Adv Syst Sci Appl 2023; **03**; 128-151
Published online at http://ijassa.ipu.ru.

Deep Feature Extraction and Weight Updated Tuned Random Forest for Piper Plant Species Recognition

A. Pravin[1*], C. Deepa[2]

[1]*Research Scholar, Department of Computer Science, Sri Ramakrishna College of Arts & Science, Coimbatore, India*

[2]*Associate Professor & Head, Department of Computer Science (Artificial Intelligence & Data Science), Sri Ramakrishna College of Arts & Science, Coimbatore, India*

Abstract: Recently, identifying plant species has become a significant research area as it is vital for securing biodiversity. Plants also possess various medicinal applications. Hence, predicting different species of plants is of utmost significance. However, determining plant species through conventional ways is a time-consuming process. That happens due to huge and distinct botanical terms. With the recent evolution of AI (Artificial Intelligence) based algorithms, researchers have undertaken various attempts to predict plant species. However, most studies averted the consideration of piper plant species which holds huge medicinal benefits. Existing research also failed to predict the plant species due to inefficient feature extraction accurately. Considering such a pitfall, this study proposes Deep CNN (Deep Convolutional Neural Network) and Inception V3 to extract features to perform all plant classification. In addition, the study proposes Deep CNN and VGG16 (Visual Geometry Group16) to extract suitable features for performing piper plant classification. Following this, the study considers PCA (Principle Component Analysis) for feature fusion as it can reduce noise in data and select relevant features for affording independent and uncorrelated data features. Finally, the study proposes WUT-RF (Weight Updated Tuned Random Forest) to classify piper and all plant species. In this process, hyperparameters of RF are tuned with convolutional likelihood weight to attain a high prediction rate. Optimal hyperparameter selection and tuning assist in improvising the performance of the proposed classifier. Performance analysis of this system about performance metrics exposes its effectiveness in plant species detection.

Keywords: Artificial Intelligence, Piper Plants, Deep Convolutional Neural Network, Inception V3, VGG16, Random Forest

1. INTRODUCTION

Plants possess several utilities in food, industry, medicine, etc. It also has a huge contribution to protecting the environment. There exists a huge plant species variety, and this count is increasing each year. As plants hold distinct benefits, identifying and classifying them is paramount. Among different plants, piper plants have significant applications in medicine, such as asthma, gonorrhoea, diarrhoea, malaria, cough, tumours, etc. (Liu, Yang, Huang, & Lin, 2022). Comprehending different plant species has become vital for several individuals, like farmers, educators, environmentalists, and other field workers. To describe a plant species, botanists need to be persuaded that the plant varies from other renowned species. Accordingly, new plant species are generally determined when a botanist expert revises the nomenclature of the entire species group by revising the taxonomy (Anubha Pearline, Sathiesh Kumar, & Harini, 2019). Due to different botanical terms, such a classification method is tedious and time-consuming. Generally, plant species are recognized through their flower, seed, fruit, leaves, etc. However, using the leaf for plant species recognition seems convenient and simple, and researchers have attempted to consider leaves for identifying

* Corresponding author: pravinresearchscholar@gmail.com

plant species. With technological progress, digital images gained significance in various areas. Concurrently, with the progress of AI (Artificial Intelligence), researchers have endeavored to develop an automated system for identifying and classifying plant species that could recognize plants to a certain extent (N. S. Kumar, Yashwanth, Srinivasulu, & Rini).

By this, an automated plant detection system is suggested that uses ML and computer vision for classifying plant leaf images. Texture and color features have been retrieved. Following this, the SVM classifier has been considered for classification. Testing has been performed on the Swedish dataset, exposing 93.26% accuracy (Kaur & Kaur, 2019). Further, leaf images are segmented and classified in a complex background through DL based method (Yang, Zhong, & Li, 2020). Initially, 2500 images having complex backgrounds were gathered. Considered images are fed into Mask R-CNN (Mask Region-based Convolutional Neural Network) (Hao et al., 2021)for training the model. Subsequently, a training set encompassing 1500 images of 15 species is fed into Deep CNN with VGG16 to train the suggested model to classify the leaf. Suitable hyperparameters for these methodologies are determined by comparing several parameter integrations. Outcomes expose that the average ME (Misclassification Error) of eighty test images is 1.15%. The average accuracy rate is explored to be 91.5%.

Further, an enhanced CNN structure is suggested for solving the leaf classification issues in small sample cases. Metric space quality and selection of supervised samples find the classification rate of the suggested K-NN (K-Nearest Neighbour) classifier. Experimental outcomes reveal that, while training samples are 20, accuracy for classification seems to be high (Wang & Wang, 2019). Furthermore, a model is suggested to analyze the leaf shape that relies on geometrical features and sinuosity coefficients. Experimentations undertaken on the LeafSnap dataset expose a better classification rate. Further, the suggested feature extraction algorithm is improvised by adding the geometrical features of the leaf, and classification rates are explored to be 93% while using RBF and 82% while considering MLP (Ganesan, 2021; Kala & Viriri, 2018).

In addition, several visual attributes are used to classify heterogeneous leaves that contrast in shape, surface, and hue. An FSST strategy selects the shape highlights for several leaf classes. Subsequently, a progressive method is considered that encompasses pre-processing to normalize the introduction and scaling of several leaves, a hue evaluating stage that includes extracting hue highlights, shape evaluation including the shape-based representation, and surface venture analysis for exposing surface instances of the leaf surface. The individual layer encompasses modules to treat conventional practice and discriminators to pick relevant modules for subsequent preparation. Further, NFC grouping is undertaken to exploit likeliness amongst broad leaves for shape and hue layers. Finally, the Euclidean metric is used to segregate surface components (Chaki, Parekh, & Bhattacharya, 2020).

The study [11] also performs piper plant recognition with all plant classifications (Pravin & Deepa, 2022), considering hyper-parameter-tuned RF (Random Forest). Deep CNN is regarded for extracting features. Empirical outcomes reveal the better performance of the suggested system with 0.94 accuracies for all the plant species, while 0.88 accuracies are attained for predicting piper plant while compared with NB (Naïve Bayes), LR (Logistic Regression) and SVM (Support Vector Machine). Conventional works have used different ML and DL-based algorithms to identify pipers and other plants. Though such methods have tried to attain better outcomes, most studies have not considered piper plant classification. Furthermore, few studies considered piper plant classification but lacked a prediction rate due to ineffective feature extraction. To avoid such negative impacts on accuracy rate, this research aims to propose suitable DL-based feature extraction and ML-based classification with hyperparameter tuning based on the below objectives.

The main objective of this study is,

- To select relevant features using the proposed Deep CNN (Deep Convolutional Neural Network) and Inception V3 for all plant classification.

- To perform feature extraction through the proposed Deep CNN and VGG16 (Visual Geometry Group16) for the piper plant classification.
- To classify piper and all plant species through the proposed WUT-RF (Weight Updated Tuned Random Forest) based on hyperparameter tuning with convolutional likelihood weight for attaining a high prediction rate.
- To evaluate the proposed system about performance metrics for confirming its effectiveness in predicting piper and all plant species.

1.1. Paper Organization

The paper is organized in the following way, with Section II discussing the conventional works for plant species recognition. Section III discusses the overall proposed system with the proper flow, algorithm, and description, Section IV states the results of the proposed system with dataset and performance metrics description, and Section V summarizes the overall study with future directions.

2. REVIEW OF EXISTING WORK

Existing researchers have attempted to use different approaches for determining plant species. These methods are discussed in this section with relevant problems.

The current study has addressed the solution for better identification of a medical herb (Piper vines) through the DBN (Deep Belief Network) network. The suggested system has been implemented on the worldwide availability of the Kerala-plants dataset, and various Piper vine images have also been collected (Pravin & Deepa, 2021). The performance of DL has exposed optimizing outcomes in the computerized vision in the current years. Thus, the endorsed work has presented a DL technique for classifying and identifying microscopic fragment pictures of the medical herb- 'Simplicia grass' through CNN (Convolutional Neutral Network), improved by the SIFT feature removal, denoted as 'MikrobatX,' that plays an important role in a microscopic categorization of medical herb 'Simplicia grass.' Crucial Simplicia grass features could be extracted by the MikrobatX using microscopic pictures of medical herb leaves. Using the MikrobatX dataset, the results of experiments have shown that the recommended model can produce satisfactory accuracy of 89.42% value for the microscopic medical grass Simplicia image issues (Rahmatulloh & Suhendy, 2022). The CNN performance and pre-trained models (VGG19 and VGG16) have been compared for the problem of leaf identification. The dataset considered in the research works contains Kerala's indigenous medical herbs. CNN has obtained a classification rate of 95.79%. VGG19 and VGG16 have procured 97.6% and 97.8% accuracy, respectively, which has been better than conventional CNN (Paulson & Ravishankar, 2020). Furthermore, with the transfer learning concept, ten pre-trained networks containing GoogLeNet, Alexnet, DenseNet201, Mobilenetv2, Inceptionv3, Resnet101, Resnet50, Resnet18, VGG19, and VGG16 have been utilized as a feature extractor. The FP (False Positive) rate of 0.1% is less than attained in all the cases(Oppong, Twum, Hayfron-Acquah, & Missah, 2022).

