Real-Time Plant Species Recognition Using Non-averaged DenseNet-169 Deep Learning Paradigm

V. Sathiesh Kumar[1(✉)] and S. Anubha Pearline[2]

[1] Department of Electronics Engineering, MIT Campus, Anna University, Chennai, India
sathieshkumar@annauniv.edu

[2] Department of Computer Science Engineering, SRM Institute of Science and Technology, Chennai, India
anubhapearl@gmail.com

Abstract. Real-time plant species recognition under unconstrained environment (viewpoint variation, changing background, scale variation, illumination changes etc.) is a challenging and time-consuming process. In this paper, a non-averaged DenseNet-169 (NADenseNet-169) CNN architecture is proposed and demonstrated to perform real-time plant species recognition. The architecture is evaluated on two datasets namely, Flavia (Standard) and Leaf-12 (custom created). The hyperparameters (optimizers, learning rate) are optimized to achieve higher performance metrics with lower computation time. From the experimental investigation, it is observed that Adam optimizer with a learning rate of 0.0001 (Batch size of 32) resulted in obtaining higher performance metrics. In case of Flavia dataset, an accuracy of 98.58% is obtained with a computational time of 3.53 s. For Leaf-12 dataset, an accuracy of 99% is obtained with a computational time of 4.45 s. The model trained on Leaf-12 dataset performed better in identifying the plant species under unconstrained environment.

Keywords: Plant species recognition · Deep learning architectures · Transfer Learning · DenseNet architecture

1 Introduction

Real-time plant species recognition through computer vision is challenging, considering the large diversity in plant species [1]. Also, the recognition accuracy is significantly affected by many factors such as camera viewpoint variation or changes in object orientation, scale changes, illumination variation, structure of leaf (simple or compound leaf), color variation due to aging or seasonal changes, arrangement of leaf in the stem and cluttered background. Conventional methods involve complex processes such as preprocessing of the input data, segmentation or region selection, feature extraction and classification. This method results in

D. Gupta et al. (Eds.): CVIP 2022, CCIS 1777, pp. 58–72, 2023.
https://doi.org/10.1007/978-3-031-31417-9_5

achieving moderate prediction accuracy with higher computation time. Introduction of deep learning method in plant species recognition, resulted in obtaining the state-of-the-art performance metrics. The training time required by the deep learning model is on the higher side when compared with the conventional method. But, the required computation time on test data by the deep learning model is lesser. Hence, utilizing the trained deep learning model in real-time scenario helps in recognizing the plant species efficiently and at a faster rate.

In recent years, pre-trained deep learning models are heavily used to recognize the plant species [2–10]. It minimizes the requirement of utilizing the high-end computing resources for implementing the deep learning-based algorithms. Among several deep learning architectures [11], the DenseNet architecture has gained wider attention due to its strong gradient flow, more diversified features, parameter and computational efficiency. It is used in different applications such as monocular depth estimation [12], remote hyperspectral sensing [13], spotting the keywords [14], Alzheimer's disease identification [15], classifying the lung diseases [16], classification of Cervical cells [17] and Cardiac phase detection [18].

In this paper, the non-averaged DenseNet-169 Convolutional Neural Network (NADenseNet-169) model is proposed and demonstrated to perform real-time plant species recognition in unconstrained environment. Also, the optimization of model hyperparameters are carried out to obtain higher performance metrics.

2 Related Works

Literatures related to the two approaches (Conventional method and Deep learning method) in the context of plant species recognition is described in the following subsections.

2.1 Conventional Methods

In this method, the input dataset images are preprocessed (resizing, contrast enhancement, histogram equalization, noise removal, blurring, sharpening etc.) and segmented (to obtain the region of interest). From the selected region, the features are extracted using image descriptors or feature descriptors. Some of the vital features extracted from the input image are shape, texture and colour features. Later, the extracted features are used in the training cum test process of the plant species recognition system.

Kheirkhah et al. [2] introduced an approach to perform the plant species recognition that involved the GIST technique. The texture feature is extracted using the GIST technique. PCA method is used as a dimensionality reduction technique. The selected features are classified using Patternnet Neural Network, k-Nearest Neighbor (k-NN), and Support Vector Machine (SVM). Wang et al. [3] presented a Maximum Gap Local Line Pattern method to perform feature extraction. Support vector machine is used to classify the extracted features. Zhang et al. [4] integrated the Sparse Representation (SR) method with the Singular Value Decomposition (SVD) technique to carry out classification of plant species.

Anubha et al. [5] studied the performance of plant species recognition system using the conventional and deep learning approaches. In conventional method, Local Binary Pattern (LBP), Haralick textures, Hu moments and Color channel statistics methods are used to extract the features from the input data. For deep learning methods, the pre-trained models (VGG-16, VGG-19, Inception-V3, Inception ResNet-V2) are used in the process of feature extraction. Machine learning classifiers (Linear Discriminant Analysis, Logistic Regression, k-Nearest Neighbor, Classification and Regression Tree, Bagging Classifier and Random Forest) are used in both approaches to recognize the plant species. The authors reported that the implementation of plant species recognition using deep learning method resulted in obtaining higher performance metrics compared to conventional methods.

