MethodsX 14 (2025) 103126

Contents lists available at ScienceDirect

MethodsX

journal homepage: www.elsevier.com/locate/methodsx

HybNet: A hybrid deep models for medicinal plant species identification ☆,☆☆

B.R. Pushpa*, S. Jyothsna, S. Lasya

Department of Computer Science, School of Computing, Amrita Vishwa Vidyapeetham, Mysuru, India

ARTICLE INFO

Method name:
HybNet

Keywords:
Medicinal Plant species
Convolutional neural network
Hybrid deep architectures
Recalibration
Classification
Transfer learning
Squeeze and excitation

ABSTRACT

Real-time plant species detection plays an important role in fields ranging from medicine to biodiversity conservation. Images captured under unconstrained environments, scale variations, different lighting conditions, leaf orientation, complicated backdrops, and leaflet structure make plant species recognition rigorous and time-consuming. Our study addresses this challenge by introducing three pioneering hybrid models, seamlessly integrating the strengths of convolution neural networks. In the first model, two deep learning models such as VGG16 and MobileNet are fused to extract features. Then, the extracted features are subjected to KNN classifier achieving an impressive 85.85 % accuracy, while the second model adopts MobileNet in conjunction with ResNet50 for feature extraction which is further classified using a deep learning classifier to achieve 88 % accuracy. The third model incorporates MobileNetV2 with the Squeeze and Excitation (SE) layers for the classification tasks. Our research highlights the immense potential of modern image processing techniques and deep learning models in comprehending and safeguarding the earth's diverse plant species. The experiments are carried out on self-created medicinal plant datasets captured in real-time conditions. From the experimentations, it is observed that hybrid model 3 reflects an improved performance of 94.24 % by utilizing recalibration efforts compared with the other two hybrid models.

- One of the significant contributions of the study lies in a focused emphasis on feature enhancement achieved through the utilization of hybrid models majorly to enrich the features.
- The feature scaling model incorporated in hybrid model 3 exhibits a superior and better performance demonstrating higher accuracy compared to the other models presented in this work.
- The deebp learning models are trained and tested on the small dataset yet achieved good accuracy.

Specifications table

Subject area:	Computer Science
More specific subject area:	*Computer Vision, Deep Learning*
Name of your method:	*HybNet*
Name and reference of original method:	*None*
Resource availability:	*NA.*

☆ **Related research article:** None
☆☆ **For a published article:** None
* Corresponding author.
E-mail address: pushpa@my.amrita.edu (B.R. Pushpa).

https://doi.org/10.1016/j.mex.2024.103126
Received 21 November 2024; Accepted 20 December 2024
Available online 21 December 2024

Background

Identification and classification of medicinal plant species play a significant role in both fields such as agronomy and traditional medicine which has a plant diversity of more than 48,655 plant species [1]. The advancement of pharmaceuticals, ecological equilibrium, and sustainable agriculture depends on a proper understanding and identification of plant species, which are the basis of human health that provide food, energy, and oxygen. The identification of plant species is a difficult and time-consuming task that requires a great deal of expertise in the field. However, it is essential for studying biodiversity and comprehending species that are unknown and have not yet been discovered. The shape, color, texture, veins, and chemical composition of leaves, flowers, fruits, and other organs are considered as essential elements for identifying plants in botany, allowing botanists to study and distinguish between one species to another species. Sometimes experienced professionals find it difficult to recognize plant species due to the vast inter-class similarities and intra-class variations that exist in the universe. Hence, in this era of technological developments, automated plant species identification using conventional machine learning and vision algorithms has become more crucial resources for tackling this complexity problem. These procedures present a possible remedy because they largely rely on leaf characteristics like shape, texture, size, venation, and color but their performance varies due to the wide diversity of plant species and also issues such as variation in viewpoint, different backgrounds, varying weather conditions, seasonal changes, variations at different growth stages and other image acquisition challenges.

Recent studies highlight the advancement of both machine learning and deep learning models for plant classification through leaf images. Computer vision techniques are extensively utilized to differentiate diseased and healthy plants [2]. Several works on the identification and classification of herbs and medicinal plants [3–5] are accomplished, classification of generic plants and specific plant species identification [6–8] and agricultural crop plant diseases prediction [9]. Image processing is an interdisciplinary area that consists of all components of digital image manipulation, evaluation, and interpretation. It is important for extracting visual information from the images. One of the important applications of image processing in plant identification is the extraction of key features such as leaf form, texture, color, and vein patterns. These features may be used as inputs for machine learning models and deep learning strategies to produce strong and dependable plant recognition models [10].

According to reports, the most often utilized techniques for automating plant categorization tasks incorporate various convolutional neural network models. Deep learning, is a subset of artificial intelligence that offers extraordinary accuracy and efficiency in this crucial research. Also, effective in automatically learning complicated and hierarchical properties from raw images compared to conventional approaches. One of deep learning's primary benefits is its ability to handle massive amounts of visual data, which enables the identification of minute changes across plant species. Deep learning algorithms can examine images of all sizes and automatically identify distinguishing traits like leaf shapes, textures, and patterns that are typically difficult for humans to notice. Unlike, traditional machine learning methods, which rely on manually constructed features, deep learning models get relevant data directly from the images. This reduces the need for manual feature engineering and boosts the flexibility and data-driven nature of the process.

Method details

In the current study, hybrid deep learning models combine the strengths of transfer learning architectures to achieve increased accuracy and flexibility. In the proposed models, features are extracted using convolutional neural networks (CNNs), which are then merged with additional deep learning architectures for more in-depth feature analysis. Through this integration, the model can incorporate complex visual signs such as leaf patterns and textures as well as temporal or sequential information. Further, hybrid deep learning models resolve data scarcity issues. Transfer learning techniques may be used to adapt pre-trained models for specific plant identification tasks with reduced datasets and enhanced model performance. Three hybrid models are proposed using a combination of deep learning models. In the first model, VGG16 and MobileNet are connected using the sequence of early fusion strategy to extract features and are classified using the KNN machine learning classifier. The second model adopts MobileNet in conjunction with ResNet50 for feature extraction which is further classified using a deep learning classifier. The third model employs the MobilenetV2 model which employs a Squeeze and Excitation (SE) block for Classification tasks.

The major contributions of the proposed work are as follows:

- Real-time datasets were systematically acquired through the usage of smartphones, ensuring the capture in authentic and dynamic environmental conditions that focused on medicinal plant species. Effective management of unbalanced samples present in the dataset by utilizing the augmentation.
- A self-collected dataset includes challenges such as images captured with varying viewpoints, varying scales, complex backgrounds, different illumination, and varying weather conditions.
- One of the significant contributions of the study lies in a focused emphasis on feature enhancement achieved through the utilization of hybrid models majorly to enrich the features resulting in expanding the versatility and innovation of the contribution.
- One of the noteworthy contributions involves the comparison of the feature scaling model (hybrid model 3) with the hybrid models employed. Both approaches focus on feature enhancement itself. Yet our investigation reveals that rescaling i.e. hybrid model 3 exhibits a superior and better performance demonstrating higher accuracy compared to the other models presented in this work.

Numerous works have been conducted on medicinal plant species recognition using machine learning and deep learning models. A review of the significant works has been explored in this section.

Waldchen et al. [11], developed digital technological advancements with the potential to enable automatic species identification, especially using deep learning CNN Network. The incorporation of machine learning methodologies is deemed crucial for pragmatic implementations in the preservation of biodiversity. Sharma et al. [12], proposed simple ANN which outperforms SVM in plant leaf classification, achieving 93.33 % accuracy by combining color, texture, and shape features. Feature extraction techniques like GLCM and HSV enhance this performance, making ANN effective for identifying medicinal plants. Pushpanathan et al. [13], introduced image processing methods for leaf detection and feature extraction, with a focus on machine learning algorithms applied to leaf images. The productivity of classifiers in identifying leaf images which are based on features such as texture, vein structure, and shape is estimated. The experiments are conducted on publicly accessible leaf databases. Islam et al. [14], employed convolutional neural networks (CNNs) and machine learning approaches to recognize indigenous flowers in Bangladesh. ReLU activation, SoftMax function, and Adam optimizer were used by the suggested CNN model for the classification. Experiments on a dataset of 1280 test images and 5120 training images of eight different flower categories resulted in 85 % classification accuracy. The study provided a useful resource for identifying local flowers in urban areas to botanists and other people.

Raghukumar et al. [15], used shape, texture, and color aspects as the main emphasis of feature extraction from the leaf images dataset. Machine learning methods like KNN and SVM were used for classification. The work assists in improving the effectiveness of Ayurvedic medicine by expediting the identification procedure. Dileep et al., [16] proposed a CNN model for automated medicinal plant classification. Several features based on leaf size, shape, color, and texture were utilized to classify with an accuracy of 96.76 % on a standard dataset that includes 40 typical medicinal plants found in Kerala, India. Wagle et al. [17], in this work focused on the detection and classification of nine plant species from the PlantVillage dataset using compact convolutional neural networks (N1, N2, N3 models) and AlexNet with transfer learning. Data augmentation significantly improves accuracy, with the models achieving over 99.45 % accuracy. The proposed N1, N2, and N3 models are more compact than AlexNet, requiring up to 34.58 % less training time while maintaining high accuracy in classifying plant species and diseases in tomato leaves. Chanyal et al. [18], addressed the growing use of less toxic and more affordable medical plants by pharmaceutical corporations which has led to recent advances in automatic medicinal plant classification. They emphasized the significance of precise identification of plant diseases by focusing on deep learning and machine learning methods for classifying plants using leaf image samples. Thanikkal et al. [19], Deep learning and machine learning algorithms are employed in early plant disease diagnosis. The work suggested future research by highlighting drawbacks and opportunities for advancement in various machine learning classifiers. Tiwari et al. [20], developed a deep-learning model for detecting plant diseases. A self-created dataset comprises leaf images of 22 plant varieties and 12 distinct crops, with a variation in intra-class and similarity among the inter-class plant species. The suggested method attained an accuracy of 97.69 % by testing with several deep neural networks and optimization strategies.

