AgriEngineering

Article

Health Detection of Medicinal Plants *Vitex negundo* and *Ricinus communis* Using Transfer Learning

Bharati Ainapure [1], Imaad Imran Hajwane [1], Maitreyee Rajesh Ekbote [1], Mohee Prashant Bansal [1], Gauri Mehul Patel [1], Bhargav Appasani [2,*], Nicu Bizon [3,*] and Alin Gheorghita Mazare [3]

[1] Department of Computer Engineering, Vishwakarma University, Pune 411046, India; ainapuressa@gmail.com (B.A.); iamimaad2kk3@gmail.com (I.I.H.); maitreyee.r.ekbote@gmail.com (M.R.E.); mohee08@gmail.com (M.P.B.); patelgauri390@gmail.com (G.M.P.)
[2] School of Electronics Engineering, Kalinga Institute of Industrial Technology, Bhubaneswar 751024, India
[3] Department of Electronics, Computers and Electrical Engineering, Pitești University Centre, The National University of Science and Technology POLITEHNICA Bucharest, 110040 Pitesti, Romania; alin.mazare@upb.ro
* Correspondence: bhargav.appasanifet@kiit.ac.in (B.A.); nicu.bizon1402@upb.ro (N.B.)

Abstract

Plant diseases pose a significant challenge to global agriculture, with early diagnosis particularly problematic for resource-limited farmers. This study introduces an AI-powered web application for classifying leaf health in two key medicinal plants, Ricinus communis (Eranda) and Vitex negundo (Nirgundi). The tool categorizes leaf images into three health states—Healthy, Medium Healthy, and Unhealthy to facilitate timely and informed decision making in crop management. A dataset comprising 2834 images of Ricinus communis and 3796 images of Vitex negundo under various conditions was created and used to train the model. Advanced transfer learning architectures, including VGG16, ResNet50, MobileNetV2, Xception, EfficientNetB0, and VGG19, were employed to enhance the classification accuracy of the system. Notably, VGG16, EfficientNetB0, and VGG19 achieved highest level of accuracy, whereas the other models showed comparatively lower level of accuracies. This work performs real-time assessment of leaf health for sustainable Ayurvedic agriculture.

Keywords: Ayurvedic medicinal plant; phytomedicine diagnostics using AI algorithms; plant pathology; herbal medicine; machine learning in agriculture

Check for updates

Academic Editor: Ray E. Sheriff

Received: 24 June 2026
Revised: 10 August 2026
Accepted: 20 August 2026
Published: 1 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Medicinal plants play a very vital role in Ayurvedic healthcare system. Ayurvedic system has been used in India from ancient ages to treat various diseases. Most of the medicines are prepared using plants like *Vitex negundo* (Nirgundi), *Ricinus communis* (Eranda), etc. Efficacy of the therapy directly depends on the quality and health of the harvested plant material [1]. Various diseases, fungal infections, and damage caused by insects and pesticides and other environmental stress-related factors cause alteration of the concentration of bioactive compounds present in the plants that are responsible for their medicinal properties. Hence, medicinal plants require close observation and maintenance of their phytochemical compositions. These damages lead to decrease in pharmacological effectiveness even if it is mild decoration in leaf health, thus affecting the quality of herbal medicines [2,3]. However, some medicinal plants share similar qualities, making it crucial to monitor leaf diseases and deficiencies carefully. Any decline in plant health can significantly reduce the

pharmacodynamic efficacy of the traditional remedies. Ayurveda advocates sustainable and natural farming practices, emphasizing prompt plant diagnosis to avoid over-reliance on chemical treatments that could alter the plant's identity [4]. Accurate predictions of leaf health are essential for maintaining the quality of Ayurvedic medicinal plants and ensuring their potency for treatment.

Recent advancements in machine learning and deep learning have revolutionized plant disease monitoring using computer-aided image-based diagnostic systems. These technologies offer unprecedented precision for the assessment of plant health. For example, a study using deep convolutional networks achieved 99.35% accuracy in identifying various plant diseases from a dataset of over 54,000 images [5]. Deep learning models are highly scalable, making them adaptable to various crops. These models can serve as comprehensive diagnostic tools for diverse plant species, regions, and climates. For example, Unmanned Aerial Vehicles (UAVs) combined with deep learning models can provide instant crop health assessments and enhance monitoring and classification efforts [6]. This technology reduces the need for manual labour, thereby increasing productivity and resource efficiency [7].

There are many studies that demonstrate the use of advanced technologies such as deep learning and transfer learning to diagnose plant diseases; however, most of them focused on plant species or food crop identification rather than plant health assessment. Hence, there is a need for development of framework that can accurately identify Ayurveda medicinal plants' leaf health using advanced machine learning techniques.

The major contributions of this work are summarized as follows:

- Development of a novel annotated dataset containing 6630 images of Vitex negundo and Ricinus communis leaves categorized into Healthy, Medium Healthy, and Unhealthy classes.
- Real-time medicinal plant leaf health assessment using web-based prototype.
- Comprehensive comparative evaluation of six transfer learning architectures such as VGG16, VGG19, ResNet50, MobileNetV2, EfficientNetB0, and Xception for medicinal plant health classification.
- Analysis of multiple performance metrics, including accuracy, precision, recall, F1-score, confusion matrices, and learning curves.

To the best of our knowledge, this is the first study to focus exclusively on the health of Eranda and Nirgundi leaves. Given their extensive use in traditional Indian medicine, this work addresses a significant research gap and encourages further exploration of their pharmacological properties. This study advances scholarly research and provides valuable resources for smart farming, agro-tech sector, and multidisciplinary research connecting pharmacology, AI, and agriculture. It has the potential to drive improvements in natural medicine production and sustainable farming practices, paving the way for future studies on medicinal plant health diagnostics.

2. Literature Review

Effective solutions for monitoring plant health require an in-depth understanding of existing methods and technologies. This section reviews the importance of Vitex negundo and Ricinus communis and the existing research on medical plant disease detection using machine learning to clearly establish the novelty of the proposed work.

Vitex negundo, commonly known as Nirgundi, is extensively used in the Indian Ayurvedic system to treat ailments such as arthritis, bronchitis, and skin disorders. Most parts of this plant are used in Indian medicine. Leaves, known for their aromatic properties, are used to relieve headaches and catarrh when burned, and their decoction is effective against febrile conditions. Juice extracted from the leaves is used to treat fetid discharges

because of its anti-inflammatory, antibacterial, and antifungal properties. The roots are used to manage diarrhea, and the flowers and fruits exhibit astringent and vermifugal properties, respectively [8,9]. Nirgundi is native to south and southeast Asia, including India, Afghanistan, Bangladesh, and Indo-Malaysia, and it is cultivated in parts of America and Europe. Research has demonstrated its efficacy in treating conditions such as cephalgia (headache), otalgia (ear pain), and rheumatism (arthritis). Additionally, experimental studies have highlighted its anticonvulsant properties in seizures [10,11].

In Ayurvedic texts, nirgundi is described as having a bitter, spicy, and alkaline taste (Rasa: Katu, Tikta, and Kashaya). It is known to pacify vata (wind) and shleshma (mucus) and is used to treat spleen disorders, intestinal worms, leprosy, poisoning, and anorexia nervosa [12,13].

Active compounds in nirgundi, such as flavonoids, are responsible for their therapeutic effects. The ongoing cultivation of the plant supports both modern and traditional herbal medicine practices. However, the overharvesting and habitat loss of Nirgundi threatens its availability. Experimental studies, such as those by Mridula et al., highlight the anti-inflammatory properties of its leaves, which were found to be more effective than the roots in reducing inflammation caused by carrageenan in albino rats. These findings underscore the potential of this plant as an alternative to standard anti-inflammatory medications [14]. Beyond its medicinal use, Nirgundi is revered for its role in rituals, reproductive health, and as an insecticide. Vitamin C is used to treat pharyngitis, tonsillitis, and orchitis and exhibits ecto-parasiticidal properties, helping to repel lice in poultry. Studies have also emphasized its efficacy in managing toxicity and skin diseases [15–17].

Ricinus communis, commonly known as castor oil plant or Eranda, belongs to the Euphorbiaceae family and has been used medicinally across cultures. Despite the presence of ricin, a toxic compound, the plant exhibits numerous therapeutic applications, including its use as a laxative and for treating fever, diarrhea, piles, cough, chest pain, constipation, obesity, and edema [18,19]. The diverse phytochemical profiles of plants include flavonoids, fatty acids, phenolic compounds, terpenoids, amino acids, and phytosterols. These compounds contribute to their wide-ranging healing effects and relevance in traditional medicine [20,21]. For instance, powdered leaves show notable insecticidal properties against pests, such as aphids, rust mites, mosquitoes, and whiteflies, making the plant valuable in agriculture. In addition, the leaves are used as dietary supplements for livestock to enhance milk production in cows and buffaloes. Ricinine, an alkaloid found in R. communis, exhibits significant pain-relieving properties and acts as a stimulant of the central nervous system (CNS). Studies suggest its potential in treating neurological conditions, improving cognitive abilities, and managing psychiatric ailments without inducing anxiety [22]. This plant has also shown efficacy against Anopheles gambiae, the primary vector for malaria. Its larvicidal activity, attributed to ricinine and 3-carboxy-4-methoxy-N-methyl-2-pyridone (C8H8NO), disrupts larval development and survival, presenting a natural insecticide option for pest and malaria vector management [23]. In addition, Eranda has potential as a natural contraceptive. A clinical study involving 50 women demonstrated that a single oral dose of 2.3–2.5 g of a seed-derived formulation (RICOM-1013-J) provided effective contraception for one year with minimal side effects. The study reported no adverse effects on renal or hepatic function, further validating the pharmacological applications of the plant [24].

The application of machine learning to monitor the health of medicinal plants is critical to ensure the potency and quality of herbal medicines. Despite the substantial research on plant health diagnostics using machine learning, relatively few studies have focused on medicinal plants. A significant barrier is the lack of comprehensive datasets on Ayurvedic medicinal plants. In [25], deep learning techniques achieved an accuracy of

98.08% in identifying medicinal plants. Similarly, ref. [26] reported 95.97% accuracy using machine learning to classify six medicinal plants, Ocimum sanctum, Mentha balsamea, Aegle marmelos, Melissa officinalis, Nepeta cataria, and Stevia rebaudiana, based on 220 × 220 pixel images. A hybrid deep neural network combining DenseNet201 and log Gabor filters was employed in [27], achieving a 97% accuracy in medicinal plant identification. The Xception model integrated with a Support Vector Machine (SVM) demonstrated 97.5% accuracy [28]. Additionally, ref. [29] utilized a deep neural network (DNN) to identify medicinal crops and achieved an accuracy of 98.68%. The MobileNet architecture used in [30] identified six different medicinal plants with an accuracy of 98.3%, whereas the DeepCNN model in [31] achieved a slightly lower accuracy of 92.7%.