2.2 Deep Learning Methods

Wang et al. [3] demonstrated a plant species recognition system by constructing a Siamese Network involving two parallel Inception Convolutional Neural Networks. It is based on few-shot learning technique. Hu et al. [19] demonstrated a Multi-Scale Fusion Convolutional Neural Network in relation to plant species recognition. In this network, the multi-scaled images are fed as an input to the convolution layers. Then, the output feature maps from the convolution layers are merged.

Tan et al. [16] proposed a D-Leaf CNN model to accomplish plant species recognition. The model is used as a feature extractor. Then, the features are classified using different classifiers (SVM, Artificial Neural Network (ANN), k-NN, Naïve-Bayes (NB) and CNN). Lee et al. [6] proposed a multi-organ classification using Hybrid Generic Organ Convolutional Neural Network (HGO-CNN) and Plant-StructNet architectures. Zhu et al. [20] treated the task of plant species recognition as an object detection problem. The authors utilized the Faster Region-CNN Inception-V2 model to detect and classify the plant species. Inception-V2 model is used as a feature extractor. He et al. [21] proposed a bi-channel model to perform plant species recognition. It consists of two pre-trained CNNs, namely, VGG-16 and SqueezeNet architectures.

Younis et al. [22] suggested the usage of modified ResNet architecture to classify the herbarium specimens. Ghazi et al. [8] performed a fusion of pre-trained GoogleNet and VGG architectures. It is done to improve the performance of the plant species recognition system. Barre et al. [9] introduced a LeafNet CNN architecture to do the task of plant recognition. Atabay [10] demonstrated the usage of custom CNN model with Exponential Linear Unit (ELU) as the activation function.

Based on the literature survey, it is observed that a significant number of works on plant species recognition has been reported. Two major approaches are followed. They are conventional method and deep learning method. Conventional methods are not able to achieve high accuracy in prediction with lower computation time (on test data). Whereas, the pre-trained deep learning models are able to attain higher performance metrics with lower computation time (test data). It is also observed that the models reported in literatures tend to perform well on standard datasets, but it is not suitable to perform real-time plant species recognition. This is mainly attributed to the dataset used in the training process. Most of the dataset does not include the images with different challenges (Scale variation, Illumination changes, Variation in camera viewpoint, Changes in object orientation, Structure of leaf - Simple leaf and Compound leaf, leaf arrangement in stem, Leaf color changes due to aging and seasonal variation, cluttered backgrounds). Hence, it becomes necessary to develop a real-time plant species recognition model with high efficiency in prediction and lower computation time.

In this paper, the real-time plant species recognition is performed using a modified version of pre-trained DenseNet-169 CNN model. To support real-time plant species recognition, a new dataset named Leaf-12 is created. The dataset includes leaf images with different scenarios such as scale variation, illumination changes, object orientation variation, different camera view point, color variation due to aging or seasonal condition, structure of leaf - simple leaf and compound leaf, arrangement of leaf in the stem and varying background. Also, the optimization of model hyperparameters are carried out to improve the performance of the recognition system.

3 Methodology

Plant species recognition is carried out using a Non-Averaged DenseNet-169 (NADenseNet-169) CNN model. The model is evaluated using two datasets namely, Flavia [23] and custom created Leaf-12.

3.1 Datasets

Flavia is a standard dataset consisting of 32 plant species as shown in Fig. 1. It is an imbalanced dataset with a number of images ranging from 50 to 77 images/class. The under-sampling technique is exploited for balancing the dataset (an equal number of images per class). Hence, 50 images/class is selected in the training cum test process of the NADenseNet-169 model.

Leaf-12 is a real-time custom created dataset containing twelve Indian plant species as shown in Fig. 2. The dataset images are captured under varying lighting condition, object orientation, camera viewpoint, scale variation, different structure of leaf- simple leaf and compound leaf, arrangement of leaf with stem, and different color background. The leaf-12 dataset contains 320 images per class.

3.2 Preprocessing of Dataset Images

The leaf images from the datasets (Flavia, Leaf-12) are resized to 300×300 pixels by maintaining the aspect ratio. Further, using the nearest interpolation method the images are resized to 100×100 pixels. Also, the pixel values in the images are normalized.

