DOI: 10.6977/IJoSI.202502_9(1).0001
M.V. Neethi, P. Raviraj, etc./Int. J. Systematic Innovation, 9(1), 1-18 (2025)

Evaluation of convolutional neural network models' performance for estimating mango crop yield

M.V. Neethi[1], P. Raviraj[2]

[1]Department of Computer Science and Engineering-Data Science, ATME College of Engineering, Affiliated to Visvesvaraya Technological University (VTU), Belagavi, India
[2]Department of Computer Science and Engineering, GSSS Institute of Engineering and Technology for Women, Affiliated to Visvesvaraya Technological University (VTU), Belagavi, India

* Corresponding author E-mail: neethi.mv@gmail.com

(Received 16 April 2024; Final version received 3 August 2024; Accepted 16 August 2024)

Abstract

In agriculture, crop yield estimation is essential; producers, industrialists, and consumers all benefit from knowing the early yield. Manual mango counting typically involves the utilization of human labor. Experts visually examine each sample to complete the process, which is time-consuming, difficult, and lacks precision. For commercial mango production to produce high-quality fruits from the orchard to the consumer, a quick, non-destructive, and accurate variety classification is required. Because of its effectiveness in computer vision, a convolutional neural network—one of the deep learning techniques—was chosen for this investigation. For yield prediction, a total of eight popular mango cultivars were utilized. A comparison with previously trained models was used to assess the proposed model. The performance of the classifiers was evaluated using evaluation metrics such as accuracy, loss, area under the receiver operating characteristic curve score, precision, recall, F1-score, sensitivity, specificity, positive predictive value, negative predictive value, and Cohen's Kappa performance measure. In terms of performance evaluation criteria, it was found that the proposed approach outperformed the pre-trained models. The suggested model achieved 98.85% accuracy in the test set, which had 800 images. This outcome demonstrates the tangible applicability of the proposed methodology for mango crop estimation.

Keywords: Convolutional Neural Networks, Deep Learning, Early Detection, Machine Learning

1. Introduction

Precise prediction of crop production levels is essential for developing efficient farming practices and preserving food security in a rapidly changing world. Crop yield affects food productivity and is crucial in ensuring food availability and safety, as recognized by policymakers, farmers, and consumers alike. Therefore, predicting crop yields offers a significant advantage in supporting financial and managerial decisions. Crop yield is the ratio of agricultural input to output, measured as the quantity of yield per unit area of cultivated land. Hence, crop yield is a typical indicator of agricultural productivity. Crop productivity can be affected by various factors, such as pests and diseases, environmental changes, farming techniques, and consumer demand.

Computer vision-based automatic crop yield estimation greatly helps agronomists to identify and categorise plant diseases, allowing them to take necessary measures at an early stage to prevent further damage to the plant (Xin Zhang et al., 2019). A simple plant disease detection system can be used as an efficient tool by small-scale farmers for early disease detection. In large-scale cultivation, the system can be mounted on autonomous vehicles that cover the entire region with continuous video coverage or image capturing which is monitored timely and accurately by the agronomist. This approach can help detect, diagnose, and treat pathological problems to a great extent. Over the decades, many types of research have been developed in the field of plant disease detection. Numerous machine learning algorithms have been created, which learn iterative from the given data and build the model without explicit programming, providing an efficient tool for plant disease detection, not only for identification.

In this study, the proposed model was compared and analyzed with nine different convolutional pre-trained neural networks, namely: AlexNet, VGG16, VGG19, DenseNet-121, ResNet-101, MobileNetV2, Xception, NasNet-Mobile, and Inception-v networks. The experiments involved tweaking the hyperparameters to improve the performance of the convolutional neural network (CNN) model. The study hypothesized that deep learning models would help phytopathologist and farmers estimate yield at an early stage.

The contents of the paper are organized as follows: Section 2 provides a survey of the related literature. Section 3 presents the dataset for experimentation and testing, as well as experiments designed to study the elements that impact the overall performance and effectiveness of the evolutionary system. Section 4 presents the results and Section 5 provides an analysis of the proposed method. The article ends with the conclusion and recommendations.

2. Literature Survey

The literature contains various works related to crop yield estimation. The development of agricultural deep learning technology, especially in yield estimation detection, has only started in recent years and remains somewhat limited. Deep learning, a subset of machine learning, helps identify and classify different plant diseases (Xin Yang et al., 2016). Deep learning utilizes multi-level depiction and generalization by employing multi-layer nonlinear modules for feature extraction and transformation, distinguishing it from traditional machine learning (Fine, 2006; Fuentes et al., 2017). CNN is a deep learning algorithm that takes an image as the input, learns the objects and patterns within the image, and differentiates it from others (Grinblat et al., 2016). A smartphone-assisted CNN model is used for crop disease detection using a publicly available dataset to identify 14 different crops and 26 different diseases, as CNN requires less pre-processing compared to other algorithms, concerning hyper-spectral images. However, CNN faces challenges in processing the high-dimensional information in multidimensional data cubes, which leads to high computational time (Mohanty et al., 2016; Paoletti et al., 2018). Nuclear discriminant analysis based on the Spectral Vegetation Index method is used for the detection and classification of yellow rust, aphids, and powdery mildew in winter wheat (Shi et al., 2017). In hyper-spectral reflecting datasets at the leaf and canopy levels, the model outperforms the traditional linear discriminant method for classifying healthy leaf and diseased wheat leaves. A leaf-based CNN model for disease recognition and classification successfully classified 13 different diseases, with precision ranging from 91% to 98% (Sladojevic et al., 2016). A dataset containing 79,265 images was introduced by Arsenovic et al., 2019). Two types of augmentation techniques were applied to increase the dataset size, and a new two-stage neural network architecture was proposed for the classification of plant diseases based on the current environment. The trained model achieved an accuracy of 93.67%. A method was developed to classify citrus disease using the ΔE colour difference algorithm to segment the diseased area. This method achieved 99.9% accuracy (Ali et al., 2017). A novel cucumber disease recognition model was developed using three pipelined procedures: segmenting diseased leaves with K-means clustering, color and shape detection, and classification of diseased leaves by sparse representation. The developed method was compared with other methods and achieved a classification performance of 85.7% (Zhang et al., 2017). A new CNN architecture with two deep classifiers and a trainable visualization method for plant disease classification was proposed. The architecture trained two classifiers in parallel, and the area over perturbation curve was used to compare the proposed method with the existing state of art method, achieving a performance of approximately 0.907 (Brahimi et al., 2019). Different plant species were classified, and a content-based data retrieval method was used to search for plant species using a deep learning approach (Gyires-Tóth et al., 2019). The EfficentNet deep learning architecture was proposed to detect plant diseases. Plant village datasets were tested on EfficientNet, achieving accuracies of 99.91% and 99.97% for the original images, and 98.42% and 99.39% for the enhanced images (Atila et al., 2021).

3. Materials and Methods

3.1. Dataset

In this study, a temporal mango crop dataset of 4,000 images, including eight different cultivars, was used. The dataset of images, captured under real cultivation conditions, contains various objects like ground, other parts of the plant, varying illumination conditions, occlusions, etc. Fig. 1 and Table 1 depict the dataset and provide detailed information about it. A large dataset is required for the robust performance of CNN models. The mango fruit dataset was built under a limited set of conditions; to improve variability and increase the dataset size. Image augmentation techniques such as rotation, reflection, translation, scaling, and shear were applied to the dataset images for both the x- and y-axes. The entire dataset was initially divided into two subsets: the training set and the testing set, with images randomly split in an 80/20 ratio. A Python script was developed to automatically divide dataset images into two sets, which includes size reduction and normalizing to a 224×224 pixel size. Fig. 2 provides an overview of the proposed workflow.

3.2. Deep Learning Models

A CNN is a deep learning algorithm that takes an input image and assigns importance to various features and objects within the image to distinguish them. The pre-processing required for a convolution network is much lower than for other classification algorithms. Several innovative ideas were have contributed to the evolution of CNNs, including optimization of parameters using various activation and loss functions, the development of standardized architectures, etc. (Khan et al., 2020; Zhou, 2020). An overview of the deep learning model is depicted in Fig. 3.

In this study, an FRCNN ResNet-50 neural network models were considered to estimate eight different cultivars of mango fruit crops and were compared with nine well-know pre-trained deep networks: AlexNet, VGG-16, VGG-19, ResNet-101, DenseNet-121, NasNet-Mobile, Inception-V3, Xception, and Mobilenet-V2.