Most existing studies rely on proprietary datasets because of the vast number of medicinal plant species, making the development of a universal dataset challenging [32]. Moreover, existing research has predominantly focused on plant identification rather than on assessing plant health. Notably, few studies have addressed the health status of Vitex negundo (Nirgundi) and Ricinus communis (Eranda) despite their significant applications in Ayurvedic medicine. This research aims to bridge this gap by focusing on health diagnostics of Nirgundi and Eranda and leveraging machine learning to enhance their medicinal utility and sustainability.

3. Proposed Methodology

The proposed methodology contains the following three major stages: 1. dataset creation, 2. application of transfer learning techniques on newly created dataset and 3. model evaluation and analysis. The primary aim of the proposed methodology was to create a thorough and trustworthy dataset for evaluating the health condition of leaves in Eranda (Ricinus communis) and Nirgundi (Vitex negundo). Figure 1 illustrates the workflow of the entire work.

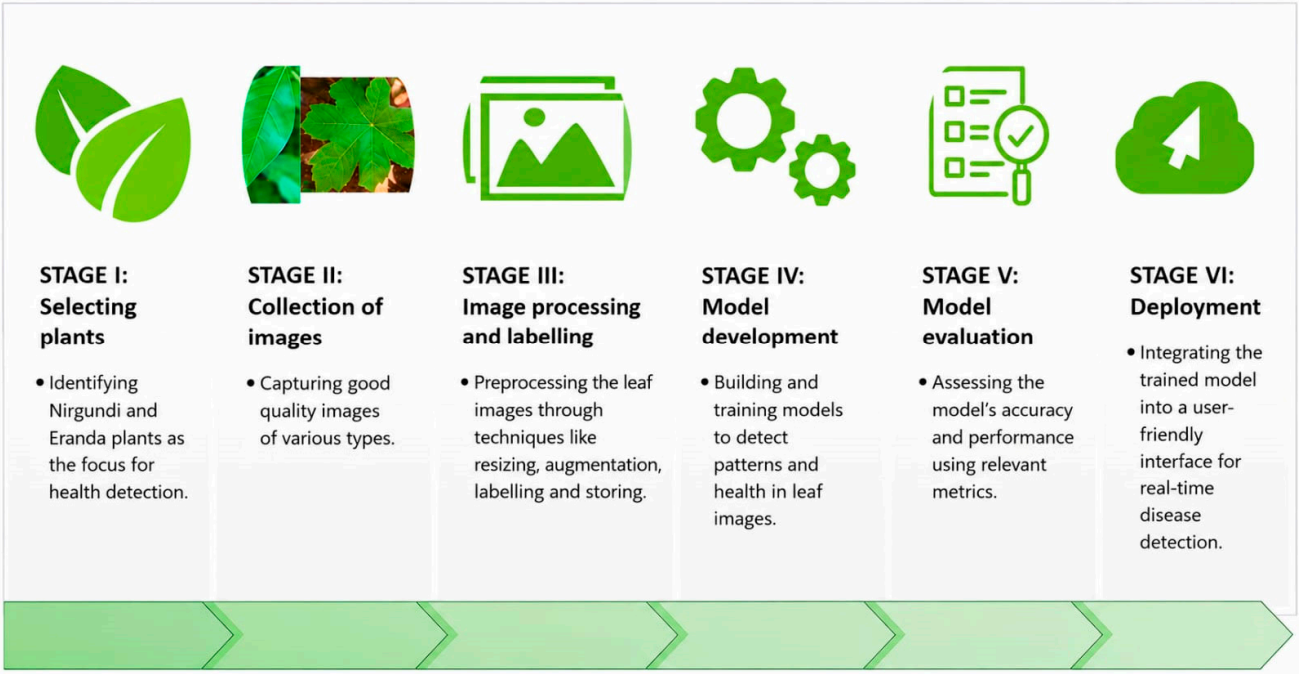

Figure 1. Methodology of the proposed work.

The process begins with data collection, in which raw data are gathered from various sources for analysis. If there are insufficient data, more data are acquired until the required threshold for effective modelling is reached. Once sufficient data were collected, data

preprocessing was performed. This step involves cleaning the data by removing noise, handling missing values, normalizing or scaling features, and converting the data into an appropriate format for the analysis. After preprocessing, the next step was to apply feature extraction techniques. These techniques help in selecting or engineering the most relevant features from the dataset, which improves the model performance by focusing on the most informative variables and reducing dimensionality.

After filtering, the dataset is ultimately refined, different models' training begins, and machine learning concepts are applied to learn patterns in the data. The models were then evaluated, where testing is performed to make models as accurate as they can. During deployment phase best performed model was selected that satisfied predefined performance criteria. If the model's performance is found to be inadequate, hyperparameter tuning is performed by optimizing key training parameters, such as the learning rate, batch size, optimizer, and number of epochs, to improve classification performance. The optimized model is then re-evaluated, and this iterative process continues until satisfactory performance is achieved. Finally, the validated model is deployed to generate predictions on new input images in real-world using web-based application.

3.1. Dataset Description

The dataset generation process includes planning, image acquisition, classification, preprocessing, and organization of the dataset. Figure 2 provides a detailed view of the steps involved in generating the dataset for the Nirgundi (Vitex negundo) and Eranda (Ricinus communis) leaves.

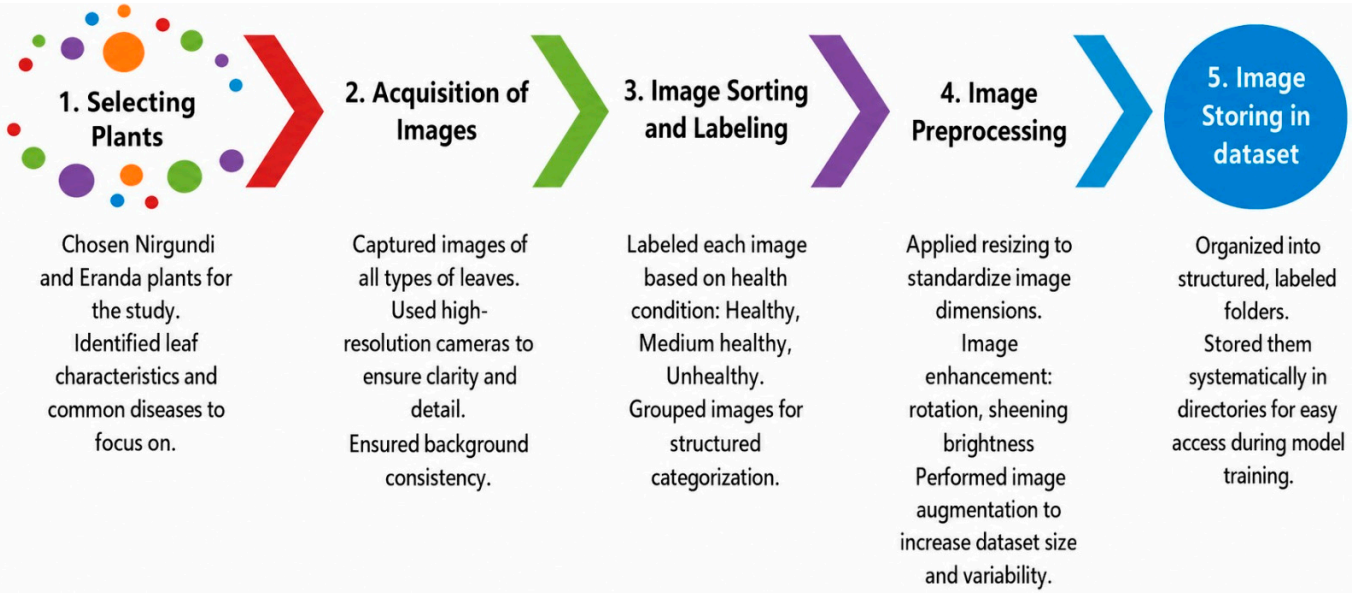

Figure 2. Flowchart of medicinal leaf dataset generation process.

In the first stage of dataset preparation, we collected 6630 high-quality images of Eranda and Nirgundi leaves from DY Patil Grounds and Botany Garden, Pune, Maharashtra, India (latitude—18.621195, 18°37′16.3020″ N and longitude—73.911491, 73°54′41.3676″ E). These images were acquired using iPhone 15. We took the images during daylight and from a variety of perspectives and different angles to minimize inconsistencies and improve accuracy. The images are in JPEG format (.jpg extension) and later resized to 1920 × 1080 pixels. Acquired leaf images of both plants were manually annotated into three health categories (Healthy, Medium Healthy, and Unhealthy) based on visible disease symptoms, including leaf discoloration, chlorosis, necrosis, curling, and tissue damage. The annotation process was carried out in consultation with an experienced

Ayurveda physician, and the assigned labels were verified using established diagnostic criteria reported in the published literature [33–38].

In addition, the dataset accounts for a wide range of leaf orientations, angles, and varied lighting scenarios, making it robust and reliable for developing trustworthy classification systems. This image labelling enables researchers to refine the algorithms and achieve high accuracy in diagnosing the specific health states of medicinal plants. The precise labelling and bifurcation of the dataset provide a robust foundation for machine learning models, enhancing their ability to predict plant health and advancing applications in agriculture and plant pathology. Leaf health categories are as follows:

1. Healthy: Leaves with no signs of disease or damage.
2. Medium Healthy: Leaves exhibiting mild stress or early-stage disease symptoms.
3. Unhealthy: Leaves significantly affected by disease or environmental stressors.

By enabling the accurate classification of plant health, this dataset serves as a valuable resource for improving predictive models and supporting advancements in sustainable agriculture. Table 1 details the distribution of the images categorized by health class for each leaf type.

Table 1. Quantitative breakdown: image count per leaf category.

Leaf	Health Classification	Image Count
Eranda	Healthy	1326
	Medium Healthy	884
	Unhealthy	624
	TOTAL	2834
Nirgundi	Healthy	1170
	Medium Healthy	1274
	Unhealthy	1352
	TOTAL	3796

Figure 3 contains sample images of the different health classes of Eranda and Nirgundi leaves from the dataset.

