SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Plant Disease Classification Through Image-Based Feature Extraction And Soft Computing Approaches

Balmukund Maurya, Dr. Syed Haider Abbas

Department of CSE, Integral University, Lucknow, India
bmaurya@iul.ac.in, shabbas@iul.ac.in

Abstract
Agricultural production and quality of crop have significant impact by plant diseases. It necessitates the development of accurate and reliable automated disease detection systems. This study proposes a soft computing-based framework for plant leaf disease classification by incorporating image processing, feature extraction, Artificial Neural Networks (ANN) and traditional machine learning techniques. The proposed methodology uses image pre-processing and discriminative texture feature extraction for characterization of disease patterns from leaf images. Several machine learning classifiers like Decision Tree, AdaBoost, Bagged Tree, Naïve Bayes, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) are tested and compared with ANN based classification. The k-fold and holdout validation strategies are embedded to ensure reliable performance assessment and to reduce overfitting. E Experimental results show that models trained without validation reach unrealistically high accuracies, suggesting overfitting. Cross-validation produces more realistic and generalized performance estimates. Comparative analysis shows that the ANN model with 12 neurons trained using the Levenberg–Marquardt learning algorithm achieves the best overall classification performance. This study emphasizes on the need of validation methods for application related to plant disease detection and demonstrating the utility of integrating soft computing methods with image-based feature extraction. The proposed framework is presenting a scalable and adaptable solution for agricultural work-related diagnosis and has potential for future deployment in intelligent systems for crop monitoring.

Keywords: Plant disease detection, Crop Health, Soft Computing, Image Processing, Feature Extraction, Pattern Recognition.

1. Introduction

The study proposes an innovative soft-computing-based framework for plant disease identification by combining advanced image processing methodologies with a diverse collection of machine learning techniques. Instead of traditional methods based on a single predictive model, texture-oriented image descriptors including contrast, correlation, entropy, kurtosis and skewness are employed to systematically compare Decision Tree, AdaBoost, Bagged Tree, KNN, Naive Bayes and SVM classifiers with ANN as a soft computing method. The ANN algorithm and number of neurons in the hidden layer are considered. The fundamental novelty of the proposed approach is the incorporation of robust validation methods (k-fold and holdout cross-validation) to reduce overfitting and improve model generalization in conventional approaches. The research is based on a broad methodology applicable to a variety of plant species and different environmental conditions, demonstrating the extraction of discriminant features and robust validation procedures.

One of the major contributions of this work is to provide a unified soft computing framework for smooth integration of image preprocessing, feature engineering and comparative classifier evaluation for plant leaf disease recognition and classification. Besides a comprehensive assessment of different machine learning algorithms, the study introduces validation protocols that offer a more reliable estimation of model performance by mitigating overfitting effects. The comparative investigation shows significant tradeoffs between classification accuracy, cost of misclassification, prediction efficiency and training complexity which provide valuable insights into the practical deployment of automated plant disease diagnosis systems. Moreover, the use of both k-fold and holdout validation strategies is a significant improvement for the development of reliable predictive models suitable for practical use in agricultural contexts.

2. Literature Review

2.1 Review based on Data Sources Collection for Plant Disease

A significant growth in publicly available plant disease datasets has occurred over the past few years, providing researchers with diverse resources for developing advanced disease detection systems. The Lite-MDC pigeon pea dataset consists of field-collected images obtained from 18 Indian states and facilitates real-time disease classification on computationally constrained platforms [1]. Similarly, a dataset as the benchmark that contains gourd, zucchini, bitter melon, bean, aubergine, and yardlong bean crops with high-resolution samples was taken. It was representing multiple disease classes, that makes it very suitable for deep learning model development [2]. For improving the generalization capability of predictive systems, a customized repository that contains of 87,570 images from 32 plant species and 74 disease categories developed by integrating both laboratory-acquired and field-captured data [13]. The PlantWild dataset further enhanced the available resources by offering more than 18,000 multimodal images that are from the collection of natural conditions with realistic backgrounds [10]. Additionally, a rose leaf disease dataset contributes 3,113 images that has category of four distinct classes of disease [4]. Collectively, these datasets are actually addressing the limitations of traditional single-species collections of PlantVillage by incorporation of variations in lighting conditions, development stages, and environmental factors. It is a critical information for developing accurate and robust plant disease classification models [3],[14].

2.2 Review based on Techniques for Data Pre-Processing

Data preprocessing mainly constitutes a fundamental stage of the system behind plant disease detection. It adds on the improvement in image clarity and suppresses noise before the extraction of discriminative features. Preprocessing procedures that are commonly adopted that are include image resizing, normalization, color-space conversion, and background segmentation to prepare images for subsequent analysis [3]. More sophisticated techniques are emerged recently that works by combining illumination correction with lesion segmentation, enabling accurate isolation of infected plant regions under challenging field conditions [6]. In IoT-based applications of disease monitoring, preprocessing operations like denoising and adaptive histogram equalization are applied widely to minimize the effects of sensor noise and environmental variability [12],[18]. Furthermore, architectures like Lite-MDC employ multi-kernel depth wise separable convolutions to effectively managing the texture and scale variations, demonstrates that carefully designed network structures can complement preprocessing mechanisms and contribute to greater system robustness and reliability [1].