Mango Cultivars

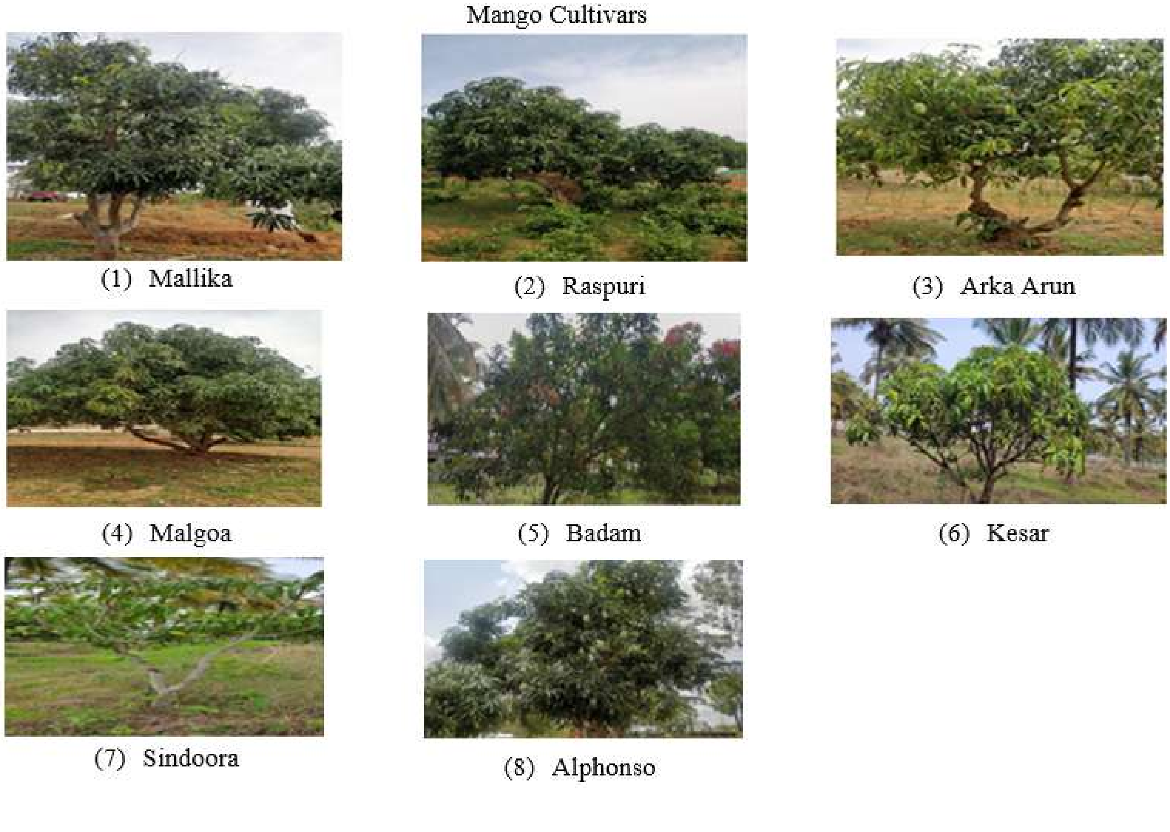


Fig. 1. Dataset description.

Table 1. Detailed information of the dataset.

SI. no.	Image dataset	Number of samples
1	Mallika	500
2	Raspuri	500
3	Arka Arun	500
4	Malgoa	500
5	Badam	500
6	Kesar	500
7	Sindoora	500
8	Alphanso	500

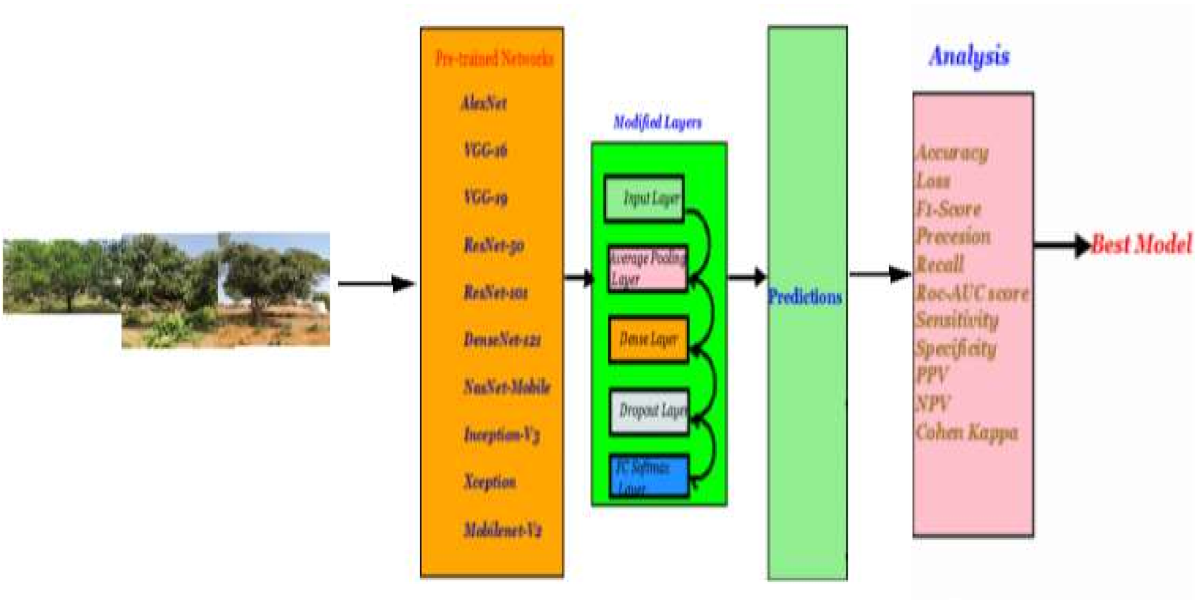


Fig. 2. Flowchart of the overall study. Abbreviations: NPV: Negative predictive value; PPV: Positive predictive value; Roc-AUC: Area under the receiver operating characteristic curve.

AlexNet, designed by Alex Krizhevsky, is a feedforward CNN architecture consisting of eight layers: five convolutional layers containing several kernels and three fully connected layers (f6, f7 and f8) (Krizhevsky et al., 2012). VGG-16 is a deep CNN with a depth of 16 layers, consisting of five convolutional blocks (13 layers) and three fully connected layers. The network consists of five larger blocks, proposed by Karen Simonyan (Simonyan et al. 2015). VGG19 is deeper than VGG 16, with 19 layers in total. It has five convolutional blocks (16 layers) followed by three fully connected layers. The ResNet CNN model is made up of residual blocks. ResNet-50 is a CNN with 50 layers, consisting of 48 convolutional layers, one MaxPool layer, and one average pool layer. The depth of ResNet-101 is 101 layers (He et al., 2016). DenseNet-121, also a CNN model, has four dense blocks with varying numbers of layers i.e., [6, 12, 24, 16]. The DenseNet-121 architecture was designed by Gao Huang (Huang et al., 2018). NasNet-Mobile is a CNN that explores an architectural building block on a small dataset and then assigns the block to a larger dataset, training the network by adding more layers into the block (Zhou & Diamos, 2018). Inception-V3 is a CNN model that mainly focuses on utilizing less computational power (Szegedy et al., 2016). It is a 48-layer deep network that stacks 11 inception modules. Each module consists of pooling layers and convolutional filters with rectified linear units as the activation function. Xception CNN stands for an extreme version of inception with a modified depth-separable convolution. The Xception architecture has 36 convolutional layers, with the initial two layers of convolution followed by depth-separable layers and fully connected layers (Chollet, 2017). MobileNet-V2 is a CNN that uses depth-wise separable convolution as efficient building blocks. It is a 53-layer deep network with 52 convolution layers and a fully connected layer (Sandler et al., 2018).

In this study, the bottom-up pathway was constructed using Fast RCNN-ResNet-50, with each of its five convolution modules (C1 through C5) containing multiple convolution layers (Yang et al. 2019). The spatial dimension was halved at each level from C1 to C5, reducing the C5 channel depth to 256-d (P5). Moreover, the initial feature map layer for object prediction was created using a 1×1 convolution filter. The nearest neighbor upsampling technique was used in a top-down manner to upsample the preceding layer by a factor of two. The pixel-by-pixel 1×1 filtered C4 and the upsampled P5 were combined to form P4. The same process was used to create P3 and P2. The network used its built-in multi-scale pyramidal structure of deep convolutional networks to construct feature pyramids and generate autonomous predictions at different levels (P2, P3, P4, P5, and P6) for multiscale object detection. The general characteristics and structural details of all the CNN models are depicted in Tables 2 and 3. All models were trained on the Imagenet dataset (Neethi & Raviraj, 2024).

3.3. Performance Evaluation of Deep Learning Models

To compare the performance of the CNN models, 10 performance indices were calculated as follows:

$$Accuracy = \frac{True\ Positive + True\ Negative}{Total\ Sample} \tag{1}$$

$$Loss = \frac{-1}{N}\sum_{i=1}^{N}\sum_{j-1}^{M} yij * \log(pij) \tag{2}$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{3}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{4}$$

$$F1 - Score = 2 * \frac{1}{\frac{1}{Precision} + \frac{1}{Recall}} \tag{5}$$

$$Sensitivity = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{6}$$

$$Specificity = \frac{True\ Negative}{True\ Positive + False\ Positive} \tag{7}$$

$$Positive\ Prediction\ Value\ (PPV) = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{8}$$

$$Negative\ Prediction\ Value\ (PPV) = \frac{True\ Negative}{True\ Negative + False\ Negative} \tag{9}$$

$$Cohen\ Kappa = \frac{P0 - P_e}{1 - P_e} \tag{10}$$

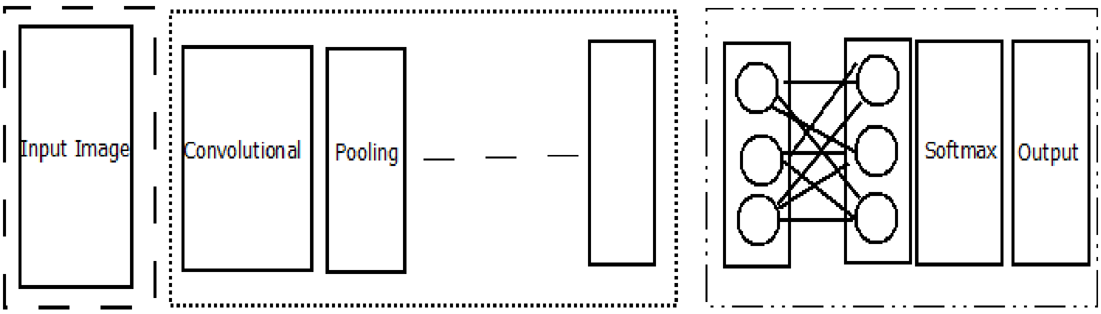


Fig. 3. Overview of the deep convolutional model.