Anubha et al. [21], conducted a comparative analysis between deep learning techniques and traditional methods. Traditional approaches utilized handcrafted features and classifiers, while deep learning employed convolutional neural network models such as VGG16 and VGG19. The results indicate that VGG CNN models outperform other methods, achieving accuracy levels of up to 99.41 % across datasets. Lee et al. [22], in this paper proposes a Plant-CNN-ViT ensemble model that combines Vision Transformer, ResNet-50, DenseNet-201, and Xception, leveraging each model's unique strengths for efficient feature extraction and classification. Evaluated on four datasets, the model achieved nearly perfect accuracy, with 100 % on three datasets and 99.83 % on the fourth, demonstrating its effectiveness in plant leaf classification. Wu et al., [23] Worked on a transfer learning method for the identification of rare and endangered plant leaves using pre-trained models. Khanna et al. [24], this study introduces PlaNet, a deep learning model for efficient identification and classification of plant diseases from leaf images. Utilizing Deep Convolutional Neural Networks (DCNN) with minimal pre-processing, PlaNet achieved 97.95 % accuracy, 0.9752 AUC, and 0.9686 F1-score in tests against 18 CNN models across various benchmark datasets. The results demonstrate PlaNet's effectiveness and real-time capabilities in differentiating healthy and unhealthy leaves, making it a flexible and scalable solution for precision agriculture. Vaidehi et al. [25], introduced the ResNet-based Convolutional Neural Networks (CNNs) for the classification of Ayurvedic plants.

Roy et al. [26], focused on employing deep neural networks (DNNs) to improve image categorization in the presence of noise. The authors proposed hybrid models by fusing convolutional neural networks (CNNs) with denoising autoencoder (DAE) approaches. The outcomes of the experiments show that these hybrid models are capable of accurately classifying images with high noise levels. Thamilselvan et al. [27], explored various hybrid classification algorithms considering their ability to deal with the problems of classifying images. The performance of conventional algorithms was investigated on various datasets to evaluate the classification accuracy. Bedi et al. [28], introduced a hybrid model that combines the architectures of convolutional neural networks (CNN) and convolutional autoencoders (CAE) to detect bacterial Spot disease in peach plants. The model obtained an outstanding accuracy of 99.35 % with the limited parameters on the Plant Village dataset. Kaur et al., [29] worked on grape plant disease diagnosis using logistic regression and transfer learning using EfficientNetB7 deep architecture to achieve a high accuracy of 98.7 %. The results highlighted the importance of effective disease detection techniques in agricultural practices. Wenjie et al. [30], presented a technique for compressing convolutional neural networks (CNNs) with limited resources. Distilled-MobileNet, distilling information from an advanced model (VGG16) to a lightweight model (MobileNet), achieved high accuracy (97.54 %) in the classification of various crop diseases. Banzi et al. [31], introduced convolutional neural network (CNN) models trained on leaf images that identified plant species with an astonishing 95.06 % success rate. This showed how CNN-based methods can be a useful tool in conservation measures aimed at protecting biodiversity. Zhao et al. [32], presented a new spatula-based plant detection system for both simple and compound leaves. They proposed a unique point landing on the global and native shape data separately, which improved the sensitivity. The experiments provide efficient and accurate recognition as the model emphasized pattern computing rather than point-to-point matching. Chaudhury et al. [33], proposed a method for identifying plant species from obscured leaf photos that outperformed state of art methods on three leaf datasets. The procedure included representing the b-Spline curve, extracting the interest points using the DCE technique, and evaluating the match quality using energy functional minimization.

Zhang et al. [34], introduced a Tranvolution detection network with GAN modules. The extraction module incorporates GAN modules and a generative model, showcasing its effective implementation on the Hybrid E203 for real-world agricultural usage. Hossain et al. [35], developed a KNN classifier-based approach for the detection and categorization of plant leaf diseases. The method effectively detects diseases in different plant species with an accuracy of 96.76 % by extracting textural information. Malik et al., [36] proposed the automatic identification of medicinal plants in Borneothat can be utilized for real-time identification. The system integrates geo-mapping features and crowdsourcing feedback, and it demonstrates significant performance over baseline models. Sonali et al. [37], established an automated deep CNN model for Aloe vera plant disease diagnosis to discriminate between healthy and unhealthy leaves, that can be used in precision agriculture to minimize crop losses and detect diseases early. Sun et al. [38], collected 10,000 images of 100 decorative plant species taken with mobile phones in their natural habitat. A recognition rate of 91.78 % was achieved on the BJFU100 dataset, using a deep learning model for plant classification that is useful for smart forestry applications. Saleem et al. [39], emphasized early plant disease diagnosis through the application of deep learning (DL) techniques and visualization approaches. They focused on the classification and exact recognition of plant disease symptoms to improve transparency in disease identification, particularly in the pre-symptomatic stages. Mohanty et al. [40], demonstrated deep learning and smartphone technology to accurately identify diseased plant species. Impressive results were obtained by training a CNN on a very large dataset, suggesting that publicly available datasets may be used for widespread disease identification.

Data acquisition

In this work, we have utilised the image samples of self-created dataset that are now publicly available in Mendeley repository that are made public access without charge. These datasets consists of eighty medicinal plant species captured using mobile phones with various resolutions. The plant leaf samples are collected from botanical gardens, medicinal plant nursery and Chandravana Garden, an Ayurvedic plant farm, in Mysore. These images have been taken randomly without setting any constraints. These images were taken in different lightings, weather conditions, low-lighting images, shadows and complex backgrounds. Each species type includes more than sixty image samples achieved through augmentation. During the data acquisition for this study, distinct mobile devices were employed to capture the real-time images with resolutions ranging from 4MP to 16MP and are stored in. jgp format. For the proposed study we have considered thirteen plant varieties from a self-created dataset as shown in Fig. 1.

Proposed method

In this section, three hybrid models are proposed using pre-trained CNN architectures, namely VGG16, MobileNet, MobileNetV2, and ResNet50 are discussed. The architecture selection is based on several key criteria such as lightweight, compatibility to merge the features extracted, and improved recognition rates and enriched features. MobileNetV2 is widely used in many real applications such as object detection, classifications, and localization tasks. MobileNet is a combination of depth-wise separable and also point-wise convolution block which makes it a lightweight deep neural network. VGG16, a CNN employed for object detection and classification purposes. Next, ResNet50, a deep neural network architecture that has gained more importance in the field of computer vision and

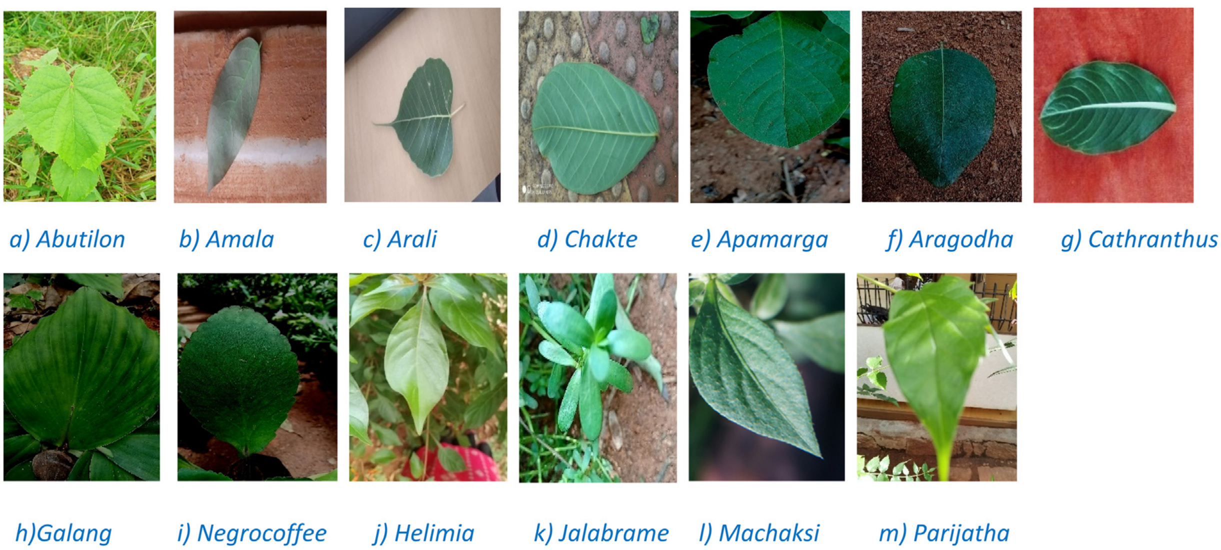

Fig. 1. a) Abutilon b) Phyllanthus emblica c) Nerium oleander d) Caesalpinia platy-loba e) Achyranthes aspera f) Cassia fistula g) Catharanthus-roseus h) Alpinia galanga i) Cassia occidentalis) Hameliapatens 11) Mimosa pudica 12) Alternanthera bettzickiana 13)Nyctanthes arbor-tristis.

Table 1
Hybrid models specification.