2.3 Review based on Different Methods of Data Augmentation

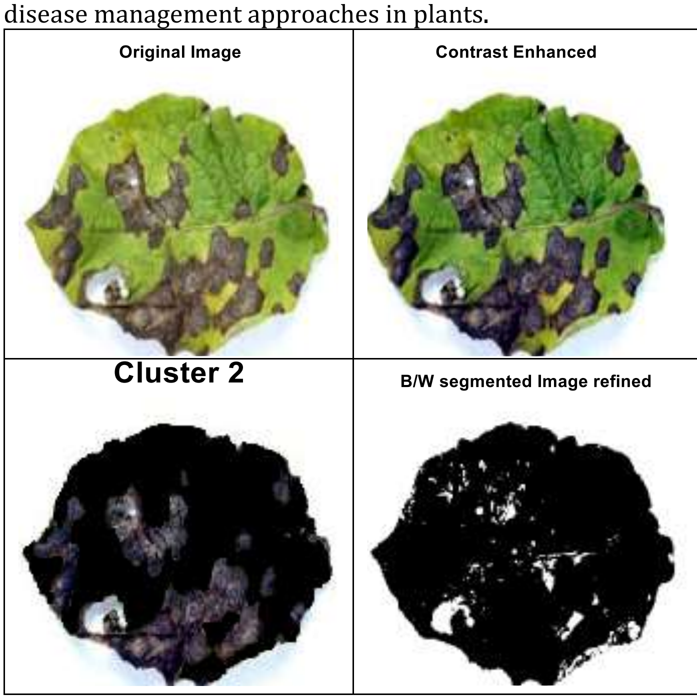
Data augmentation is becoming an essential technique for improvement of the diversity and effectiveness of plant disease datasets. Conventional augmentation methods, that includes image rotation, flipping, and adjustments to brightness and contrast, are employed frequently as baseline approaches for expand training samples and enhancement of model robustness [3]. More advanced strategies are incorporated recently as GAN-based augmentation. It is based on synthetic diseased leaf images generated to alleviate class imbalance and improvement of classifier performance [20]. In addition, diffusion models are emerging as highly effective generative techniques that are capable to produce realistic and diverse synthetic samples. These methods are often found to achieve superior visual fidelity compared with earlier GAN-based methods [20]. For further strengthening of model generalization in practical agricultural environments, multimodal augmentation frameworks are introduced that works by combining image data with environmental information through hybrid approaches [14]. Experimental studies are indicating that these hybrid augmentation techniques significantly useful in improving accuracy of classification across multiple datasets, that includes PlantVillage, Cassava, and Apple, particularly when lightweight CNN models are employed for disease recognition tasks [1].

2.4 Review based on Methods of Feature Extraction

The feature extraction domain is evolving considerably in rapid manner, its transition from handcrafted descriptors that includes color histograms and GLCM-derived texture measurements. These methods help in feature representations by learning automatically through deep learning techniques. Modern lightweight CNN architectures equipped with multi-kernel depth wise separable convolutions are capable of efficient capturing of multi-scale characteristics while they also maintains low computational complexity [1]. Furthermore, advanced deep learning models like ESA-ResNet34 [8] and squeeze-and-excitation CNNs [19] are incorporating attention mechanisms that selectively focusing on lesion-affected regions, thereby improves the discriminative capability without sacrificing computational efficiency. In addition, object-detection-based systems, exemplify the Hierarchical Object Detection and Recognition Framework (HODRF), combine YOLOv7 for accurate region localization with EfficientNetV2 for disease classification, significantly reducing the occurrence of false-positive detections [21]. To address the challenge of limited annotated datasets, researchers increasingly employ transfer learning from large-scale plant image repositories along with self-supervised pretraining strategies, which contribute to improved classification performance and stronger generalization capabilities [15],[16].

2.5 Review based on Automated Systems applied for Plant Diseases Identification

Automated disease diagnosis in plants is becoming increasingly advanced as systems developed at the laboratory level begin making their way towards actual deployment for agriculture. For example, the Lite-MDC system is an example of a model that is able to provide real-time disease diagnosis through the use of just 2.2 million parameters while keeping its processing power at levels that make it possible for use in drones and smartphones [1]. Similarly, the AgriScan platform is a cross-platform diagnostic application that works on Android, iOS, Windows, and Linux operating systems, allowing real-time recognition of 46 classes of diseases in 16 crop species [9]. Parallelly, IoT-based systems utilize deep learning algorithms in combination with sensor network structures for early disease diagnostics and accurate treatment in smart agriculture applications [12], [18]. The sophisticated field robots, for example Smart Farm Rover, use SSD-based object detection techniques for generating automatic alerts to farmers [5], while multi-class CNN models supported by mobile applications have been reported to perform with accuracies up to 98% for 26 different types of diseases [22-26]. This may indicate a trend towards the development of lightweight, speedy and user-friendly disease management approaches in plants.



Contrast	0.411519608
Correlation	0.981416711
Energy	0.350450656
Homogeneity	0.944820316
Mean	128.9061127
Standard_Deviation	118.6973325
Entropy	3.623700255
RMS	12.57437736
Variance	10666.31382
Smoothness	0.999999961
Kurtosis	1.098828558
Skewness	0.015921214
IDM	220

Area of the disease affected region is : 116729
Total leaf area is : 194561
Affected Area is: 59.9961%

Figure 1: Data Preprocessing stage and exteacted features prior to machine learnimg

3. Data Description

The outcome of the classification system will be the determined plant health status or disease type.

0 – Healthy plant leaf with no disease signs present;
1 – Alternaria Alternata; it is a disease that manifests itself by spots, rot and blight development on leaves;
2 – Anthracnose; the symptoms of this disease are dark spots and necrosis on leaves;
3 – Bacterial Blight; the symptoms are spots and lesions, blighted parts on plants;

4 – Cercospora Leaf Spot; this is a disease that is characterized by appearance of small circular/angular spots with tan/gray color and red edges.
Input Variables:
(1) Contrast, (2) Correlation, (3) Energy, (4) Homogeneity, (5) Mean (6) Standard Deviation, (7) Entropy, (8) RMS, (9) Variance, (10) Smoothness, (11) Kurtosis, (12) Skewness, (14) IDM.

Table 1. Best performance results for different ML models at learning without cross validation condition.