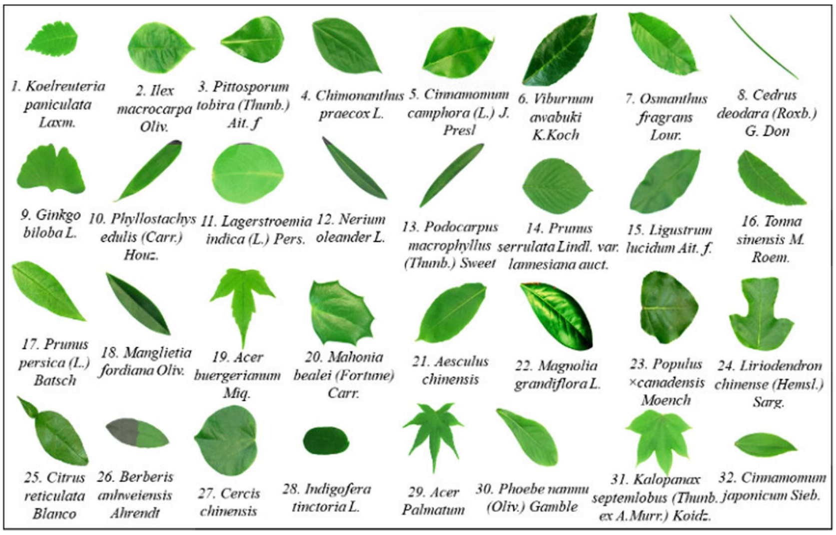

Fig. 1. Sample images in Flavia dataset with its botanical names.

Fig. 2. Sample images in custom created Leaf-12 dataset with its botanical names.

3.3 NADenseNet-169 CNN Model

The block diagram representation of NADenseNet-169 CNN model is shown in Fig. 3. The proposed Non-Averaged DenseNet-169 (NADenseNet-169) architecture is the modified version of the DenseNet-169 model. Traditional DenseNet-169 model is altered by removing the Global Average Pooling (GAP) layer. In traditional DenseNet-169 model, the GAP layer comes in front of final fully connected layer (FCL). GAP uses an average of feature maps obtained from the previous convolution layer. It results in loss of information. Then, the information obtained after the GAP layer is being propagated to FCL. FCL is used in the process of classification. In the modified version of DenseNet-169 (NADenseNet-169), the GAP layer is completely eliminated. So, the features extracted from the earlier layers in DenseNet-169 CNN model is directly propagated to FCL, for the purpose of classification. It resulted in achieving higher performance metrics with lower computation time. This model also eliminates the vanishing gradient problem [11]. It also incorporates the advantages of ResNet, ResNeXt, FractalNet, and Highway network. The proposed NADenseNet-169 CNN has 1×1 convolutions similar to Network-in-Network architecture, Inception-V3, and Inception ResNet-V2. The number of parameters used by the proposed NADenseNet-169 CNN (12,963,744 trainable parameters) is lesser than ResNet architecture (24,583,200 trainable parameters).

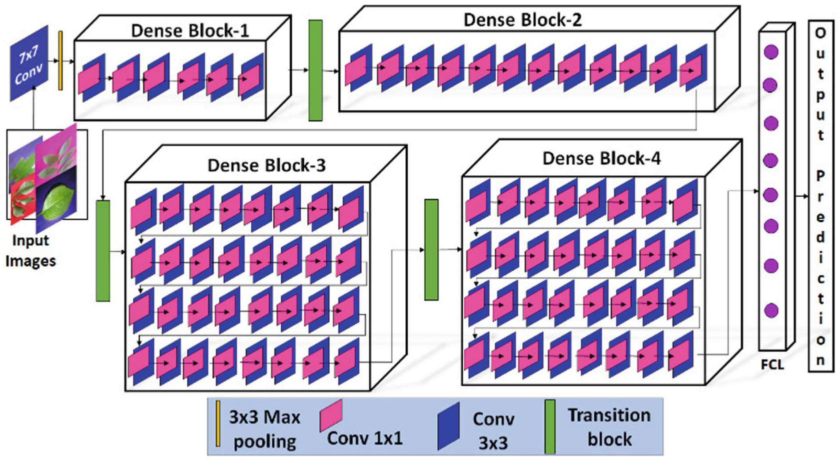

Fig. 3. Plant species recognition using NADenseNet-169 CNN model.

The preprocessed images from the dataset (Flavia and custom created Leaf-12) are fed into NADenseNet-169 model. The model consists of 7×7 convolution layer, 3×3 max-pooling layer, 4 dense blocks, 3 transition layers and a fully connected layer. The features are extracted through the combination of convolution layer, dense blocks and transition layers. The macro-level features are extracted by using the convolution layer with a kernel matrix of size 7×7 and a stride value of 2. The micro-level features are extracted using the dense blocks. Each dense block possesses multiple convolution layers. Four dense block with increasing layers (6, 12, 32, 32) on progression is inbuilt into the architecture. The dense block contains 1×1 convolution followed by 3×3 convolution. 1×1 convolutions are used to reduce the number of input feature maps before it is processed by 3×3 kernels. The feature information in one layer is propagated to all the subsequent layers. It increases the number of input channel in the subsequent layers. In between the dense blocks, the transition layer is embedded into the architecture. Transition layer consists of 1×1 convolution layer and average pooling layer (size $= 2 \times 2$, stride value of 2). This layer is used in the process of dimensionality reduction. The final classification is performed by a fully connected layer (FCL).

