Comprehensive Multilayer Convolutional Neural Network for Plant Disease Detection

Radhika Bhagwat[1], Yogesh Dandawate[2]
Department of Technology, Savitribai Phule Pune University Pune, India[1]
Department of Information Technology, MKSSS's Cummins College of Engineering for Women, Pune, India[1]
Electronics and Telecommunication Engineering, Vishwaskarma Institute of Information Technology, Pune, India[2]

Abstract—**Agriculture has a dominant role in the world's economy. However, losses due to crop diseases and pests significantly affect the contribution made by the agricultural sector. Plant diseases and pests recognized at an early stage can help limit the economic losses in agriculture production around the world. In this paper, a comprehensive multilayer convolutional neural network (CMCNN) is developed for plant disease detection that can analyze the visible symptoms on a variety of leaf images like, laboratory images with a plain background, complex images with real field conditions and images of individual disease symptoms or spots. The model performance is evaluated on three public datasets -Plant Village repository having images of the whole leaf with plain background, Plant Village repository with complex background and Digipathos repository with images of lone lesions and spots. Hyperparameters like learning rate, dropout probability, and optimizer are fine-tuned such that the model is capable of classifying various types of input leaf images. The overall classification accuracy of the model in handling laboratory images is 99.85%, real field condition images is 98.16% and for images with individual disease symptoms is 99.6%. The proposed design is also compared with the popular CNN architectures like GoogleNet, VGG16, VGG19 and ResNet50. The experimental results indicate that the suggested generic model has higher robustness in handling various types of leaf images and has better classification capability for plant disease detection. The obtained results suggest the favorable use of the proposed model in a decision support system to identify diseases in several plant species for a large range of leaf images.**

Keywords—Crop diseases; plant disease detection; hyperparameters; deep learning; convolutional neural network

I. Introduction

Agriculture has a huge impact on the economic development of the country. Factors like climatic changes, the ever-increasing population and the widespread of crop diseases highly affect the contribution made by the agricultural sector [1]. Crop diseases are of profound concern, and hence, to control the corresponding losses, timely and effective solutions are very important. However, plant disease detection using visual symptoms is intricate. Due to the huge variety and diversity in plants and their diseases, diagnosis using visual symptoms can lead to misguided treatments. This traditional method is also time-consuming and costly. In this reference, many researchers along with agriculture professionals have suggested numerous automated plant disease detection techniques [2, 3].

The conventional machine learning procedures for automatic plant disease detection utilize multiple stages like pre-processing, segmentation, feature extraction and classification with various image processing approaches used at each stage [4-9]. One of the major constraints of traditional machine learning methods is that they need domain expertise to extract relevant features. In the past decade, developments in areas of computer vision, computing technology, machine learning, etc. led to accelerated progress in multiple applications and over the last few years, deep learning has given a new advancement to the traditional machine learning techniques to overcome part of the complexities in many domains. Deep Learning algorithms learn the relevant features during training from the raw input data thus eliminating the requirement of domain knowledge for feature extraction.

Deep learning approaches are now being predominantly used in various computer vision and pattern recognition applications like healthcare, text or handwriting generation, image recognition, etc. [10, 11]. In agriculture applications too, deep learning methods have gained huge popularity [12, 13], especially in plant disease diagnosis. Authors in [14] suggested a method for rice disease identification using deep Convolutional neural network (CNN) using infected and healthy leaves and stems. The proposed method could distinguish ten rice diseases to achieve an accuracy of 95.48%. Work in [15] used diseased and healthy leaf images taken under controlled conditions to train deep CNNs. They compared two CNN architectures to identify 26 diseases in 14 plant species. While in [16] the author trained several CNN architectures (VGG, Overfeat, AlexNet, GoogLeNet, and AlexNetOWTBn) to detect and diagnose diseases of plants and found that VGG gave the highest classification of 99.53%. Authors in [17] used transfer learning with VGG16 for disease detection in millet crops. Their proposed approach gave a classification accuracy of 95%, precision of 90.5%, 94.5% recall and 91.75% F1 score. The study in [18] used SVM for segmenting the disease symptoms and used VGG16 along with conditional Convolutional generative adversarial network to get 90% classification accuracy for tea leaf diseases. Authors in [19] suggested an infield wheat disease localization and detection procedure with several instance learning techniques. The suggested model gave higher accuracy as compared with two traditional CNN architectures. Fine-tuning the existing six CNN architectures for analyzing their performance for healthy and diseased images of 38 classes is proposed in [20]. The study suggested DenseNet gives better performance compared to other architectures. The use of the Caffe deep learning framework for the recognition of plant diseases was done in [21]. Their model discriminated 13 different infections and achieved 91% to 98% precision for different classes with 96.3% overall average classification accuracy.

