kecman, Support Vector Machines - An Introduction, in: L. Wang (Ed.), Support Vector Machines: Theory and Applications, Springer, Berlin, Heidelberg, 2005: pp. 1–47. https://doi.org/10.1007/10984697_1.

[31] S. Taheri, M. Mammadov, Learning the Naive Bayes Classifier with Optimization Models, Int. J. Appl. Math. Comput. Sci. 23 (2013), 787–795. https://doi.org/10.2478/amcs-2013-0059.

[32] A.J. Myles, R.N. Feudale, Y. Liu, et al. An Introduction to Decision Tree Modeling, J. Chemometrics 18 (2004), 275–285. https://doi.org/10.1002/cem.873.

[33] Y. Liu, Y. Wang, J. Zhang, New Machine Learning Algorithm: Random Forest, in: B. Liu, M. Ma, J. Chang (Eds.), Information Computing and Applications, Springer, Berlin, Heidelberg, 2012: pp. 246–252. https://doi.org/10.1007/978-3-642-34062-8_32.

[34] Y. Dani, M.A. Ginting, Classification of Predicting Customer Ad Clicks Using Logistic Regression and K-Nearest Neighbors, JOIV: Int. J. Inform. Vis. 7 (2023), 98–104. https://doi.org/10.30630/joiv.7.1.1017.

[35] J. Liu, Y. Xu, T-Friedman Test: A New Statistical Test for Multiple Comparison with an Adjustable Conservativeness Measure, Int. J. Comput. Intell. Syst. 15 (2022), 29. https://doi.org/10.1007/s44196-022-00083-8.

[36] J. Ma, D. Xia, Y. Wang, et al. A Comprehensive Comparison among Metaheuristics (MHs) for Geohazard Modeling Using Machine Learning: Insights from a Case Study of Landslide Displacement Prediction, Eng. Appl. Artif. Intell. 114 (2022), 105150. https://doi.org/10.1016/j.engappai.2022.105150.