Table 2. Characteristics of convolutional neural networks used.

Network	Depth	Parameters	Trained parameters	Input layer size	Output layer size	Epochs
AlexNet	8	21,619,464	32,776	224 by 224	7 by 7	100
VGG-16	16	14,718,792	4,104	224 by 224	7 by 7	100
VGG-19	19	20,028,488	4,104	224 by 224	7 by 7	100
FRCNN ResNet-50	50	23,604,104	16,392	224 by 224	7 by 7	100
ResNet-101	101	42,674,568	16,392	224 by 224	7 by 7	100
Inception-V3	159	21,819,176	16,392	224 by 224	7 by 7	100
NasNet-Mobile	-	4,278,172	8,456	224 by 224	7 by 7	100
MobileNet-V2	53	2,268,232	10,248	224 by 224	7 by 7	100
DensNet-121	121	7,045,704	8,200	224 by 224	7 by 7	100
Xception	71	20,877,872	16,392	224 by 224	7 by 7	100

Table 3. Structural details of the convolutional neural networks.

Network	Number of convolutional layers	Number of fully connected layers	Pooling	Softmax layer	Filters
AlexNet	5	3	3	1	256
VGG-16	13	3	5	1	512
VGG-19	16	3	5	1	512
FRCNN ResNet-50	16 (residual blocks)	1	1	1	2048
ResNet-101	33 (residual blocks)	1	1	1	2048
DensNet-121	4 (dense blocks)	1	1	1	1024
NasNet-Mobile	-	1	1	1	1056
Inception-V3	11 (inception blocks)	1	1	1	2048
Xception	36	1	1	1	2048
MobileNet-V2	7 (bottleneck)	1	1	1	1024

In this study, positive and negative classes were assigned to mango fruit and non-mango fruit regions. Hence, true positive and true negative represent the number of correctly diagnosed mango and non-mango regions, respectively. False positive and false negative represent the number of incorrectly diagnosed mango and non-mango regions, respectively. To evaluate the overall performance of the CNN model, the area under the receiver operating characteristic curve (Roc-AUC) and Cohen' kappa scores were calculated.

4. Results

The performance of all 10 CNN model is depicted in Table 4. In this study, the training dataset consists of 4,000 images, and the validation dataset consists of 800 images. The networks were able to classify mango and non-mango regions of mango fruit crops, with a Roc-AUC score in the range of 0.96 to 0.99. The MobileNet-V2 CNN model outperformed all the other models. The performance of all the 10 CNN models was measured using accuracy, loss, Roc-AUC score, precision, recall, F1-score, sensitivity, specificity, positive predictive value (PPV), negative predictive value (NPV), and Cohen's Kappa performance measure. The best model is considered based on the overall performance of all these measures.

The proposed FRCNN ResNet-50 network achieved a RoC-AUC score of 0.999 (accuracy: 0.9863; loss: 0.07; precision: 0.99; recall: 0.99; F1-score: 0.99; sensitivity: 0.94; specificity: 0.90; PPV: 0.96; NPV: 0.95; Cohen's Kappa: 0.98) in the training dataset and a RoC-AUC score of 0.989 (accuracy: 0.9815; loss: 0.04; precision: 0.98; recall: 0.98; F1-score: 0.98; sensitivity: 0.92; specificity: 0.87; PPV: 0.88; NPV: 0.97; Cohen's Kappa: 0.97) in the validation dataset. The time taken for each epoch was 2 seconds per step and 119 iterations per epoch.

The MobileNet-V2 network also achieved better results after the FRCNN ResNet-50, with a RoC-AUC score of 0.98 (accuracy: 0.97; loss: 0.09; precision:

Table 4. Overall performance of all the convolutional model.

Model	Data	Acc	Loss	Roc-AUC score	Pre	Rec	F1	Cohen's kappa	Sen	Spe*	PPV	NPV
AlexNet	Train	0.8245	0.4532	0.9657	0.85	0.82	0.81	0.8223	0.87	0.79	0.8901	0.6785
	Val	0.80	0.4328	0.9566	0.83	0.82	0.82	0.8103	0.9012	0.8345	0.9432	0.7012
NasNet-Mobile	Train	0.9356	0.1695	0.9980	0.96	0.95	0.95	0.9366	0.94	0.76	0.8293	0.9206
	Val	0.9275	0.1410	0.9977	0.95	0.94	0.94	0.9285	0.97	0.64	0.9552	0.7293
DenseNet	Train	0.9497	0.1588	0.9980	0.96	0.95	0.96	0.9393	0.9601	0.6765	0.8562	0.8943
	Val	0.945	0.1354	0.9983	0.96	0.95	0.94	0.9371	0.99	0.6765	0.9833	0.7071
VGG-16	Train	0.8986	0.2764	0.9934	0.90	0.89	0.90	0.87	0.8909	0.7209	0.7002	0.9010
	Val	0.8862	0.2918	0.9903	0.90	0.89	0.89	0.8783	0.91	0.67	0.8815	0.7338
VGG-19	Train	0.8844	0.3631	0.9905	0.88	0.87	0.87	0.934	0.9064	0.6395	0.8776	0.7076
	Val	0.8687	0.3542	0.991	0.88	0.87	0.87	0.901	0.97	0.55	0.6830	0.9482
Inception-V3	Train	0.9505	0.1170	0.9993	0.97	0.94	0.95	0.9518	0.9878	0.5851	0.9442	0.8716
	Val	0.946	0.1324	0.9991	0.96	0.95	0.95	0.9385	1.0	0.58	1.0	0.71
Xception	Train	0.7110	0.8199	0.9674	0.76	0.61	0.71	0.6444	0.98	0.103	0.8235	0.7586
	Val	0.5912	0.8982	0.9526	0.71	0.59	0.52	0.5328	0.96	0.06	0.8571	0.5052
ResNet-101	Train	0.96	0.1337	0.9987	0.97	0.96	0.97	0.9518	0.7901	0.9619	0.8791	0.9289
	Val	0.963	0.1096	0.9987	0.97	0.96	0.96	0.9585	0.7605	0.99	0.9870	0.8048
MobileNetV2	Train	0.9371	0.1942	0.9974	0.95	0.93	0.94	0.9238	0.9688	0.6024	0.8714	0.8742
	Val	0.9312	0.1741	0.9979	0.95	0.93	0.93	0.9214	0.99	0.6024	0.9821	0.6875
FRCNN-ResNet 50	Train	0.9715	0.0912	0.9993	0.98	0.97	0.98	0.9657	0.9844	0.7802	0.9461	0.9274
	Val	0.9685	0.0797	0.9996	0.97	0.97	0.97	0.9642	0.99	0.76	0.9870	0.8048

Abbreviations: Acc: Accuracy; F1: F1-Score; NPV: Negative predictive value; PPV: Positive predictive value; Pre: Precision; Rec: Recall; Sen: Sensitivity; Spe: Specificity; Val: Validation.

0.98; recall: 0.98; F1-score: 0.98; sensitivity: 0.92; specificity: 0.95; PPV: 0.96; NPV: 0.95; Cohen's Kappa: 0.98) and a RoC-AUC score of 0.989 (accuracy: 0.98; loss: 0.05; precision: 0.98; recall: 0.98; F1-score: 0.98; sensitivity: 0.92; specificity: 0.96; PPV: 0.95; NPV: 0.89; Cohen's Kappa: 0.97) in the training and validation datasets, respectively. The time taken for each epoch was 6 seconds per step, with 119 iterations per epoch.

The ResNet-101 network achieved a RoC-AUC score of 0.96 and an accuracy of 0.71, with overall performance much lower compared to other networks. The time taken for each epoch was 11 seconds per step, with 119 iterations per epoch. The NasNet-Mobile network also achieved good results, close to the Xception network. DenseNet, Vgg-19, Inception-V3, and AlexNet networks performed well in classifying healthy and diseased leaves, with accuracy in the range of 0.91–0.94. VGG-16 achieved an accuracy of 0.88.

Fig. 4 shows the overall performance of all the networks using a radar plot. The training and validation process is shown in Fig. 5. A confusion matrix was calculated to simplify the understanding of the performance of all the networks on both training and validation datasets, as shown in Fig. 6. In the confusion matrix, the eight classes of mango crops were represented by class 0 to class 7: class 0 — Mallika, class 1—Raspuri, class 2—Arka Arun, class 3—Malgoa, class 4—Badam, class 5—Kesar, class 6—Sindoora, and class 7—Alphanso (Table 1). By observing all the model's confusion matrices, it can be noted that the major misclassifications were between class 0 and class 1, i.e., Mallika and Raspuri. The model failed to classify the images captured under field conditions due to the presence of other parts of the plant, the ground, and other background objects. Even after tuning hyperparameters, the results did not show considerable variance. The reputable results were achieved for 100 epochs. The classification accuracy of each mango fruit crop class for all the convolutional models is depicted in Table 5, and the respective chart is shown in Fig. 7. Receiver operating characteristic curve for each class are plotted and presented in Fig. 8 for all the convolutional models.