Models	Feature description	Execution time
Hybrid Model 1 (VGG16+MobileNet)	1536 features per input instance, concatenated feature maps have a spatial dimension of 7 × 7 resulting in 76,608 features per input instance.	9 min, 35 s
Hybrid Model 2 (ResNet50+ MobileNet)	3072 features per input instance Concatenated feature maps have a spatial dimension of 7 × 7 resulting in 1,50,528 features per input instance.	35 min
Hybrid Model 3 (MobileNetV2 + SE block)	1280 features per input instance Concatenated feature maps have a spatial dimension of 7 × 7 resulting in 62,720 features per input instance.	30 min

image recognition tasks and extends to object detection, segmentation, and facial recognition, solidifying its widespread adoption and influence in the realm of deep learning. The three proposed hybrid models incorporate a fusion of deep learning models by hyperparameter tuning to extract deep features that are subjected to machine learning models and neural network classifiers. The experiments are carried out to compare the performance of proposed models with the state of art methods on the self-created datasets. The three proposed hybrid models with detailed feature descriptions and the model execution time are presented in Table 1.

Preprocessing

Preprocessing is a technique for removing undesirable noise and improving the image by employing image processing methodologies. Addressing noise in real-time images is one of the most common challenges in various fields, the presence of noise can manifest a random variation in values, artifacts, or unwanted interference, making it crucial to understand. One of the most effective approaches to eliminate these noises is image blurring by applying blurring filters.

The two preprocessing techniques used in this study are Gaussian blurring and median blur to reduce the noise and remove the high-frequency values, creating a smoother and more simplified version of the images. Gaussian blurring is one of the widely used blurring techniques that involve the image with the Gaussian Kernal. This kernel assigns more weight to the central pixels and less weight to the pixels, which is far away from it to create a more smoother image. Gaussian blur is more effective at reducing noise and preserving the edge information. The two-dimensional Gaussian function $G(x, y)$is represented in the Eq. (1).

$$\boldsymbol{G}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{2\boldsymbol{\pi}\boldsymbol{\sigma}^2}\boldsymbol{e}^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

G(x,y) Gaussian function at the point(x,y), x & y are the independent variables that represent the coordinates of a point in the two-dimensional space, σ is the standard deviation of the Gaussian distribution controlling the spread or width of the distribution, $\boldsymbol{\pi}$ is the constant equal to 3.14159, $\boldsymbol{e}^{-\frac{x^2+y^2}{2\sigma^2}}$ is the exponent and represents the radial distance from the center. The value obtained from this function will create a convolutional matrix that it will be applied to every pixel of the original image. If the image is larger, it is said to have more computations, and we have to perform this on every pixel.

Median blur replaces each pixel with the median value of the pixel values within its kernel that is robust to outliers and excels in reducing noise as shown in Eq. (2).

$$\boldsymbol{B}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{\boldsymbol{N}}\sum_{\boldsymbol{i}=-\boldsymbol{k}}^{\boldsymbol{k}}\sum_{\boldsymbol{j}=-\boldsymbol{k}}^{\boldsymbol{k}}\boldsymbol{I}(\boldsymbol{x}+\boldsymbol{i}, \boldsymbol{y}+\boldsymbol{j}) \tag{2}$$

B (x, y) represents the output of the averaging operation at the coordinates (x, y), and N is the normalization factor equal to the total number of elements in the neighborhood. It ensures that the averaging operation is normalized and doesn't depend on the size of the neighborhood, i & j are indices used for the summation, ranging from are -k to k, $\boldsymbol{I}(\boldsymbol{x}+\boldsymbol{i}, \boldsymbol{y}+\boldsymbol{j})$ denotes the intensity value of the input image at the coordinates, k is a parameter that defines the size of the neighborhood. The neighborhood is a square of size(2k+1) ×(2k+1) centered around the points.

After completing the process of blurring the next crucial step in our data preprocessing is data augmentation to replicate original images with some variations using data augmentation techniques including rotation, scaling, flipping, and noise addition. This process prevents overfitting and ensures the robustness, reliability, and flexibility of our classification model for real-world applications. A further benefit of augmentation is that it broadens the dataset and reduces the risk of data scarcity. The various preprocessing operations performed in the proposed work are shown in Fig. 2.

The transformations such as Scaling, translation, and rotation of an image can be simply defined in such a way that the parallel lines in an image will remain parallel even after the transformation as shown in the Eq. (3).

$$\mathbb{R} = \begin{pmatrix} \cos\boldsymbol{\theta} & -\sin\boldsymbol{\theta} \\ \sin\boldsymbol{\theta} & \cos\boldsymbol{\theta} \end{pmatrix} \tag{3}$$

This matrix represents a 2D rotation transformation in a Euclidean space where the element in the first row and first column $\boldsymbol{cos\theta}$ represents the cosine angle of the rotation. The elements in the 1st row and the 2nd column -$\sin\boldsymbol{\theta}$ represents the negative sign of the angle rotation. The element in the second row, first column, $\sin\boldsymbol{\theta}$, represents the sine of the angle of rotation. The element in the

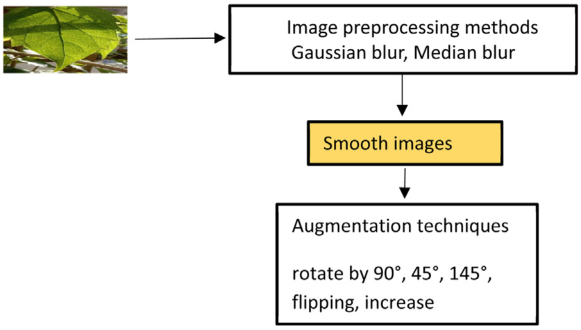

Fig. 2. Pre-processing methods.

2nd row and 2nd column $\cos\theta$ represents the cosine of the angle of rotation. Furthermore, in two dimensions, the standard rotation matrix rotates the column of vector values as given in Eq. (4).

$$\begin{bmatrix} x^1 \\ y^1 \end{bmatrix} = \begin{bmatrix} \cos\theta - \sin\theta \\ \sin\theta \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \tag{4}$$

$$\begin{aligned} x^1 &= x\cos\theta - y\sin\theta \\ y^1 &= x\sin\theta + y\cos\theta \end{aligned} \tag{5}$$

x^1, y^1 are the new coordinates obtained by applying rotation to the points x, y using the transformation equations represented in Eq. (5) where Θ represents the angle of rotation over the original image. Accordingly, the image dataset which was collected in different orientations is subjected to a series of rotations so that it would explore more different prospectives. First, the original image is rotated 90 °, which turns the image on its side, stimulating a right-angle turn, next the rotation follows 45 ° to introduce a change in perspective. Finally, the rotation was minus 45 °, which is essentially a 45-degree counterclockwise rotation, providing a perspective shift in an opposite direction. Then the horizontal flipping technique was where each pixel that is positioned along the horizontal axis will be reversed.

The other augmentation technique involves the brightness of an image by adding a constant value of the pixel intensity to each color channel, that is red, green, and blue (RGB channels). This will just make the original image appear brighter which results in a new pixel value by adding the old pixel value with the offset value being, to decrease the brightness offset value will be subtracted from the original image. This process will also allow us to independently control both contrast and brightness through the alpha parameter and the beta parameter of any given image $f(x)$. It can be mathematically represented to apply a linear transformation to all the pixel values found in the original image, and Alpha denotes the scale of the pixel values to adjust the contrast, and beta is an additive term to increase and decrease the brightness. Here $g(x)$represents the new image obtained and $f(x)$ represents the original image α *and* β are two parameters accordingly controlling the contrast and brightness of the image as shown in the Eq. (6).

$$g(x) = \alpha * f(x) + \beta \tag{6}$$

Alpha is a constant multiplier, referred to as a coefficient or scaling factor. Beta, the bias or an offset added to the scaled function alpha multiplied with $f(x)$ to provide an additional constant contribution to the required output g(x). The results of different augmentation techniques are shown in Fig. 3.

Hybrid model 1

In the Proposed Hybrid Model 1, the input images of size 224*224 are subjected to pre-trained models VGG16 and MobileNetV2. The layers are set as non-trailable and only the output layers are considered for the feature extraction process then feature maps are made to combine various features such as batch size, height, width, and channels yielding a new channel axis as shown in Eq. (7). The resulting tensor has the shape of [b, h, w, 2c], where 2c signifies channels from VGG16 and MobileNetV2 that are concatenated. Concatenating the feature maps from both the deep learning networks, resulted in a more diverse and enriched representation of features. Now these concatenated feature maps are flattened into one-dimensional vectors that are required for further processing by the subsequent layers. The extracted combined features are applied to classification using K–Nearest Neighbors (KNN).

$$\text{Merged features [b, h, w, c]} = \text{[vf [b, h, w, c], mf [b, h, w, c]} \tag{7}$$

$\boldsymbol{b, h, w}$ ***and*** $\boldsymbol{c}$ represent batch, height, width, and channel dimension and $\boldsymbol{vf}$, $\boldsymbol{mf}$ represent VGG16 and MobileNetV2 model. The Eq. (7) represents the concatenation of two sets of features, where "vf" stands for a set of features with dimensions [b, h, w, c], and "mf" represents another set of features with the same dimensions. The resulting Merged features [b, h, w, c] tensor has a shape of [b, h, w, 2c], indicating that the features from both sets are combined along the last dimension and this can also be represented as a modular feature extraction model as shown in Fig. 4. The extracted features from the training dataset are subjected to the KNN classifier. The specific choice of K= 13 furthermore, emphasized the robustness of the KNN algorithm in handling the multi-dimensional feature