3.4 Training Algorithm

The pre-trained weights of ImageNet is employed as initial weights. The hyper-parameter setting employed are 50 epochs, learning rate ($\eta = 0.0001$), Adam optimizer, and a batch size of 32. Adam optimizer [24] is an adaptive optimizer which updates the new weights (θ_{t+1}) based on the learning rate, η and bias-corrected moments ($\hat{m}_t$, $\hat{v}_t$). The optimized learning rate (0.0001) is determined for NADenseNet-169 architecture. The accuracy in predicting the plant species is highly affected when the learning rate is set with a value of 0.001 and 0.00001. The Fully Connected Layer (FCL) with a softmax activation function is used. The categorical cross-entropy loss function is used in the model [25]. The algorithm to train a fine-tuned NADenseNet-169 CNN model is specified in algorithm 1.

Algorithm 1: Fine-tuned NADenseNet-169 model to perform plant species recognition

Input: Images, I_N; ImageNet weights, IW; ImageNet biases, IB; Epochs, ep=50; Batch Size, BS=32; η=0.0001
Output: Predicted label, y_i

Step 1: Read N number of I_N trainingimages
Step 2: Resize I_N images to 100x100x3
Step 3: Pass I_N through the Dense blocks, Transition blocks,and FCL
Step 4: Train the CNN using Adam optimizer
for j=1 to ep:
Update weights, IW and biases, IB
End
Step 5: Test the images to predict the label y_i

3.5 Hardware Setup and Software Tools

The utilized hardware setup to train the NADenseNet-169 model on Flavia and Leaf-12 datasets are detailed below. It includes a Windows-10 64-bit OS running on a Intel Core i7-790 CPU combined with NVIDIA Titan X GPU with 3584 CUDA cores. The programming is carried out in Python 3.5 along with Keras (Tensorflow as backend), Scikit-learn, H5py, Numpy, OS, Matplotlib, and Seaborn packages.

4 Results and Discussion

Two datasets namely, Flavia (Standard) and Leaf-12 (custom developed) are used to determine the performance of the NADenseNet-169 model. The images in the dataset are randomly chosen in the ratio of 70:30 to get separated into train-test datasets. The proposed NADenseNet-169 model performance is examined using the metrics [8] such as top-1 accuracy, top-5 accuracy, precision, recall and F1-score. The performance metrics obtained by using the NADenseNet-169 model are discussed in Subsect. 4.1. Comparison between the NADenseNet-169 model with and without Global Average Pooling (GAP) layer is described in Subsect. 4.2.

4.1 Performance Measure of NADenseNet-169 Model

The performance metrics pertaining to different deep learning architectures namely, VGG-16, VGG-19, Inception-V3, ResNet50, Inception ResNet-V2, Xception, MobileNet, DenseNet-121, DenseNet-201 and the proposed NADenseNet-169 are described in this section. The above specified models are evaluated in relation to plant species recognition.

Flavia Dataset
Table 1 lists the performance metrics obtained by utilizing the different deep learning models on Flavia dataset. From the table data, it is observed that the proposed NADenseNet-169 model resulted in obtaining higher performance metrics (Top-1 accuracy = 98.58%, Top-5 accuracy = 99.84%, Precision = 0.99, Recall = 0.99, F1-Score = 0.99) when compared with other advanced deep learning models (VGG-16, VGG-19, Inception-V3, ResNet-50, Inception ResNet-V2, Xception, MobileNet, DenseNet-121, DenseNet-201). This improvement in performance metrics is mainly attributed to the characteristics of NADenseNet-169 model such as dense connection, feature reuse property and non-averaging of final feature maps.

The DenseNet (121 and 201) models resulted in a comparable accuracy to the NADenseNet-169 model, but other performance metrics (precision, recall, F1-score) are relatively lower than NADenseNet-169 model. The removal of global average pooling (GAP) layer from the conventional DenseNet-169 model, resulted in the improvement of performance metrics (as detailed in Sect. 4.3). By utilizing the GAP layer, the final feature map gets averaged out. Fully Connected Layer (FCL) is used to classify the input data. Also, the computation time of DenseNet-201 model (4.37 s, mainly related to the extra layer - GAP

Table 1. Performance Metrics obtained by using different deep learning architectures on Flavia Dataset.

CNN Models	Top-1(%)	Top-5(%)	Precision	Recall	F1-Score	Time(s)
VGG-16	96.88	99.38	0.97	0.97	0.97	0.46
VGG-19	97.29	99.38	0.98	0.97	0.97	0.54
Inception-V3	97.50	99.17	0.98	0.97	0.97	1.75
ResNet-50	98.12	99.79	0.98	0.98	0.98	1.43
Inception ResNet-V2	98.17	99.79	0.98	0.98	0.98	4.60
Xception	98.17	99.79	0.98	0.98	0.98	1.03
MobileNet	97.29	99.58	0.97	0.97	0.97	0.70
DenseNet-121	98.12	99.79	0.98	0.98	0.98	2.43
DenseNet-201	98.38	99.79	0.98	0.98	0.98	4.37
Proposed NADenseNet-169	**98.58**	**99.84**	**0.99**	**0.99**	**0.99**	**3.53**

layer) is high when compared to NADenseNet-169 model (3.53 s). The computation time of the model is mainly related to the number of tunable parameters and layers. The list of parameters in each model is specified in Table 2. The deep learning architectures such as DenseNet-201, DenseNet-121, ResNet-50, Inception ResNet-V2 and Xception produced accuracies greater than 98%. The proposed NADenseNet-169 resulted in high values for both Top-1 and Top-5 accuracies compared to other deep learning architectures.