A Python Imaging Library (PIL) (Pillow 11.1.0) framework was used to apply uniform preprocessing steps to ensure consistency across the dataset. Images were cropped and resized to 1920 × 1080 p× and then to 224 × 244 p× saved in JPEG format with. jpg extension to bring all photographs in the dataset to have the same size. For application with computer vision and machine learning, this format is optimized for efficient storage and processing of images. The images were saved with a resolution of 92 dots per inch (dpi) so that the leaves' visual representations were held at high-quality detail. Brightness and saturation of the leaves was adjusted to improve general visibility and consistency. Such a process includes:

- **Rotation**: Images within a given range are rotated at random to enable the model to learn from various leaf orientations, thereby making it more robust to real-world variations.
- **Segmentation**: This is the division of images into smaller focused sections. Thus, the focused parts of the leaves were studied in detail. It comes in handy when slight changes in health status are observed that cannot be seen in the whole view.
- **Translation**: Shifting images in various directions generates new perspectives on leaves, enabling the model to better understand different spatial representations.

Ayurvedic Medicinal Leaves

Sr	Erenda Leaves (*Ricinus communis*)	Sample 01	Sample 02
01	Healthy		
02	Medium Healthy		
03	Unhealthy		

(**a**)

Ayurvedic Medicinal Leaves

Sr	Nigundi Leaves (*Vitex negundo*)	Sample 01	Sample 02
01	Healthy		
02	Medium Healthy		
03	Unhealthy		

(**b**)

Figure 3. Sample images of (**a**) Eranda medicinal leaves and (**b**) Nirgundi medicinal leaves.

After image enhancement, the next step was to use data augmentation, which is the process of increasing the diversity and comprehensiveness of the dataset. The image was then converted into a NumPy array, which was used to augment it with parameters such as rotation, shifting, and shearing, using an ImageDataGenerator. After augmentation, all images were saved to the output folder, and there was a significant diversity in the dataset for other tasks, such as further training on machine learning models.

Each image is categorically maintained in various folders with the name of the health category, which makes it easy to access. For easier identification and organization, each category folder assigns new sequential names to the images within them. Figure 4 shows the directory structure of the dataset that can be used to store the images in a proper and more appropriate manner.

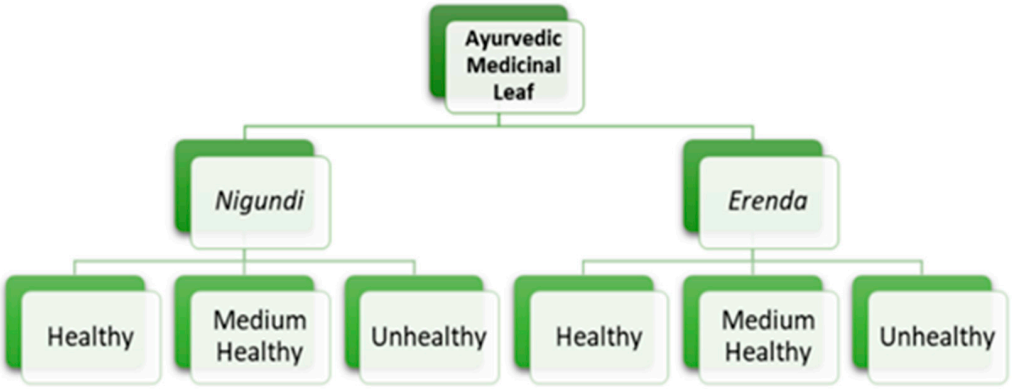

Figure 4. Directory structure of the Indian medicinal leaf images.

These techniques enhance the dataset with diversified representations of leaf health, which profoundly improves the model's ability to classify health classes. In summary, using this broad approach, machine learning models will have a comprehensive training dataset that will boost their performance in health class assignments on Nirgundi and Eranda leaves.

3.2. Transfer Learning Analysis

The proposed work used six different transfer learning techniques such as VGG16, VGG19, ResNet50, MobileNetV2, EfficientNetB0, and Xception for medicinal plant leaf disease classification. Transfer learning is a machine learning technology in which a pre-trained model is applied to answer a new but related task more effectively. Transfer learning uses information from an existing model trained on a large general dataset rather than building information from scratch. This method is particularly useful for our proposed work as this requires a limited amount of data.

Before starting the model training, we divided the dataset randomly into the following three splits: training set with 70%, validation set with 15% and testing set with 15%. A stratified sampling strategy was employed to preserve the proportion of each health category (Healthy, Medium Healthy, and Unhealthy) across all subsets. This ensured that each subset maintained a representative class distribution while preventing class imbalance. The test set was kept completely independent and was used only for the final performance evaluation, whereas the validation set was utilized for hyperparameter tuning and model selection.

3.2.1. Model 1: VGG16 Architecture

In this study, VGG16 architecture was used to classify medicinal plant leaves into the following three health categories: Healthy, Medium Healthy, and Unhealthy. The VGG16 network, which was pre-trained on ImageNet, was fine-tuned for the leaf classi-

fication task. Normalized leaf images of size 224 × 224 × 3 pixels are provided as input to the network. Figure 5 shows in VGG16 network used in this study. The initial convolutional layers capture low-level features such as edges, vein patterns, colour gradients, and texture, whereas the deeper layers extract high-level disease-related characteristics, including yellowing, necrotic spots, discoloration, fungal symptoms, and insect damage. The convolution operation is mathematically represented as below in Equation (1).

$$\mathrm{Output}(\mathrm{i},\mathrm{j}) = \sum_{\mathrm{m}=0}^{\mathrm{k}-1} \sum_{\mathrm{n}=0}^{\mathrm{k}-1} \mathrm{Input}(\mathrm{i}+\mathrm{m},\mathrm{j}+\mathrm{n}) \cdot \mathrm{Kernel}(\mathrm{m},\mathrm{n}) + \mathrm{b} \quad (1)$$

where Input (*i,j*) is the pixel value at (*i,j*), Kernel (*m,n*) is the applied filter, *b* is the bias term, and *k* is the kernel size (e.g., 3 × 3 × 3 for VGG16). Each convolution layer followed a dense layer with 256 units, a ReLU activation function for non-linearity, and a dropout layer with a 0.5 rate to mitigate overfitting using Equation (2).

$$\mathrm{ReLU}(\mathrm{x}) = \max(0,\mathrm{x}) \quad (2)$$

New layers were added on top of these frozen layers: a global average pooling layer to reduce the dimensionality of the feature map as shown in below equation.

$$\mathrm{Output}_c = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \mathrm{FeatureMap}_c(i,j) \quad (3)$$

where *H* is the height of the feature map representing a specific feature, *W* is the width of the feature map based on the image size, and *c* is the channel index.

The model concludes with a Softmax output layer designed for multiclass classification as shown in Equation (4).

$$\mathrm{Softmax}(z_j) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \quad (4)$$

where z^j is the logit of class *j* and *K* is the total number of classes.

Adam optimizer was used to combine the concepts of both momentum and RMSProp to adaptively modify the learning rate. The update rule for each parameter θ_t is expressed as follows:

$$\theta_t = \theta_{t-1} - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \quad (5)$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1}, \quad \hat{v}_t = \frac{v_t}{1-\beta_2} \quad (6)$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \quad (7)$$

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t \quad (8)$$

where g_t is the gradient at time step *t*; β_1, and β_2, are the decay rates for the moving averages, typically set to 0.9 and 0.999, respectively; α is the rate of learning which is commonly set to 0.001; and ϵ is a small constant (typically 10^{-7}) to prevent division by zero.

Sparse categorical cross-entropy was selected as the loss function, suitable for multi-class problems, and accuracy was used as the primary evaluation metric. To evaluate the generalization, the model was trained over multiple epochs with continuous monitoring of the training and validation accuracy. Following the training phase, the classification performance of the model was assessed by comparing its predictions with true labels. The final test accuracy is measured using a separate test dataset. The F1 score was calculated alongside the accuracy to provide a balanced measure of recall and precision, which is particularly useful for handling unbalanced datasets. A comprehensive classification report is generated to present the performance metrics for each class. To further evaluate the

performance of the model, a confusion matrix was computed to identify misclassifications among health categories. Additionally, the training and validation accuracy trends over epochs were plotted to illustrate the learning process and detect potential overfitting. These evaluations collectively ensure a comprehensive analysis of the model's performance and reliability.

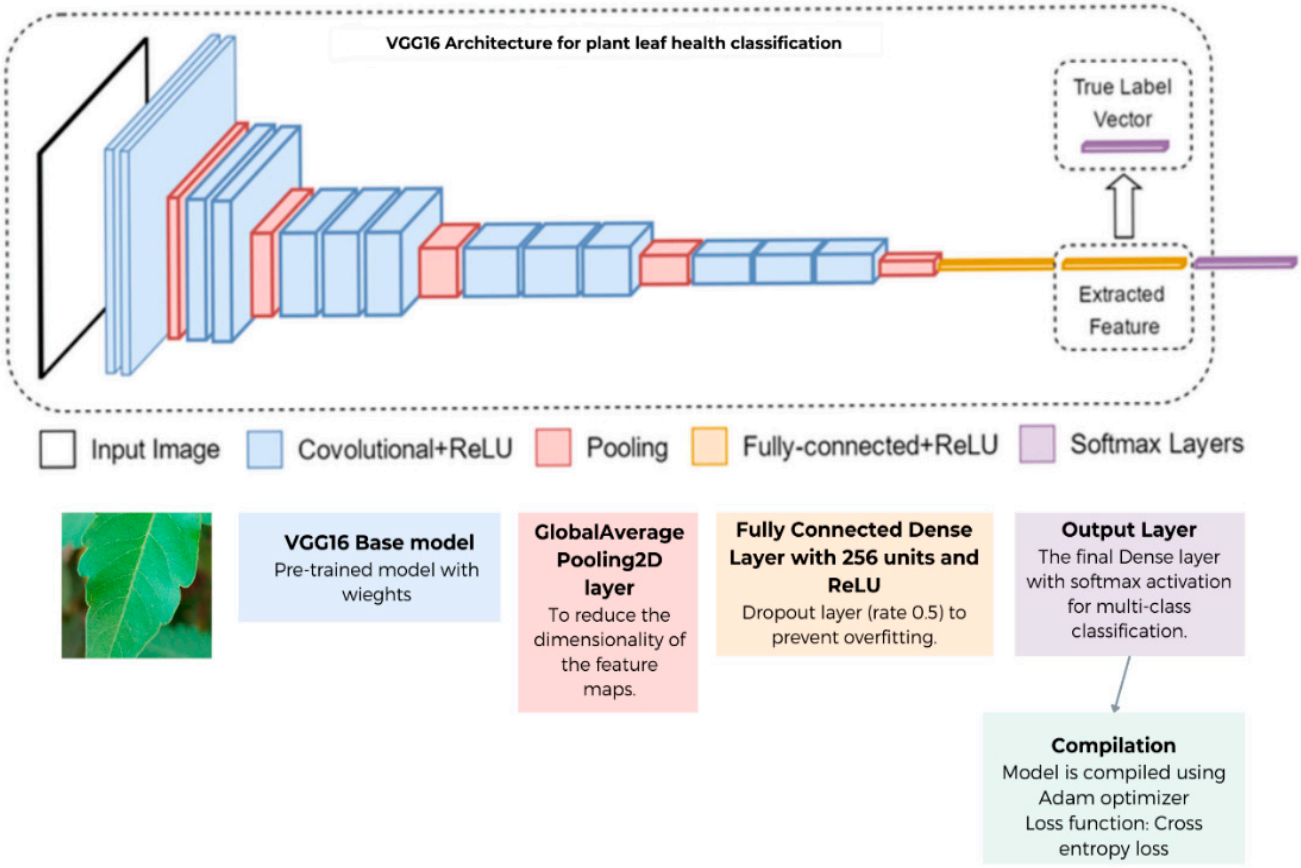

Figure 5. Architecture of VGG16 for proposed work.