Most of the models proposed in the literature are either designed for particular crop species [22-24] or are designed for the specific type of images e.g. images captured under controlled laboratory conditions with a plain background[25,26]. In this work, a comprehensive multilayer convolutional neural network (CMCNN) model is proposed for plant disease detection. The work aims on developing a generic model capable of processing a variety of leaf images like, images that are captured under controlled conditions with a plain background, images taken in uncontrolled conditions with real field complex background and images having lone lesions and spots. For this purpose, the use of three public datasets having healthy and infected leaf images is done. The work thereby aims to overcome all the challenges related to these input images. The proposed deep learning architecture is extensively assessed for various hyperparameters and is fine-tuned to process a variety of leaf images. Experimental results give an average classification accuracy of 99.85%, 98.16% and 99.6% for plain background images, complex background images and for images with lone lesions and spots respectively. The model output is also compared against the state of art techniques and the results show that the proposed CMCNN design outperforms other methods in terms of classification accuracy and computational efficiency. The overall experimental results suggest the potential use of the proposed model for handling a huge variety of input images for efficient plant disease detection.

The remainder of this paper is organized in the following manner: Materials and methods used for experimentation are set out in Section II. The results of the experiments and related discussions are presented in Section III and the Conclusion is contained in Section IV.

II. Materials and Methods

A. Datasets

The proposed work uses three database repositories. First is the Plant Village [27] repository for laboratory conditions images with plain background, second is Plant Village repository for real field condition images with complex background and third is the Digipathos repository (Database for plant disease symptoms (PDDB)) for images with lone lesions and spots., ([28]-[30]). The Plant Village database is divided into two categories to study the model performance on the individual type of images.

The Plant Village dataset having plain background images has 38 classes. This dataset has images that have viewpoint and disease severity variations. The Plant Village dataset with complex background images has 11 classes. The images in this dataset have occlusions, variations in shadows, lighting conditions along with viewpoint and disease severity changes. The Digipathos dataset has 53 classes. It has images of individual lesions and spots indicating the disease symptoms. Fig. 1 shows the sample for laboratory condition images with a plain background, real field condition with complex background and images with lone lesions. Table I gives detailed information about the datasets.

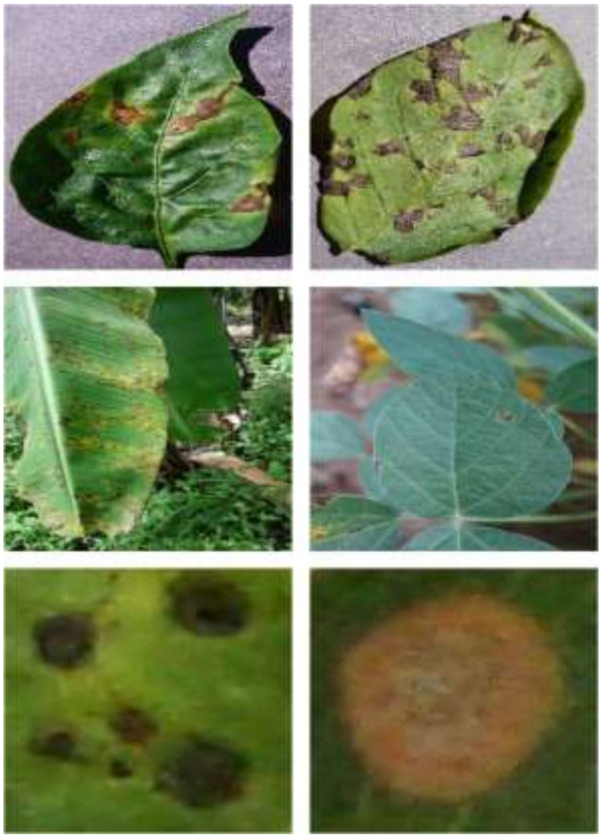

Fig. 1. Sample Images for (a) Plain Background, (b) Complex Background, (c) Lone Lesions.

TABLE I. Information about Datasets

Dataset	Type of Image	Background	Crops	Classes	Diseased Classes	Healthy Classes	Images (Number)
Dataset1 (PlantVillage)	Whole Leaf	Plain	14	38	27	11	54,308
Dataset2 (PlantVillage)	Whole Leaf	Complex	3	11	8	3	26,347
Dataset3 (Digipathos)	Disease symptoms	-	15	53	53	0	43,106

B. Proposed Architecture

Convolutional Neural networks belong to the family of deep learning. The major advantage of CNN lies in its ability to learn the best features for given samples during the training process, as compared to the traditional algorithms that require domain knowledge for creating the feature set. CNN models are normally a stack of Convolutional layers, pooling layer and fully connected layer. The CNN architecture can be configured depending upon the utilization. Several CNN architecture variants like AlexNet, GoogleNet, VGG16, VGG19, ResNet, DenseNet, etc. have been suggested in past few years for various applications. These architectures differ in terms of their structural details.