Table 2. Model's trainable parameters

Model	Number of layers	Number of Trainable parameters
VGG-16	16	14,862,176
VGG-19	19	20,171,872
Inception-V3	48	21,833,920
ResNet-50	50	24,583,220
Inception ResNet-V2	164	54,294,636
Xception	71	21,396,808
MobileNet	28	3,501,920
DenseNet-121	121	7,248,800
DenseNet-201	201	18,300,300
Proposed NADenseNet-169	**169**	**12,963,744**

The larger F1-score value of 0.99 indicates that the number of misprediction is significantly reduced on employing the proposed NADenseNet-169 model in real-time plant species recognition. The performance of the proposed NADenseNet-169 model is compared with other existing literatures (Flavia dataset). This is represented as a bar graph in Fig. 4. The proposed NADenseNet-169 layer outperforms conventional method as well as deep learning method. From the Fig. 4 data, it is observed that the proposed NADenseNet-169 model is highly suitable to be used in the plant species recognition system as compared with other conventional and deep learning methods.

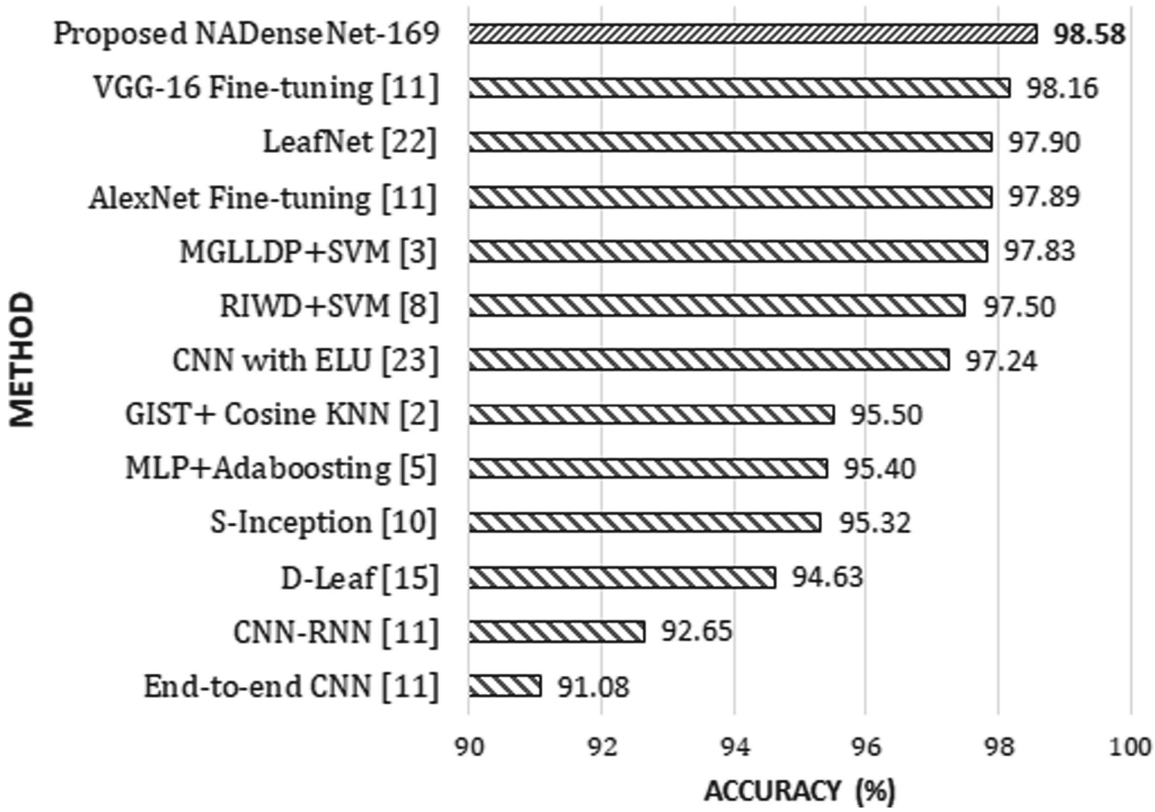

Fig. 4. Comparison of plant species prediction accuracy obtained by using the proposed NADenseNet-169 model with other methods.

Leaf-12 Dataset

The performance metrics obtained by implementing different deep learning architectures on Leaf-12 dataset are listed in Table 3. It is observed that the proposed pre-trained NADenseNet-169 model resulted in achieving higher performance metrics (Top-1 accuracy = 99%, Top-5 accuracy = 100%, Precision = 0.99, Recall = 0.99 and F1-score = 0.99) as compared with other advanced deep learning models (VGG-16, VGG-19, Inception-V3, ResNet-50, Inception ResNet-V2, Xception, MobileNet, DenseNet-121, DenseNet-201). This improvement in performance metrics is mainly attributed to the characteristics of NADenseNet-169 model such as dense connection, property of feature reuse and non-averaging of final feature maps. Similar trend is also visualized while training the model on Flavia dataset.