3.2.2. Model 2: ResNet50 Architecture

This is the second architecture that utilizes ResNet50 to classify leaf health. Transfer learning was performed using pre-trained weights from ImageNet. This helped us to adapt effectively to our specific task while working with a much smaller dataset. ResNet50 is a member of the "Residual Network," a class of networks that introduced skip connections (identity mappings). Figure 6 illustrates the flow of the ResNet50 model. The input image was processed through a series of convolutional layers combined with weight layers and ReLU activation functions. The model incorporates skip connections that provide identity mapping, thus ensuring the preservation of crucial information throughout the network.

The network was fine-tuned to classify the leaves. Some pre-trained layers captured a variety of features. Initially, they were frozen to retain their learned weights. New layers with the addition of a dimensionality reduction layer using global average pooling were added. A dense layer of 256 units was also added to the ReLU to induce non-linearity. A dropout rate of 0.5 used to decrease overfitting. The output layer was the final SoftMax (Equation (4)) multiclass classification layer for the ResNet. Each of its convolutional layers is defined by the following:

$$Y = \mathrm{Conv}(X, W) + b \tag{9}$$

where Y is the output feature map, Conv (X, W) is the convolution operation applied to input X using weights W, and b is the bias term. After each convolutional layer, batch normalization is often applied, which is defined as below:

$$\Upsilon_{\text{norm}} = \frac{\Upsilon - \mu}{\sigma} \cdot \gamma + \beta \tag{10}$$

where Y is the output of the convolutional layer, μ is the batch mean, σ is the standard deviation, γ is a scale parameter, and β is a shift parameter.

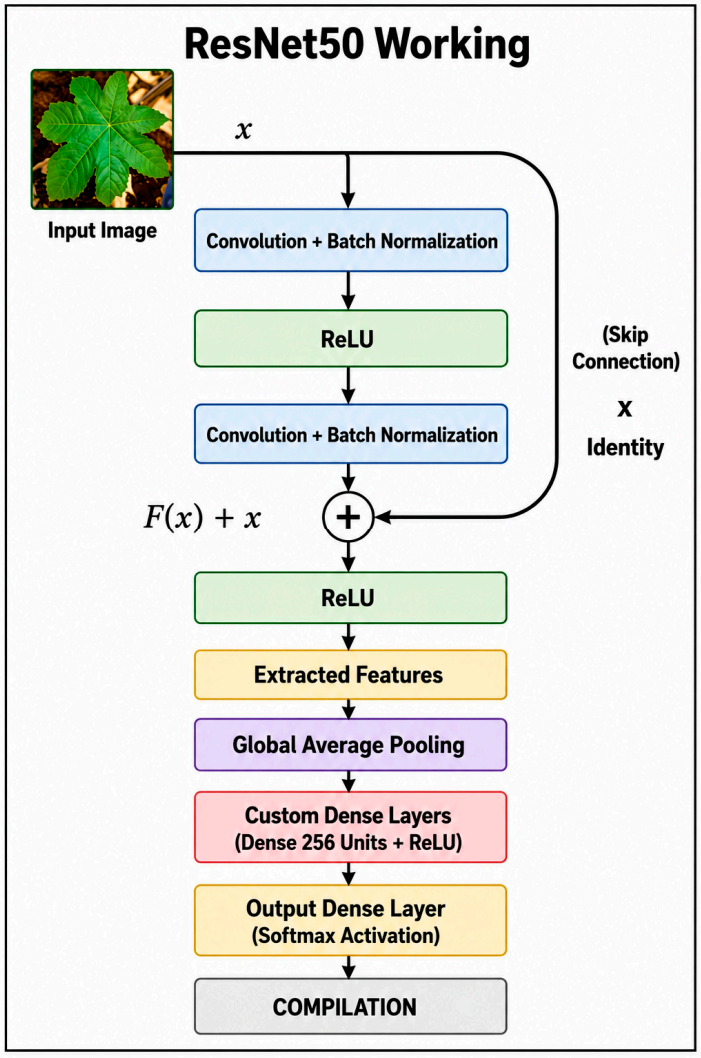

Figure 6. ResNet50 structure.

Training uses the sparse categorical cross-entropy loss function that is appropriate for our multi-class job and the Adam optimizer, which is also known for its effectiveness and versatility in controlling gradients. Accuracy was the primary evaluation metric used to monitor performance. To measure the generalization, we monitored the training and validation accuracy of the model over several epochs. During the training process, we monitored both the loss and the accuracy of the training and validation datasets. After training, we tested the classification performance of the model by comparing the predictions to actual labels and computing the final test accuracy on another test set.

To evaluate the performance of the model, a confusion matrix was computed to represent the correct classifications versus misclassifications across the health categories. The confusion matrix was further visualized as a heatmap to improve interpretability.

3.2.3. Model 3: MobileNetV2 Architecture

The architecture designed as part of this experiment, which was used for classifying leaf health by applying transfer learning, is a type named MobileNetV2, which is a mobile structure intended specifically for mobile devices and is used extensively in embedded vision application cases. MobileNetV2 is particularly designed in accordance with mobile as well as resource-constrained settings where high accuracy would become possible with low complexity by using efficient building blocks through which the model could fit in a smaller space together with minimized latencies, making this technology mainly beneficial for real-time applications on low-power devices.

This method employs pre-trained weights on ImageNet, allowing the use of a much smaller, domain-specific dataset, and still achieves high classification performance. MobileNetV2's skip-connected inverted residual block enhances the flow of features across the network. This architecture captures valuable features by maintaining the residual information to improve the handling of complex data patterns in the network. The linear bottleneck layer decreases the dimensions of each block, eliminating redundant feature extraction, and resulting in meaningful representations.

The MobileNetV2 model was used minus the fully connected layers for this specific leaf classification task. The base model architecture utilizes depthwise separable convolutions to capture salient features, decreasing computation without loss in accuracy. The model was modified for our classification challenge by adding new layers after the pre-trained layers were frozen during the initial training, while retaining their learned features. This consisted of a dense layer of 256 units that used ReLU activation to introduce non-linearity, and a global average pooling layer to compress the feature maps. A dropout layer with a rate of 0.5, which is overfitting, was used to prevent a commonly known problem in machine learning. For the output layer, we used the Softmax activation function to classify the leaves into different health classes using the number of target classes.

Figure 7 shows the MobileNetV2 structure, in which two major categories—depthwise and pointwise of convolutions—appear. In Section (a) of the figure, a standard sequence of convolution is shown, where pointwise and depthwise separable convolutions help reduce the cost of computation by processing the input image. Section (b) of the figure presents an inverted residual block that combines convolution layers with skip connections. The blocks shave irrelevant features while conserving important features. Thus, the architecture permits MobileNetV2 to be both efficient and accurate for classification. An Adam optimizer used to handle sparse gradients. This model was built using efficiently and adaptively the sparse categorical cross-entropy appropriate loss function for multiclass classification tasks with integer-target labels. Accuracy was set as the primary evaluation criterion to evaluate the performance of the model at different training stages.

To measure the extent to which the model learns and generalizes to the latest set of data, the training process was iterated over the dataset for a few epochs and both training and validation accuracies were tracked. Subsequently, the result of the model was evaluated on a different test set to determine the final testing accuracy. In addition to accuracy, another evaluation metric that was utilized was the F1 score. It is an equilibrium between precision and recall and is pertinent in light of possible class imbalances in the dataset. Therefore, MobileNetV2 aided the rapid development of a reliable model for leaf health classification, effectively utilizing transfer learning to achieve robust performance even with a limited dataset.

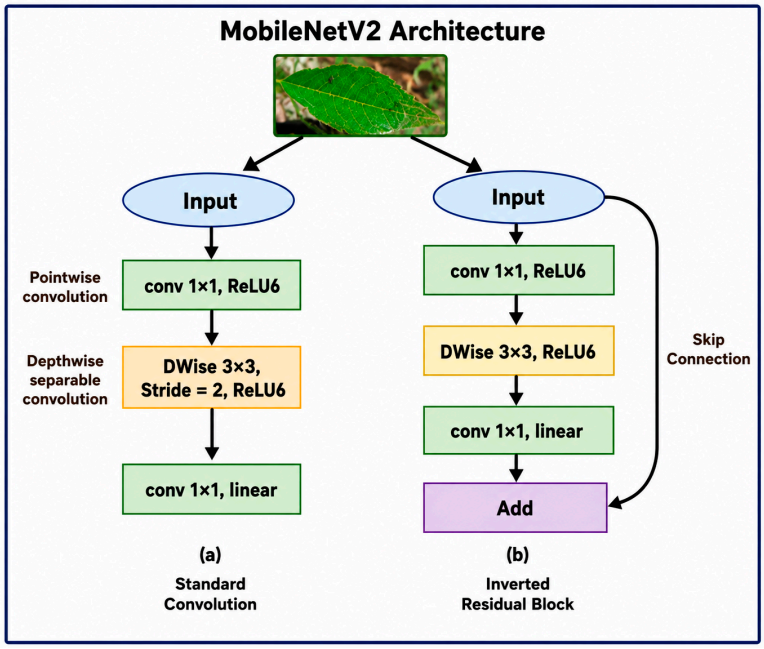

Figure 7. MobileNetV2 structure.

3.2.4. Model 4: Xception Architecture

Xception, the fourth architecture evaluated, replaces standard Inception modules with depthwise separable convolutions, decoupling spatial and channel-wise feature learning to improve parameter efficiency. Using ImageNet pre-trained weights, the frozen base was extended with a global average pooling layer, a 256-unit dense layer with ReLU activation, a dropout layer (rate 0.5), and a Softmax classification layer for the Healthy, Medium Healthy, and Unhealthy categories. The model was compiled with the Adam optimizer and sparse categorical cross-entropy loss, and training and validation accuracy and loss were monitored over the training epochs to assess convergence and generalization.