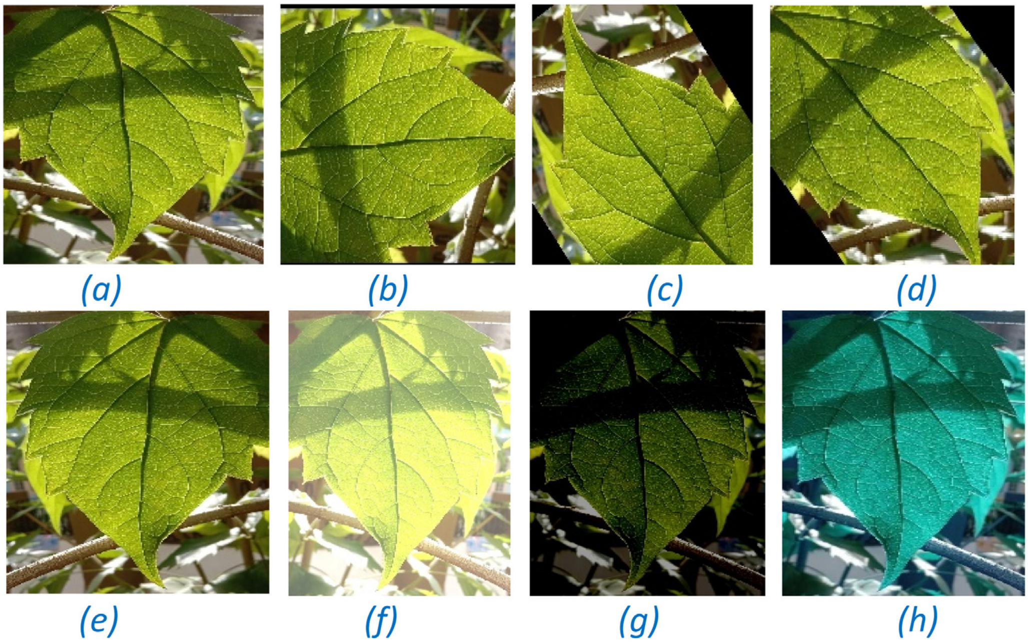

Fig. 3. Augmentation outputs. (a) Original Image. (b) Image rotated by 90˚. (c) Image rotated by 45. (d) Image rotated by -45˚. (e) Rotate flipping. (f) Increased Brightness (g) Lower Brightness(h) Hue image.

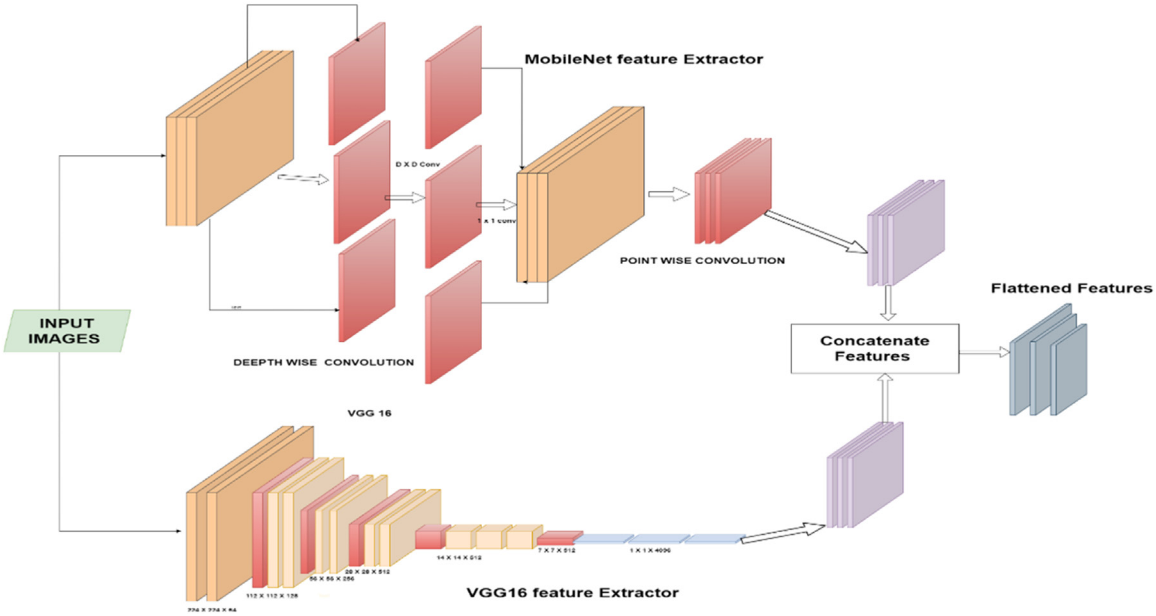

Fig. 4. Feature extraction architecture model for a hybrid model 1(modular representation).

space generated by this hybrid feature extraction model. Further contributing more valuable insights to the efficiency of this image classification task.

Hybrid model 2

The proposed hybrid model 2 consists of a fusion of two deep learning models such as ResNet50 and MobileNetV2 that accept input images of size 224 *224. ResNet50 is utilized with its pre-trained weights on the ImageNet dataset, the ResNet50 model is imported from Keras applications. It's configured to exclude its top classification layers and all top layers of this ResNet50 model are frozen to retain the pre-trained weights like ResNet50, MobileNet is configured to exclude its top classification layer. The features are

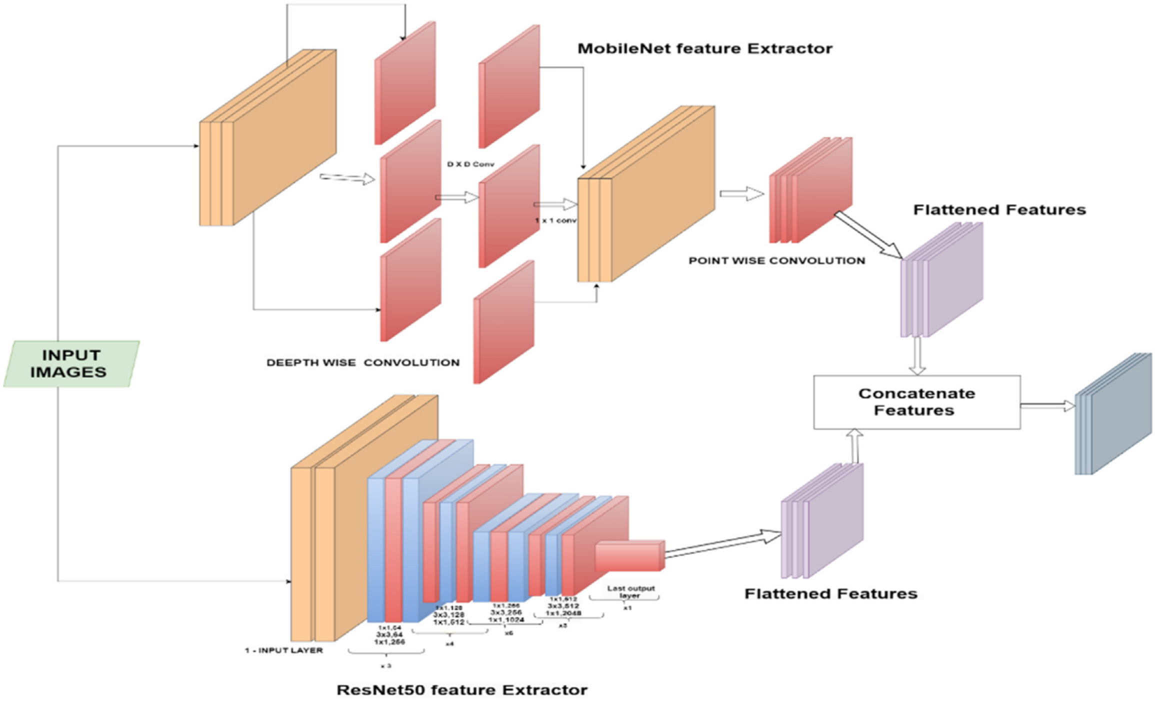

Fig. 5. Feature extraction architecture model for a hybrid model 2 (modular representation).

extracted from this hybrid model. The extracted features are reshaped into 1D-feature vectors by leveraging the strengths of both the pre-trained models as shown in Eq. (8). The next step involves the construction of a custom-based layer which is completely built on a neural network. The structure of the neural network is elaborated upon, such as the size of the input layer and the incorporation of Batch Normalization and ReLU activation function in the hidden layers. A custom classification model is constructed on top of the concatenated features. It consists of two Dense layers with 128 and 64 units respectively, using ReLU activation functions, batch normalization, and increased regularization strength (l2 regularization with a factor of 0.1). Dropout is also applied with a rate of 0.5 to reduce overfitting. The model is compiled using the Adam optimizer and categorical cross-entropy loss function. It is then trained on the combined features for 100 epochs, using the training data and its corresponding one-hot encoded labels. There are four Dense layers (two explicitly defined Dense layers and two additional Batch Normalization layers). Additionally, there are two activation layers (ReLU) and one Dropout layer, making a total of seven additional layers on top of the pre-trained ResNet50 and MobileNet models for feature extraction. Validation data are used to monitor the model's performance during the training period. After the training process, the model is assessed using the test data which is 20 % of the total dataset.

$$\text{Merged} - \text{features}\ [\text{b, h, w, c}] = [\text{rs}\ [\text{b, h, w, c}],\ \text{mf}\ [\text{b, h, w, c}] \tag{8}$$

b, h, w *and* c represent batch, height, width, and channel dimension and rs, mf represent ResNet50 and MobileNetV2 model. This formula represents the concatenation of two sets of features, where "rs" stands for a set of features with dimensions [b, h, w, c], and "mf" represents another set of features with the same dimensions. The resulting "Merged features [b, h, w, c] " tensor has a shape of [b, h, w, 2c], indicating that the features from both sets are combined along the last dimension and this can also be represented as modular feature extraction model as shown in Fig. 5.