Conventional research has also exposed new cases of medical herb dataset termed DeepHerb dataset containing 2515 leaf pictures from 40 species of Indian plants. The efficiency of a dataset has been exposed by comparing it with pre-trained deep CNN structures such as VGG19, VGG16, Xception, and InceptiveV3. The work has been focused on accepting the transfer learning method on the suggested pre-trained model to undertake feature extraction, and classification has been performed using ANN (Artificial Neutral Network) and SVM (Support Vector Machine). The hyperparameters of SVM have been tuned additionally through the optimization of Bayesian to attain a better performance model. The recommended DeepHerb model studied from ANN and Xception outperformed by an accuracy of 97.5% (Roopashree & Anitha, 2021). The current study possesses images of leaves encompassing nine various herbs with 32 various categories of the PlantVillage database. Besides, CNN has been used to identify a plant leaf. Transfer learning has been used for a pre-trained network of AlexNet for several amounts of information for network training, and the outcomes have been evaluated with SVM and DL categorizers. With 91.15% accuracy, AlexNet has performed better than SVM, providing 89.69% and 88.96% for circular basis linear kernel and functional kernel, respectively (Wagle, 2021).

Additionally, the current study has recommended a CNN-based process denoted as D-Leaf. After pre-processing the leaf images, the features have been extracted using three CNN models, specifically Fine-tuned AlexNet, D-Leaf, and pre-trained AlexNet. The features were then categorized using five ML methods, specifically ANN, SVM, CNN, K-NN (K-Nearest Neighbour), and NB (Naïve Bayes). The D-Leaf model has accomplished comparative testing of 94.88% accuracy compared to 93.26% (AlexNet) and 95.54% (fine-tuned Alexnet) models.

The current study has concluded that compared to CMM (Conventional Morphological Method), CNN has been better for retrieving features of herb species (Wei Tan, Chang, Abdul-Kareem, Yap, & Yong, 2018). Besides, Rtsd-net (Real-Time Strawberry Detection) has been performed through DNN (Deep Neural Networks) on the embedded system, which has exposed satisfactory performance (Zhang et al., 2022). To enhance prediction performance, the research by (Dönmez, 2022) has suggested a model for classifying haploid and diploid maize seeds. Initially, deep features have been attained from CNN. These features have been selected and fused by varying CNN model combinations. Such integrated features were then utilized in traditional classifier methodology training and testing stages. By empirical outcomes, it has been found that the performance rate has been about 96.74%. In addition, crop species and weeds have been classified using textual features of RGB images compared to the SVM and DL-based VGG16 model. A feature selection approach, namely ReliefF, has been employed for selecting suitable features to perform prediction (Sunil et al., 2022). Performance metrics, namely kappa score, F1-score, and accuracy, have been considered to assess the suggested model's reliability and performance.

Further, PLS (Partial Least Square) regression (Goyal & Kumar, 2021) has been performed to select suitable features from the retrieved deep feature dataset (Saeed et al., 2021). The recommended framework has included three major stages. At the initial stage, deep features have been extracted through a pre-trained VGG19-CNN model (Rajesh & Bhaskari, 2021).

Subsequently, a PLS-based parallel fusion methodology has been endorsed that integrates the features retrieved from FC (Fully Connected) layers. The convenient features chosen through PLS have been incorporated into the ensemble tree classifier. The suggested system has accomplished accuracy at a rate of 90%. As DL algorithms have exposed better performance, the study (Crisóstomo de Castro Filho et al., 2020) has intended to assess methodologies for detecting rice farming in the southern part of Brazil, for which it has used Bi-LSTM and LSTM. Comparison has been performed with conventional ML algorithms, namely SVM, NB, K-NN, and RF (Random Forest). Outcomes show that LSTM and Bi-LSTM models have exposed better performance than traditional ML. Accordingly, the study (Alajrami & Abu-Naser, 2020) has recommended a solution for assisting individuals in determining the tomato kinds. A model has been constructed through the DL-based CNN model that has been trained and tested. The suggested trained model has been used for predicting the kind of tomato images with the suggested network encompassing 4-max pooling and CNN layers. Testing accuracy is 93%. Furthermore, a multiscale CNN has been suggested to recognize plant leaves at multiple scales (Hu, Chen, Yang, Zhang, & Cui, 2018). Experimentations have revealed that the suggested model is better than several conventional plant leaf identification algorithms.

Nevertheless, the research (Wäldchen & Mäder, 2018) has stated that, despite modern ML algorithms exposing slow performance, in the future, a proliferation of these algorithms will be found to solve the issues of predicting plant species. Under this, the study (Feng et al.,

2019) has intended to expose the employment of SVMs and RF in identifying multi-feature crop kinds like spectroscopy, phonological parameters, and vegetation index. Training of the sample and verification of accuracy have been undertaken through the use of ML algorithms encompassing SVM, maximum likelihood methodologies, and RF. Outcomes have been significant. Further, for classifying plant species from occluded images of leaves, the research (Chaudhury & Barron, 2018) has used a dataset comprising varied leaf kinds (M. Kumar, Gupta, Gao, & Singh, 2019). Initially, 2D-contour points have been indicated as β-spline curves. Following this, the DCE (Discrete Contour Evolution) approach has extracted interest points on such curves. Following this, the parameters for similarity transformation have been calculated for an individual open curve. Then, distinct open curves were overlaid with inverse similarity curves, and used the Frechet metric for computing the measure of match quality and retaining ideal η-matched curves. The function has used global curvature, local curvature, string-cut features, and shape-context descriptors. Energy function has been minimized by using a convex and concave relaxation model. Experimentations on three accessible leaf image datasets have exposed the better performance of the suggested model.

2.1. Problem Identification

The main issues that have been identified through the evaluation of the above conventional studies are listed below,

- Only a few studies have endeavored to perform piper plant classification. However, these studies lacked accuracy. For instance, the study (Rahmatulloh & Suhendy, 2022) has exposed 89.42% accuracy while using MikrobatX.
- Conventional works have attempted to employ various algorithms for classifying different kinds of plant species. For instance, the research (Wagle, 2021) has used AlexNet and exposed 91.15% accuracy compared with SVM, which affords 89.69% and 88.96% for linear kernel and RBF (Radial Basis Function). The study (Wei Tan et al., 2018) has exposed 94.88% accuracy while utilizing the D-leaf model. Further, the research (Saeed et al., 2021) used PLS to select features with pre-trained VGG19 and exposed them with 90% accuracy. On the contrary, the study (Alajrami & Abu-Naser, 2020) aimed to classify tomato plants and accomplished a 93% prediction rate using CNNs.
- Suitable feature extraction approaches have to be selected for data pre-processing that could be suitable for the DL model (Wagle, 2021).
- Complex and recently successful DL algorithms could enhance the classifier performance (Chaudhury & Barron, 2018; Sunil et al., 2022).
- Although the suggested model in (Kaur & Kaur, 2019) has exposed better performance, it still requires enhancement in executing DL or NN-based methods. In the future, the existing limitation has to be resolved by retrieving suitable cultivated features and executing an enhanced classifier or hybrid classifier. Lastly, the plant species have to be identified by considering a real-time dataset.

3. PROPOSED METHODOLOGY

The research attempts to classify all plants and piper plant species based on suitable ML and DL-based feature extraction and classification. Though existing studies have tried to perform this prediction, they need to be more accurate. Hence, this study intends to resolve the existing pitfalls by proposing methods, and its sequential processes are shown in Figure 1 and Figure 2.

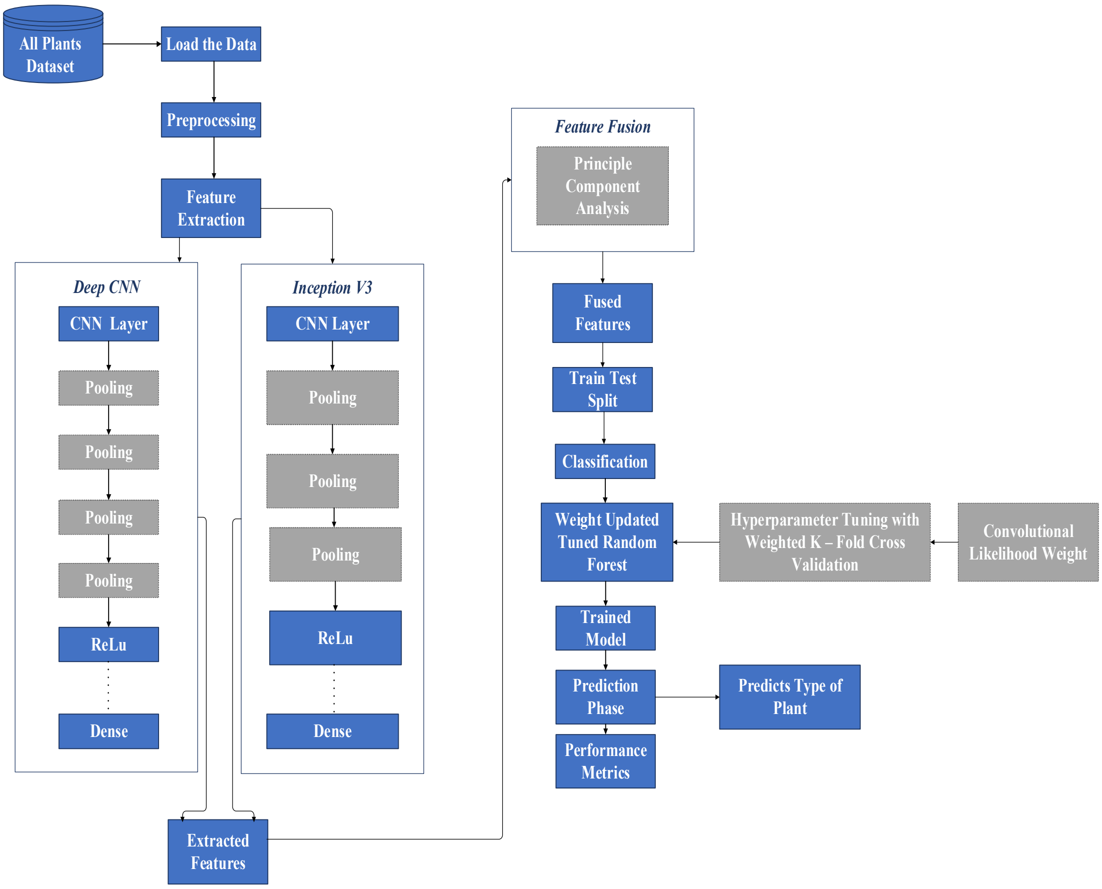

Figure 1. Overall view of the proposed work for all plants classification

As described in Figure 1, the dataset is initially loaded. Following this, pre-processing is performed to eliminate irrelevant and unwanted redundant data to improvise further processing. Then, significant features are extracted by Deep CNN and Inception V3. This phase assists in constructing the proposed model with minimum machine effort and enhances the learning speed and generalization stages in the ML process. Extracted features are then fused using PCA (Principle Component Analysis). This stage tries to extract highly discriminative information from various input features, thereby avoiding redundant information. The fused features are then fed into the train and test split for classification. In this phase, WUT-RF is used for classification, wherein hyperparameter tuning is performed with the convolutional likelihood weight. Finally, the kind of plant is predicted by the trained model, and the efficacy of the proposed system is assessed through performance metrics. Further, the piper plant classification is performed based on the sequential process, as shown in Figure 2.