3.2.5. Model 5: EfficientNetB0 Architecture

EfficientNetB0 served as the fifth model, using compound scaling to balance network depth, width, and input resolution for an optimal accuracy-to-efficiency ratio. Built on mobile inverted bottleneck blocks with squeeze-and-excitation attention, the architecture emphasizes informative channels while suppressing redundant ones. The pre-trained ImageNet backbone was frozen, and a global average pooling layer, a 256-unit dense layer with ReLU activation, a dropout layer (rate 0.5), and a Softmax output layer were added for the three health categories. Training used the Adam optimizer with sparse categorical cross-entropy loss, and accuracy and loss trends were tracked across epochs.

3.2.6. Model 6: VGG19 Architecture

VGG19, the sixth and final architecture, extends the VGG16 design with three additional convolutional layers, deepening the hierarchical feature extraction while retaining uniform 3×3 filters throughout the network. This extra depth allows the model to learn more refined disease-related patterns such as subtle discoloration and lesion boundaries. The frozen ImageNet-pre-trained base was followed by a global average pooling layer, a 256-unit dense layer with ReLU activation, a dropout layer (rate 0.5), and a Softmax layer for classifying leaves into the three health categories. The Adam optimizer and sparse categorical cross-entropy loss were used during training, with accuracy and loss monitored across the training and validation sets.

4. Model Evaluation

Model calibration was performed by fine-tuning the varying hyperparameters, such as the number of images taken for each batch and the dimensions of the images. It uses a batch size of 32, resizing the images to a standard dim of 224 × 224 pixels across the dataset. The training, validation, and test datasets were then created using TensorFlow's image dataset utilities, which made it easy to load the images from the aforementioned directory. The following are some important metrics for evaluating the efficiency of a model.

(1) Accuracy: Accuracy represents the ratio of correctly classified instances (both positive and negative) to total number of instances as shown in Equation (11). The overall performance of the model is evaluated. Mathematically, it is calculated as the sum of true positives (TPs) and true negatives (TNs) divided by the total number of instances (the sum of TP, TN, false positives (FPs), and false negatives (FNs)).

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \tag{11}$$

(2) Precision: Precision quantifies the proportion of true positive predictions of all positive predictions made by the model as derived in Equation (12). This indicates that the number of predicted positive cases was positive. This is particularly important when false positives are expensive.

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{12}$$

(3) Recall (Sensitivity): Recall, also known as sensitivity, measures the model's ability to identify all actual positive instances in the dataset using Equation (13). This is the ratio of true positives to the sum of true positives and false negatives, highlighting how many of the actual positive cases were correctly identified.

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{13}$$

(4) F1 Score: The F1 Score (Equation (14)) is the harmonic mean of the precision and recall, offering a single metric that balances the trade-off between these two measures. It is particularly useful for datasets with class imbalances, as it considers both false positives and false negatives in its calculation.

$$\text{F1} - \text{score} = 2 * \frac{(\text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})} \tag{14}$$

(5) Confusion Matrix: The confusion matrix is a two-dimensional table that summarizes the performance of a classification model by showing the counts of true positives, true negatives, false positives, and false negatives for each class. This breakdown allows for detailed insights into model behaviour across different categories.

Multiple epochs were utilized in the training, with one epoch defined as one full run through the training dataset. The model was then found to be enhanced with an increased number of epochs, which allowed it to learn the more complex features that existed in the data while observing the validation accuracy to avoid overfitting. This process clearly illustrates the point at which one had to have a balance between the complexity and training time of the model to give excellent results to the picture classification problem.

4.1. Results and Discussion

In this research, six architectures of convolutional neural networks, VGG16, ResNet50, MobileNetV2, Xception, EfficientNetB0, and VGG19, were tested by classifying the leaf

health based on their accuracy and loss metrics during training and validation over five epochs. Table 2 summarizes the results of the same.

Table 2. Training and validation accuracy of different models.

Model	Epoch No.	Training		Validation		Architectural Reason for Performance
		Accuracy	Loss	Accuracy	Loss	
VGG16	1	0.6958	1.2638	0.9307	0.2389	High training and validation accuracies with low losses indicate effective convergence. The sequential convolutional architecture learns hierarchical leaf features efficiently and generalizes well without noticeable overfitting.
	2	0.8890	0.2910	0.9699	0.1223	
	3	0.9325	0.1793	0.9779	0.0831	
	4	0.9601	0.1173	0.9829	0.0630	
	5	0.9713	0.0853	0.9893	0.0432	
ResNet50	1	0.7961	0.5357	0.9448	0.1699	Nearly identical training and validation accuracies demonstrate excellent generalization. Residual skip connections added in the architecture stabilize optimization and enable effective learning of complex disease features.
	2	0.9424	0.1700	0.9689	0.1034	
	3	0.9631	0.1054	0.9829	0.0666	
	4	0.9772	0.0696	0.9789	0.0569	
	5	0.9849	0.0482	0.9880	0.0388	
MobileNetV2	1	0.5432	1.1595	0.6647	0.7845	Moderate training and validation accuracies suggest limited feature learning. The lightweight architecture prioritizes computational efficiency over representational capacity, reducing classification performance.
	2	0.6978	0.7685	0.7309	0.6320	
	3	0.7484	0.6416	0.8062	0.5257	
	4	0.7948	0.5454	0.8454	0.4446	
	5	0.8258	0.4704	0.8484	0.4020	
Xception	1	0.3844	3.6509	0.5663	1.1119	Low training accuracy with relatively higher validation accuracy indicates insufficient convergence rather than overfitting. The deeper separable-convolution architecture requires more training data and epochs to achieve optimal performance.
	2	0.4865	1.1978	0.5873	0.9784	
	3	0.5314	1.0886	0.6687	0.8378	
	4	0.5656	0.9899	0.7078	0.7552	
	5	0.5863	0.9568	0.6938	0.8008	
EfficientNetB0	1	0.8103	0.4923	0.9538	0.1729	A good validation accuracy with less validation loss indicates superior generalization. Compound scaling balances model capacity and feature extraction, producing highly discriminative representations.
	2	0.9403	0.1821	0.9749	0.0989	
	3	0.9696	0.1083	0.9940	0.0461	
	4	0.9806	0.0699	0.9819	0.0571	
	5	0.9821	0.0577	0.9910	0.0342	
VGG19	1	0.6646	1.4997	0.8976	0.2902	Slightly slower convergence than VGG16 due to additional layers but achieves higher final training accuracy with low loss. Higher validation accuracy and lower validation loss indicate improved feature representation.
	2	0.8763	0.3202	0.9568	0.1487	
	3	0.9302	0.1980	0.9608	0.1123	
	4	0.9461	0.1459	0.9749	0.0850	
	5	0.9590	0.1136	0.9809	0.0694	

The accuracy of the VGG16 model remarkably increased throughout the epochs, concomitant with the training and validation accuracy. By the fifth epoch, it increased to 97.13% from its first training accuracy of 69.58% in the first epoch. The ability of the model to standardize well on unknown data was evident by the significant increase in the validation accuracy, which increased from 93.07% to 98.93%. As the training loss decreased from 1.2638 to 0.0853 and the validation loss decreased from 0.2389 to 0.0432, the loss metrics indicated a persistent downward trend. VGG16 achieved high classification accuracy because of its simple and sequential architecture composed of multiple 3×3 convolutional filters. These small kernels effectively capture local texture patterns, leaf edges, colour transitions, and disease spots while preserving spatial information. Since medicinal leaf health classification depends largely on subtle visual characteristics rather than complex object structures, the hierarchical feature extraction of VGG16 is well suited for this application.

ResNet50 also showed excellent performance metrics, with the training accuracy having increased by 98.49% at the fifth epoch from a mere 79.61% in the first epoch. This was an indication of the tremendous capacity in learning, as the validation accuracy increased from 94.48% to 98.80%. The validation loss decreased from 0.1699 to 0.0388, and the training loss decreased dramatically from 0.5357 to 0.0482. The results highlighted that ResNet50

was not only good in terms of accuracy but also had low loss values, indicating a well-calibrated model with minimal overfitting tendency. ResNet50 demonstrated comparable performance owing to its residual learning framework. The skip connections alleviate the vanishing gradient problem, enabling deeper feature learning without degradation in training performance. Residual blocks allow the network to learn both low-level and high-level representations simultaneously, making the model robust to variations in illumination, leaf orientation, and background conditions.

MobileNetV2, on the other hand, reported comparatively lower accuracy values for all epochs. The training accuracy remained at 54.32% before escalating to its highest value of 82.58% during the fifth epoch. Additionally, the validation accuracy was also enhanced, which increased from 66.47% to 84.84%. This pattern was also observed in the loss measures, which indicated a reduction in training loss from 1.1595 to 0.4704 and loss of validation from 0.7845 to 0.4020. Despite the improved accuracy of MobileNetV2, its performance was not as high as that of VGG16 and ResNet50. It may be due to the fact that MobileNetV2 was designed keeping in mind efficiency and speed rather than great accuracy.

Xception showed the weakest and most erratic convergence among the six models that converged, with training accuracy rising only from 38.44% to 58.63% and validation accuracy from 56.63% to 69.38% over five epochs; its depthwise separable convolutions appear to need more training data and epochs than were available here to fit the leaf-health task well. EfficientNetB0, by contrast, converged quickly and reliably, with training accuracy improving from 81.03% to 98.21% and validation accuracy from 95.38% to 99.10% while validation loss fell to 0.0342, reflecting the benefit of its compound scaling of depth, width, and resolution for this classification problem. VGG19 followed a pattern similar to VGG16 but converged slightly more slowly because of its additional convolutional layers, with training accuracy increasing from 66.46% to 95.90% and validation accuracy from 89.76% to 98.09% by the fifth epoch.

Visual results of leaf health assessment of the proposed work are shown in Figure 8. These assessments are captured from web application of the work.

4.2. Performance Metrics

Table 3 presents the performance metrics of the VGG16, ResNet50, MobileNetV2, Xception, EfficientNetB0, and VGG19 models, respectively showing exhibit heterogeneous levels of performance. VGG16 performed extraordinarily well on precision and F1 scores as well as recall across all classes, whose scores were almost 1.00; this is indicative of very good classification strengths. ResNet50 also performed excellently; both precision and recall were observed to be high, especially for the “Unhealthy” class. On the other hand, MobileNetV2 had a reasonable performance, significantly for the “Medium Healthy” class, whose precision dropped to 0.71. On average, VGG16, EfficientNetB0 and VGG19 led the way in the task of predicting leaf health, whereas MobileNetV2 offered at least theoretical opportunities to gain, mainly with regard to recall and precision for some classes.