The paper focuses on developing a comprehensive multilayer CNN architecture for plant disease detection optimized for handling a variety of leaf images. Fig. 2 shows the suggested CNN architecture.

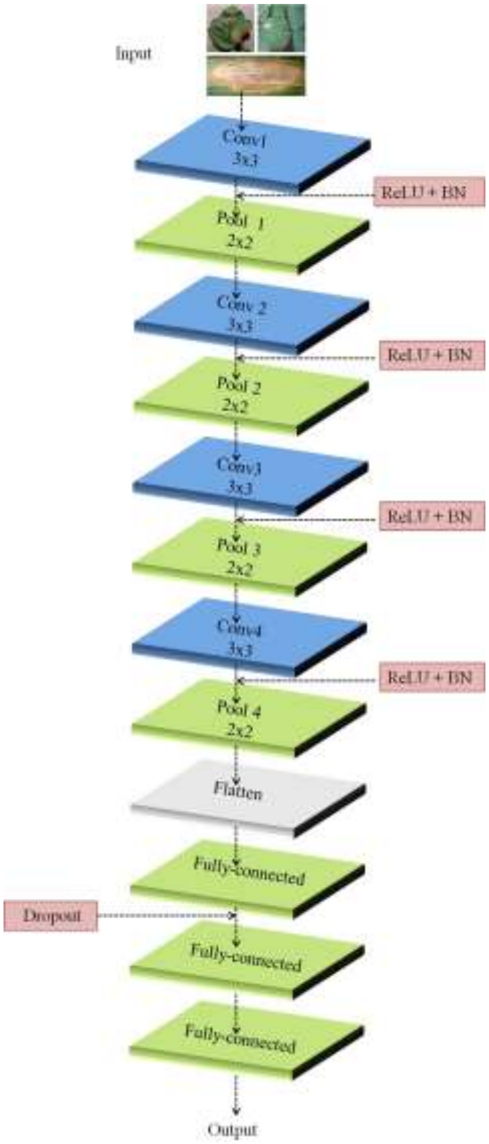

Fig. 2. Proposed CMCNN Architecture.

The model comprises four Convolutional layers each with ReLU activation function, batch normalization (BN) and max-pooling layer and three fully connected (dense) layers with softmax activation for the last dense layer. The convolutional layer is the principal unit of CNN architecture. It is responsible for extracting relevant features from the input data using Convolutional kernels. Initial convolution layers are responsible for capturing the low level features, and the deeper Convolutional layers extract the high level features. This together gives a network that has a detailed understanding of the input images in the dataset. The first convolutional layer used in the proposed work uses 32 filters with a kernel size of 3 while the last convolutional layer uses 192 filters with a kernel size of 3. Thus convolutional layers convolve the input image with several kernels to get various feature maps.

It is then activated by a non-linear activation function that helps to capture complex relations in the data. The convolutional layers in the proposed architecture utilize the ReLU activation function which is represented by:

$$\sigma(x) = \max(0, x) \tag{1}$$

The function returns a zero for any negative value of x, while for any positive input it returns that value. It is the most used activation function as it subdues the vanishing gradient problem and also helps the model to learn fast and give a better performance.

The activation process is succeeded by Batch normalization (BN) and pooling. Batch normalization helps in keeping the input of intermediate layers in the same range throughout the training process to avoid internal covariate shift. Pooling layers lessens the dimensionality of the feature map while retaining the most pertinent features. Two of the most frequently used pooling techniques are Max pooling and Average pooling, however, max-pooling gives better invariant features and helps in convergence [31]. Max-pooling is used in the work with a filter size of 2x2.

After a sequence of Convolutional and pooling layers, the extracted feature map is converted into a 1D array for simple data handling. This is succeeded by fully connected layers where each neuron is attached to every neuron in the next layer. Dropout is used with fully connected layer to avoid overfitting. The final layer in the proposed architecture is the dense or fully connected layer with Softmax classifier whose output is in the form of probabilities representing each class. The expression for softmax function for k classes is as follows:

$$S_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \tag{2}$$

Where z is the input vector to softmax classifier and z_j is the jth element of the vector.

III. Experimental Results

In this work, three database repositories, each with different kinds of images are utilized to assess the generalization capability and performance of the model in handling various challenges (occlusions, illumination variations, viewpoint and disease severity variations) in the input images. The datasets are divided into training, validation and testing sets to better analyze the model. Table II specifies the train, test and validation ratios used for experimentation.

A. Implementation Details

All the experimentation for the proposed model is executed using the Keras, Scikit-learn and OpenCV library with Tensorflow backend using the python programming language with NVIDIA Tesla K80 GPU.