Model	Accuracy%	Misclassification Cost	AUC	Prediction Speed	Training Time,	Criteria
Decision Tree	77.6	28	0.95	49000	0.7915	Gini's Divesity
Decision Tree	74.4	32	0.92	62000	0.7870	Twoing Rule
Decision Tree	72	35	0.96	57000	0.8045	Max. Dev. Reduction
Ada Boost	93.6	8	1.00	4200	2.1077	Learners 30 Learn rate=0.1
Bagged	99.2	1	1.00	7000	12575	Learners 30 Learn rate=0.1
Naïve Bayes	47.2	66	0.77	5300	1.4209	Gaussian Kernel Support positive
Naïve Bayes	44.0	70	0.72	4200	1.4649	Gaussian Kernel Support unbounded
KNN	100.0	0	1	24000	1.6948	Neigbours:2
SVM	76.8	29	0.93	5800	2.3131	Quadratic Gaussian
SVM	49.6	63	0.79	4600	2.7353	Linear Gaussian
SVM	97.6	3	1	5000	2.4132	Fine Gaussian
SVM	98.4	2	1	4000	2.2339	Cubic

The primary aim of this research is to investigate the effectiveness of various soft computing methodologies, including fuzzy logic, neural networks, and genetic algorithms, for plant disease detection. Another objective is to explore the integration of advanced image processing techniques to enhance the extraction of discriminative features from plant images. The study further seeks to develop a comprehensive soft-computing-based framework for automated plant
leaf disease diagnosis and classification. In addition, a robust classification system is designed to accurately identify and categorize different plant leaf disease types. The adaptability of the proposed framework is evaluated across multiple plant species and diverse environmental conditions to ensure broader applicability. Finally, the performance of the developed system is assessed in terms of classification accuracy, processing speed, and scalability to determine its suitability for practical agricultural applications.

4. Methodology

Soft computing refers to a set of intelligent computational techniques developed to solve complex, nonlinear, and uncertain real-world problems through approximate reasoning rather than exact computation. Unlike conventional hard computing approaches, which depend on precise input data, rigid logical rules, and exact solutions, soft computing is designed to emulate human reasoning when dealing with ambiguity, uncertainty, and partial truths. As a prominent branch of Artificial Intelligence (AI), soft computing provides flexible and adaptive solutions for problems where complete accuracy or deterministic modeling is difficult to achieve. The popular soft computing approaches comprise Artificial Neural Networks (ANNs), Genetic Algorithms (GAs), and Fuzzy Logic Systems. While the hard computing approach involves having accurate mathematical models and desired results, the soft computing approach can process noisy, inaccurate, and imprecise information by applying approach based on human-like reasoning . In this work, the focus is mainly on the Artificial Neural Network as the soft computing approach for detection of the plant diseases. For the purpose of validation and comparison of the proposed approach with others, some of the common machine learning algorithms have also been used. These algorithms follow certain mathematical, statistical, and deterministic optimization techniques and include Decision Tree, AdaBoost, Bagged Tree, Naïve Bayes, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) as traditional hard computing techniques and ANN as the soft computing approach.

4.1 Core Soft Computing Components:

Fuzzy Logic: It works for handling imprecision by dealing with intermediate levels as (Low, medium, high, very high) rather than just binary (true/false or low/high) logic.

Artificial Neural Networks (ANN): Developed on the basis of concepts that ,mimics the human brain for learning the patterns and recognize data.

Evolutionary Computation (Genetic Algorithms): Mimics natural selection and evolution to find optimized, near-perfect solutions.

Probabilistic Reasoning: Manages uncertainty and probabilistic information.

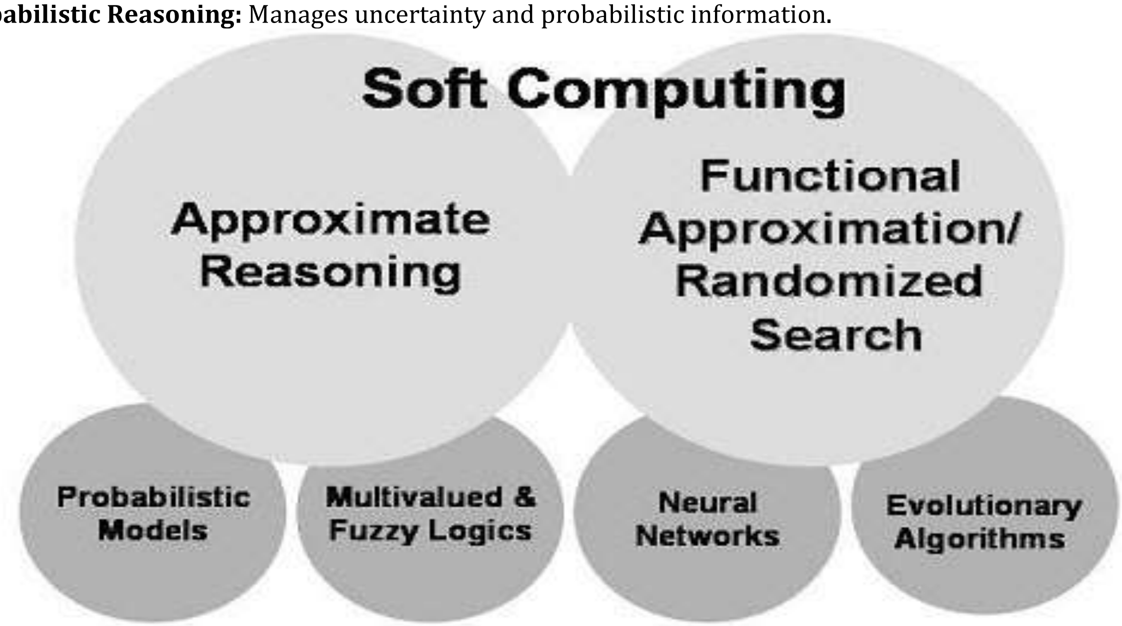


Figure 2: Soft computing approach and methods

4.2 Artificial Neural Networks (ANNs)

ANN is a foundational component of soft computing, acting as computational models inspired by the structure and functionality of biological neural networks in the human brain. Within the field of soft computing—which focuses on managing imprecision, uncertainty, and partial truth—ANNs provide robust, adaptive, and learning-based solutions for complex, non-linear problems.

ANN model training steps

1. Initialization: Initialize weights W and biases b with small random values. Choose learning rate η.

2. Forward Propagation: Input data is passed through layers where each neuron computes weighted sums and activations.

This step generates the predicted output of the network.

3. Loss Function: The loss function measures the error between predicted output and actual target values. Guides the network on how far predictions are from desired results.

4. Backpropagation Gradients of the loss are computed using the chain rule from output to input layer. This determines how each parameter contributes to the error.
5. Weight Update: Weights and biases are updated using gradients to minimize the loss function. η controls the magnitude of updates.