4.3. Confusion Matrix

In Figure 9, confusion matrices are for all six models illustrated showing classification of Healthy, Moderately Healthy, and Unhealthy leaves of Eranda and Nirgundi medicinal plants. Among all models, VGG16 demonstrated the highest classification consistency, with almost all samples correctly classified along the principal diagonal and only a few misclassifications between adjacent health categories. VGG19, ResNet50, and EfficientNetB0 also achieved excellent performance, exhibiting strong diagonal dominance with only minor confusion, indicating that their deep feature extraction capabilities effectively captured disease-specific texture, colour, and structural characteristics of the leaves. MobileNetV2,

designed as a lightweight architecture using depthwise separable convolutions, produced comparatively more off-diagonal predictions, particularly between the Medium Healthy and Unhealthy classes, suggesting a reduction in discriminative feature representation due to its compact design. Xception exhibited increased confusion across several classes despite employing depthwise separable convolutions, indicating that the model struggled to distinguish subtle inter-class variations in the medicinal leaf dataset. Overall, the progressive reduction in off-diagonal elements from Xception to MobileNetV2, EfficientNetB0, ResNet50, VGG19, and finally VGG16 demonstrates that deeper architectures with higher representational capacity were better able to learn discriminative pathological features, resulting in superior classification accuracy and robustness.

Table 3. Performance accuracy of different models.

Model	Leaf	Health Classification	Precision	Recall	F1-Score	Support
VGG16	**Eranda**	Healthy	0.99	0.99	0.99	199
		Medium Healthy	0.99	1.00	1.00	133
		Unhealthy	1.00	0.99	0.99	94
	Nirgundi	Healthy	1.00	0.99	1.00	175
		Medium Healthy	0.99	1.00	0.99	191
		Unhealthy	1.00	0.99	0.99	203
ResNet50	**Eranda**	Healthy	1.00	0.98	0.99	199
		Medium Healthy	0.98	1.00	0.99	133
		Unhealthy	1.00	1.00	1.00	94
	Nirgundi	Healthy	1.00	0.99	0.99	175
		Medium Healthy	0.98	0.99	0.99	191
		Unhealthy	0.99	0.99	0.99	203
MobileNetV2	**Eranda**	Healthy	0.93	0.93	0.93	199
		Medium Healthy	0.88	0.90	0.89	133
		Unhealthy	0.96	0.83	0.89	94
	Nirgundi	Healthy	0.97	0.89	0.93	175
		Medium Healthy	0.71	0.96	0.82	191
		Unhealthy	0.96	0.74	0.84	203
Xception	**Eranda**	Healthy	0.62	0.95	0.75	199
		Medium Healthy	0.86	0.47	0.61	133
		Unhealthy	0.99	0.76	0.86	94
	Nirgundi	Healthy	0.69	0.82	0.75	175
		Medium Healthy	0.74	0.72	0.73	191
		Unhealthy	0.71	0.53	0.61	203
EfficientNetB0	**Eranda**	Healthy	1.00	0.98	0.99	199
		Medium Healthy	0.97	1.00	0.99	133
		Unhealthy	1.00	0.98	0.99	94
	Nirgundi	Healthy	1.00	1.00	1.00	175
		Medium Healthy	0.99	1.00	1.00	191
		Unhealthy	1.00	1.00	1.00	203
VGG19	**Eranda**	Healthy	1.00	1.00	1.00	199
		Medium Healthy	0.99	1.00	0.99	133
		Unhealthy	1.00	0.98	0.99	94
	Nirgundi	Healthy	1.00	0.99	1.00	175
		Medium Healthy	0.97	0.99	0.98	191
		Unhealthy	0.99	0.98	0.98	203

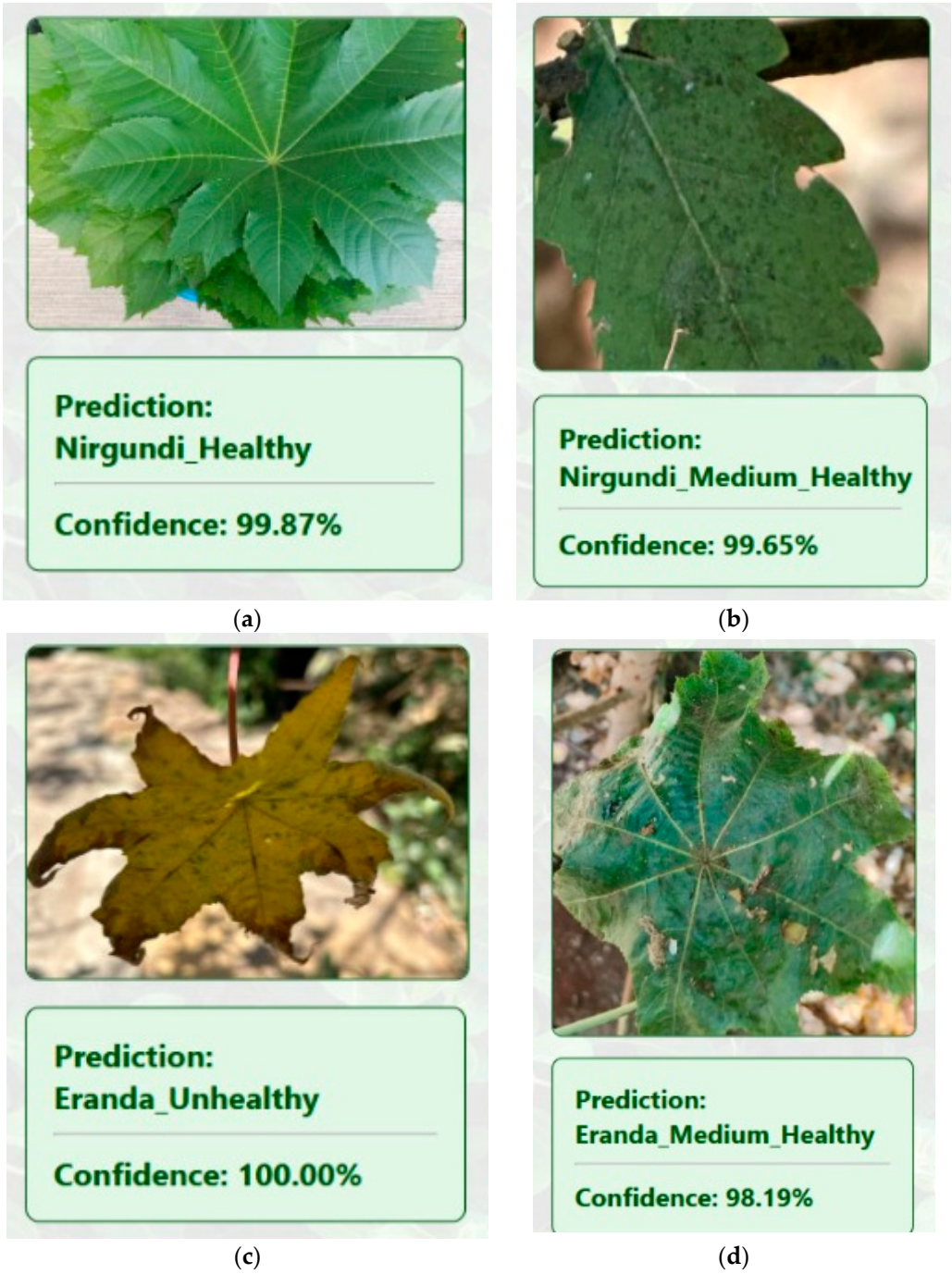

Figure 8. Visual results of leaf health assessment using the proposed web application: (**a**) Healthy Nirgundi leaf, (**b**) Medium Healthy Nirgundi leaf, (**c**) Healthy Eranda leaf, and (**d**) Medium Healthy Eranda leaf.

4.4. Statistical Analysis

To evaluate the robustness of the experimental results, each transfer learning model was trained and evaluated five independent times using different random initialization seeds while keeping the dataset split fixed. The mean classification accuracy and standard deviation were computed across all runs. Statistical significance between the best-performing model and competing architectures was assessed using a paired Student's *t*-test at a significance level of 0.05. This analysis verifies that the observed performance improvements are statistically reliable rather than resulting from random variation.

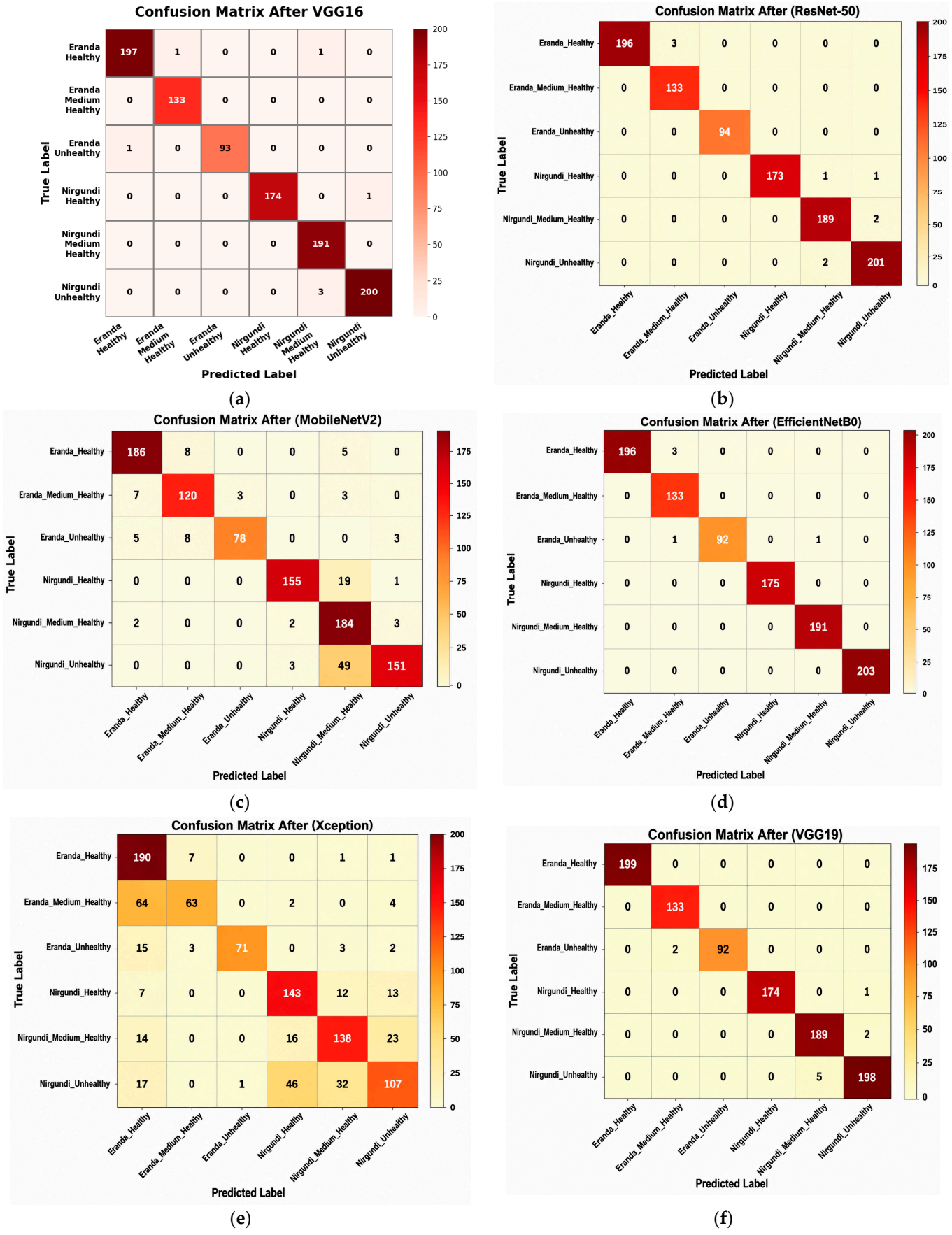

Figure 9. Confusion matrix for predicted v/s actual for (**a**) VGG16, (**b**) ResNet50, (**c**) MobileNetV2, (**d**) EfficientNetB0, (**e**) Xception, and (**f**) VGG19 models.