5. Discussion

In this study, an FRCNN model was proposed and compared with well known CNNs to estimate mango fruit crop yields. The results showed that proposed method could efficiently estimate mango crop yield with the highest accuracy. Although the proposed model and MobileNet-V2 both gave the same accuracy results, the proposed model could diagnose plant disease with higher sensitivity, lower specificity,

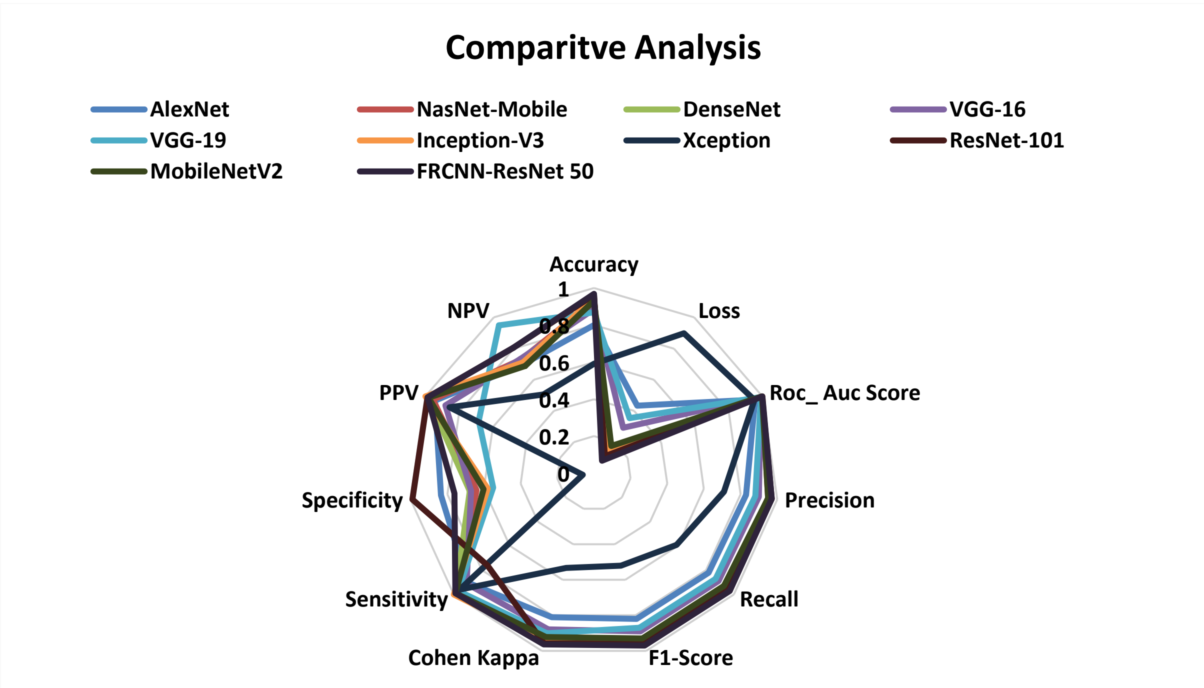
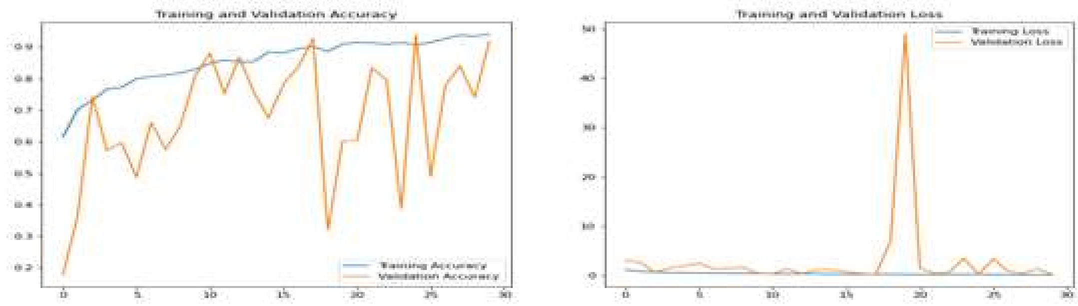
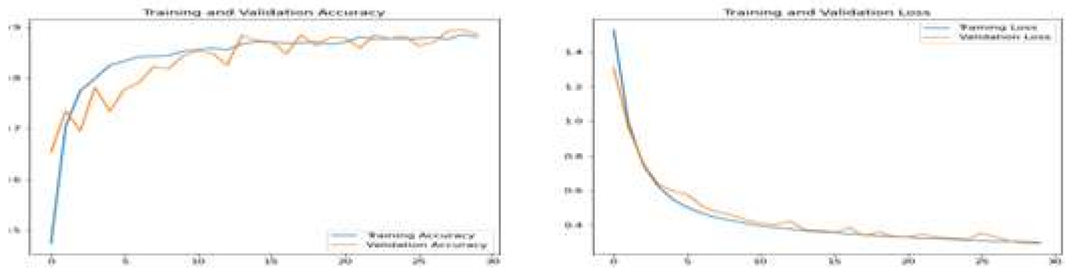


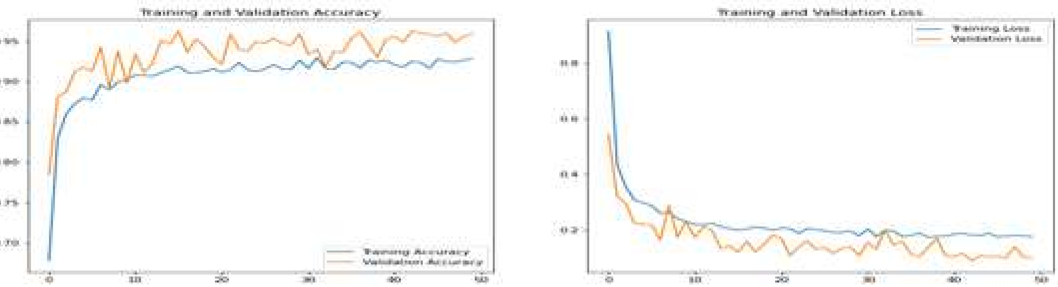
Fig. 4. Radar plot of the 10 individual networks. Abbreviations: NPV: Negative predictive value; PPV: Positive predictive value; Roc_Auc: Area under the receiver operating characteristic curve.



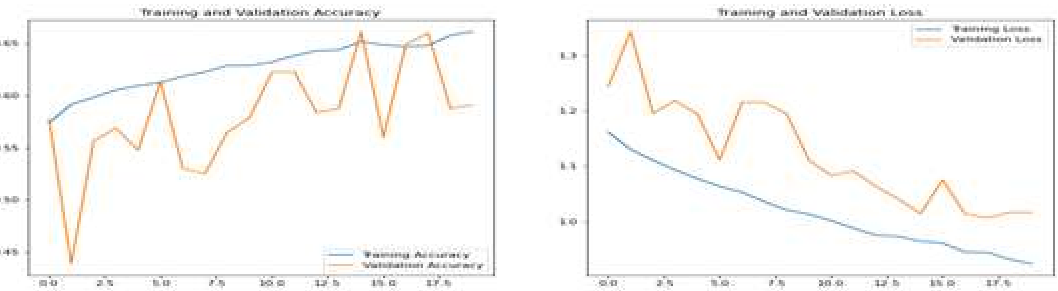
1. AlexNet



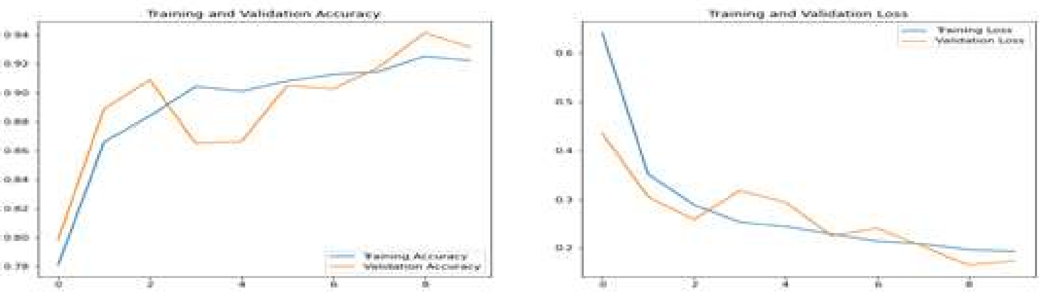
2. Vgg16



3. Vgg19

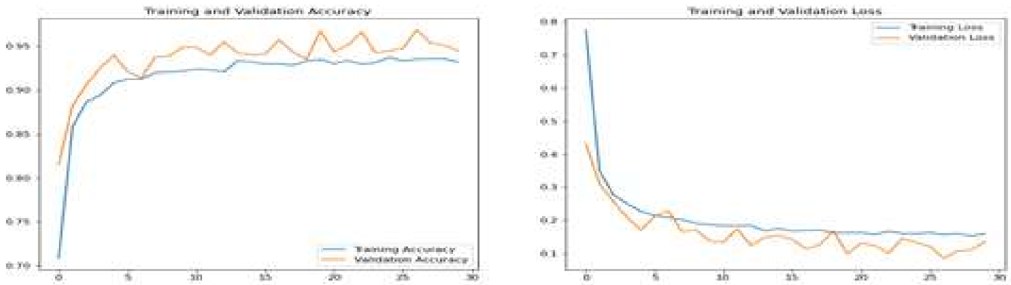


4. ResNet-101

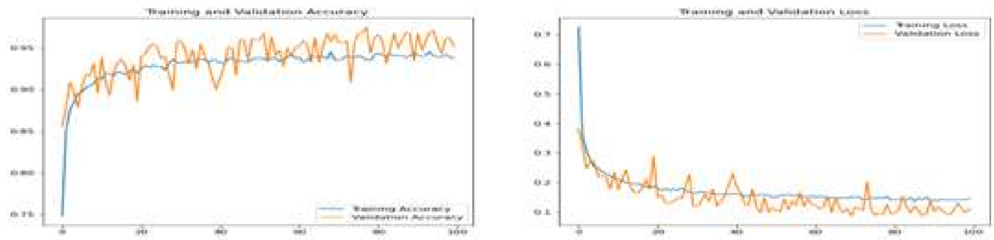


5. DesnseNet

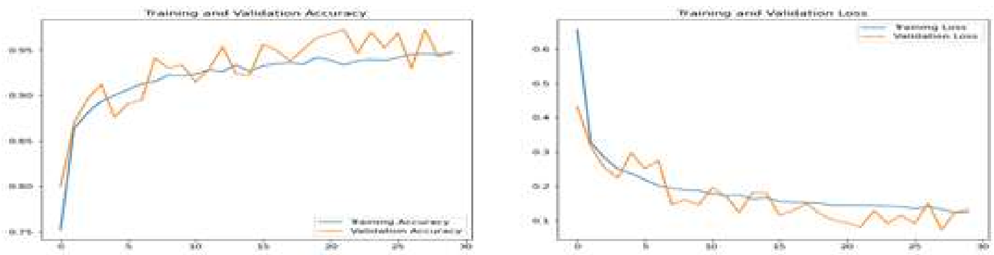
(Fig. 5)