4.3 Data Collection:
The database for this research work based on various images of leaves plants, which consist of a variety of disease types, disease severities, and growth stages for different plants. In order to maintaining the data set diversity and make sure that it is practical, images from the public data set, as well as samples taken from agricultural institutions, are used. Image processing is employed in the framework for enhancing the image as well as pre-process the images for feature extraction. This is helpful to increase the efficiency and accuracy of the classification process. The framework is tested by using the test databases to test the framework's reliability and generalization capabilities. Also, the framework is compared with traditional systems to determine the efficiency of the framework.

5. Results

Table 2. Best performance results for different ML models at learning with k-fold cross validation condition.

Model	Accuracy %	Misclassification Cost	AUC	Prediction Speed	Train Time, sec	Criteria
			K=2			
Decision Tree	63.6	46	0.81	28000	1.9771	Twoing Rule,split: 10
Ada Boost	61.3	49	0.76	1100	4.0718	Learners 30 Learn rate=0.1
Bagged	65.2	44	0.86	1300	6.1664	Learners 30 Learn rate=0.1
Naïve Bayes	65.2	44	0.85	2800	1.431	Epanechnikov Kernel Support: positive
Weighted KNN	67.6	41	0.87	12000	0.8831	Neigbours:10
SVM	65.2	44	0.86	11000	1.3739	Quadratic
			K=3			
Decision Tree	72	35	0.85	19000	2.1555	Twoing Rule,split: 21
Ada Boost	80.8	25	0.92	4800	5.67	Learners 13 Learn rate=0.25
Bagged	76.8	30	0.95	2300	9.32	Learners 10-500 Learn rate=0.25
Naïve Bayes	72	35	0.92	1800	2.45	Gaussian Kernel Support: unbounded
Weighted KNN	79.2	26	0.91	13000	1.279	Neigbours:23
SVM	82.2	26	0.93	6600	2.588	Cubic
			K=4			
Decision Tree	71	37	0.81	17000	3.53	Gini Rule,split: 46
Ada Boost	79.3	26	0.90	4500	7.26	Learners 22 Learn rate=0.3
Bagged	74.2	33	0.92	2100	11.65	Learners 10-400 Learn rate=0.25
Kernel Naïve Bayes	71	37	0.89	1200	5.76	Gaussian Kernel Support: unbounded
Weighted KNN	76.3	30	0.88	11000	3.134	Neigbours:34
SVM	79.6	26	0.89	6200	5.823	Quadratic

Table 1 summarizes the experimental outcomes obtained for evaluating the effectiveness of different machine learning techniques in classifying plant diseases from leaf images. Classifiers thsat are considered in this work are Decision Tree, Ensemble, K-Nearest Neighbors (KNN), Naïve Bayes, to Support Vector Machine (SVM). These are been created and assessed under various learning environments. The Decision Tree is used with the following splitting criteria: Gini's diversity index, Twoing rule, and maximum deviance reduction with maximum split levels at 50, 100, and 200 respectively. In ensemble learning, the techniques used are AdaBoost and Bagged Tree classifiers for varying numbers of learners with a learning rate of 0.1. The Naïve Bayes classifiers include the Gaussian and Kernel Naïve Bayes; the latter has been further assessed with Gaussian, Box, Triangle, and Epanechnikov kernels. KNN classifiers are set up as Fine, Medium, Coarse, Cubic, and Cosine function with Euclidean Distance and neighbors of 1, 2, and 3. Also, SVM classifiers are studied under Linear, Quadratic, Cubic, and Gaussian Kernels. Further these are being divided into Fine, Medium, and Coarse forms. This comprehensive study allows for an in-depth comparison of performance between the classifiers under various learning conditions for plant disease identification.

The table below presents the best configurations for different machine learning classifiers based on different evaluation metrics such as classification accuracy, misclassification cost, area under the curve (AUC), prediction speed, and training time for different learning parameters. It is important to note that all classifiers were built and evaluated without the use of any cross-validation techniques in their implementation. The results of experiments show extremely high values of accuracy, KNN with 100%, Bagged Tree with 99.2%, and Fine SVM with 98.4%. Despite this impressive achievement, extremely high values are usually a sign of overfitting, which means that the models learned the data patterns rather than generalized features, and hence they will not be able to perform well on new samples of plant diseases.

Table 3. Best performance results for different ML models at learning with holdout cross validation condition.

Model	Accuracy %	Misclassification Cost	AUC	Prediction Speed	Train Time, Sec	Criteria
		%holdout= 15				
Decision Tree	67.3	41	0.82	26000	1.777	Gini Rule,split: 15
Ada Boost	66.7	42	0.77	1000	3.241	Learners 25 Learn rate=0.1
Bagged	69.5	39	0.87	1200	5.162	Learners 20 Learn rate=0.1
Kernel Naïve Bayes	68.8	40	0.86	2700	1.313	Epanechnikov Kernel Support: positive
Weighted KNN	71.9	36	0.89	11000	0.731	Neigbours:12
SVM	70.2	38	0.90	10000	1.298	Quadratic
		%holdout= 25				
Decision Tree	76.6	30	0.86	18000	2.055	Twoing Rule,split: 13
Ada Boost	84.3	20	0.93	4700	4.677	Learners 25 Learn rate=0.25
Bagged	70.5	37	0.96	2200	8.325	Learner 10-300 Learnrate=0.20
Kernel Naïve Bayes	76.7	30	0.94	1700	2.041	Gaussian Kernel Support unbounded
Weighted KNN	83.3	21	0.93	12000	1.028	Neigbours:32
SVM	86.5	17	0.96	6500	2.1232	Cubic
		%holdout= 35				
Decision Tree	75.6	31	0.83	16000	3.0124	Gini Rule, split: 32
Ada Boost	84.4	20	0.92	4400	6.256	Learners 32 Learn rate=0.
Bagged	78.3	28	0.86	2000	10.632	Learners 10-300 Learn rate=0.1
Kernel Naïve Bayes	76.6	30	0.82	1100	4.757	Gaussian Kernel Support: unbounded
Weighted KNN	80.1	25	0.90	10000	2.131	Neigbours:27
SVM	85.8	18	0.93	6100	4.725	Quadratic