4.5. Real-Time Deployment of the Application

Proposed work developed AI-based web application for classifying leaf health concerning the following two very important Ayurvedic medicinal plants: Eranda (Ricinus communis) and Nirgundi (Vitex negundo). This tool categorizes the leaf images into three states that include "Healthy," "Medium-healthy," and "Unhealthy" as shown in Figure 10. This application helps Ayurveda medicinal practitioners, and Pharmacognosist to identify right leaves to prepare medicine. This also helps medicinal plant former to identify crop-related problems as early as possible in the decision-making process.

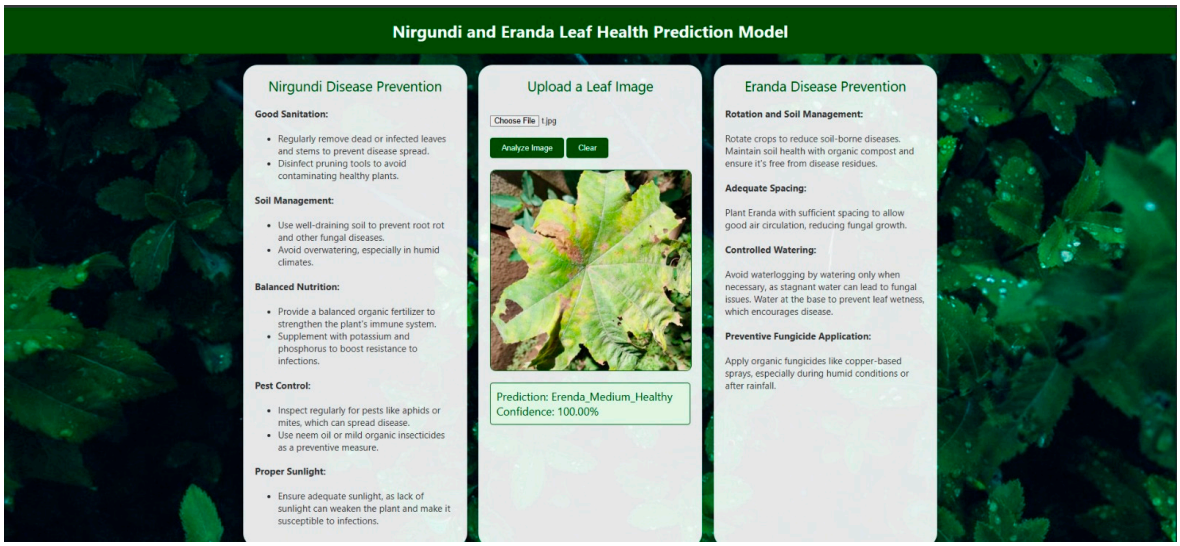

Figure 10. AI-based web application for classifying leaf health.

5. Comparative Analysis

Table 4 shows statistical analysis of the proposed framework in which mean accuracy, standard deviation and p-values are calculated for each of the models considering VGG16 as base reference. Table 5 shows the performance comparison of a few machine learning and deep learning classification models' performances using multiple sources of paper and contrasting these with newly designed models designed for this study.

Table 4. Summary of statistical analysis.

Model	Mean Accuracy (%)	Standard Deviation	*p*-Value (vs. VGG16)
VGG16	99.30	0.23	— (Reference)
VGG19	98.99	0.31	0.042
ResNet50	99.10	0.37	0.031
EfficientNetB0	99.50	0.35	0.028
MobileNetV2	87.84	0.91	<0.001
Xception	71.56	1.15	<0.001

It shows significant metrics, such as accuracy, precision, recall, and F1-score, to provide a basic idea of model performance and reliability. This paper discusses the methodologies involved in the research, such as ResNeXt-50, GoogLeNet, Multi-Layer Perceptron (MLP), Artificial Neural Network (ANN), and Convolution Neural Network (CNN), along with their respective models, such as Random Forest and SVM. These benchmark results, collected from different studies, provide a basis for measuring the performance of the proposed VGG16, ResNet50, MobileNetV2, Xception, EfficientNetB0, and VGG19 models. It can be said that the proposed models performed well, as EfficientNetB0 achieved 99.50%, VGG16 achieved 99.30%, ResNet50 achieved 99.10%, and VGG19 achieved 98.99% mean

accuracy (averaged over five runs, as seen in Table 4), thereby proving their competitive strengths and improvements. This indicates that they can be used effectively for strong leaf disease classification in real-world applications.

Table 5. Comparative analysis for classification accuracy and metrics.

Ref. No.	Methodology	Accuracy	F1	Precision	Recall
[25]	ResNeXt-50 (Augmented)	98.08%	-	-	-
	GoogLeNet (Augmented)	97.04%	-	-	-
	ShuffleNet (Augmented)	96.33%	-	-	-
[26]	MLP (220 × 220)	95.87%	-	0.961	0.959
	LB (220 × 220)	95.04%	-	0.951	0.950
	B (220 × 220)	94.21%	-	0.944	0.942
	SLg (220 × 220)	93.38%	-	0.935	0.934
	RF (220 × 220)	92.56%	-	0.927	0.926
[27]	DenseNet201 and log Gabor filter	97%	0.981	0.982	0.983
[28]	Xception + ANN	97.5%	-	-	-
[29]	DNN	98.68%	-	-	-
[30]	MobileNet	98.33%	-	0.99	0.97
[31]	Ayur-PlantNet (DeepCNN)	92.27%	-	-	-
Proposed	ResNet 50	99.10%	0.99	0.99	0.99
	VGG 16	99.30%	0.99	0.99	0.99
	MobileNet V2	87.84%	0.88	0.90	0.88
	Xception	71.56%	0.71	0.74	0.72
	EfficientNetB0	99.50%	0.99	1.00	0.99
	VGG 19	98.99%	0.99	0.99	0.99

6. Conclusions

This study presented a framework to evaluate leaf health of two Ayurvedic medicinal plants, Eranda (*Ricinus communis*) and Nirgundi (*Vitex negundo*). This study applied six state-of-the-art transfer learning-based models, including VGG16, ResNet50, MobileNetV2, Xception, EfficientNetB0, and VGG19, to classify the leaf into the following three categories: Healthy, Medium Healthy and Unhealthy, using a newly developed dataset comprising 6630 annotated leaf images collected under controlled conditions. Experimental results demonstrated that EfficientNetB0 achieved the highest mean accuracy of 99.50% (averaged over five runs), closely followed by VGG16 (99.30%), ResNet50 (99.10%), and VGG19 (98.99%), confirming the effectiveness of transfer learning for medicinal plant health assessment. Furthermore, a web-based application was developed to demonstrate the practical deployment of the trained models for real-time leaf health classification. However, even though this study established a benchmark for AI-based health assessment of Ayurveda medicinal plants, certain limitations remained untouched, which includes limited dataset collected under controlled environments. This has limited model's generalization in real world environments. In future, the study may focus on diverse medicinal plant leaf dataset collection in varied environments, creating more robust deep learning models and external validations.

Author Contributions: Conceptualization, B.A. (Bharati Ainapure), I.I.H., M.R.E., M.P.B. and G.M.P.; methodology, B.A. (Bharati Ainapure), I.I.H., M.R.E., M.P.B. and G.M.P.; software, B.A. (Bharati Ainapure), I.I.H., M.R.E., M.P.B. and G.M.P.; validation, B.A. (Bhargav Appasani) and N.B.; formal analysis, B.A. (Bharati Ainapure), I.I.H., M.R.E., M.P.B. and G.M.P.; investigation, B.A. (Bhargav Appasani) and N.B.; resources, B.A. (Bharati Ainapure); data curation, B.A. (Bhargav Appasani); writing—original draft preparation, B.A. (Bharati Ainapure), I.I.H., M.R.E., M.P.B. and G.M.P.; writing—review and editing, B.A. (Bhargav Appasani), A.G.M. and N.B.; visualization, B.A. (Bharati Ainapure); supervision, N.B.; project administration, B.A. (Bhargav Appasani) and A.G.M.; funding acquisition, N.B. and A.G.M. All authors have read and agreed to the published version of the manuscript.

Funding: The research was fully supported by the PubArt programme of the National University of Science and Technology POLITEHNICA Bucharest.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study were generated by the authors and made openly available in IEEE Dataport at https://doi.org/10.21227/rjxb-3p27.