The DenseNet (121 and 201) models resulted in a comparable accuracy to the NADenseNet-169 model, but the other performance metrics (precision, recall, F1-score) are relatively lower than NADenseNet-169 model. The removal of global average pooling (GAP) layer from the conventional DenseNet-169 model resulted in the improvement of performance metrics. The computational time of DenseNet-201 model (5.73 s, mainly related to the extra layer - GAP layer) is high when compared to NADenseNet-169 model (4.45 s). The higher values of performance metrics (Precision, Recall, F1-Score, Top-1 and Top-5 accuracies) indicates that the proposed model is well suited for real-time plant species recognition.

Table 3. Different models performance metrics obtained by utilizing Leaf-12 dataset.

CNN Models	Top-1 (%)	Top-5 (%)	Precision	Recall	F1-Score	Time(s)
VGG-16	98.44	100	0.98	0.98	0.98	0.96
VGG-19	98.09	99.91	0.98	0.98	0.98	1.17
Inception-V3	97.92	99.83	0.98	0.98	0.98	2.44
ResNet-50	98.78	100	0.99	0.99	0.99	2.10
Inception ResNet-V2	98.87	99.83	0.99	0.99	0.99	5.62
Xception	98.26	100	0.98	0.98	0.98	1.57
MobileNet	98.52	99.91	0.99	0.99	0.99	0.82
DenseNet-121	98.26	100	0.98	0.98	0.98	3.17
DenseNet-201	98.26	100	0.98	0.98	0.98	5.73
Proposed NADenseNet-169	**99**	**100**	**0.99**	**0.99**	**0.99**	**4.45**

The performance metrics obtained by the model on custom created Leaf-12 dataset is in correlation with the values obtained on using the standard Flavia dataset. Since the proposed model (NADenseNet-169) performance metrics is on a higher scale, it becomes highly suitable to be used in the plant species recognition system. The Leaf-12 dataset has more variety (illumination changes, scale variation, camera viewpoint variation, changes in object orientation, structure of leaf - simple leaf and compound leaf, arrangement of leaf in the stem, cluttered background) incorporated into it. Hence, the model trained on the images of Leaf-12 dataset is highly adaptable to be used in real-time scenarios.

4.2 NADenseNet-169 Architecture with GAP Layer

The performance metrics are also computed for NADenseNet-169 architecture with Global Average Pooling (GAP) layer. A GAP layer of size 7×7 is considered. The obtained performance metrics of NADenseNet-169 model are compared with DenseNet-169 architecture (NADenseNet-169 with GAP layer) and shown in Fig. 5. It is observed that the addition of the GAP layer to the model resulted in lowering of performance metrics with increased computation (Flavia: 3.62 s, Leaf-12: 4.60 s) time. This trend is visualized irrespective of the two datasets (Flavia, Leaf-12) considered in the studies.

4.3 Real-Time Prediction of Plant Species

The non-averaged DenseNet-169 CNN (NADenseNet-169) model is proposed and demonstrated to perform real-time plant species recognition under unconstrained environment such as variation in viewpoint, cluttered background, scale changes, illumination variation etc. The model is trained on custom created Leaf-12 dataset.

The real-time images shown in Fig. 6 and Fig. 7. From Table 3 data, it is observed that the other deep learning models such as VGG-16, VGG-19, Inception-V3, ResNet-50, Xception, MobileNet, DenseNet-121 resulted in comparable

Performance Metrics	Flavia		Leaf-12	
	With GAP	Without GAP	With GAP	Without GAP
Top-1 Accuracy (%)	98	**98.58**	97.31	**99**
Top-5 Accuracy (%)	99.24	**99.84**	99.91	**100**
Precision	0.98	**0.99**	0.98	**0.99**
Recall	0.98	**0.98**	0.97	**0.99**
F1-Score	0.98	**0.98**	0.97	**0.99**
Computation Time(s)	3.62	**3.53**	4.60	**4.45**

Fig. 5. Comparison of performance metrics obtained by using the pre-trained DenseNet-169 model with and without GAP layer.

Fig. 6. Real-time leaf prediction using NADenseNet-169 model and other deep learning models. Correct predictions are highlighted in red box. (a) Jungle Flame, (b) Rose and (c) Indian Borage. (Color figure online)

accuracy with lower computation time in-relation to the proposed NADenseNet-169 model. But, these models resulted in a large number of misprediction, when it is tested on real-time images with variable environments. This is visualized in Fig. 6.