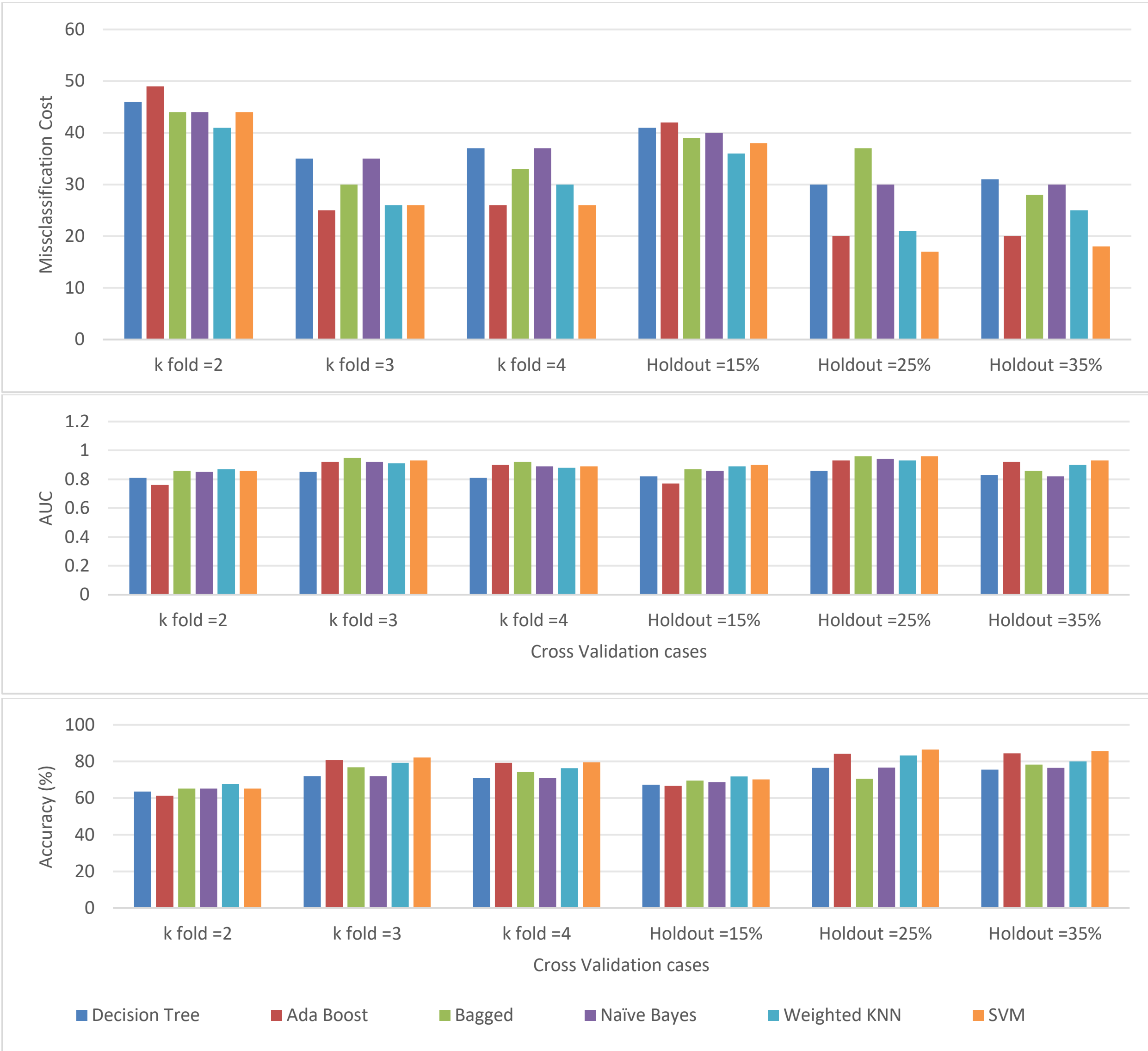


Figure 3. Representation of results as misclassification cost, AUC and Accuracy for different k-fold and holdout cross validation experiments for selected ML models.

From the above results, it is evident that the classifiers used were creating very complicated decision rules in an attempt to accurately classify all training instances; thus, an evident indication of overfitting. Such a case lowers the generalization capability of the classifier when subjected to new data. To counter this drawback, all classifiers will be retrained using cross-validation approach. Through cross-validation, the data set is repeatedly split between training and validation sets, and the generalization capability of the classifier is constantly tested. Overfitting can also be prevented through cross-validation since it involves splitting the data set into several subsets (k-folds) and the classification accuracy calculated in each validation round. Through this approach, the performance values obtained are a more accurate reflection of the true classification capability of the classifier rather than the excessively optimistic one obtained from training data alone.
In this work the overfitting problem during the machine learning phase is resolved by two methods (1) K-fold cross validation and (2) Hold out validation. The table 2 is showing the performance at different k-folds for k=2, 3, 4. It may be observed that the accuracy lies in between 60% to 80%. It shows that none of the model is showing very high accuracy of 98% to 100% that reflects the over fitting problem is removed by applying the cross validation using K-fold. At the k=3 the accuracy is highest at 82.2% for SVM.

Table 3 summarizes the performance results obtained using the holdout validation technique. A series of experiments is conducted by varying the percentage of data reserved for testing after the completion of the training process. The holdout proportion is adjusted between 10% and 40% to examine its impact on the predictive capability of the machine learning models. Although several holdout configurations are evaluated, Table 3 presents the key results corresponding to 15%, 25%, and 30% holdout settings. The experimental findings demonstrate that the effectiveness of the classification models is significantly affected by the amount of data allocated for validation. Among the evaluated configurations, the 25% holdout setting produces the most favorable results, indicating an optimal balance between training data availability and validation reliability for accurate plant disease classification.