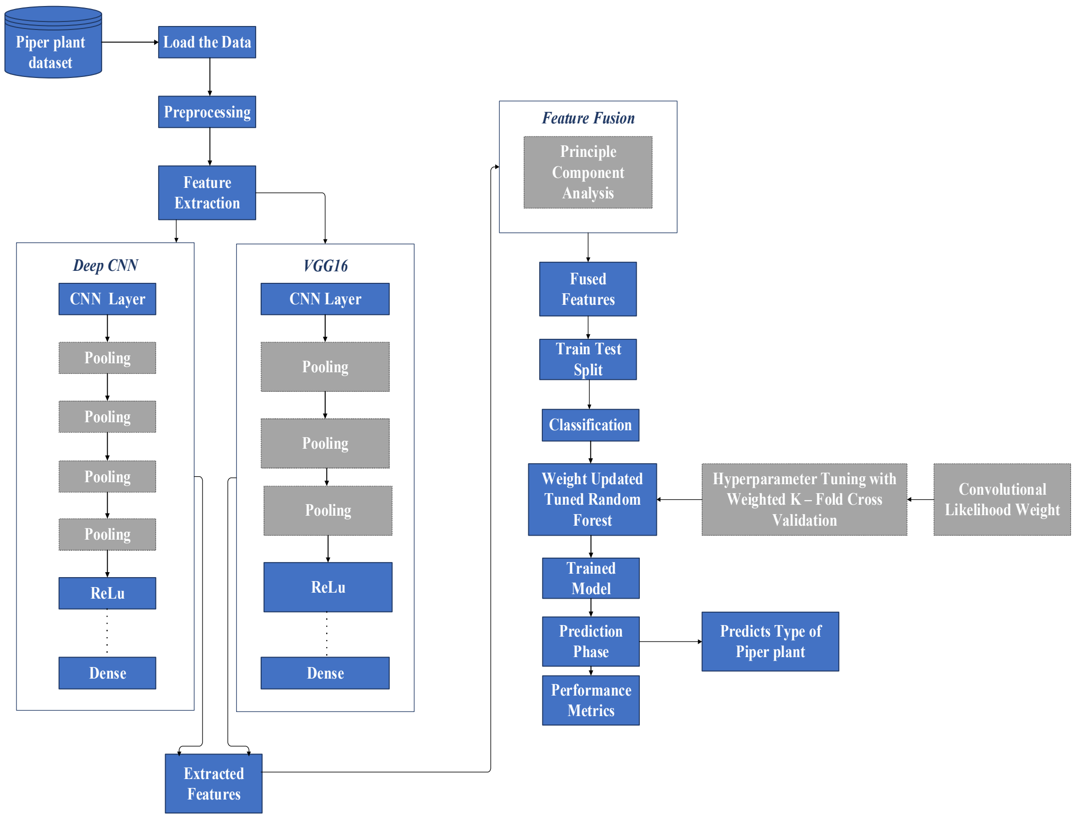

Figure 2. Overall view of the proposed work for piper plants classification

As shown in Figure 2, the dataset is initially loaded, and pre-processing is undertaken to eliminate irrelevant data. Subsequently, feature extraction is performed by Deep CNN and VGG16, wherein relevant features are attained, and then feature fusion is undertaken by PCA. Lastly, piper plant species classification is accomplished by WUT-RF, and the performance of this stage is assessed through performance metrics. The overall flow of the present research is explored in Figure 3.

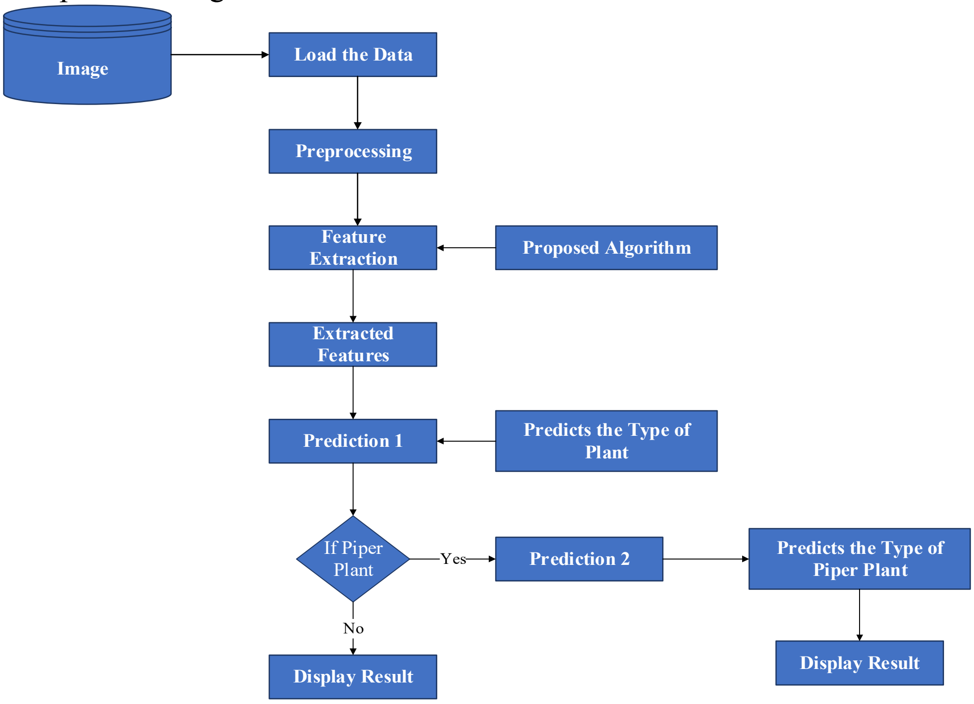

Figure 3. Overall proposed flow

As shown in Figure 3, the dataset is loaded, and pre-processing is performed. Then, feature extraction is undertaken by the proposed algorithm. Based on the extracted features, the type of plant is predicted. For example, if it is a piper plant, then the type of piper plant is predicted. If not, the plant type is displayed as an outcome.

3.1. Feature Extraction-Deep CNN, Inception V3, VGG16

Feature extraction holds significance in accomplishing better classification performance. The main intention of this process lies in determining informative and compact feature sets for improving classifier efficacy. This study proposes Deep CNN and Inception V3 for all plant species classification and Deep CNN with VGG16 for piper plant classification.

3.1.1. Deep CNN (Deep Convolutional Neural Network)

Typically, Deep CNNs are FFNNs (Feed Forward Neural Networks) employed for adjusting the parameters encompassing biases and weights of NN to minimize the cost-functional value. The process involved in extracting image features from deep CNN layers is called Deep feature extraction. In this case, the extracted features are termed as deep-features. Dimension of convolutional layer outcome could be computed by equation.1,

$$\mathrm{Dimension}(\mathrm{Conv}(a,k)) = \left(\frac{a_w - b_w}{s} + 1\right), \left(\frac{a_h - b_h}{s} + 1\right), b_c \tag{1}$$

Input width is given by a_w, and height is represented by b_h for the first convolutional layer. In addition, b_w, b_h, b_c indicate the width, height, and channels for the convolutional layer-kernel filter. An initial max-pooling layer is incorporated to minimize the first convolutional layer result dimensions. Moreover, dimensions corresponding to the max-pooling layer result are computed by equation.2,

$$\mathrm{Dimension}(\mathrm{Pooling}(n,k)) = \left(\frac{a_w - b_w}{s} + 1\right), \left(\frac{a_h - b_h}{s} + 1\right), a_c \tag{2}$$

In equation.2, a_w, a_h, a_c indicates the input's width, height, and channels (n). Whereas b_w, b_h, b_c indicate the width, height, and channels of filter b within the max-pooling layer. The outcome of an initial max-pooling layer is fed as input to the subsequent convolutional layer. Further, ReLU activation is utilized on all convolutional layers. That is performed by equation.3,

$$\mathrm{Relu}(x) = \mathrm{Max}(0,x) \tag{3}$$

Following this, equation.4 denotes the output of neuron (z_j) of the initial dense layer. Moreover, i indicates the inputs of the first dense layer that ranges from (1-512), while j represents the resultant overall layers ranging from (1-2048) and is given by,

$$z_j = \mathrm{ReLu}\left(0, \sum_i^{512} \mathrm{bias}_j + x_i w_i\right) \tag{4}$$

In equation.4, bias_j represents the bias value for the jth node. Moreover, this layer uses the Softmax function for plant leaf classification. Softmax (σ) of the ith neuron of the dense layer is given by equation.5,

$$\mathrm{softmax}\,(\sigma(z_i)) = \frac{e^{z_i}}{\sum_{j=1}^{59} e^{z_i}} \tag{5}$$

The resultant input image class could be exposed through an equation.6,

$$\mathrm{Output\ Class}\ (z_{out}) = \max(z_1, z_2, \dots \dots z_{59}) \tag{6}$$

The resultant value from (z_1 to z_{59}) indicates the overall piper and all plant leaf classes.