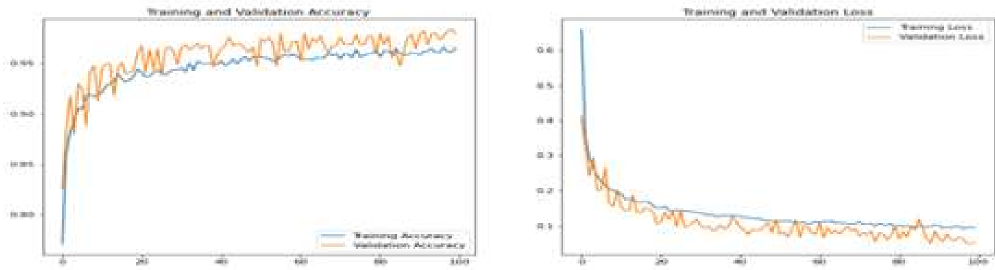
6. NasNet-Mobile



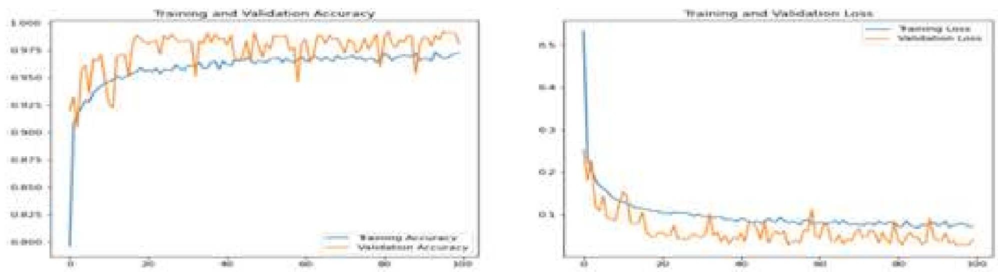
7. Inception-V3



8. Xception

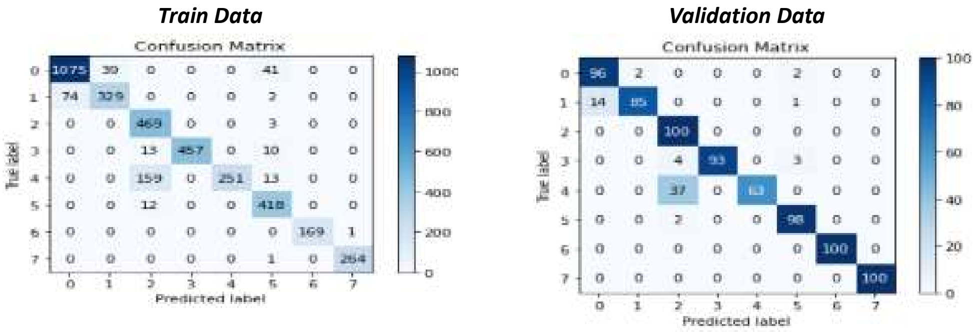


9. MobileNet-V2



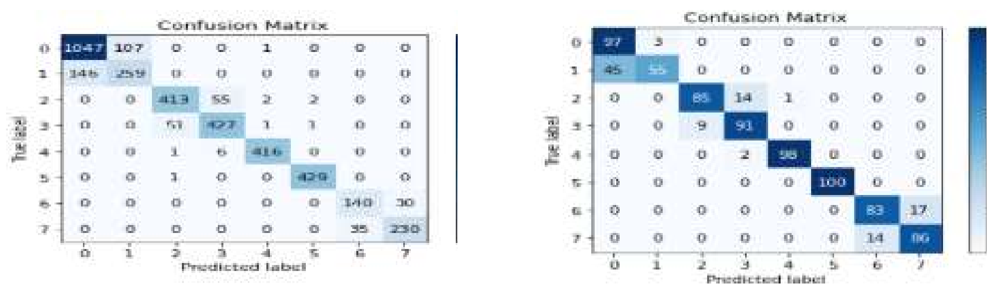
10. FRCNN ResNet-50

Fig. 5. Accuracy (right) and loss (left) plots of 10 convolutional neural networks for training and validation datasets.

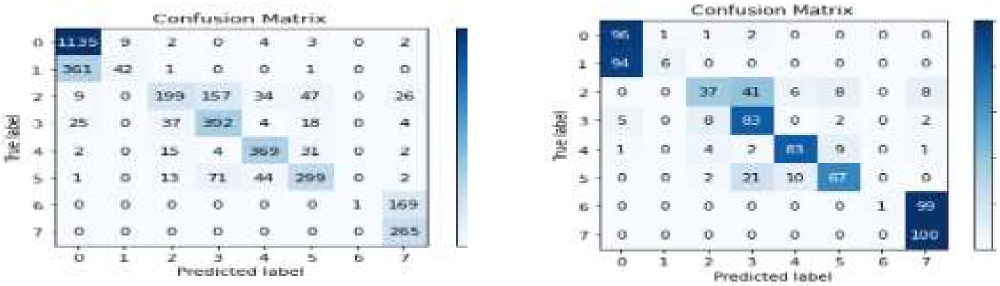


1. AlexNet

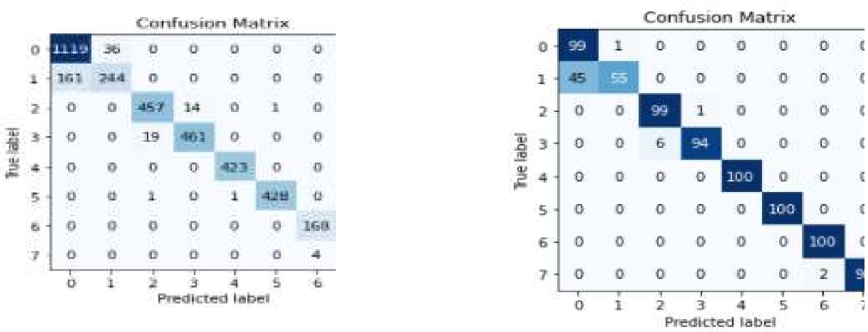
2. VGG-16



3. VGG-19

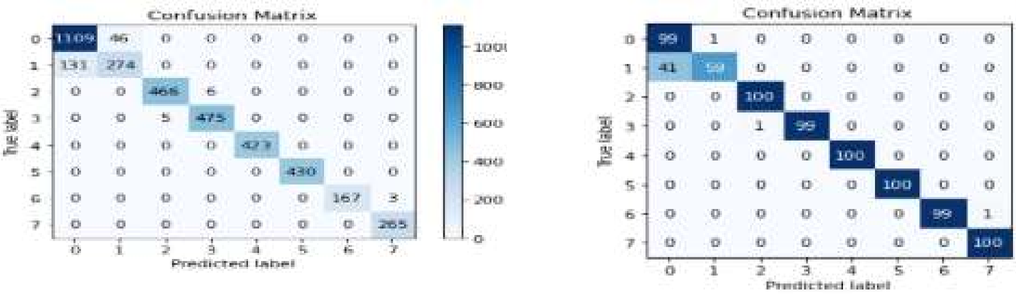


4. ResNet-101

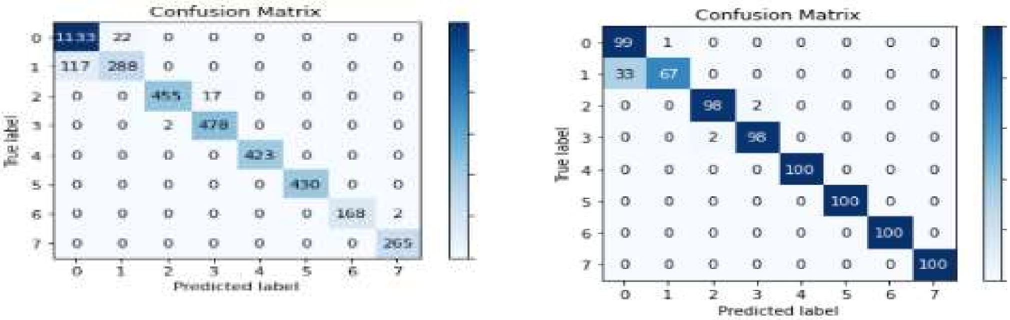


5. DenseNet-101

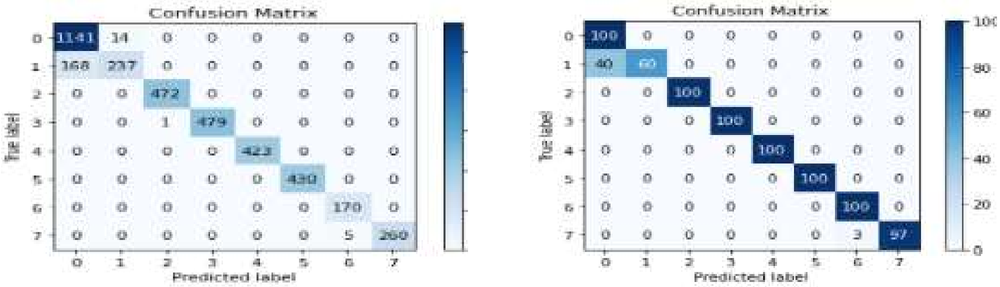
(Fig. 6)