Hybrid model 3

In this section, hybrid model-3 is investigated to analyze the effectiveness of MobileNetV2 with the Squeeze and Excitation (SE) layers for the classification tasks. In the first step, the MobileNetV2 model is pre-trained on the ImageNet dataset. The future model's top layers are set to false, and the input shape (224,224,3) is specified according to the model's compatibility. Subsequently, SE layers are defined to create a SE block. The link between the two is achieved by connecting the last layer of the base model to the SE block. Features extracted from MobileNetV2 are given as the input tensor from the last layer of the MobileNetV2 to SE layers, and then Global Average pooling is used to calculate the average value for each channel across the spatial dimensions as shown in Eq. (9). This results in a global representation of the feature map that is reshaped to transform the global average pooling output into a tensor shape hence making it suitable for future subsequent dense layers. Here in this context, it has 2 dense layers, the first dense layer reduces the dimensionality of the data, and the ReLU activation function is used to introduce non-linearity as shown in Eq. (10)–(12). Similarly, the second dense layer has a sigmoid activation function which generates a channel-wise scaling factor.

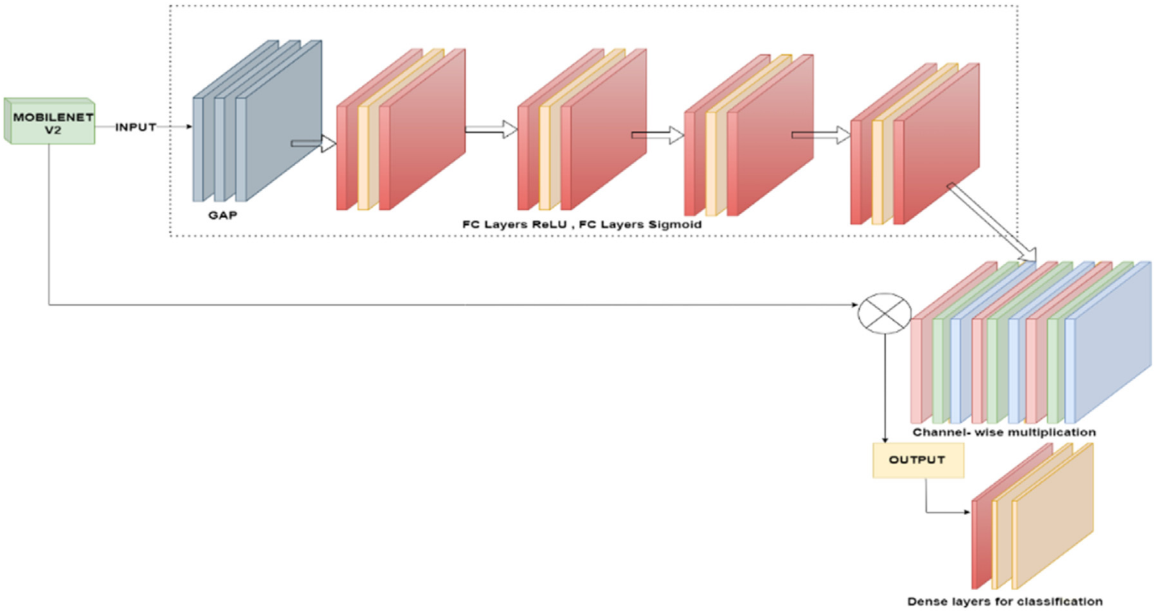

Fig. 6. Hybrid model 3 modular representation.

The final step includes element-wise multiplication of the original input layer with the output from the SE block which is effectively scaling each channel based on its importance as shown in Eq. (10). Combining these an integrated model is created which is used to extract features from the training dataset and testing dataset respectively. These extracted features are reshaped to make it more suitable for classification tasks. Further, classification is carried out using three densely connected layers. The first layer has 512 units with the ReLU activation function which is followed by a dropout of (0.5) to prevent overfitting, second layer has 256 units again with the ReLU function and a dropout layer. The last output layer has 13 units with SoftMax activation function for multiclass classification tasks. This model was compiled using Adam optimizer, a categorical cross-entropy loss function, and with 100 epochs.

$$\boldsymbol{GAP}\left(\boldsymbol{channel}_{i}\right) = \frac{1}{\boldsymbol{H}\cdot\boldsymbol{W}}\sum_{\boldsymbol{x}=1}^{\boldsymbol{H}}\sum_{\boldsymbol{y}=1}^{\boldsymbol{W}}\text{Feature\_Map}(\boldsymbol{x},\boldsymbol{y},\text{channel\_i}) \tag{9}$$

Here GAP represents the global average polling result for the specific channel number i. H and W are the height and width of the feature map. x and y are spatial indices representing their respective position over a feature map. $x, y,$ c channel_i is the value of the feature map at spatial coordinates. (x, y) for respective channels and the modular representation is given in Fig. 6.

$$\text{Squeeze(ci)} = \text{ReLU (Weight squeeze} * \text{GAP (ci)} + \text{Bias squeeze)} \tag{10}$$

$$Excitation(ci) = Sigmoid\ (Weight\_excitation \cdot Squeeze(ci) + Bias\ excitation) \tag{11}$$

$$OutputFeatureMap\ (x,\ y,\ ci)\ = FeatureMap\ (x,\ y,\ ci).\ excitation(ci) \tag{12}$$

Experimentation and results

The proposed hybrid models are experimented using the 11th Gen Intel(R) Core (TM) i5–1135G7,2.42 GHz processor, operating at a base frequency of 2.40 gigahertz. and a maximum turbo frequency of 2.42 gigahertz with 15.7 GB RAM. The The accuracy attained using our three proposed hybrid models and state-of-the-art pre-trained neural network models that are trained on the self-created real-time medicinal plant species dataset is listed in Table 2.

From Fig. 7 it is observed that the fusion of deep learning CNN models produced better accuracy. This indicates that integrating different deep architectures with diverse features to create a feature extractor model inferred a more elevated performance. Comparing the three hybrid models, hybrid model 1, a fusion of VGG16 with MobileNet achieved an accuracy of 85.85 %, model 2 fusion of MobileNet with ResNet50 resulted in 88 %, and hybrid model 3 at an impressive 94.24 % accuracy that projects a notable advantage for hybrid models compared to the existing state-of-the-art approaches.

Table 2
Classification accuracy obtained using hybrid models and state of art methods.

sl.no	MODEL	Accuracy
1	MobileNet + XG-Boost	80.60 %
2	VGG16+KNN	83.98 %
3	Hybrid Model-1	85.85 %
4	Hybrid Model-2	88 %
5	Hybrid model-3	94.24 %

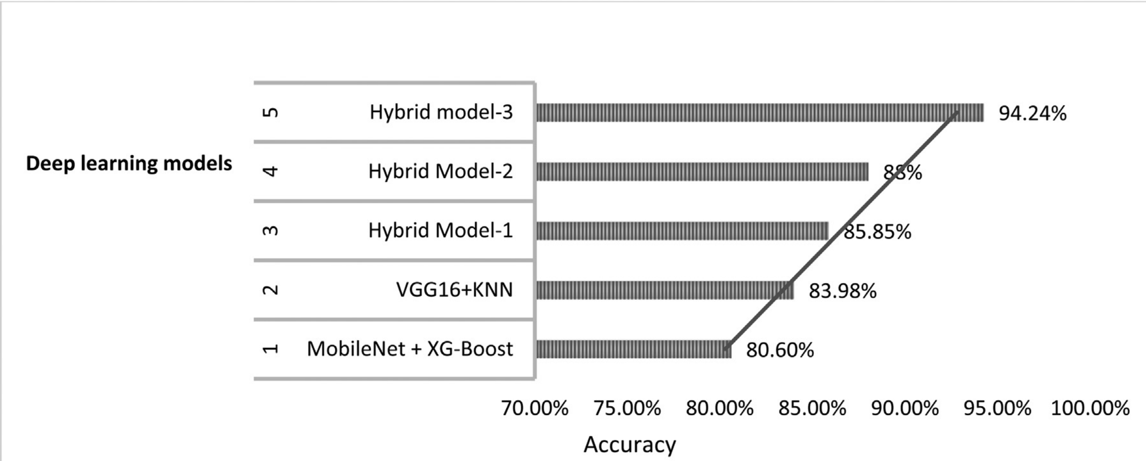

Fig. 7. Performance of hybrid models and state of art methods with respect to accuracy.

Hybrid model 1

The confusion matrix and learning curve in Fig. 8 illsutrate the hybrid model 1(VGG16 + MobileNet) performance across thirteen different classes indicating successful classification for most of the classes. Overall, the model demonstrates varying degrees of accuracy and misclassification across different classes by achieving an accuracy of 85.85 %. The learning curve for the hybrid model 1 with K nearest neighbor as a classifier consists of an upward trend indicating a progressive improvement in both training and cross-validation curves starting from 55 % and gradually increasing to 85 % and more thus this upward trajectory suggests that the model benefits from more data or increased complexity. The final accuracy of 85.85 % attests to the effectiveness of the chosen parameters showing a well-optimized model for the successful classification of the test dataset.

Hybrid model 2

The performance of Hybrid Model 2, (ResNet50 + MobileNet) concerning training, validation accuracy and confusion matrix are represented in Figs. 9. As for the training and validation loss and accuracy curves, a notable trend emerged. The training loss curve displayed a remarkable constancy, indicating the model's effective convergence and mastery of the training data. In contrast, the validation loss curve exhibited slight variations, suggesting that the model maintained a degree of adaptability for previously unseen examples. Interestingly, the training accuracy remained consistently high, indicating the model's proficiency in learning the training data, while the validation accuracy displayed nuanced fluctuations. The macro and weighted averages reinforce the overall effectiveness of the ResNet50 + MobileNet hybrid model, showcasing its ability to achieve a balanced performance across multiple classes. These findings collectively underscore the model's robustness and competence in this image classification tasks.

The confusion matrix highlights the model's proficiency in classifying certain categories, with particularly high precision and recall values better than Hybrid Model 1. This specificity indicates the model's adeptness in distinguishing fine-grained features within these classes. The overall accuracy yields 88 %, showcased enhanced interpretability through the confusion matrix analysis. A unique feature was the model's ability to capture long-range dependencies via residual connections from ResNet50, coupled with Mobile Net's efficiency in handling computational complexities. The confusion matrix unveiled the model's sensitivity to certain class transitions better than Hybrid Model 1.