3.1.2. Inception V3

Inception V3 indicates a pre-trained CNN image recognition framework. It represents a network version already trained upon several million images from the ImageNet dataset. This model typically uses several methods to optimize the network to attain better adaptation of the model. As a result, it possesses a deep network compared to the Inception-V1 and Inception-V2 models. However, its speed is maintained.

Furthermore, it utilizes auxiliary classifiers and is computationally cheap. Due to this advantage, this study considers Inception V3 for feature extraction. Its overall architecture is shown in Figure 4, encompassing different layers of Inception V3 with suitable dimensions as considered in this research.

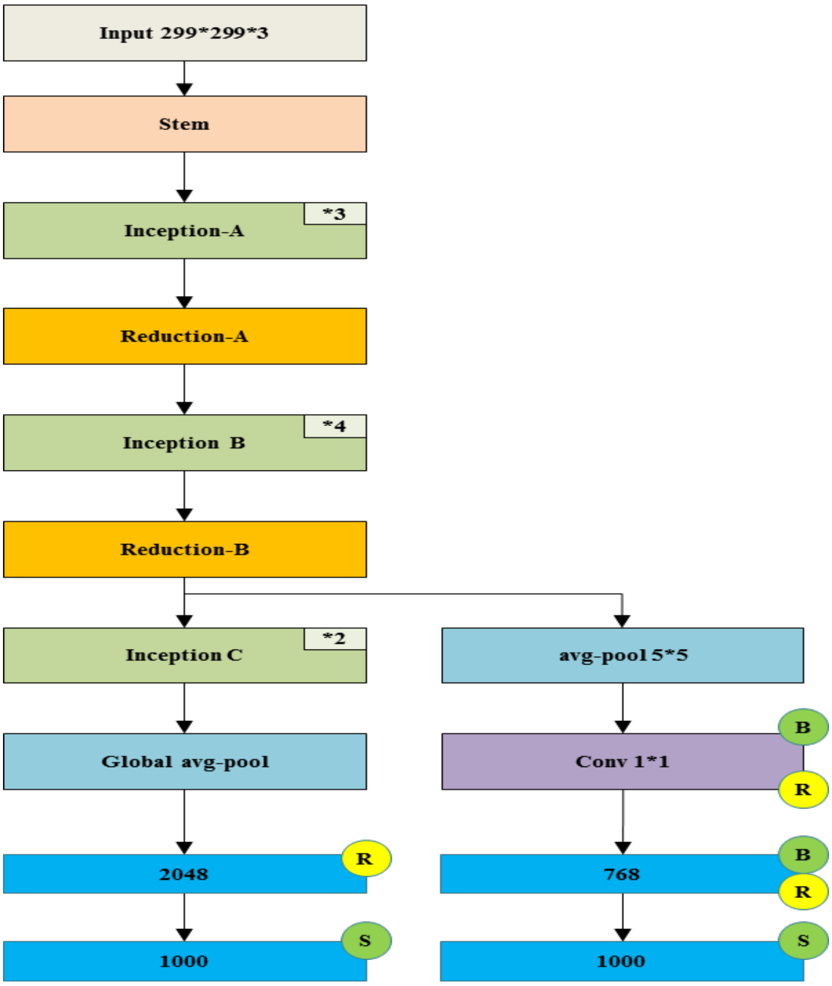

Figure 4. Inception V3 architecture

3.1.3. VGG16 (Visual Geometry Group16)

VGG16 is typically a CNN model comprising 16 convolutional layers. It possesses various benefits, like uniform architecture. Similar to AlexNet, it possesses convolutions of (3*3) dimensions. It could be trained upon 4GPUs for two to three weeks. Currently, it is a preferred option to extract features from an image. It finds applicability in several DL-based image classification issues and remains a building block to perform learning and undertakes easy execution. The main merit of this model is that it is optimal for benchmarking, and pre-trained networks seem publicly accessible. Its overall architecture is depicted in Figure 5.

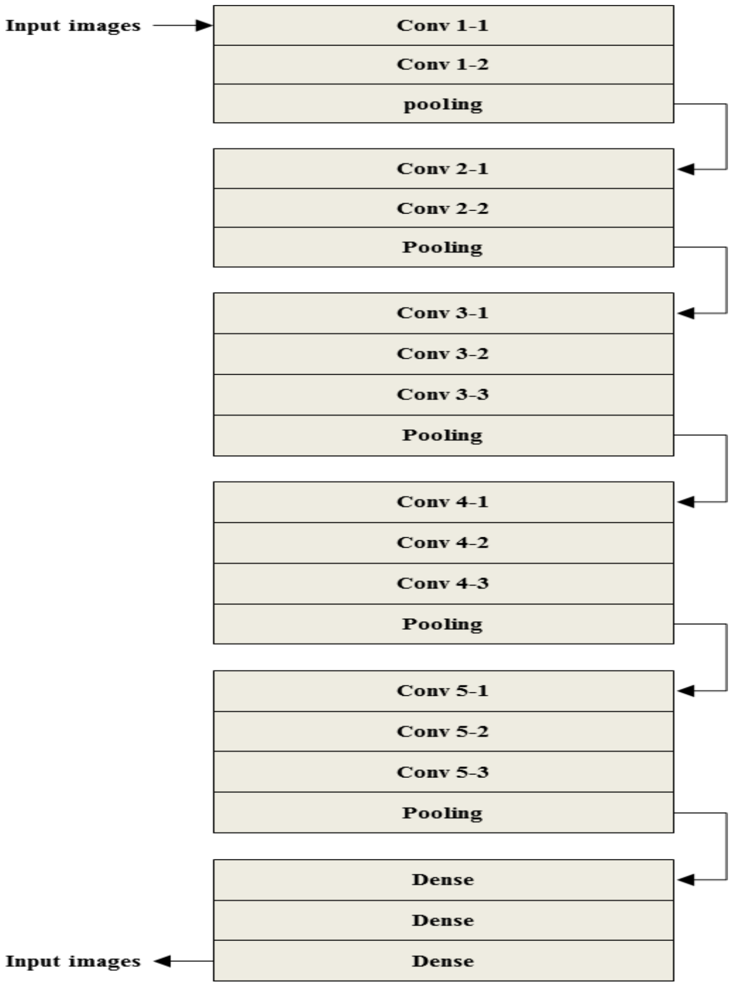

Figure 5. VGG16 model architecture

3.2. Feature Fusion-PCA (Principle Component Analysis)

The present study considers PCA for feature fusion. It is an ML algorithm generally used to minimize feature dimensionality while maintaining significant data properties. Functioning in data with high dimensions could make classification computationally expensive. It might also lead to low performance in comparison with common feature space. Hence, minimizing feature space avoids feature redundancy and affords computational advantages that enhance the classifier's performance. This study selects PCA for feature fusion as it could enhance performance at a low cost. Other benefits include data noise reduction, feature selection, and the ability to produce uncorrelated and independent data features. The overall mathematical formulation of this process is discussed below,

Assume $U = \{u_1, u_2, u_3 \ldots, u_R\}$ represents the actual dataset with size R. Co-variance matrix (C) is given as per equation.7,

$$C = \frac{1}{R}\sum_{m=1}^{R}(u_m - u')(u_m - u')^T \tag{7}$$

In equation.7, u' indicates the mean of actual samples given by equation.8,

$$u' = \frac{1}{R}\sum_{m=1}^{R} u_m \tag{8}$$

Furthermore, Eigenvectors are calculated from the alternation of the covariance matrix (C). Then, it is sorted by the Eigenvalue in descending order as $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_p$ wherein p indicates the eigenvectors. Finally, the overall process is given in Algorithm-I.

Algorithm-I: PCA (Principal Component Analysis)
Step 1: Calculate the mean of the feature vector $\mu = \frac{1}{R}\sum_{k=1}^{R} u_m$, where u_m represents a pattern $(k = 1 \text{ to } p)$, p: number of patterns, x: feature matrix **Step 2:** Determine the covariance matrix $C = \frac{1}{R}\sum_{m=1}^{R}(u_m - u')(u_m - u')^T$ // T: Matrix transposition **Step 3:** Calculate the covariance matrix's Eigenvalues (λ_1) and Eigenvectors (v_i). $Cv_i = \lambda_1 v_i, (i = 1,2,3, \dots f) f = \text{features}$ **Step-4:** Estimate Eigenvectors of high value $(\sum_{i=1}^{s}\lambda_1)(\sum_{i=1}^{q}\lambda_1)^{-1} \geq \theta$, s: number of high valued λ_i selected **Step-5:** Extract features of low dimension from the raw feature-matrix as $P = V^T x$ Where V represents the principal component matrix and x denotes the feature matrix.

As shown in Algorithm-I, the mean of the feature vector is initially computed. Following this, the covariance matrix is computed. Then, Eigen values and Eigen vectors for the corresponding covariance matrix are calculated. Next, eigenvectors are estimated as per Step 4. Finally, the features with low dimensions are extracted from raw features.

3.3. Classification- WUT-RF (Weight Updated Tuned Random Forest)

The present study considers RF as it is capable of affording better performance. It could also deal with big data having numerous variables operating in thousands. It could perform automatic dataset balance while a class seems more infrequent than any other data class. This algorithm includes additional randomness to the model while developing trees. During node split, it searches optimal features within random feature subsets rather than for a significant feature. Hence, it minimizes overfitting issues in DTs (Decision Trees) and alleviates variance, enhancing accuracy. The process while considering RF for the classification process is depicted in Pseudocode-I.