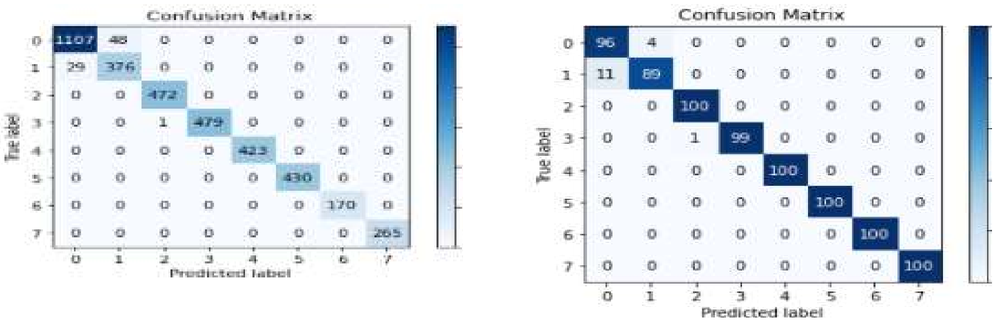
6. NasNet-Mobile



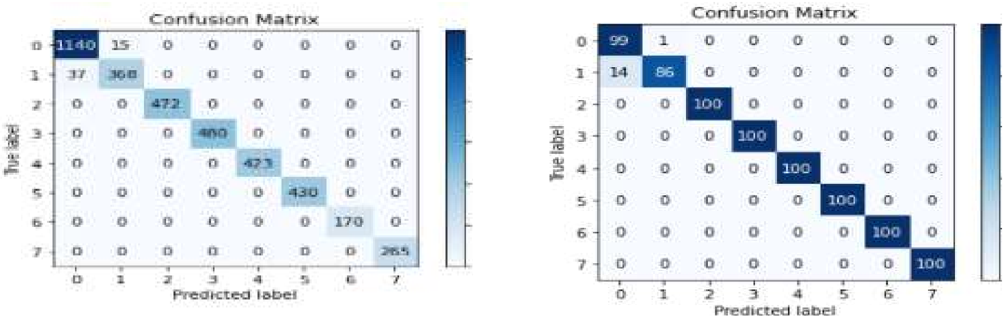
7. Inception-V3



8. Xception



9. MobileNet-V2



10. FRCNN-ResNet-50

Fig. 6. Confusion matrices of 10 convolutional neural networks for training and validation datasets.

Table 5. Classification accuracy of the fruit crop leaves dataset for each class.

Mango Fruit Culti-vars	Accuracy (%)	AlexNet (%)	VGG-16 (%)	VGG-19 (%)	ResNet-101 (%)	DenseNet-121 (%)	NasNet-Mobile (%)	Inception-V3 (%)	Xception (%)	MobileNet-V2 (%)	FRCNN-ResNet-50 (%)
Mallika	100	96	91	97	96	99	100	99	100	96	99
Raspuri	100	85	67	55	68	55	59	67	60	89	86
Arka Arun	100	100	83	85	37	99	100	98	100	100	100
Malgoa	100	93	96	91	83	94	99	98	100	99	100
Badam	100	63	99	98	83	100	100	100	100	100	100
Kesar	100	98	100	100	67	100	100	100	100	100	100
Sindoora	100	100	75	83	100	100	99	100	100	100	100
Alphanso	100	100	98	86	100	98	100	100	97	100	100
Overall Accuracy of the Model	100	91.3	88.6	86.87	59.1	93.1	94.5	95.2 5	94.6	98.0	98.85

and greater PPV when compared to the MobileNet-V2 network. The proposed model has a depth convolution structure, and the network becomes lighter as it goes deeper. Its inverted residual block helps improve efficiency and boosts the robustness of the model.

The MobileNet-V2 network achieved an accuracy of 85.12% using 4,000 images of five different classes of fruits. In one study, fruit images were used to detect the diseases present in the images, and the results were compared with MobileNetV1, InceptionV3, and DenseNet121 (Xiang et al., 2019). The classification accuracy of all eight classes of fruit crops for MobileNet-V2 is as follows: class 0—diseased coffee, 99%; class 1—grape black rot, 86%; class 2—grape esca, 100%; class 3—grape healthy, 100%; class 4—grape leaf blight, 100%; class 5—healthy coffee, 100%; class 6 —diseased mango, 100%; class 7—healthy mango, 100%. A

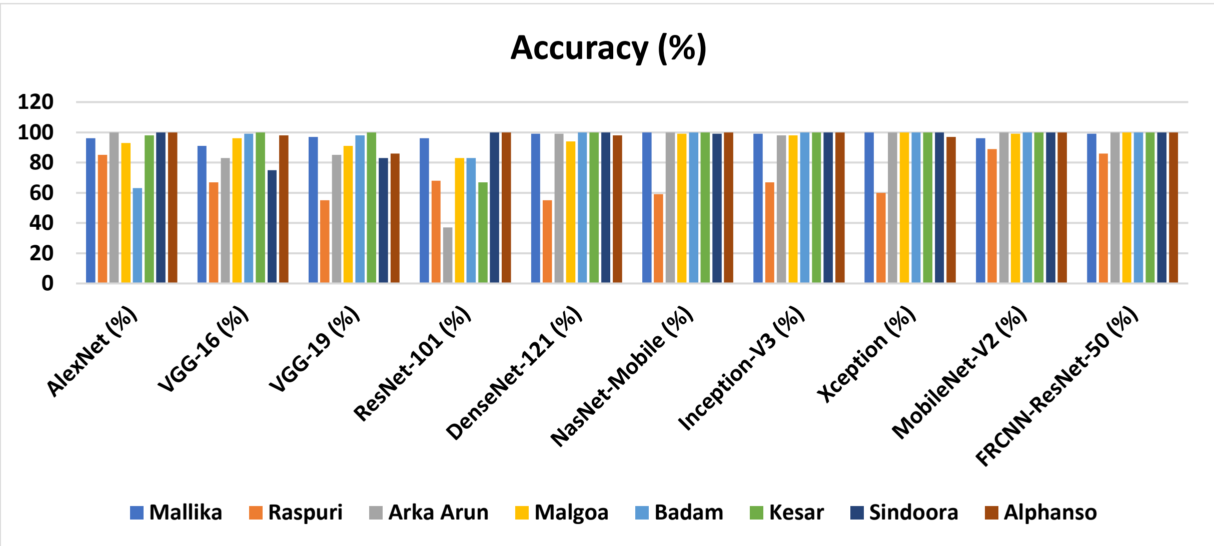
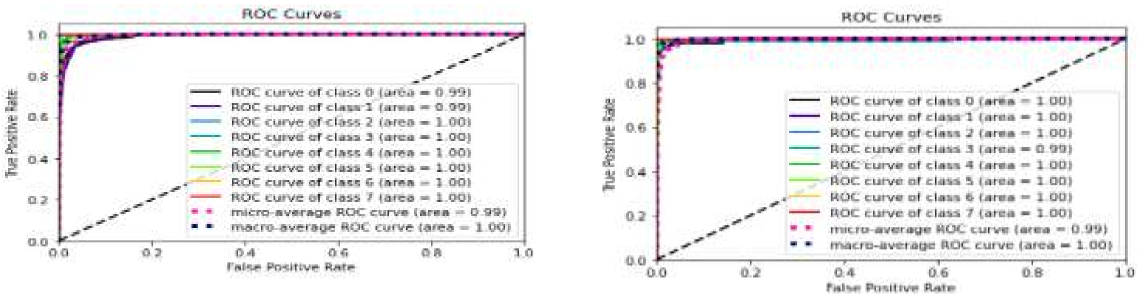
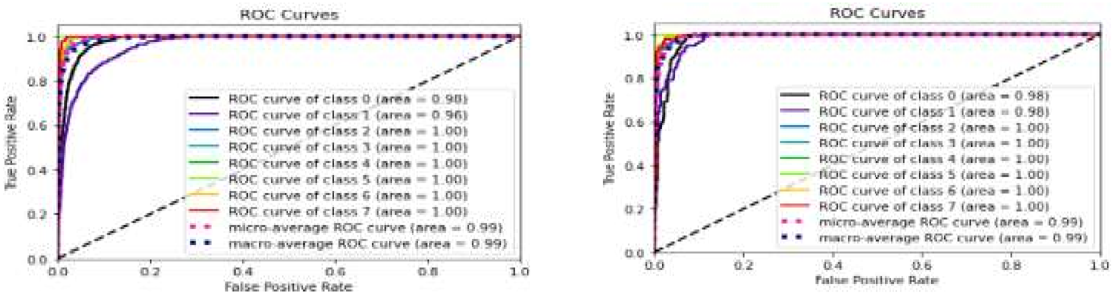


Fig. 7. Classification accuracy chart of the fruit crop leaves dataset for each class.