In Fig. 10, there are t-SNE (t-distributed Stochastic Neighbor Embedding) visualizations which are used to represent the features extracted from hybrid model 1 and hybrid model 2 in the 2D space. These two resulting clusters of 13 different classes are distinguished by their unique colors. These visualizations benefited to differentiate between the classes immediately. Despite the absence of distinct clusters in the representation, the major goal here is to not cluster, but rather to represent the features that have been extracted from

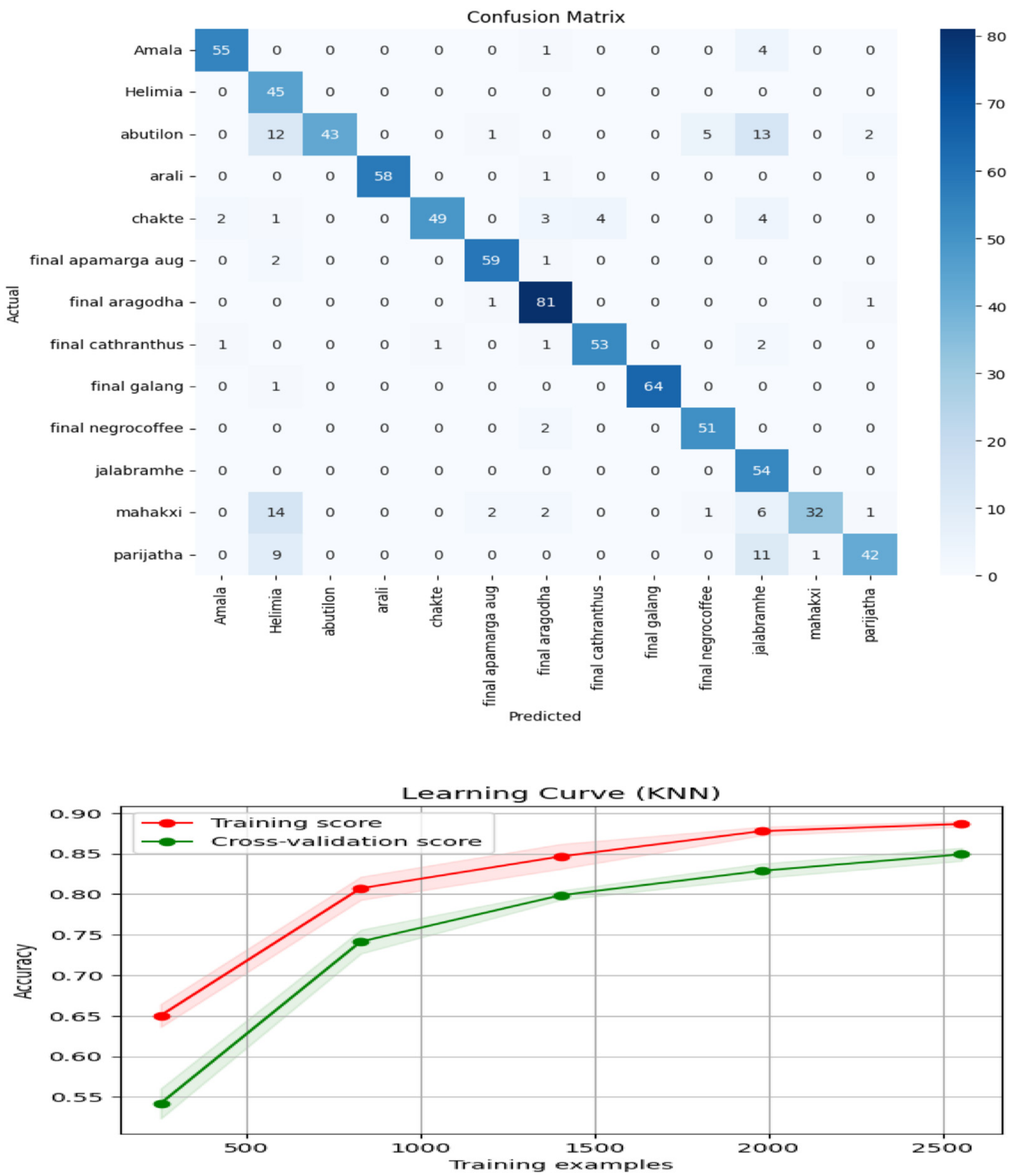

Fig. 8. Confusion matrix and learning curve for Hybrid Model 1.

both the hybrid model. It provides a comprehensive and exclusive overview of the feature span across the 13 different classes of self-created data set offering more insightful distribution and relationship among them. This visualization can also be instrumental in gaining a more qualitative understanding of the feature space and which further facilitates an interpretation analysis of hybrid models feature extraction.

Hybrid model 3

In the Hybrid Model 3, the training loss and accuracy graph for the MobileNetV2 model is presented in Fig. 11 which reveals a compelling learning trajectory. The graph exhibits a significant and consistently increasing trend in accuracy, reaching an impressive 94.24 %. This robust upward trajectory in accuracy implies that the scaling model has profoundly influenced the model's ability to discern crucial features within the images. The concurrent decline in training loss suggests that the model is adeptly minimizing errors and converging toward an optimal solution. The coherence between the ascending accuracy and descending loss demonstrates the effectiveness of the scaling model in enhancing the model's discriminative capabilities. The considerable performance improvement underscores the utility of this preprocessing technique in accentuating relevant image features and contributing to the model's overall accuracy. This model yielding an accuracy of 94.24 %, showcased enhanced interpretability through the confusion matrix analysis.

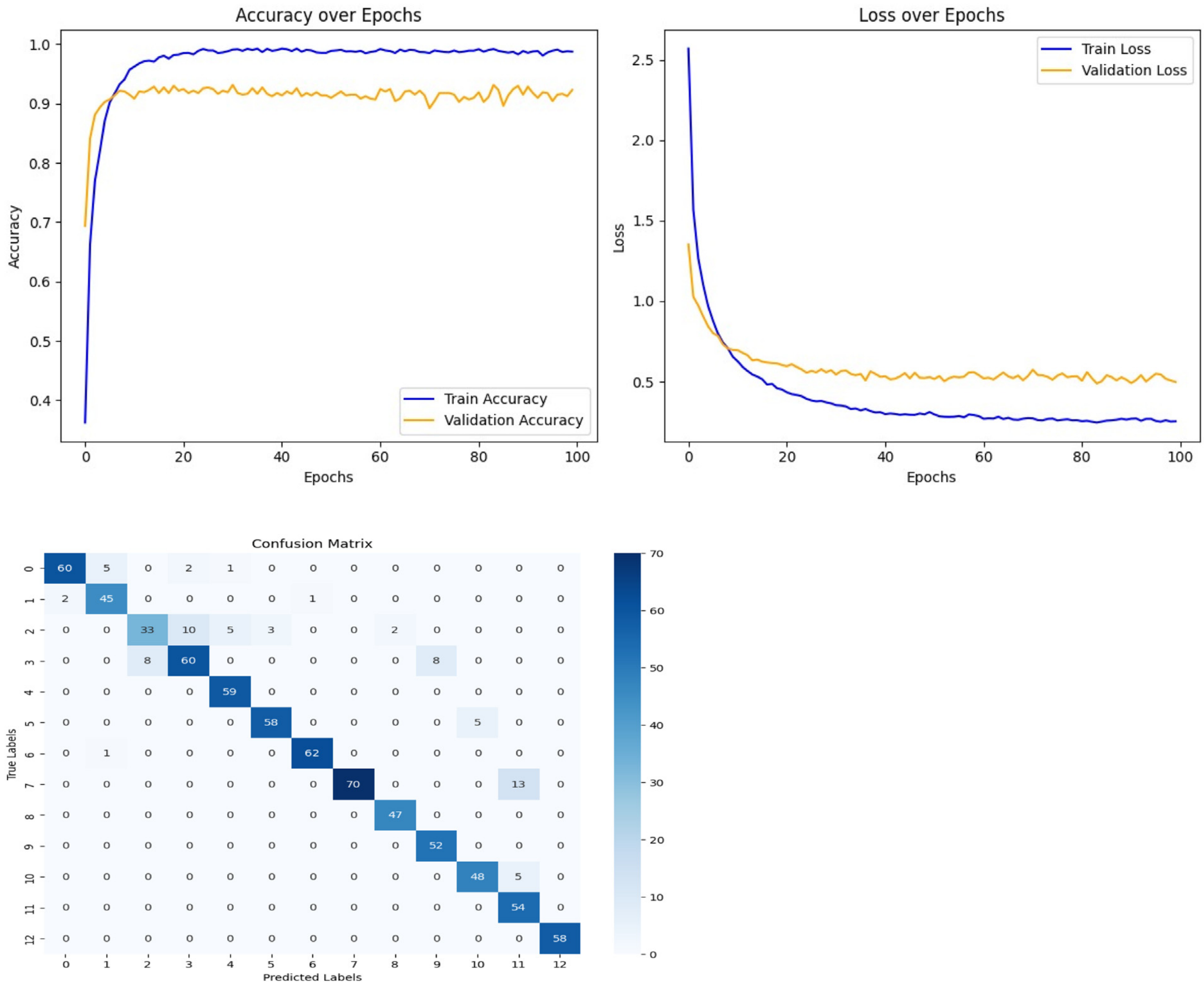

Fig. 9. Training, validation accuracy and confusion matrix of hybrid Model -2.