Pseudocode-I: RF (Random Forest)

piper plant images and other **images** in the dataset
A is training set $TS := (a_1, b_1 \dots \dots, (a_n b_n)$
training set(TS) = Piper plants and other plants images
F − feature,
N(t) = no of trees
Forest = E
begin
function RandomForest (G, R)
$I \leftarrow \emptyset$
for i $\in 1, \dots . E$ **do**
$\mathbf{G^{(i)}} \leftarrow$ **A bootstrap sample from TS**
$\mathbf{h_i} \leftarrow$ **Randomized Tree Learn** $(\mathbf{G^{(i)}}, \mathbf{R})$
$\mathbf{I} \leftarrow \mathbf{I} \cup (\mathbf{h_i})$
end for
return I
end function
function RandomizedTreeLearn(G, R)
At each node:
$F \leftarrow$ very small subset of R
split on best feature in F

Return the learned tree
end function
Classification Pseudocode
assume TS = p
Prediction 1
"predicts the type of plants."
if p == 0 then
Display("The plant is identified as piper plant : ")
Prediction 2
"predicts the type of piper plants."
else results are p = !0
Display("The plant is predicted as which type of piper plant : "
Endif
End for
End

In this study, K-fold cross-validation encompasses a dataset split into K partitions. The main notion behind this process is cross-validation with a search algorithm wherein hyperparameters are inputted prior to model training. Then, by integrating random search, the model fits individual pairs of varied hyperparameter sets in individual cross-validation. This overall process is depicted in Pseudocode-IIwhere require indicates the input to be afforded, and RandomSample indicates the function which considers a random set from hyperparameter.

Pseudocode-II:K-fold cross-validation with random search
require K_1, K_2 − K_1number of outer folds and $k_2$inner folds
TS − dataset of plant images with features x and output features y
P_{set} −→ set of hyperparameters with different values
M−→ single model estimator
for i = 1 to K_1 split do
split TS into TS_i^{train}, $TS_i^{test}$for the ith split
for j = 1 to K_2 split do
split TS_i^{train} into TS_j^{train}, $TS_j^{test}$for the jth split
for each p in the random sample( ) **Pseudo**
Train M on TS_j^{train} with hyperparameter set p
Compute test rate $E_j^{test}$for M with TS_j^{test}
Select the optimal hyperparameter set with the convolution$_{weight}$(con) vw fro
where E_j^{test} is best

Through the K-Fold cross-validation with random search, optimal hyperparameters are selected. The overall process involved in hyperparameter tuning with weighted cross-fold validation is given by Pseudocode-III, that finally affords the classification of piper and all plant types. In this process, selecting optimal hyperparameters and tuning assists in enhancing the classifier performance.

Pseudocode-III: RF hyperparameter tuning with Weighted cross fold validation
piper plant images and other **images** in the dataset
A is training set TS := $(a_1, b_1 \ldots \ldots, (a_n b_n)$
training set(TS) = Piper plants and other plants images
k fold cross − validation with the random search
require K_1, K_2 − K_1number of outer folds and $k_2$inner folds
TS − dataset of plant images with features x and output features y
P_{set} −→ set of hyperparameters with different values

M−→ single model estimator
F − feature,
N(t) = no of trees
Forest = E
begin
function RandomForest (G, R)
$I \leftarrow \emptyset$
for $i \in 1, \ldots, E$ **do**
$G^{(i)} \leftarrow$ **A bootstrap sample from TS**
$h_i \leftarrow$ **Randomized Tree Learn** $(G^{(i)}, R)$
$I \leftarrow I \cup (h_i)$
end for
for $i = 1$ to K_1 split do
split TS into TS_i^{train}, TS_i^{test} for the ith split
for $j = 1$ to K_2 split do
split TS_i^{train} into TS_j^{train}, TS_j^{test} for the jth split
return I
end function
function RandomizedTreeLearn(G, R)
At each node:
For each p in the random sample( ) **pseudo**
Train M on TS_j^{train} with hyperparameter set p
Compute test rate E_j^{test} for M with TS_j^{test}
$F \leftarrow$ very small subset of R
split on best feature in F
Return the learned tree
end function
Select the optimal hyperparameter set with the convolution$_{weight}$ ($conv_w$)
where E_j^{test} is best
assume TS with learning tree = p
Prediction 1
"predicts the type of plants."
if p == 0 then
Display("The plant is identified as piper plant : ")
Prediction 2
"predicts the type of piper plants."
else results are p = ! 0
Display("The plant is predicted as which type of piper plant : ")
Endif
End for
End

4. RESULTS AND DISCUSSION

The outcomes procured through the execution of the proposed system are presented in this section with dataset description and performance metrics.

4.1. Dataset Description

Dataset considered in this research includes all plant kinds and piper plant kinds. The normal plant kind dataset is depicted in Table 1, while the specific piper plant type dataset is shown in Table 2. For the normal plant type dataset, different kinds of plants exist, which include piper, peach, apple, strawberry, etc., as shown in Table 1, with a total of 1607 images.

Table 1. All plant dataset

Class	Plant Classes	Image Count
1	Piper Plant	172
2	Apple Plant	260
3	Peach Plant	180
4	Strawberry Plant	200
5	Cherry Plant	160
6	Grape Plant	130
7	Corn Plant	151
8	Paddy Plant	150
9	Rice Plant	209
10	Soya Plant	160
11	Potato Plant	152
12	Tomato Plant	132

Furthermore, for the piper plant dataset, various piper plant kinds are regarded as shown in Table 2, namely Piper Mullesa, Piper Adunucum, Piper Argyrites, etc., with 2056 images.

Table 2. Piper Plant dataset

Class	Piper Classes	Image Count
1	Piper Mullesa	102
2	Piper Nigrum	100
3	Piper Adunucum	100
4	Piper Argyrites	101
5	Piper Umbellatum	103
6	Piper Excelsum	107
7	Piper Parmatum	103
8	Piper Orantum	102
9	Piper porphyrophyllum	102
10	Piper sylvaticum	102
11	Piper longum	100
12	Piper Auritum	102
13	Pepper Bell	180
14	Piper Sarmentosum	103
15	Piper betle	100

Moreover, the sample images for all plants, including a piper and other plant kinds, are exposed in Figure 6, while the piper plant leaf dataset is depicted in Figure 7.

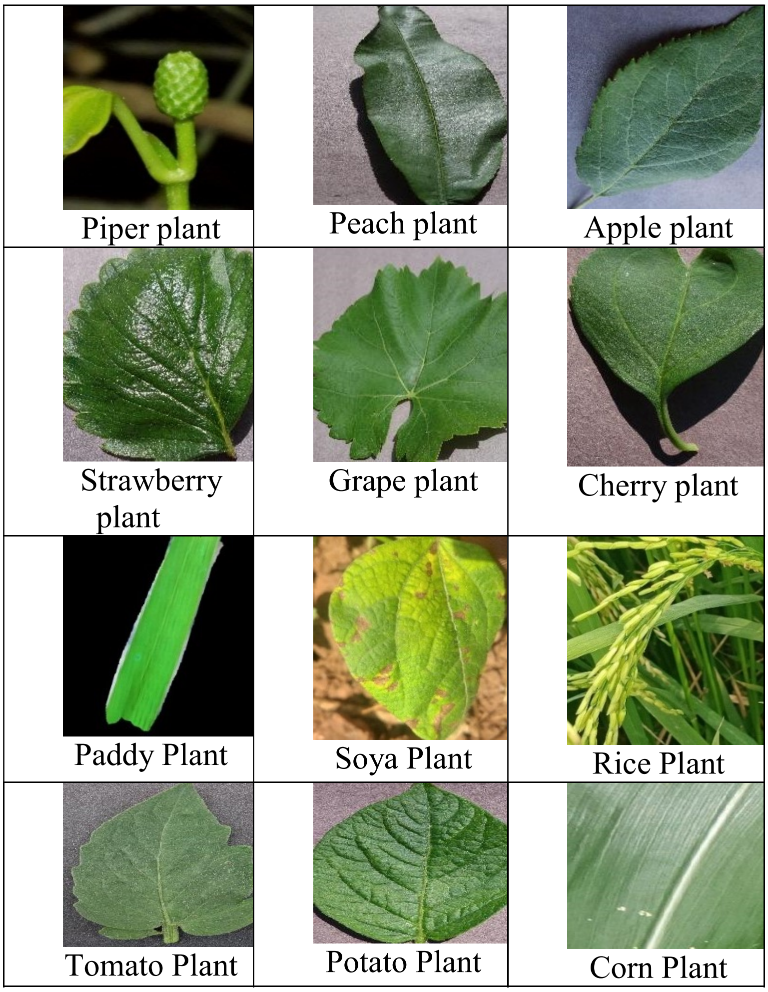

Figure 6. All plants dataset samples

In addition, the samples of the piper plant dataset are explored in Figure 7.