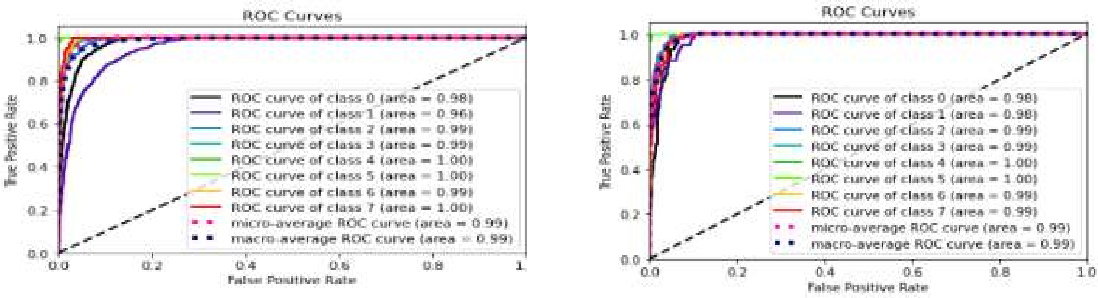
Train Data ***Validation data***



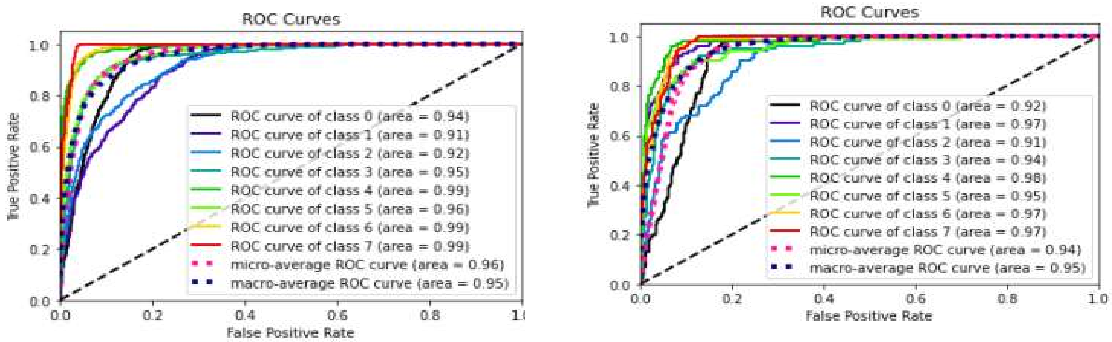
1. AlexNet



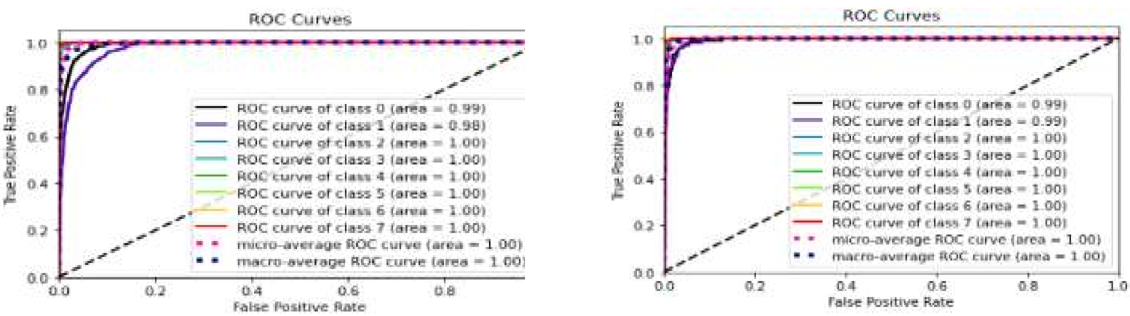
1. VGG-16



2. VGG-19



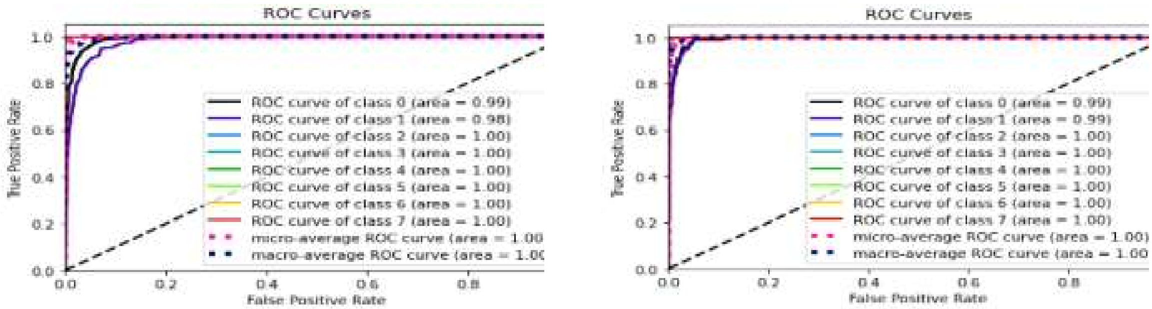
3. ResNet-101



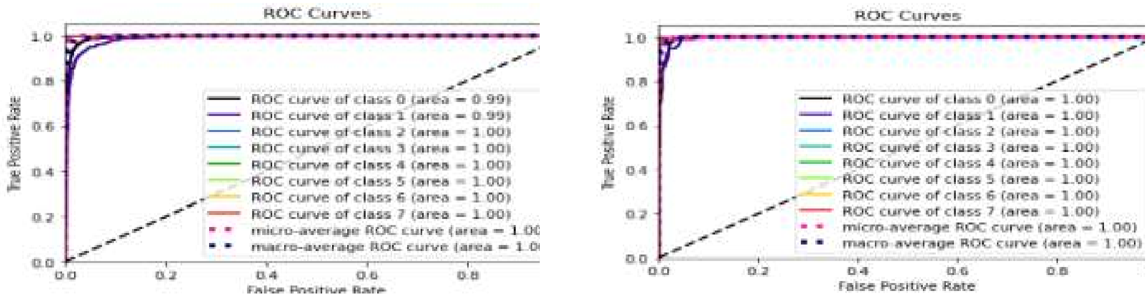
4. DenseNet-101

(Fig. 8)

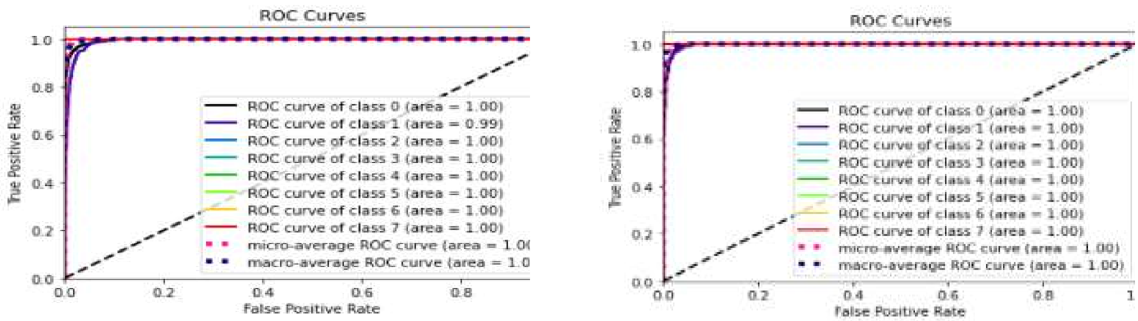
DOI: 10.6977/IJoSI.202502_9(1).0001
M.V. Neethi, P. Raviraj, etc./Int. J. Systematic Innovation, 9(1), 1-18 (2025)



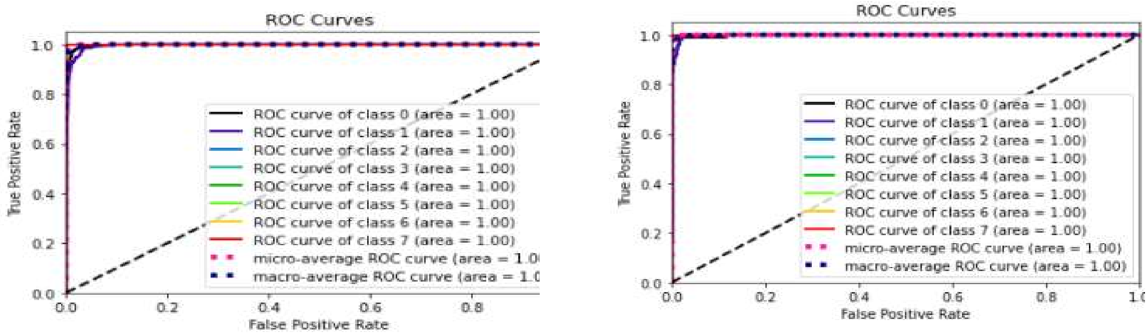
5. NasNet-Mobile



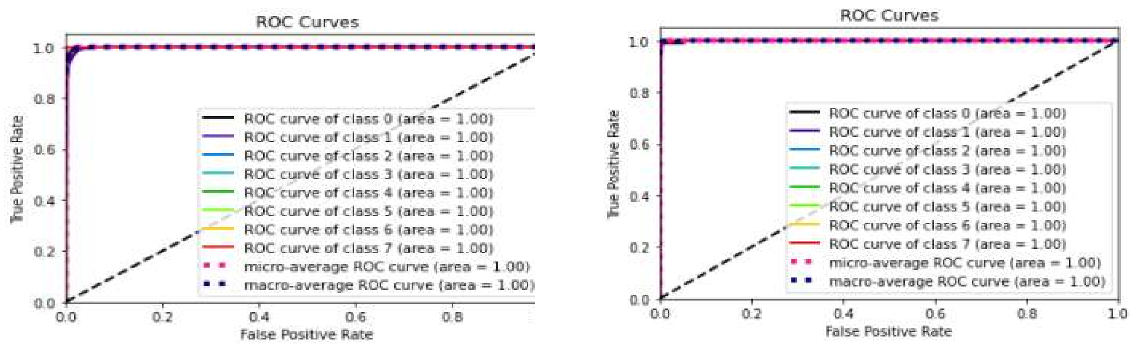
6. Inception-V3



7. Xception



8. MobileNet-V2



9. FRCNN ResNet-50

Fig. 8. Receiver operating characteristic curve of 10 convolutional neural networks for training and validation datasets.

convolutional network was constructed to classify 22 different plant disease classes. The classification accuracy of the network ranged from 33% to 98%, with an average accuracy of 86.2%. The network failed to classify a few classes due to the smaller number of training samples (Dyrmann et al., 2016). The Tomato Diseases and Pests dataset, which contains challenging images of diseases and pests, was used. This dataset includes several inter-and extra-class variations, such as infection status and location in the plant. VGGNet and ResNet were combined to form deep learning meta-architectures formed to train the tomato images, achieving an accuracy of 80% (Fuentes et al., 2017; Saleem et al., 2019). Hybrid convolutional models have been shown to help to detect plant diseases, offering a new direction for plant disease detection (Punam & Gole, 2021), which could be adopted in future studies. Better data retrieval systems assist in retrieving suitable plant disease datasets, and texture techniques with machine learning algorithms can help select the best dataset (Dhingra & Bansal, 2020). The limitation of this study is that the networks could not classify images captured under field conditions, and pre-processing was required for images from field conditions. Additionally, the dataset size should be increased, and more images need to be used for training.