Acknowledgments: We would like to thank Vivek Choudhary, Sumatibhai Shah Ayurved Mahavidyalaya, Pune, Krishna Kokate and Ankita Ainapure from Shashwat Multispecialty Ayurveda Hospital, Pune, India, for their constant support during leaf annotation and further guidance in achieving this research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Bhelkar, N.S.; Bolavarapu, S. Machine Learning Techniques for Medicinal Plants Identification Using Leaf Features: Review. *Front. Health Inform.* **2024**, *13*, 520–526.
2. Mitra, D.; Gupta, S. Comparison of Proposed Plant Leaf Diseases Detection Algorithm with Existing State-Of-the-Art Techniques. In Proceedings of the 2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 3–4 November 2023; pp. 472–477. [CrossRef]
3. El-Saadony, M.T.; Saad, A.M.; Mohammed, D.M.; Korma, S.A.; Alshahrani, M.Y.; Ahmed, A.E.; Ibrahim, E.H.; Salem, H.M.; Alkafaas, S.S.; Saif, A.M.; et al. Medicinal plants: Bioactive compounds, biological activities, combating multidrug-resistant microorganisms, and human health benefits—A comprehensive review. *Front. Immunol.* **2025**, *16*, 1491777. [CrossRef] [PubMed] [PubMed Central]
4. Sahu, R.; Gupta, P.K.; Mishra, A.; Kumar, A. Ayurveda and in silico Approach: A Challenging Proficient Confluence for Better Development of Effective Traditional Medicine Spotlighting Network Pharmacology. *Chin. J. Integr. Med.* **2023**, *29*, 470–480. [CrossRef] [PubMed]
5. Mohanty, S.P.; Hughes, D.P.; Salathé, M. Using Deep Learning for Image-Based Plant Disease Detection. *Front. Plant Sci.* **2016**, *7*, 1419. [CrossRef] [PubMed]
6. Bouguettaya, A.; Zarzour, H.; Kechida, A.; Taberkit, A.M. Deep learning techniques to classify agricultural crops through UAV imagery: A review. *Neural Comput. Applic.* **2022**, *34*, 9511–9536. [CrossRef] [PubMed]
7. Jung, M.; Song, J.; Shin, A.Y.; Choi, B.; Go, S.; Kwon, S.-Y.; Park, J.; Park, S.G.; Kim, Y.-M. Construction of deep learning-based disease detection model in plants. *Sci. Rep.* **2023**, *13*, 7331. [CrossRef] [PubMed]
8. "*Vitex negundo* L.," India Biodiversity Portal. Available online: https://indiabiodiversity.org/species/show/32833 (accessed on 30 January 2025).
9. Orwa, C.; Mutua, A.; Kindt, R.; Jamnadass, R.; Anthony, S. Agroforestree Database: A Tree Reference and Selection Guide. Version 4.0. 2009. Available online: https://apps.worldagroforestry.org/treedb/AFTPDFS/Vitex_negundo.PDF (accessed on 30 January 2025).
10. Alam, M.I.; Gomes, A. Snake venom neutralization by Indian medicinal plants (*Vitex negundo* and *Emblica officinalis*) root extracts. *J. Ethnopharmacol.* **2003**, *86*, 75–80. [CrossRef] [PubMed]
11. Garg, G.; Bharadwaj, A.; Chaudhary, S.; Gupta, V. Chemical profiling of bioactive compounds in the methanolic extract of wild leaf and callus of *Vitex negundo* using gas chromatography-mass spectrometry. *World J. Exp. Med.* **2024**, *14*, 88064. [CrossRef] [PubMed]
12. Zheng, C.-J.; Li, H.-Q.; Ren, S.-C.; Xu, C.-L.; Rahman, K.; Qin, L.-P.; Sun, Y.-H. Phytochemical and Pharmacological Profile of *Vitex negundo*. *Phyther. Res.* **2015**, *29*, 633–647. [CrossRef] [PubMed]

13. Kj, S.; Nk, M.; Pg, A. Ayurveda, yoga, and acupuncture therapies in alleviating the symptom score among patients with spinal cord injury—A systematic review. *J. Ayurveda Integr. Med.* **2023**, *14*, 100749. [CrossRef] [PubMed]
14. Mridula, M.K. An Experimental Study on the Anti-inflammatory Effect of Roots with that of Leaves of Nirgundi (*Vitex negundo* Linn, Family Verbenaceae). *Int. J. Green Pharm.* **2016**, *5*, 199–205.
15. Uniyal, S.; Singh, K.; Jamwal, P.; Lal, B. Traditional use of medicinal plants among the tribal communities of Chhota Bhangal, Western Himalaya. *J. Ethnobiol. Ethnomed.* **2006**, *2*, 14–21. [CrossRef] [PubMed]
16. Sharma, D.; Radha; Kumar, M.; Andrade-Cetto, A.; Puri, S.; Kumar, A.; Thakur, M.; Chandran, D.; Pundir, A.; Prakash, S.; et al. Chemical Diversity and Medicinal Potential of *Vitex negundo* L.: From Traditional Knowledge to Modern Clinical Trials. *Chem. Biodivers.* **2023**, *20*, e202301086. [CrossRef] [PubMed]
17. Sharma, K.; Yadav, M.; Attri, K. Pharmacognostic Studies on Methanolic Extract of Leaves of *Vitex negundo* Linn. *Pharmacogn. J.* **2020**, *12*, 551–556. [CrossRef]
18. Abdul, W.M.; Hajrah, N.H.; Sabir, J.S.M.; Al-Garni, S.M.; Sabir, M.J.; Kabli, S.A.; Saini, K.S.; Bora, R.S. Therapeutic role of *Ricinus communis* L. and its bioactive compounds in disease prevention and treatment. *Asian Pac. J. Trop. Med.* **2018**, *11*, 177. [CrossRef]
19. Abomughaid, M.M.; Teibo, J.O.; Akinfe, O.A.; Adewolu, A.M.; Teibo, T.K.A.; Afifi, M.; Al-Farga, A.M.H.; Al-kuraishy, H.M.; Al-Gareeb, A.I.; Alexiou, A.; et al. A phytochemical and pharmacological review of *Ricinus communis* L. *Discov. Appl. Sci.* **2024**, *6*, 315. [CrossRef]
20. Srivastava, P.; Jyotshna, N.; Gupta, N.; Maurya, A.K.; Shanker, K. New anti-inflammatory triterpene from the root of *Ricinus communis*. *Nat. Product. Res.* **2013**, *28*, 306–311. [CrossRef] [PubMed]
21. Gupta, R.; Chaudhary, A.K.; Sharma, R. Analgesic and Anti-inflammatory Potential of *Ricinus communis* Linn.: Evidence from Pharmacology to Clinical Studies. *Curr. Pharmacol. Rep.* **2023**, *10*, 27–67. [CrossRef]
22. Ferraz, A.C.; Angelucci, M.E.M.; Costa, M.L.D.; Batista, I.R.; de Oliveira, B.H.; Cunha, C.D.A. Pharmacological evaluation of ricinine, a central nervous system stimulant isolated from *Ricinus communis*. *Plant Physiol.* **1999**, *63*, 367–375. [CrossRef] [PubMed]
23. Wachira, S.W.; Otieno, G.O.; Nyunja, J.C.; Kamanza, J.I. Toxicity of six plant extracts and two pyridone alkaloids from *Ricinus communis* against the malaria vector *Anopheles gambiae*. *Parasites Vectors* **2014**, *7*, 312. [CrossRef] [PubMed]
24. Okwuasaba, F.; Chandra Das, S.S.; Isichei, C.; Ekwenchi, M.; Onoruvwe, O.; Olayinka, O.; Uguru, V.; Dafur, S.; Ekwere, E.O.; Parry, O. Pharmacological studies on the antifertility effects of RICOM-1013-J from *Ricinus communis* var minor and preliminary clinical studies on women volunteers. *Phyther. Res.* **1997**, *11*, 547–551. [CrossRef]
25. Kumar, G.; Kumar, V. Herbal Plants Leaf Image Classification Using Deep Learning Models Based on Augmentation Approach. In *Proceedings of the 4th International Conference on Communication & Information Processing (ICCIP), Pune, India, 31 July 2022*; Elsevier: Amsterdam, The Netherlands, 2022. [CrossRef]
26. Naeem, S.; Ali, A.; Chesneau, C.; Tahir, M.H.; Jamal, F.; Sherwani, R.A.K.; Hassan, M.U. The Classification of Medicinal Plant Leaves Based on Multispectral and Texture Feature Using Machine Learning Approach. *Agronomy* **2021**, *11*, 263. [CrossRef]
27. Oppong, S.O.; Twum, F.; Hayfron-Acquah, J.B.; Missah, Y.M. A novel computer vision model for medicinal plant identification using Log-Gabor filters and deep learning algorithms. *Comput. Intell. Neurosci.* **2022**, *2022*, 1189509. [CrossRef] [PubMed]
28. Roopashree, S.; Anitha, J. DeepHerb: A Vision Based System for Medicinal Plants Using Xception Features. *IEEE Access* **2021**, *9*, 135927–135941. [CrossRef]
29. Tiwari, V.; Joshi, R.C.; Dutta, M.K. Deep neural network for multi-class classification of medicinal plant leaves. *Expert Syst.* **2022**, *39*, e13041. [CrossRef]
30. Kavitha, S.; Kumar, T.S.; Naresh, E.; Kalmani, V.H.; Bamane, K.D.; Pareek, P.K. Medicinal Plant Identification in Real-Time Using Deep Learning Model. *SN Comput. Sci.* **2024**, *5*, 73. [CrossRef]
31. Pushpa, B.R.; Rani, N.S. Ayur-PlantNet: An Unbiased Light Weight Deep Convolutional Neural Network for Indian Ayurvedic Plant Species Classification. *J. Appl. Res. Med. Aromat. Plants* **2023**, *34*, 100459. [CrossRef]
32. Mulugeta, A.K.; Sharma, D.P.; Mesfin, A.H. Deep learning for medicinal plant species classification and recognition: A systematic review. *Front. Plant Sci.* **2024**, *14*, 1286088. [CrossRef] [PubMed]
33. Fuentes, A.; Asgher, S.A.; Dong, J.; Jeong, Y.; Lee, M.H.; Kim, T.; Yoon, S.; Park, D.S. Comprehensive plant health monitoring: Expert-level assessment with spatio-temporal image data. *Front. Plant Sci.* **2025**, *16*, 1511651. [CrossRef] [PubMed] [PubMed Central]
34. Shrotriya, A.; Sharma, A.K.; Prabhu, S.; Bairwa, A.K. An Approach Toward Classifying Plant-Leaf Diseases and Comparisons With the Conventional Classification. *IEEE Access* **2024**, *12*, 117379–117398. [CrossRef]
35. Khan, A.; Nawaz, U.; Ulhaq, A.; Robinson, R.W. Real-time plant health assessment via implementing cloud-based scalable transfer learning on AWS DeepLens. *PLoS ONE* **2020**, *15*, e0243243. [CrossRef] [PubMed] [PubMed Central]
36. Boa, E. *An Illustrated Guide to the State of Health of Trees Recognition and Interpretation of Symptoms and Damage*; Food and Agriculture Organization of the United Nations: Rome, Italy, 2003.

37. Islam, M.; Azad, A.; Arman, S.E.; Alyami, S.A.; Hasan, M.M. PlantCareNet: An advanced system to recognize plant diseases with dual-mode recommendations for prevention. *Plant Methods* **2025**, *21*, 52. [CrossRef] [PubMed] [PubMed Central]
38. Asani, E.O.; Osadeyi, Y.P.; Adegun, A.A.; Viriri, S.; Ayoola, J.A.; Kolawole, E.A. MPD-APP: A mobile-enabled plant diseases diagnosis application using convolutional neural network toward the attainment of a food secure world. *Front. Artif. Intell.* **2023**, *6*, 1227950. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.