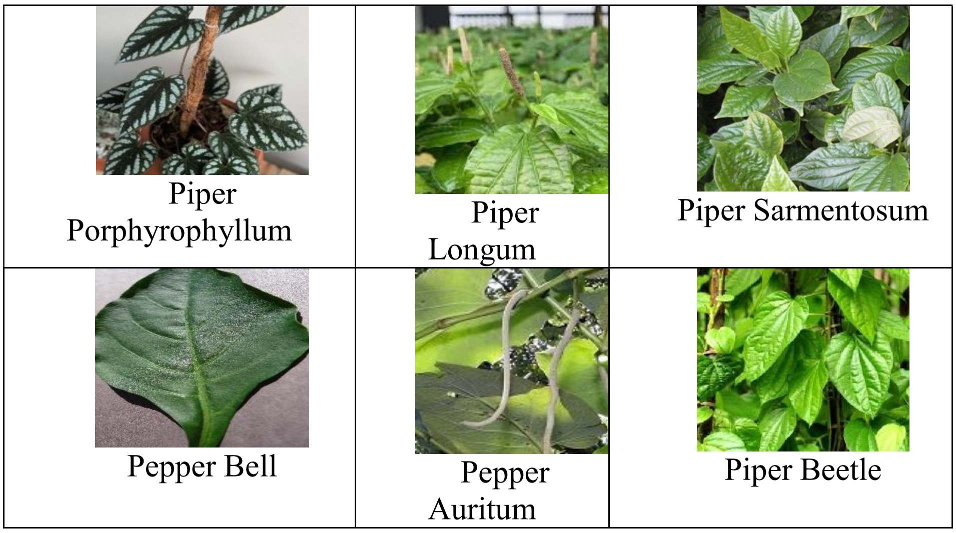

Figure 7. Piper plant dataset samples

4.2. Performance Metrics

The metrics regarded in this study for validating the performance of the proposed system are,

i) Accuracy

It is described as the calculation of overall correct classification. It is indicated by equation.9,

$$\text{Accuracy} = \frac{Tr_{Ne} + Tr_{Po}}{Tr_{Po} + Tr_{Ne} + Fa_{Po} + Fa_{Ne}} \qquad (9)$$

ii) Precision

It is claimed as the correct classification calculation and is represented per equation.10,

$$\text{Precision} = \frac{Tr_{Po}}{Tr_{Po} + Fa_{Po}} \qquad (10)$$

As represented in equation.9 and equation.10, Tr_{Po} indicates True Positive, Fa_{Po} is False Positive, Tr_{Ne} denotes True Negative, and Fa_{Ne} indicates False Negative.

iii) Recall

It indicates the proportion of retrieved and relevant images to the proportion of the relevant image. It is given by equation.11,

$$\text{Recall} = \frac{Rel_{Image} \cap Ret_{Image}}{Rel_{Image}} \qquad (11)$$

As shown in the equation.11, Rel_{Image} represents the relevant image, while, Ret_{Image} denotes the retrieved image.

iv) F-measure

It is also termed F1-score. It could be claimed as the harmonic mean of Recall and Precision. It is represented by equation.12,

$$F - \text{measure} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \qquad (12)$$

4.3. Experimental results

The outcomes that have been attained through the execution of the proposed system are discussed in this section. Initially, the image is selected for prediction, as shown in Figure 8.

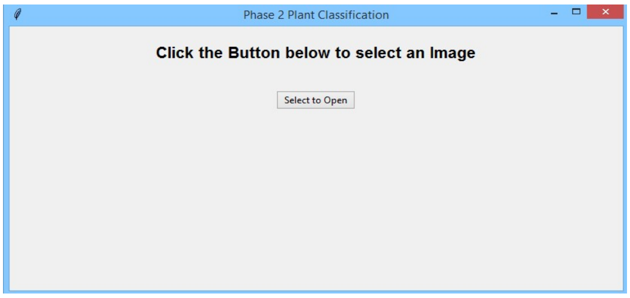

Figure 8. Selection of Image

Based on the selected image, the plant species are classified as all plants with suitable species, as shown in Figure 9.

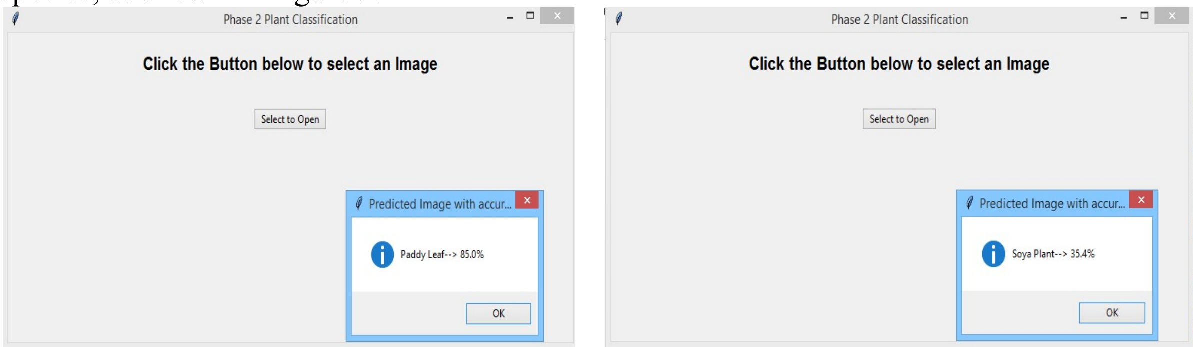

Figure 9. Plant type classification

As shown in Figure 9, the paddy leaf is predicted at a rate of 85%, while the Soya plant is identified at 35.4%. Similarly, the piper plants are also classified by the image taken as input, and the results are depicted in Figure 10.

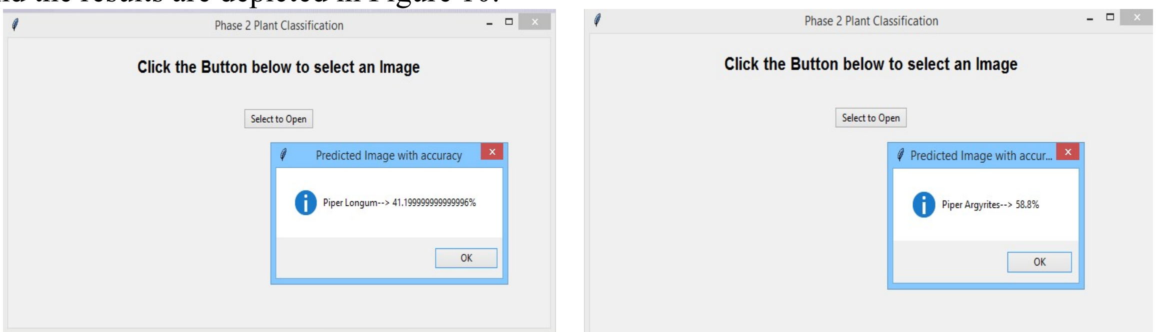

Figure 10. Piper plant type classification

As shown in Figure 10, the Piper Longum is predicted at a rate of 41.79%, while Piper Argyrites is identified at 58.8%.

4.4. Performance Analysis

The performance of the proposed system is evaluated, and the corresponding outcomes are discussed in this section. The accuracy score for piper plant classification is presented in Table 3.

Table 3. Accuracy for the piper plant classification

	ACCURACY SCORE
PIPER PLANT	98.82591093

From Table 3, it is found that the accuracy score for classifying piper plants is 98.825%. Further, the classification performance for the piper plant's performance metrics is tabulated in Table 4.

Table 4. Classification performance for piper plants

Classes	Precision	Recall	F1-score	Support
0	0.99	1	0.99	276
1	1	0.94	0.97	573
2	0.99	0.99	0.99	321
3	1	1	1	540
4	1	1	1	321
5	1	1	1	306
6	1	1	1	285
7	0.95	0.97	0.96	299
8	0.97	0.99	0.98	296
9	0.99	0.99	0.99	301
10	0.99	0.98	0.99	306
11	1	1	1	252
12	1	1	1	294
13	0.98	0.98	0.98	303
14	0.93	1	0.96	267

From Table 4, it is exposed that the proposed system's precision, recall, F1-score, and support rate have shown better performance in the range of 0.93-0.99. Further, the overall average accuracy for piper plant classification is exposed in Figure 11.

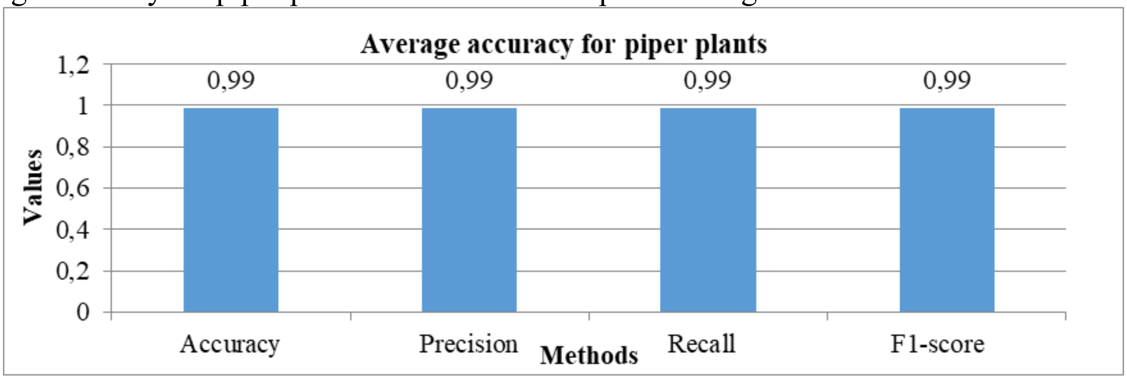

Figure 11. Average accuracy rate for classifying piper plants

Figure 11 shows that the accuracy rate is exposed to be 0.99, while the precision, recall, and F1-score rate is explored to be 0.99.The accuracy score for all plant species classifications is given in Table 5.

Table 5. Accuracy for all plant classification

	ACCURACY SCORE
ALL PLANT	97.14975117

From Table 5, it is found that the accuracy score for classifying all plants is 97.149%. Further, classification performance for all plants about performance metrics is tabulated in Table-6.