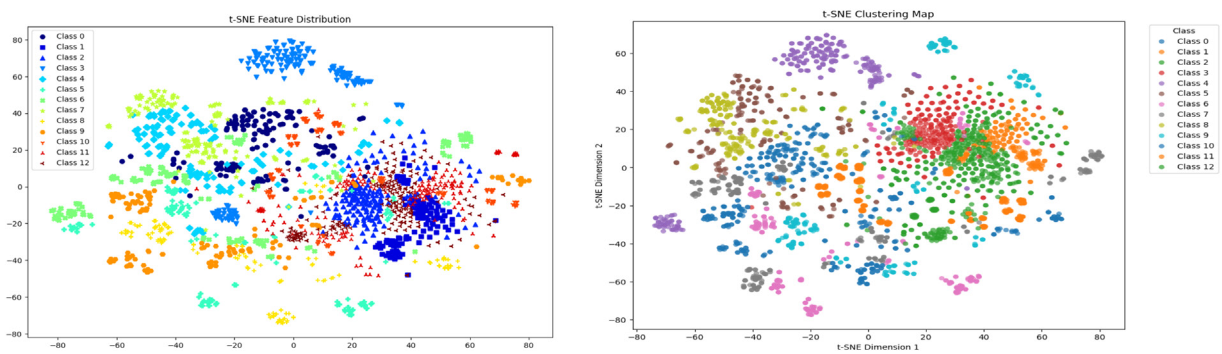

Fig. 10. t-SNE used to represent features extracted from hybrid models 1 and 2.

A unique feature was the model's ability to capture rescaled feature vectors via MobileNet with SE block, which further led to much higher precision, accuracy, and recall compared to the other two hybrid models.

Table 2 provides the comparison of the accuracy percentage achieved by the proposed hybrid methods with other state-of-the-art methods in the literature that have utilized conventional approaches (ML), deep learning models (DL), and combining ML and DL approaches. It is observed that the proposed hybrid models yield better accuracy and hybrid model 3 (MobileNetV2+SE block)

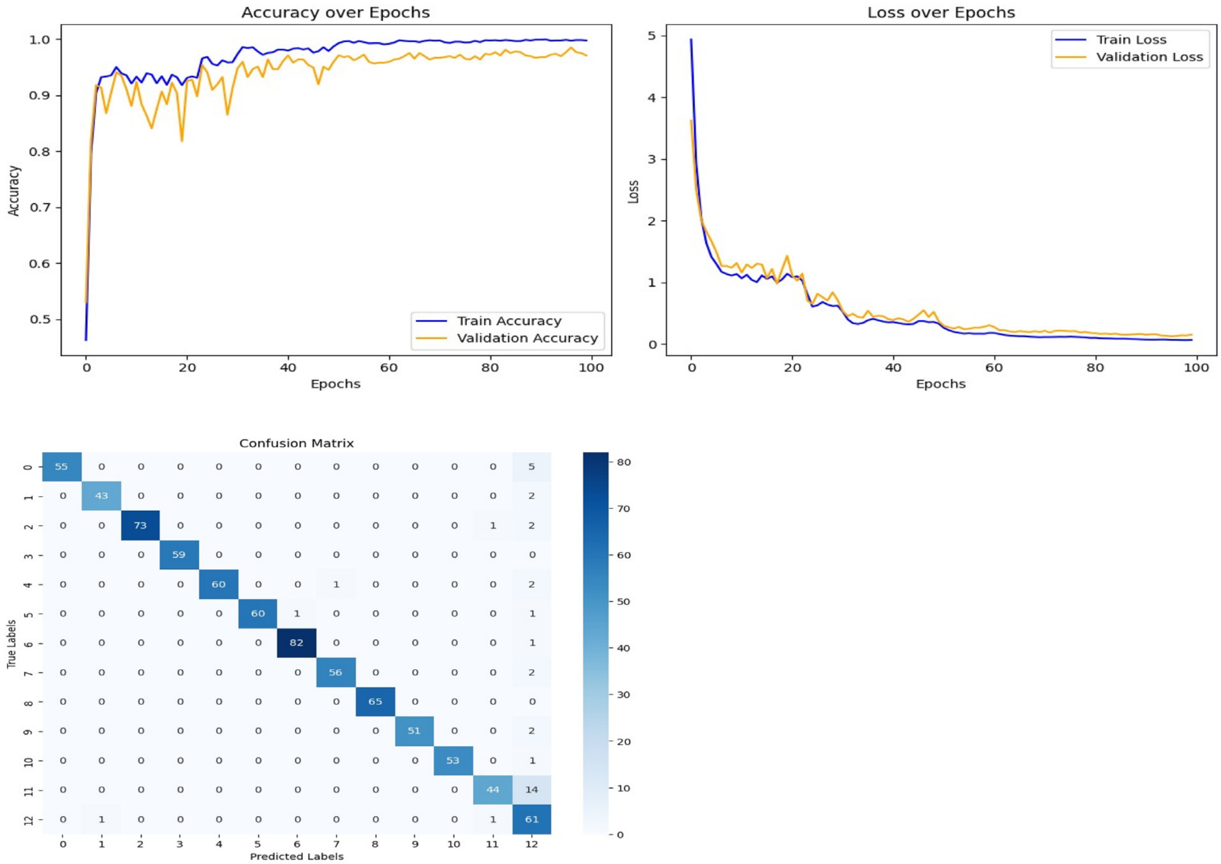

Fig. 11. Training loss, accuracy and confusion matrix for hybrid model 3.

Table 3
Comparison of accuracy of the hybrid models with state-of-the-art methods.

Authors	Dataset	Image samples	Model	Accuracy
[41]	Flavia	1907 images	Ten-layer CNN	87.92 %
[42]	Self-Created	25 Herbal plants	Six-classical ML used -MLP	82.51 %
[43]	Self-Created	6000 images	Ayur-PlantNet	92.27 %
[44]	Self-Created	45 different leaf plants	CNN	93.75 %
[45]	PlantCLEF2017, ExpertLifeCLEF2018, iNaturalist2018	1,21,501 leaf images	ViT-Large/16	91.15 % & 83.54 %
[46]	DanangForestPlant	10,527 images	PlantKViT, Resnet152, ConvNeXt	93 %,89 %,76 %
[47]	Flavia	1907 images	SVM, KNN, RF	90 %,88 %,84 %
[48]	plant village dataset	54,303 images	CNN	87.62 %
[49]	Self-Created	32 plant leaf images	SVM, KNN, ANN	94 %
[50]	Self-Created	34,123 images	CNN	71.3 %
PROPOSED WORK	Self-Created	4000 images	Hybrid model 1 Hybrid model 2 Hybrid model 3	85.85 %, 88 % 94.24 %

resulted in a good performance metric- accuracy of 94.24 %. It is evident that proposed hybrid models produce extreme results with limited and real-time dataset as shown in Table 3.

Conclusion & future work

The proposed study investigated three novel deep-learning hybrid models on real-time medicinal plant species recognition. The hybrid models utilized VGG 16, MobileNet, MobileNetV2, and ResNet50 as feature extractors. The ultimate feature vector is obtained by merging the features that were extracted from each of the models separately. Then, this ultimately extracted feature vector is classified using machine learning and neural network classifiers. Regardless of the dataset sample size limitation, hybrid model 3 performed better than other hybrid models and the state-of-the-art outcomes, according to the experimental results. It is good to perform a feature channel rescaling that improves the model's performance in comparison with the other two hybrid models and the regular model. The proposed models result in achieving a good recognition rate of plant species that are appropriate in recognizing the plant species that are captured in real-time and in unconstrained environments that pose certain image challenges. The models developed are lightweight CNNs that are suitable for building a mobile application for recognizing the variety of medicinal plant species, furthermore, research would focus on expanding the datasets to include more rare medicinal plant species to enhance the capability and the robustness of the proposed models in the real-world scenarios, ultimately contributing to more effective medicinal plant identification and biodiversity conservation efforts. As a future scope, the model can be enhanced further to best fit to train more number of medicinal plant species with image challenges.

Limitations

- In order to employ the proposed method for the image samples with varied backgrounds, it is required to devise a model that can work efficiently to detect the plant species accurately.
- The dataset utilised in the proposed samples is limited with respect to the number of plant species considered and the image samples.
- In the proposed models, the images are directly trained on the proposed hybrid deep learning models, the performance of the model can be still increased by applying the segmentation to extract the region of interest.

Ethics statements

Human subjects do not appear in our research work nor animal experiments or even social media data for that matter.

Supplementary material *and/or* additional information [OPTIONAL]

None

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

B.R. Pushpa: Conceptualization, Methodology, Data curation, Supervision, Formal analysis. **S. Jyothsna:** Conceptualization, Methodology, Data curation, Validation, Formal analysis. **S. Lasya:** Methodology, Software, Formal analysis, Validation, Visualization, Writing – original draft.

Data availability

Data will be made available on request.

Acknowledgments

None

References

[1] R. Gowthami, N. Sharma, R. Pandey, A. Agrawal, Status and consolidated list of threatened medicinal plants of India, Genetic Resour. Crop. Evolut. 68 (6) (2021) 2235–2263.
[2] K.K. Sherly, A. Sonia, Hybrid CNN models for plant species recognition and disease detection, in: Science and Information Conference, Springer Nature Switzerland, Cham, 2024, pp. 35–50.
[3] S. Kavitha, T.S. Kumar, E. Naresh, V.H. Kalmani, K.D. Bamane, P.K. Pareek, Medicinal plant identification in real-time using deep learning model, SN Comput. Sci. 5 (1) (2023) 73.
[4] S.N. Antunes, M.T. Okano, I.D.A. Nääs, W.A.C. Lopes, F.P.L. Aguiar, O. Vendrametto, … M.E. Fernandes, Model development for identifying aromatic herbs using object detection algorithm, AgriEngineering 6 (3) (2024) 1924–1936.
[5] A.H. Uddin, Y.L. Chen, B. Borkatullah, M.S. Khatun, J. Ferdous, P. Mahmud, … L.Y. Por, Deep-learning-based classification of Bangladeshi medicinal plants using neural ensemble models, Mathematics 11 (16) (2023) 3504.
[6] A. Haridasan, J. Thomas, E.D. Raj, Deep learning system for paddy plant disease detection and classification, Environ. Monit. Assess. 195 (1) (2023) 120.
[7] A. Pal, V. Kumar, AgriDet: Plant Leaf Disease severity classification using agriculture detection framework, Eng. Appl. Artif. Intell. 119 (2023) 105754.
[8] A. Pavithra, G. Kalpana, T. Vigneswaran, Deep learning-based automated disease detection and classification model for precision agriculture, Soft Comput. (2023) 1–12.
[9] W.B. Demilie, Plant disease detection and classification techniques: a comparative study of the performances, J. Big Data 11 (1) (2024) 5.
[10] J. Yao, S.N. Tran, S. Garg, S. Sawyer, Deep learning for plant identification and disease classification from leaf images: multi-prediction approaches, ACM Comput. Surveys 56 (6) (2024) 1–37.