The study provides the following contributions for the literature:

- Introduces a novel FRCNN model that enhances the estimation of mango crop yield and the diagnosis of plant diseases, contributing to the body of knowledge on agricultural applications of CNNs.
- Contributes to the literature by demonstrating how an FRCNN model can achieve higher sensitivity and PPV in plant disease diagnosis compared to other models.
- The study contributes by identifying key limitations in current approaches, such as the need for larger datasets and better handling of field-captured images, which can guide future developments in the field.
- The study suggests that hybrid convolutional models and improved data retrieval systems could significantly advance the field of plant disease detection.

6. Conclusion

An efficient approach to estimating the harvest of mango fruits was proposed in this work. The dataset used to train the proposed approach consisted of images of eight different mango varieties. Nine pre-trained models, of which only the final layer was altered, were compared with the proposed FRCNN model. Each model's classification performance was improved by fine-tuning the model. To improve accuracy and reduce error rates, the models were subjected to validation and optimization. The proposed model's accuracy was determined to be 98.85%. The results demonstrated that pre-trained models were unable to accurately estimate mango yield. As a result, a novel CNN architecture was proposed and applied in this study.

Adapting the CNN model presented in this work improves mango fruit yield estimation performance. This method will assist individuals with limited knowledge of mango crop yield estimation, as it can be difficult to receive proper guidance from agriculturists for manual variety determination and categorization. At the same time, it will enable precise, quick, and reliable classification. By utilizing various deep learning techniques, we hope to expand the scope of this study and include more types and data in the future.

Acknowledgments

The support by GSSS Institute of Engineering and Technology for Women, Affiliated to Visvesvaraya Technological University (VTU), Belagavi, India.

References

Ali, H., Lali, M.I., Nawaz, M.Z., Sharif, M., & Saleem, B.A. (2017). Symptom-based automated detection of citrus diseases using color histogram and textural descriptors. Computers and Electronics in Agriculture, 138, 92–104.

Arsenovic, M., Karanovic, M., Sladojevic, S., Anderla, A., & Stefanović, D. (2019). Solving current limitations of deep learning-based approaches for plant disease detection. Symmetry, 11, 939.

Atila, Ü., Uçar, M., Akyol, K., & Uçar, E. (2021). Plant leaf disease classification using EfficientNet deep learning model. Ecological Informatics, 61, 101182.

Bedi, P., & Gole, P. (2021). Plant disease detection using a hybrid model based on convolutional autoencoder and convolutional neural network. Artificial Intelligence in Agriculture, 5, 90–101.

Brahimi, M., Mahmoudi, S., Boukhalfa, K., &

Moussaoui, A. (2019). Deep interpretable architecture for plant diseases classification. 2019 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), 111–116.

Chollet, F. (2016). Xception: Deep learning with depthwise separable convolutions. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1800–1807.

Dhingra, S., & Bansal, P. (2020). Employing divergent machine learning classifiers to upgrade the preciseness of image retrieval systems. Cybernetics and Information Technologies, 20, 75–85.

Fine, T.L., Lauritzen, S.L., Jordan, M., Lawless, J., & Nair, V. (1999). Feedforward neural network methodology. Springer-Verlag, Berlin, Heidelberg.

Fuentes, A., Yoon, S., Kim, S. C., & Park, D. S. (2017). A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors, 17(9), 2022.

Grinblat, G.L., Uzal, L.C., Larese, M.G., & Granitto, P.M. (2016). Deep learning for plant identification using vein morphological patterns. Computers and Electronics in Agriculture, 127, 418–424.

Gyires-Tóth, B.P., Osváth, M., Papp, D., & Szűcs, G. (2019). Deep learning for plant classification and content-based image retrieval. Cybernetics and Information Technologies, 19(1), 88–100.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770–778.

Heredia, I., Iglesias, L.L., Vykozlov, V., & Orviz, P. (2019). Plants classification engine. DIGITAL.CSIC.

Huang, G., Liu, Z., & Weinberger, K.Q. (2016). Densely connected convolutional networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2261–2269.

Khan, A., Sohail, A., Zahoora, U., & Qureshi, A.S. (2019). A survey of the recent architectures of deep convolutional neural networks. Artificial Intelligence Review, 53, 5455–5516.

Krizhevsky, A., Sutskever, I., & Hinton, G.E. (2012). ImageNet classification with deep convolutional neural networks. Communications of the ACM, 60, 84–90.

Mohanty, S.P., Hughes, D.P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Computers and Electronics in Agriculture, 7, 1419.

Neethi, M.V., & Raviraj, P. (2024). Fast region-based convolutional neural network ResNet-50 model for on-tree mango fruit yield estimation. Indonesian Journal of Electrical Engineering and Computer Science, 33, 1084.

Sandler, M., Howard, A.G., Zhu, M., Zhmoginov, A., & Chen, L. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 4510–4520.

Saleem, M.H., Potgieter, J., & Arif, K.M. (2019). Plant disease detection and classification by deep learning. Plants, 8.

Shi, Y., Huang, W., Luo, J., Huang, L., & Zhou, X. (2017). Detection and discrimination of pests and diseases in winter wheat based on spectral indices and kernel discriminant analysis. Computers and Electronics in Agriculture, 141, 171–180.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. CoRR, abs/1409.1556.

Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., & Stefanovic, D. (2016). Deep neural networks-based recognition of plant diseases by leaf image classification. Computational Intelligence and Neuroscience, 6, 1–11.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2015). Rethinking the inception architecture for computer vision. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2818–2826.

Xiang, Q., Wang, X., Li, R., Zhang, G., Lai, J., & Hu, Q. (2019). Fruit image classification based on MobileNetV2 with transfer learning technique. Proceedings of the 3rd International Conferencen Computer Science and Application Engineering.

Yang, K., Qinami, K., Fei-Fei, L., Deng, J., & Russakovsky, O. (2020). Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the ImageNet hierarchy. Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (FAT '20)*, 547–558.

Yang, J., Sun, J., Ren, Y., Li, S., Ding, S., & Hu, J. (2023). GACP: Graph neural networks with ARMA filters and a parallel CNN for hyperspectral image classification. International

Journal of Digital Earth, 16(1), 1770–1800.

Yang, X., & Tingwei, G. (2017). Machine learning in plant disease research. European Journal of Biomedical Research, 6–9.

Zhang, S., Wu, X., You, Z., & Zhang, L. (2017). Leaf image-based cucumber disease recognition using sparse representation classification. Computers and Electronics in Agriculture, 134, 135–141.

Zhang, X., Han, L., Dong, Y., Shi, Y., Huang, W., Han, L., González-Moreno, P., Ma, H., Ye, H., & Sobeih, T. (2019). A Zhou, D.X. (2020). Theory of deep convolutional neural networks: Downsampling. Neural Networks, 124, 319–327.

deep learning-based approach for automated yellow rust disease detection from high-resolution hyperspectral UAV images. Remote Sensing, 11(13), 1554.

Zhou, Y., & Diamos, G. (2018). Neural architect: A multi-objective neural architecture search with performance prediction. arXiv preprint arXiv:1804.09081.

AUTHOR BIOGRAPHIES

Neethi M. V. completed her doctorate degree in Computer Science and Engineering in the area of machine learning and received an M.Tech. degree in Computer Cognition Technology from University of Mysore, India, in 2015. She also received her B.E. from Visvesvaraya Technological University, India, in 2013. She is currently an assistant professor in the Depertment of Computer Science and Engineering—Data Science at ATMECE, VTU, Karnataka, India. Her research areas include machine learning, data mining, and image processing. She can be contacted at the following email addresses: neethi.m@gmail.com or neethimv_cd@atme.edu.in.

Raviraj P. completed his doctorate degree in Computer Science and Engineering in the area of Image Processing. He holds the position of Director-IQAC and Professor in the Department of Computer Science and Engineering at GSSS Institute of Engineering and Technology for Women, Mysore, Karnataka. He has 19 years of teaching and research experience. He has published more than 92 papers in international journals and conferences. Five research scholars have completed their Ph.D. under his guidance at various universities. He is currently guiding Ph.D. research scholars in the areas of image processing, pervasive and cloud computing, bio-inspired algorithms, and robotics. He has received a project grantof Rs.5 Lakhs from the VGST, Govt. of Karnataka, for the "Underwater Robotic Fish for Surveillance and Pollution monitoring." He has received the awards and recognitions such as the "Rhastriya Gaurav Award, 2015," "Shri P.K. das Memorial Best Faculty Award, 2012," and "Young Achiever Award, 2016." He can be contacted at the email address: raviraj@gsss.edu.in.