The plant species in home garden is used to acquire the real-time leaf images. The proposed model is tested with different settings such as illumination variation, rotation, camera viewpoint and scale changes. These real-time images are

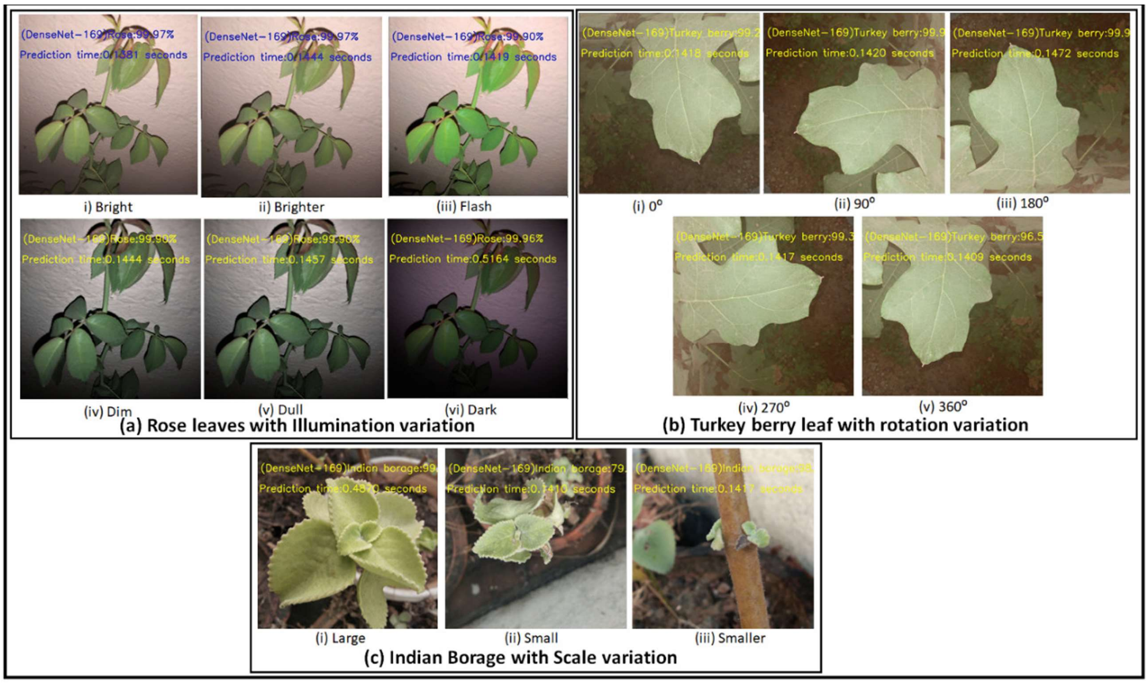

Fig. 7. Prediction of plant species using NADenseNet-169 model with unconstrained settings such as (a) Illumination, (b) Rotation and (c) Scale variations.

used in the process of plant species recognition by different architectures (VGG-16, VGG-19, Inception-V3, ResNet50, Inception ResNet-V2, Xception, MobileNet, DenseNet-121, DenseNet-201 and proposed NADenseNet-169). Figure 7 shows a few examples of prediction of Jungle flame, Rose and Indian Borage plant species by the proposed NADenseNet-169 model and other deep learning architectures (mispredictions shown in the figure). From Fig. 6, it is observed that the NADenseNet-169 model efficiently predicts the plant species compared to other architectures.

Figure 7 shows the prediction results obtained by using the proposed NADenseNet-169 CNN model for images captured under varying illumination condition (Fig. 7(a)), orientation variation (Fig. 7(b)) and scale variation (Fig. 7(c)). It is observed that the proposed NADenseNet-169 CNN model identifies the plant species efficiently in unconstrained environment.

On a summary, it is identified that the NADenseNet-169 model trained on Leaf-12 dataset is highly suitable to perform real-time plant species recognition. As a future work, the proposed model will be tested by considering large number of plant species. Architecture level modification will also be carried out to minimize the computation time.

5 Conclusion

Non-Averaged DenseNet-169 CNN model is proposed and demonstrated to perform plant species recognition. From the experimental investigation, it is observed that the NADenseNet-169 model performed better (Flavia dataset:

accuracy = 98.58%, computation time = 3.53 s; Leaf-12 dataset: accuracy = 99%, computation time = 4.45 s) compared to traditional image processing method and other deep learning approaches. The above-specified metrics are obtained with the optimized hyperparameters (Adam optimizer, learning rate = 0.0001, Batch size = 32). Also, the performance of the architecture gets weakened (lower accuracy and higher computation time) by inserting a Global Average Pooling (GAP) layer with a size of 7×7 into the architecture. The NADenseNet-169 model attains higher accuracy with comparable computation time, while testing it on the non-augmented datasets. The number of misprediction on real-time images is significantly reduced by utilizing NADenseNet-169 model. Hence, the proposed NADenseNet-169 model trained on Leaf-12 dataset becomes highly suitable to be utilized in the real-time plant species recognition system.

Acknowledgements. The authors would like to thank NVIDIA for providing Titan X GPU under the University Research Grant Programme.