[11] J. Wäldchen, P. Mäder, Plant species identification using computer vision techniques: a systematic literature review, Arch. Comput. Meth. Eng. 25 (2018) 507–543.
[12] G. Sharma, A. Kumar, N. Gour, A.K. Saini, A. Upadhyay, A. Kumar, Cognitive framework and learning paradigms of plant leaf classification using artificial neural network and support vector machine, J. Exp. Theor. Artif. Intell. 36 (4) (2024) 585–610.
[13] K. Pushpanathan, M. Hanafi, S. Mashohor, et al., Machine learning in medicinal plants recognition: a review, Artif. Intell. Rev. 54 (2021) 305–327.
[14] S. Islam, M.F. Ahmed Foysal and N. Jahan, "A computer vision approach to classify local flower using convolutional neural network," 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2020, pp. 1200–1204, doi: 10.1109/ICICCS48265.2020.9121143.
[15] A.M. Raghukumar and G. Narayanan, "Comparison Of machine learning algorithms for detection of medicinal plants," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2020, pp. 56–60, doi: 10.1109/ICCMC48092.2020.ICCMC-00010.
[16] M.R. Dileep and P.N. Pournami, "AyurLeaf: a deep learning approach for classification of medicinal plants," TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON), Kochi, India, 2019, pp. 321–325, doi: 10.1109/TENCON.2019.8929394.
[17] S.A. Wagle, R. Harikrishnan, S.H.M. Ali, M. Faseehuddin, Classification of plant leaves using new compact convolutional neural network models, Plants 11 (1) (2021) 24.
[18] H. Chanyal, R.K. Yadav, D.K.J. Saini, Classification of medicinal plants leaves using deep learning technique: a review, Int. J. Intellig. Syst. Appl. Eng. 10 (4) (2022) 78–87.
[19] J.G. Thanikkal, A.K. Dubey, M.T. Thomas, An efficient mobile application for identification of immunity boosting medicinal plants using shape descriptor algorithm, Wireless Personal Commun. (2023) 1–17.
[20] V. Tiwari, R.C. Joshi, M.K. Dutta, Deep neural network for multi-class classification of medicinal plant leaves, Expert Systems 39 (8) (2022) e13041.
[21] S. Anubha Pearline, V. Sathiesh Kumar, S. Harini, A study on plant recognition using conventional image processing and deep learning approaches, J. Intellig. Fuzzy Syst. 36 (3) (2019) 1997–2004.
[22] C.P. Lee, K.M. Lim, Y.X. Song, A. Alqahtani, Plant-CNN-ViT: plant classification with ensemble of convolutional neural networks and vision transformer, Plants 12 (14) (2023) 2642.
[23] L. Wu, J. Yang, Z. Gu, J. Guo, X. Zhang, Rare and endangered plant leaf identification method based on transfer learning and knowledge distillation, Int. J. Agric. Environ. Inf. Syst. (IJAEIS) 12 (4) (2021) 1–24.
[24] M. Khanna, L.K. Singh, S. Thawkar, M. Goyal, PlaNet: a robust deep convolutional neural network model for plant leaves disease recognition, Multim. Tools Appl. 83 (2) (2024) 4465–4517.
[25] M.V. Vaidehi, M.V. Vinod, ResNet based classification in CNN for ayurvedic plant categorization using deep learning, Design Eng. (2021) 1507–1516.
[26] \ & S.S. Roy, M. Ahmed, M.A.H Akhand, Noisy image classification using hybrid deep learning methods, J. Inf. Commun. Technol. 17 (2) (2018) 233–269.
[27] \ & P. Thamilselvan, J.G.R Sathiaseelan, Image classification using hybrid data mining algorithms-a review, in: In 2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), IEEE, 2015, pp. 1–6.
[28] P. Bedi, P. Gole, Plant disease detection using hybrid model based on convolutional autoencoder and convolutional neural network, Artif. Intell. Agric. 5 (2021) 90–101.
[29] P. Kaur, S. Harnal, R. Tiwari, S. Upadhyay, S. Bhatia, A. Mashat, A.M. Alabdali, Recognition of leaf disease using hybrid convolutional neural network by applying feature reduction, Sensors 22 (2) (2022) 575.
[30] Q.I.U. Wenjie, Y.E. Jin, H.U. Liangqing, Y.A.N.G. Juan, L.I. Qili, M.O. Jianyou, Y.I. Wanmao, Distilled-mobilenet model of convolutional neural network simplified structure for plant disease recognition, Smart Agriculture 3 (1) (2021) 109.
[31] J. Banzi, T. Abayo, Plant species identification from leaf images using deep learning models (CNN-LSTM architecture), Tanzania J. Forestry and Nature Conserv. 90 (3) (2021) 93–103.
[32] C. Zhao, S.S. Chan, W.K. Cham, L.M. Chu, Plant identification using leaf shapes—A pattern counting approach, Pattern Recognit. 48 (10) (2015) 3203–3215.
[33] A. Chaudhury, J.L. Barron, Plant species identification from occluded leaf images, IEEE/ACM Trans. Comput. Biol. Bioinf. 17 (3) (2018) 1042–1055.
[34] Y. Zhang, S. Wa, L. Zhang, C. Lv, Automatic plant disease detection based on tranvolution detection network with GAN modules using leaf images, Front. Plant Sci. 13 (2022) 875693.
[35] E. Hossain, M.F. Hossain, M.A. Rahaman, A color and texture based approach for the detection and classification of plant leaf disease using KNN classifier, in: 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), IEEE, 2019, pp. 1–6.
[36] O.A. Malik, N. Ismail, B.R. Hussein, U. Yahya, Automated real-time identification of medicinal plants species in natural environment using deep learning models—A case study from borneo region, Plants, 11 (15) (2022) 1952.
[37] S. Sonali, S.S. Dhotre, Improved deep learning-based classifier for detection and classification of aloe barbadensis miller disease, Int. J. Intellig. Syst. Appl. Eng. 12 (2s) (2024) 239–254.
[38] Y. Sun, Y. Liu, G. Wang, H. Zhang, Deep learning for plant identification in natural environment, Comput. Intell. Neurosci. 2017 (2017).
[39] M.H. Saleem, J. Potgieter, K.M. Arif, Plant disease detection and classification by deep learning, Plants 8 (11) (2019) 468.
[40] S.P. Mohanty, D.P. Hughes, M. Salathé, Using deep learning for image-based plant disease detection, Front. Plant Sci. 7 (2016) 1419.
[41] J. Liu, S. Yang, Y. Cheng and Z. Song, "Plant leaf classification based on deep learning," 2018 Chinese Automation Congress (CAC), Xi'an, China, 2018, pp. 3165–3169, doi: 10.1109/CAC.2018.8623427..
[42] G. Kumar, V. Kumar, A.K. Hrithik, Herbal plants leaf image classification using machine learning approach, in: Intelligent Systems and Smart Infrastructure, CRC Press, 2023, pp. 549–558.
[43] B.R. Pushpa, N.S. Rani, Ayur-PlantNet: An unbiased light weight deep convolutional neural network for Indian Ayurvedic plant species classification, J. Appl. Res. Med. Aromatic Plants 34 (2023) 100459.
[44] D.B. Valdez, C.J.G. Aliac, L.S. Feliscuzo, Medicinal plant classification using convolutional neural network and transfer learning, in: 2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET), IEEE, 2022, pp. 1–6.
[45] L. Picek, M. Šulc, Y. Patel, J. Matas, Plant recognition by AI: Deep neural nets, transformers, and kNN in deep embeddings, Front. Plant Sci. (2022) 2788.
[46] N. Van Hieu, N.L.H. Hien, L. Van Huy, N.H. Tuong, P.T.K. Thoa, PlantKViT: a combination model of vision transformer and KNN for forest plants classification, JUCS: J. Univ. Comput. Sci. 29 (9) (2023).
[47] K. Malarvizhi, M. Sowmithra, D. Gokula Priya, B. Kabila, Machine learning for plant species classification using leaf vein morphometric, Int. J. Eng. Res. Technol. (IJERT) 10 (04) (2021).
[48] A. Agarwal, R. Venkat, Prediction of leaves using convolutional neural network, Inf. Syst. 9 (4) (2020) 35–38.
[49] U. Kumar, S. Yadav, E. Tripathi, A hybrid approach for automated plant leaf recognition using hybrid texture features and machine learning-based classifiers, Int. J. Distrib. Artif. Intellig. (IJDAI) 13 (2) (2021) 27–39.
[50] R. Akter, M.I. Hosen, CNN-based leaf image classification for Bangladeshi medicinal plant recognition, in: In 2020 Emerging Technology in Computing, Communication and Electronics (ETCCE), IEEE, 2020, pp. 1–6.