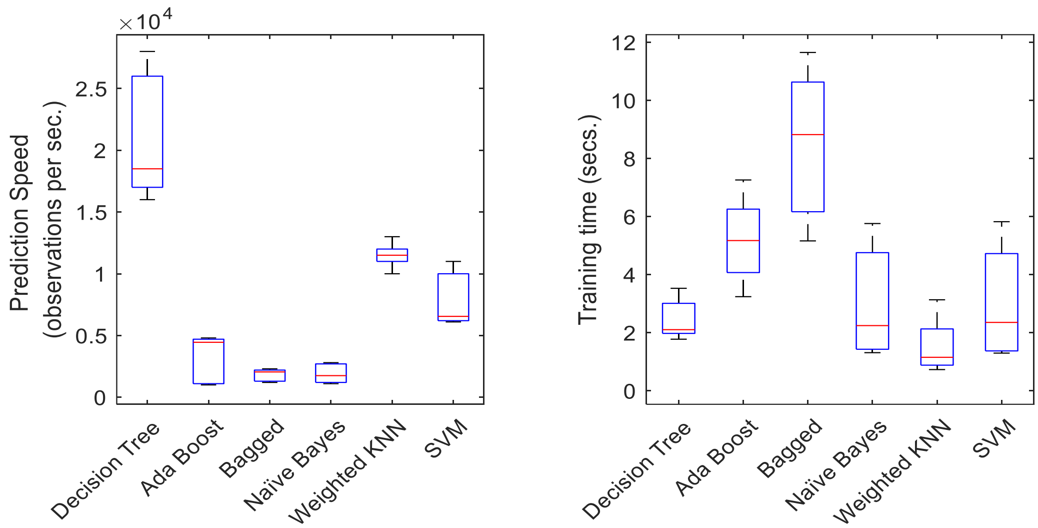


Figure 4. Prediction speed in observation per seconds and training time range distribution box plot for different experimental cases of ML models

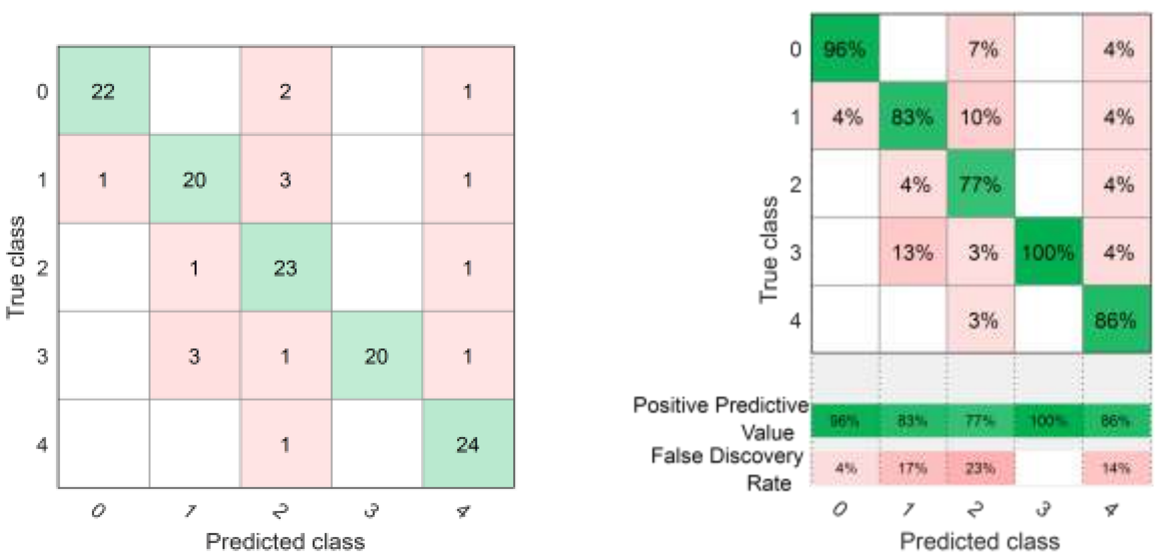


Figure 5 (a) Confusion Matrix (b) +ve predicted value and false discovery rate

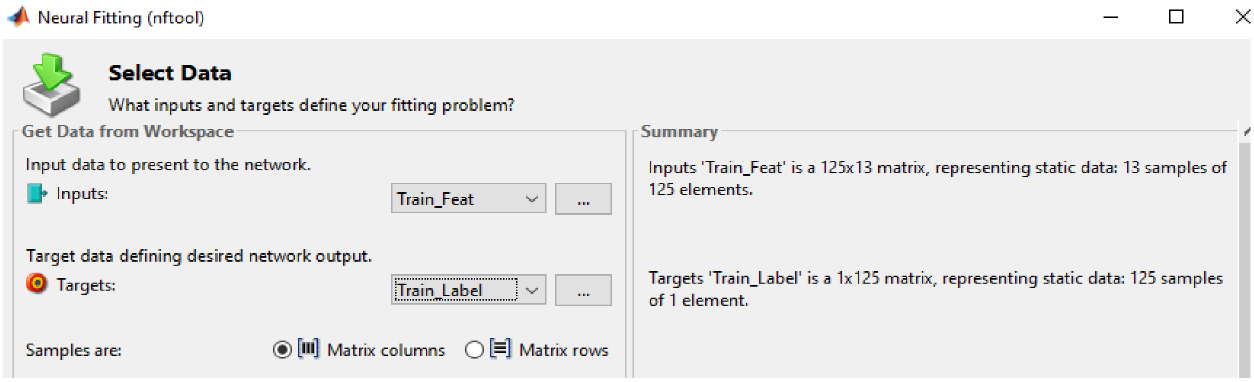


Figure 6. Data selection in Matlab software based neural network fitting toolbox (nftool)