Table 6. Classification performance for all plants

Classes	Precision	Recall	F1-score	Support
0	1	1	1	780
1	1	1	1	480
2	0.99	1	1	453
3	1	1	1	390
4	1	1	1	450
5	1	0.99	1	540
6	1	1	1	780
7	1	0.99	1	456
8	1	1	1	627
9	1	1	1	480
10	0.77	0.98	0.86	600
11	0.97	0.7	0.81	595

Table 6 shows that the precision, recall, F1-score, and support rate of the proposed system have shown better performance in the range of 0.7-0.99. Further, the overall average accuracy for all plant classifications is exposed in Figure 12.

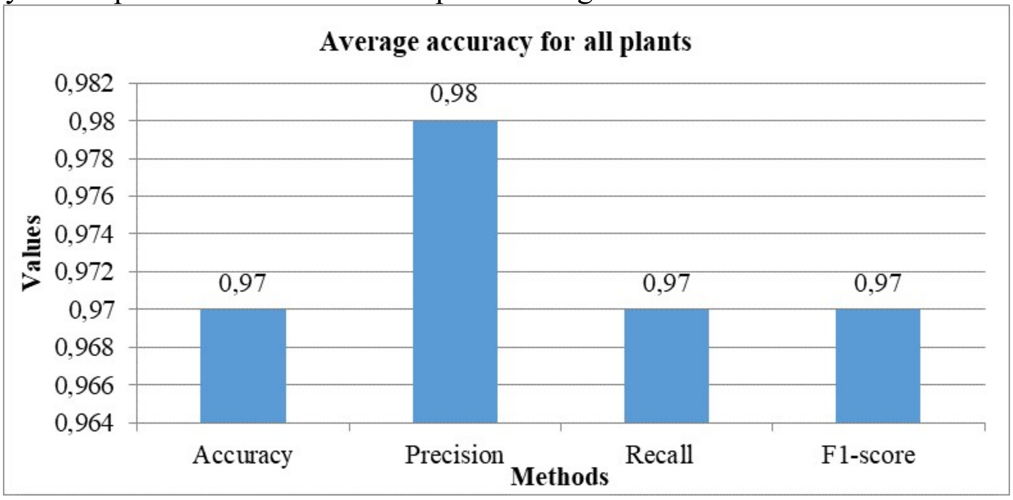

Figure 12. Average accuracy rate for classifying all plants

Figure 12 shows that the accuracy rate is exposed to be 0.97, while the precision, recall, and F1-score rate is explored to be 0.98, 0.97, and 0.97. Additionally, test analyses for all plant types are exposed in Table 7.

Table 7. Test Analysis

Test-analysis	Range	K-values
Landis and Koch	Substantial	0.96
Fleiss	Intermediate to Good	0.87
Cicchetti	Good	0.94
Cramer	Strong	0.86
Matthews	Moderate	0.86
Scott-PI	Perfect Agreement	0.8812

Table 7 shows that the proposed system has a higher K-value of 0.96 for the Landis and Koch test than other test analyses of Fleiss exposing 0.87, Cramer exploring 0.86, Scott-PI showing 0.8812, Matthews exposing 0.86 and Cicchetti exploring 0.94 as k-value. Furthermore, a confusion matrix is procured for the piper plant and all plant classifications.

This assists in defining the classifier performance. It also affords information regarding classification errors. Finally, it reflects how the classifier needs to be more disordered and clear in making predictions. Accordingly, the confusion matrix of piper plant classification is exposed in Figure 13.

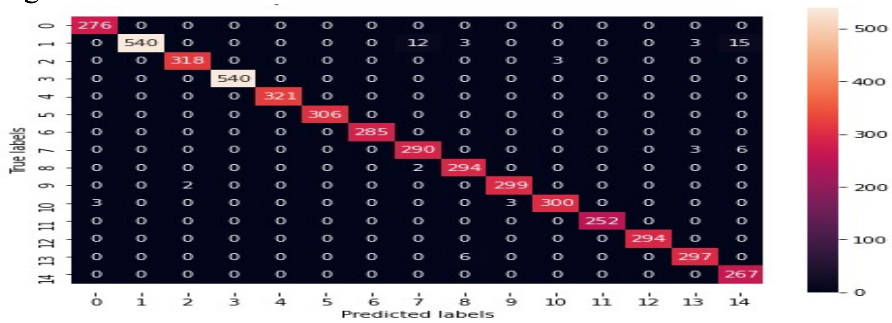

Figure 13. Confusion matrix of piper plant classification

From Figure 13, it is exposed that the correct classification rate is found to be high than the misclassification rate. In this case, the correct classifications could be seen in a diagonal. Therefore, the proposed system is exposed to be better due to the high rate of correct classification than the misclassification rate. In addition, the confusion matrix of all plant classifications is exposed in Figure 14.

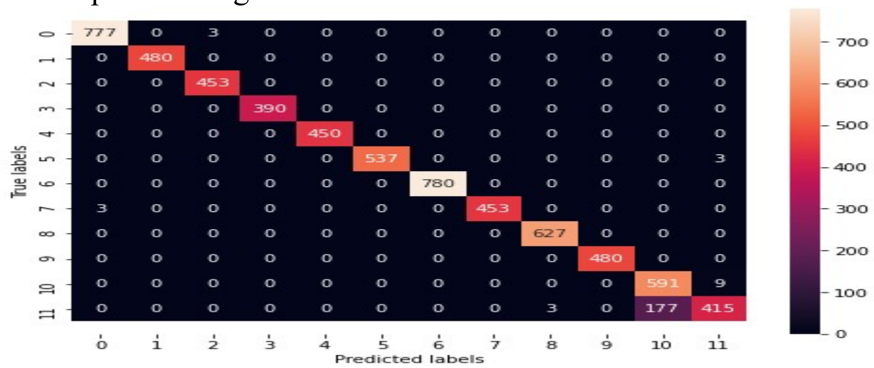

Figure 14. Confusion matrix of all plant classification

From Figure 14, the correct classification rate is higher than the misclassification rate. In such cases, the classifications performed correctly could be seen in diagonal. Due to such a high rate of correct classification in comparison to the misclassification rate, the proposed system is found to be better.

4.5. Internal Comparison

The present research has considered a real-time dataset; thus, it is impossible to compare the present study with conventional works. However, the proposed system is internally compared during the implementation phase, and the corresponding outcomes that have been attained are exposed in this section. Initially, the comparison was undertaken for piper plant classification. Therefore, existing algorithms, namely XGBoost, SVM, and RF, have been regarded in this case. The attained results are exposed in Table 8, with its graphical depiction in Figure 15.

Table 8. Analysis of piper plant classification about performance metrics

Piper plants	Accuracy	Precision	Recall	F1-score
XGBoost	0.56	0.56	0.56	0.57
Support vector classifier	0.24	0.24	0.231	0.24
Random forest	0.65	0.65	0.64	0.64
Proposed algorithm	**0.99**	**0.99**	**0.99**	**0.99**

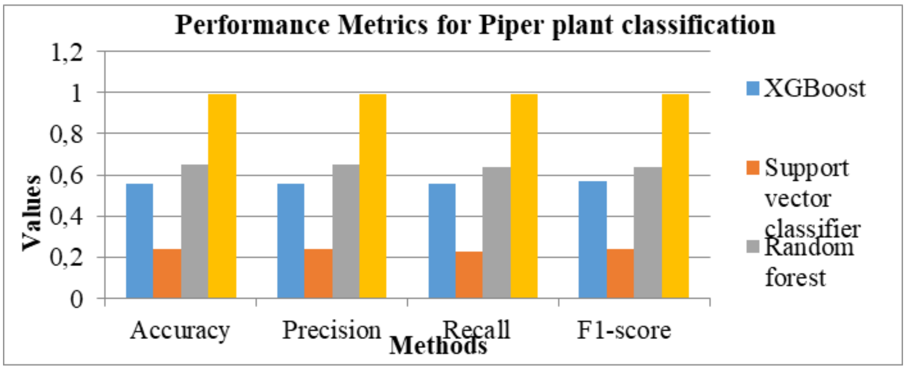

Figure 15. Analysis of piper plant classification

Table 8 and Figure 15 show that existing algorithms like RF have exposed 0.65 as accuracy, while SVM has explored 0.24 and XGBoost has revealed 0.56. Similarly, the precision, recall, and F1-score rate have been different for the considered algorithms. However, the proposed system has explored high classification performance for piper plant classification by exposing 0.99 as the accuracy rate, 0.99 as the precision rate, 0.99 as the recall rate, and 0.99 as the F1-score rate. Similarly, the comparison has been undertaken for all plant classifications. Again, existing algorithms, namely XGBoost, SVM, and RF, have been regarded. The attained results are exposed in Table 9, with its graphical depiction in Figure 16.