References

1. Wäldchen, J., Mäder, P.: Plant species identification using computer vision techniques: a systematic literature review. Arch. Comput. Methods Eng. **25**, 507–543 (2018)
2. MostajerKheirkhah, F., Asghari, H.: Plant leaf classification using GIST texture features. IET Comput. Vis. **13**(4), 369–375 (2019)
3. Wang, X., Du, W., Guo, F., Hu, S.: Leaf recognition based on elliptical half gabor and maximum gap local line direction pattern. IEEE Access **8**, 39175–39183 (2020)
4. Zhang, S., Zhang, C., Wang, Z., Kong, W.: Combining sparse representation and singular value decomposition for plant recognition. Appl. Soft Comput. J. **67**, 164–171 (2018)
5. Anubha Pearline, S., Sathiesh Kumar, V., Harini, S.: A study on plant recognition using conventional image processing and deep learning approaches. J. Intell. Fuzzy Syst. **36**(3), 1997–2004 (2019)
6. Lee, S.H., Chan, C.S., Remagnino, P.: Multi-Organ Plant Classification Based on Convolutional and Recurrent Neural Networks. IEEE Trans. Image Process. **27**(9), 4287–4301 (2018)
7. Raj, A.P.S.S., Vajravelu, S.K.: DDLA: dual deep learning architecture for classification of plant species. IET Image Process. **13**(12), 2176–2182 (2019)
8. Mehdipour Ghazi, M., Yanikoglu, B., Aptoula, E.: Plant identification using deep neural networks via optimization of transfer learning parameters. Neurocomputing **235**, 228–235 (2017)
9. Barré, P., Stöver, B.C., Müller, K.F., Steinhage, V.: LeafNet: a computer vision system for automatic plant species identification. Ecol. Inform. **40**, 50–56 (2017)
10. Atabay, H.A.: A convolutional neural network with a new architecture applied on leaf classification. IIOAB J. **7**(5), 226–331 (2016)
11. Khan, S., Rahmani, H., Shah, S.A.A., Bennamoun, M.: A guide to convolutional neural networks for computer vision. Synthesis Lectures Comput. Vis. **8**(1), 1–207 (2018)
12. Liu, J., Zhang, Y., Cui, J., Feng, Y., Pang, L.: Fully convolutional multi-scale dense networks for monocular depth estimation. IET Comput. Vision **13**(5), 515–522 (2019)

13. Zhang, C., Li, G., Du, S.: Multi-scale dense networks for hyperspectral remote sensing image classification. IEEE Trans. Geosci. Remote Sens. **57**(11), 9201–9222 (2019)
14. Zeng, M., Xiao, N.: Effective combination of DenseNet and BiLSTM for keyword spotting. IEEE Access **7**, 10767–10775 (2019)
15. Cui, R., Liu, M.: Hippocampus analysis by combination of 3-D DenseNet and shapes for Alzheimer's disease diagnosis. IEEE J. Biomed. Heal. Informatics **23**(5), 2099–2107 (2019)
16. Tan, T., et al.: Optimize transfer learning for lung diseases in bronchoscopy using a new concept: sequential fine-tuning. IEEE J. Transl. Eng. Heal. Med. **6**, 1–8 (2018)
17. Lin, H., Hu, Y., Chen, S., Yao, J., Zhang, L.: Fine-grained classification of cervical cells using morphological and appearance based convolutional neural networks. IEEE Access **7**, 71541–71549 (2019)
18. Dezaki, F.T., et al.: Cardiac phase detection in echocardiograms with densely gated recurrent neural networks and global extrema loss. IEEE Trans. Med. Imaging **38**(8), 1821–1832 (2018)
19. Hu, J., Chen, Z., Yang, M., Zhang, R., Cui, Y.: A multiscale fusion convolutional neural network for plant leaf recognition. IEEE Signal Process. Lett. **25**(6), 853–857 (2018)
20. Zhu, X., Zhu, M., Ren, H.: Method of plant leaf recognition based on improved deep convolutional neural network. Cogn. Syst. Res. **52**, 223–233 (2018)
21. He, G., Xia, Z., Zhang, Q., Zhang, H., Fan, J.: Plant species identification by bi-channel deep convolutional networks. J. Phys: Conf. Ser. **1004**, 012015–6 (2018)
22. Younis, S., Weiland, C., Hoehndorf, R., Dressler, S., Hickler, T., Seeger, B., Schmidt, M.: Taxon and trait recognition from digitized herbarium specimens using deep convolutional neural networks. Bot. Lett. **165**(3), 377–383 (2018)
23. Wu, S.G., Bao, F.S., Xu, E.Y., Wang, Y.X., Chang, Y.F., Xiang, Q.L.: A leaf recognition algorithm for plant classification using probabilistic neural network. In: ISSPIT 2007–2007 IEEE International Symposium on Signal Processing and Information Technology, pp. 11–16 (2007)
24. Ruder, S.: An overview of gradient descent optimization algorithms (2016). arXiv preprint arXiv:1609.04747 (2016)
25. Rusiecki, A.: Trimmed categorical cross-entropy for deep learning with label noise. Electron. Lett. **55**(6), 319–320 (2019)