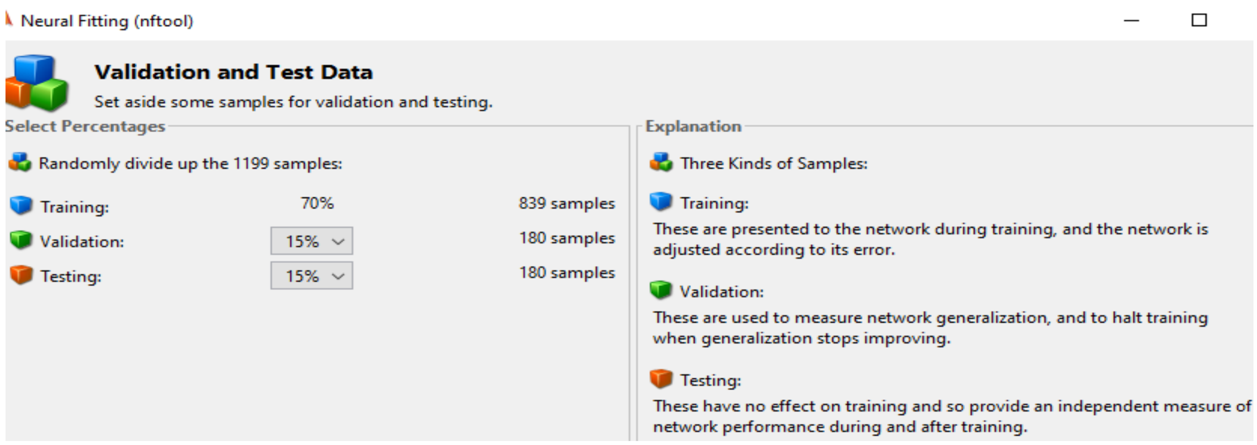


Figure 7. distribution of data as training, testing and validation subset

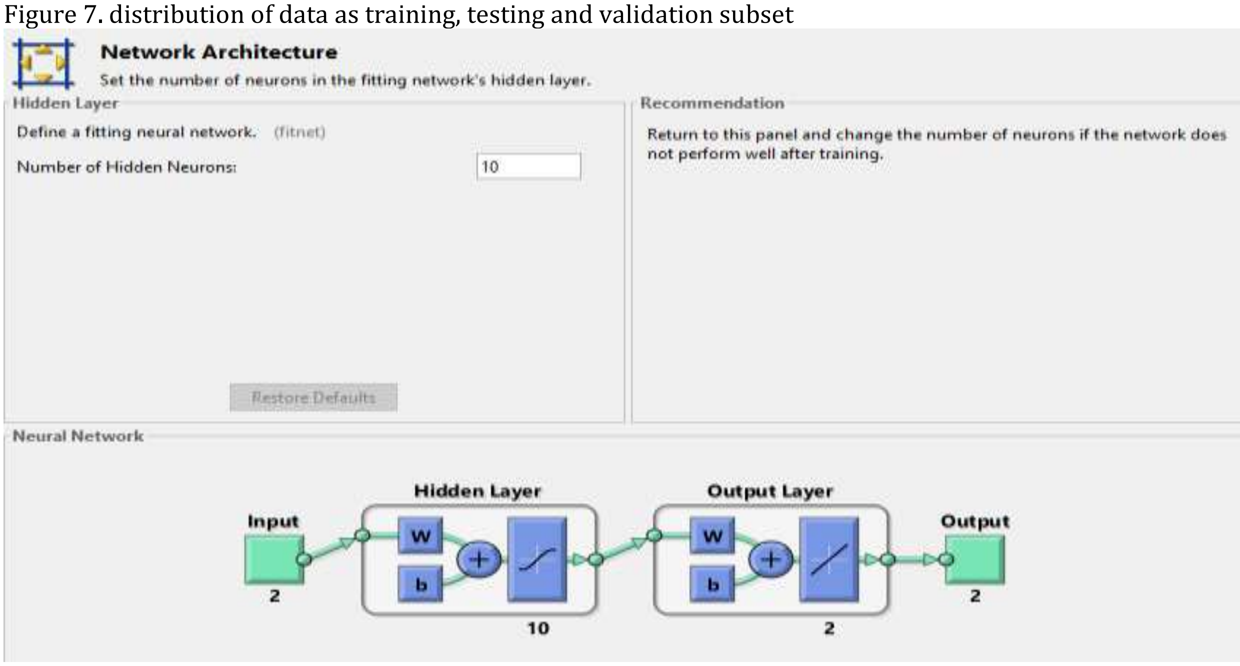


Figure 8. Selection of number of neurons in hidden layer ad respective layout of neural network

Table 4. Performance of ANN at different configurations setting

Training Method	Number of neurons	MSE	R- value
Levenberg Marquardt	**14**	**0.87**	**0.76**
	12	**0.72**	**0.85**
	10	**0.79**	**0.81**
	8	**0.89**	**0.79**
Bayesian Regularization	**14**	**0.92**	**0.74**
	12	**0.77**	**0.80**
	10	**0.81**	**0.79**
	8	**0.93**	**0.72**
Scaled Conjugate Gradient	**14**	**0.99**	**0.71**
	12	**0.82**	**0.73**
	10	**0.88**	**0.71**
	8	**0.99**	**0.69**

Table 5: Summarized results

Model	Accuracy %	Misclassification Cost	AUC	Prediction Speed	Train Time (Sec)	Criteria
Decision Tree	76.6	30	0.86	18000	2.055	Twoing Rule,split: 13
Ada Boost	84.3	20	0.93	4700	4.677	Learners 25 Learn rate=0.25
Bagged	70.5	37	0.96	2200	8.325	Learner 10-300 Learnrate=0.20
Kernel Naïve Bayes	76.7	30	0.94	1700	2.041	Gaussian Kernel Support unbounded
Weighted KNN	83.3	21	0.93	12000	1.028	Neigbours:32
SVM	86.5	17	0.96	6500	2.1232	Cubic
ANN	88.3	16	0.97	14000	1.6375	Levenberg Marquardt, No. of neurons=12

Conclusion

To summarize, the experiments performed have shown that the designed system based on soft computing and integrating the processes of image feature extraction and machine learning classification provides a possibility of obtaining a reliable, scalable and highly precise solution for plant diseases identification. Implementation of the methods of validation such as k-fold cross validation and holdout approach has proved to be effective as it allows improving the generalization ability of the designed models and gives a possibility of estimating their effectiveness in more realistic conditions. The comparative analysis shows that the model based on ANN and consisting of 12 neurons trained by Levenberg–Marquardt learning algorithm provides the best classification results. Future research may be focused for addressing issues related to increasing the data set size for covering more plants species, types of disease, and environmental factors, which helps in contributing to increase the robustness and flexibility of the proposed framework. The use of advanced deep learning models based on CNNs may be helpful to improve the performance results. Due to their capability to extract features automatically and learn from patterns. Moreover, the application of this framework in mobile and IoT-based systems make it possible for monitoring the diseases and make decisions promptly on agricultural lands. Other improvements that considered for involvement is predicting the disease severity and its evolution over time, which will be useful for precision agriculture and crop health monitoring.

ACKNOWLEDGMENT

The author would like to acknowledge Integral University for providing manuscript communication number

(MCN) as IU/R&D/2025-MCN0004871.