Table-9. Analysis of all plant classifications about performance metrics

All plants	Accuracy	Precision	Recall	F1-score
XGBoost	0.75	0.75	0.75	0.78
Support vector classifier	0.52	0.52	0.53	0.53
Random forest	0.8	0.81	0.81	0.81
Proposed algorithm	0.97	0.98	0.97	0.97

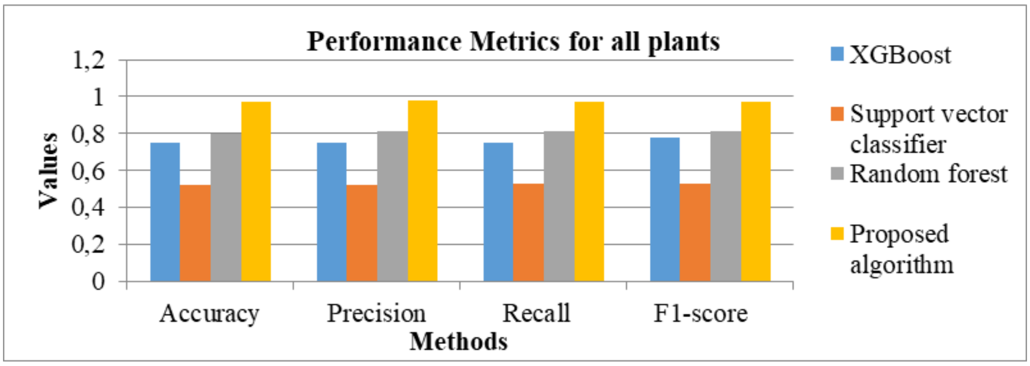

Figure 16. Analysis of all plant classification

Table 9 and Figure 16 show that existing algorithms like RF have exposed 0.8 as accuracy, while SVM has explored 0.52 and XGBoost has revealed 0.75. Similarly, the precision, recall, and F1-score rate have been different for the considered algorithms. However, the proposed system has explored high classification performance for all plant classifications by showing 0.97 as the accuracy rate, 0.98 as the precision rate, 0.97 as the recall rate, and 0.97 as the F1-score rate. Furthermore, the proposed Deep CNN can adjust the parameters to minimize the cost-functional value. Furthermore, inception V3 can perform network optimization to adapt the model better, while VGG16 is easy to execute. In addition, PCA can reduce data noise and select suitable features for affording uncorrelated and independent data features. Lastly, the proposed classifier is capable of minimizing overfitting issues. Such advantages have made the proposed system expose better performance in plant species prediction.

5. CONCLUSION

The research intended to recognize plant species, including a piper and all plant kinds, by considering suitable ML and DL-based algorithms. To achieve this, the study proposed Deep CNN and Inception V3 for extracting the features to classify all plant species. At the same time, Deep CNN and VGG16 extract relevant features to classify piper plant species. Further, for feature fusion, PCA was utilized. At the same time, WUT-RF was proposed for classification that relied on tuning the hyperparameters of RF with convolutional likelihood weight for obtaining a high prediction rate. Finally, a test analysis assessed the performance of the proposed approach. The analysis exposed that the proposed system revealed a higher K-value of 0.96 for the Landis and Koch test than other test analyses.

Additionally, a confusion matrix was attained. The matrix found that the proposed system exposed high classification rates for different classes compared to misclassification for classifying piper and all plant types. As this study regarded a real-time dataset, it was impossible to compare with conventional studies. However, the comparison was internally undertaken with existing algorithms like XGBoost, SVM, and RF. The outcomes found that the proposed system showed a high classification rate for piper plant prediction at a rate of 0.99, while it showed 0.97 for classifying all plant kinds. Though better results were attained, there is a scope for future enhancement. In the future, the performance of the proposed models can be evaluated for various agricultural applications like plant disease diagnosis. Another dimension involves undertaking experimentations when numerous datasets become publicly accessible.

REFERENCES

1. Alajrami, M. A., & Abu-Naser, S. S. (2020). Type of tomato classification using deep learning. *International Journal of Academic Pedagogical Research (IJAPR),* **3**(12).
2. Anubha Pearline, S., Sathiesh Kumar, V., & Harini, S. (2019). A study on plant recognition using conventional image processing and deep learning approaches. *Journal of Intelligent & Fuzzy Systems,* **36**(3), 1997-2004.
3. Chaki, J., Parekh, R., & Bhattacharya, S. (2020). Plant leaf classification using multiple descriptors: A hierarchical approach. *Journal of King Saud University-Computer and Information Sciences,* **32**(10), 1158-1172. https://doi.org/10.1016/j.jksuci.2018.01.007
4. Chaudhury, A., & Barron, J. L. (2018). Plant species identification from occluded leaf images. *IEEE/ACM transactions on computational biology and bioinformatics,* **17**(3), 1042-1055.
5. Crisóstomo de Castro Filho, H., Abílio de Carvalho Júnior, O., Ferreira de Carvalho, O. L., Pozzobon de Bem, P., et al. (2020). Rice crop detection using LSTM, Bi-LSTM, and machine learning models from sentinel-1 time series. *Remote Sensing,* **12**(16), 2655.
6. Dönmez, E. (2022). Enhancing classification capacity of CNN models with deep feature selection and fusion: A case study on maize seed classification. *Data & Knowledge Engineering,* **141**, 102075.
7. Feng, S., Zhao, J., Liu, T., Zhang, H., Zhang, Z., et al. (2019). Crop type identification and mapping using machine learning algorithms and sentinel-2 time series data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing,* **12**(9), 3295-3306.

8. Ganesan, S. (2021). Automated Plant Disease Identification and Detection with Multi-features. *Proc. of Sustainable Communication Networks and Application,* Erode, India, 331–338.
9. Goyal, N., & Kumar, N. (2021). On solving leaf classification using linear regression. *Multimedia Tools and Applications,* **80**(3), 4533–4551.
10. Hao, Z., Lin, L., Post, C. J., Mikhailova, E. A., Li, M., et al. (2021). Automated tree-crown and height detection in a young forest plantation using mask region-based convolutional neural network (Mask R-CNN). *ISPRS Journal of Photogrammetry and Remote Sensing,* **178**, 112–123.
11. Hu, J., Chen, Z., Yang, M., Zhang, R., & Cui, Y. (2018). A multiscale fusion convolutional neural network for plant leaf recognition. *IEEE Signal Processing Letters,* **25**(6), 853–857.
12. Kala, J. R., & Viriri, S. (2018). Plant specie classification using sinuosity coefficients of leaves. *Image Analysis & Stereology,* **37**(2), 119–126.
13. Kaur, S., & Kaur, P. (2019). Plant species identification based on plant leaf using computer vision and machine learning techniques. *Journal of Multimedia Information System,* **6**(2), 49–60.
14. Kumar, M., Gupta, S., Gao, X.-Z., & Singh, A. (2019). Plant species recognition using morphological features and adaptive boosting methodology. *IEEE Access,* **7**, 163912–163918.
15. Kumar, N. S., Yashwanth, N., Srinivasulu, D., & Rini, R. C. (2022). Plant leaf disease diagnosis using artificial intelligence. *International Research Journal of Modernization in Engineering Technology and Science*, **4**(5), 2660–2663.
16. Liu, K.-H., Yang, M.-H., Huang, S.-T., & Lin, C. (2022). Plant species classification based on hyperspectral imaging via a lightweight convolutional neural network model. *Frontiers in Plant Science,* **13**, 1–17.
17. Oppong, S. O., Twum, F., Hayfron-Acquah, J. B., & Missah, Y. M. (2022). A Novel Computer Vision Model for Medicinal Plant Identification Using Log-Gabor Filters and Deep Learning Algorithms. *Computational Intelligence and Neuroscience,* **2022**, 1–21.
18. Paulson, A., & Ravishankar, S. (2020). AI Based Indigenous Medicinal Plant Identification. *Proc. of 2020 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA)*, Cochin, India (57-63).
19. Pravin, A., & Deepa, C. (2021). A Identification of Piper Plant Species Based on Deep Learning Networks. *Turkish Journal of Computer and Mathematics Education (TURCOMAT),* **12**(10), 6740–6749.
20. Pravin, A., & Deepa, C. (2022). Piper Plant Classification using Deep CNN Feature Extraction and Hyperparameter Tuned Random Forest Classification. *Transdisciplinary Journal of Engineering & Science,* **13**, 1–1.
21. Rahmatulloh, A., & Suhendy, H. (2022). MikrobatX: Deep Learning Approach for Microscopic Identification and Classification of Medicinal Leaf Simplicia Fragments Using Sift Feature Extraction. *SSRN Electronic Journal*.
22. Rajesh, K., & Bhaskari, D. L. (2021). Optimum Dataset Size for Ayurvedic Plant Leaf Recognition Using Convolution Neural Networks. *Proc. of Data Management, Analytics and Innovation*, Kolkata, India, 249–261.
23. Roopashree, S., & Anitha, J. (2021). DeepHerb: A Vision Based System for Medicinal Plants Using Xception Features. *IEEE Access,* **9**, 135927–135941.

24. Saeed, F., Khan, M. A., Sharif, M., Mittal, M., Goyal, L. M., & Roy, S. (2021). Deep neural network features fusion and selection based on PLS regression with an application for crops diseases classification. *Applied Soft Computing,* **103**, 107164.
25. Sunil, G., Zhang, Y., Koparan, C., Ahmed, M. R., Howatt, K., et al. (2022). Weed and crop species classification using computer vision and deep learning technologies in greenhouse conditions. *Journal of Agriculture and Food Research,* **9**, 100325.
26. Wagle, S. A. (2021). Comparison of Plant Leaf Classification Using Modified AlexNet and Support Vector Machine. *Traitement du Signal,* **38**(1).
27. Wäldchen, J., & Mäder, P. (2018). Machine learning for image based species identification. *Methods in Ecology and Evolution,* **9**(11), 2216–2225.
28. Wang, B., & Wang, D. (2019). Plant leaves classification: A few-shot learning method based on siamese network. *IEEE Access,* **7**, 151754–151763.
29. Wei Tan, J., Chang, S.-W., Abdul-Kareem, S., Yap, H. J., & Yong, K.-T. (2018). Deep learning for plant species classification using leaf vein morphometric. *IEEE/ACM transactions on computational biology and bioinformatics,* **17**(1), 82–90.
30. Yang, K., Zhong, W., & Li, F. (2020). Leaf segmentation and classification with a complicated background using deep learning. *Agronomy,* **10**(11), 1721.
31. Zhang, Y., Yu, J., Chen, Y., Yang, W., Zhang, W., & He, Y. (2022). Real-time strawberry detection using deep neural networks on embedded system (rtsd-net): An edge AI application. *Computers and Electronics in Agriculture,* **192**, 106586.