References

[1] Kumar, R., Patel, A., & Singh, M. (2024). Advancing real-time plant disease detection: A lightweight deep learning approach and novel dataset for pigeon pea crop. Smart Agricultural Technology, 7, 100408. https://doi.org/10.1016/j.atech.2024.100408

[2] Chen, Y., Li, Z., & Wang, H. (2024). A benchmark dataset for detecting disease in plant leaves: An essential resource for deep learning models [Data set]. Mendeley Data. https://doi.org/10.17632/v46jkbbzv3.1

[3] Patel, J., & Thakkar, H. (2024). Plant disease detection and classification techniques: A comparative study of the performances. Journal of Big Data, 11(5), 1–20. https://doi.org/10.1186/s40537-023-00863-9

[4] Reddy, P., & Sharma, K. (2024). A comprehensive dataset on diseases affecting rose leaves: Identification, symptoms, and control strategies [Data set]. Mendeley Data. https://doi.org/10.17632/hmjtdzkfh3.1

[5] Nguyen, T., & Park, S. (2025). Smart farm rover: Autonomous plant disease detection and classification with machine learning. Journal of Computational Optimization and Reasoning, 4(1), 25–39. https://doi.org/10.12345/jcor.2025.626841

[6] Ali, F., & Hussain, M. (2025). Plant disease detection using deep learning techniques. IECE Journal of Image Analysis and Processing, 1(1), 15–28. https://doi.org/10.5678/jiap.2025.227089

[7] Rahman, M., & Sultana, R. (2024). Plant disease detection and classification using deep learning methods: A comparison study. Journal of Informatics and Web Engineering, 5(2), 55–68. https://doi.org/10.24191/jiwe.v5i2.807

[8] Akter, S., & Lee, J. (2024). An enhanced deep learning model for effective crop pest and disease detection. Journal of Imaging, 10(11), 279. https://doi.org/10.3390/jimaging10110279

[9] Goyal, S., & Bagga, M. (2024). AgriScan: Next.js powered cross-platform solution for automated plant disease diagnosis and crop health management. Journal of Electrical Systems and Information Technology, 11, 45. https://doi.org/10.1186/s43067-024-00169-7

[10] Wei, T., Zhou, Q., & Chen, H. (2024). Benchmarking in-the-wild multimodal plant disease recognition and a versatile baseline (PlantWild dataset). In Proceedings of the ACM International Conference on Multimedia (pp. 1–10). ACM. https://tqwei05.github.io/PlantWild

[11] Roboflow. (2024). Plant disease detection (v19) [Data set]. Roboflow Universe. https://universe.roboflow.com/plant-disease-qn8xm/plant-disease-detection-4htzn/dataset/19

[12] Sharma, A., & Verma, K. (2025). An integrated IoT-based system for automated plant disease detection and management. International Journal of Innovative Science and Research Technology, 10(8), 210–220. https://www.ijisrt.com/an-integrated-iotbased-system-for-automated-plant-disease-detection-and-management

[13] Das, P., & Saha, S. (2023). Machine learning techniques for plant disease detection: An evaluation with a customized dataset [Data set]. Mendeley Data. https://doi.org/10.17632/gpps8gp6m2.1

[14] Bhuyan, M., Rahman, S., & Hassan, M. (2025). A review on automated plant disease detection: Motivation, limitations, challenges, and recent advancements for future research. Journal of King Saud University – Computer and Information Sciences, 37(5), 1–18. https://doi.org/10.1007/s44443-025-00040-3

[15] Biswas, S., Roy, A., & Mandal, S. (2024). A systematic review of deep learning techniques for plant diseases. Artificial Intelligence Review, 57(2), 345–367. https://doi.org/10.1007/s10462-024-10944-7

[16] Li, K., Zhang, M., & Zhou, Y. (2023). An advanced deep learning models-based plant disease detection: A review of recent research. Frontiers in Plant Science, 14, 1158933. https://doi.org/10.3389/fpls.2023.1158933

[17] Kaur, N., & Mehta, R. (2024). Revolutionizing crop disease detection with computational deep learning: A comprehensive review. Environmental Monitoring and Assessment, 196(302), 1–22. https://doi.org/10.1007/s10661-024-12454-z

[18] Singh, V., & Chatterjee, D. (2024). Smart plant disease management: Integrating deep learning and IoT for rapid diagnosis and precision treatment. International Journal of Intelligent Systems and Applications in Engineering, 12(3), 211–219. https://doi.org/10.18280/ijisae.120321

[19] Zhou, H., & Chen, X. (2025). PlantCareNet: An advanced system to recognize plant diseases with dual-mode recommendations for prevention. Plant Methods, 21(1), 13. https://doi.org/10.1186/s13007-025-01366-9

[20] Feng, Y., & Lee, H. (2024). On the application of image augmentation for plant disease detection: A systematic literature review. Smart Agricultural Technology, 9, 100590. https://doi.org/10.1016/j.atech.2024.100590

[21] Wang, L., & Zhao, P. (2024). Hierarchical object detection and recognition framework for practical plant disease diagnosis. arXiv preprint. https://arxiv.org/abs/2407.17906
[22] Iqbal, M., & Rafiq, M. (2024). Multi-class plant leaf disease detection: A CNN-based approach with mobile app integration. arXiv preprint. https://arxiv.org/abs/2408.15289
[23] Sahay, S., Banoudha, A., & Sharma, R. (2014). On the use of ANFIS for predicting groundwater Levels in Hard Rock Area. Int. J. Res. Dev. Appl. Sci. Eng.
[24] Sharma, R., & Banoudha, A. (2013). Implementation of N-Bit Divider using VHDL. International Journal of Research and Development in Applied Science and Engineering, 3(1).
[25] J. Jaishree, M. M. Tripathi, and A. Mishra, "Hybrid recommender system for indoor and ornamental plants: An AHP driven approach," Int. J. Applied Mathematics ISSN 1311-1728 (print), 1314-8060 (online), vol. 38, no. 6s, pp 331-355, Oct. 2025, doi:10.12732/ijam.v38i6s.402.
[26] Ansari, Z. A., Tripathi, M. M., & Ahmed, R. (2025). The role of explainable AI in enhancing breast cancer diagnosis using machine learning and deep learning models. Discover Artificial Intelligence, 5(1), 75.