The model training is done in a supervised method. The initial values of the weights and biases are arbitrarily selected and the new values are updated using back-propagation of the gradient. The loss function used in the work is cross-entropy and the optimizer used is Adam. A batch size of 32 is selected. A learning rate of 1e-4 is selected to improve the model fitting. The training, validation and test sets are shuffled randomly to enhance the model stability. Once the processes of training and validation are over, the trained model is checked for the test dataset. The selected parameters and configuration details are given in Table III.

B. Performance Analysis of Datasets

The performance of the developed model is extensively validated on the three image datasets. Table IV, Table V and Table VI show the precision, recall and F1 score obtained for the test datasets of Dataset1, Dataset2 and Dataset3 respectively. The class name in the tables (Plant_Disease) represents the plant along with the disease or (Plant_Healthy) healthy plant.

The weighted average is considered for evaluation due to the imbalanced datasets. Dataset2 with real field complex background obtained the least overall weighted precision of 0.92 as compared to 0.97 and 0.94 for dataset1 and dataset3 respectively.

TABLE II. TRAINING, TESTING AND VALIDATION RATIOS

Database	Type of images	Training set(70%)	Validation set (15%)	Testing set (15%)
Plant Village (Dataset 1)	Whole leaf Laboratory conditions with plain background	38016	8146	8146
Plant Village (Dataset 2)	Whole leaf real field conditions with complex background	18443	3952	3952
Digipathos (Dataset 3)	Lesions and spots	30174	6466	6466

TABLE III. CONFIGURATION AND PARAMETER DETAILS

Parameters	Selected Value
Convolutional Layers	4 (Kernel size= 3x3)
Max Pooling	4 (Kernel size= 2x2)
Activation Function	ReLU
Learning Rate	1e-4
Optimizer	Adam
Batch Size	32
Dropout	0.5

TABLE IV. EVALUATION FOR DATSET1(WHOLE LEAF LABORATORY CONDITIONS WITH PLAIN BACKGROUND)

Class	Class Name	Precision	Recall	F1 Score
P_0	Apple _Scab	0.99	0.95	0.97
P_1	Apple_ Cedar Apple Rust	1	0.84	0.91
P_2	Apple_ Healthy	1	0.9	0.95
P_3	Apple_ Frogeye Spot	0.95	0.98	0.96
P_4	Blueberry_ Healthy	0.97	0.99	0.98
P_5	Cherry_ Healthy	0.95	1	0.98
P_6	Cherry_ Powdery Mildew	1	0.98	0.99
P_7	Corn_ Gray Leaf Spot	0.93	0.88	0.9
P_8	Corn_ Common Rust	1	0.99	1
P_9	Corn_ Healthy	0.93	0.96	0.95
P_10	Corn_ Northern Leaf Blight	0.99	1	1
P_11	Grape_ Black Rot	0.93	0.97	0.95
P_12	Grape_ Esca Black Measles	0.98	0.95	0.96
P_13	Grape_ Healthy	0.99	0.99	0.99
P_14	Grape_ Leaf Blight	0.97	0.98	0.97
P_15	Orange_ Citrus Greening	0.99	1	1
P_16	Peach_ Bacterial Spot	0.94	0.99	0.97
P_17	Peach_ Healthy	0.94	0.94	0.94
P_18	Pepper bell_ Bacterial Spot	0.97	0.96	0.97
P_19	Pepper bell_ Healthy	0.95	0.98	0.96
P_20	Potato_ Early Blight	0.96	0.99	0.98
P_21	Potato_ Healthy	0.94	0.95	0.95
P_22	Potato_ Late Blight	1	0.68	0.81
P_23	Raspberry_ Healthy	0.98	0.94	0.96
P_24	Soybean_ Healthy	0.99	0.99	0.99
P_25	Squash_ Powdery Mildew	0.99	0.99	0.99
P_26	Strawberry_ Healthy	0.99	0.97	0.98
P_27	Strawberry_ Leaf Scorch	0.95	0.98	0.97
P_28	Tomato_ Bacterial Spot	0.99	0.98	0.98
P_29	Tomato_ Early Blight	0.88	0.85	0.86
P_30	Tomato_ Healthy	0.95	0.9	0.92
P_31	Tomato_ Late Blight	0.94	0.96	0.95
P_32	Tomato_ Leaf Mold	0.99	0.93	0.96
P_33	Tomato_ Septoria Leaf Spot	0.93	0.98	0.96
P_34	Tomato_ Spider Mites	0.92	0.93	0.93
P_35	Tomato_ Target Spot	1	1	1
P_36	Tomato_ Mosaic Virus	0.98	0.94	0.96
P_37	Tomato_ Yellow Leaf Curl Virus	1	0.98	0.99
	Weighted Average	**0.97**	**0.97**	**0.97**

TABLE V. Evaluation for Dataset 2 (Whole Leaf Real Field Conditions with Complex Background)

Class	Class Name	Precision	Recall	F1 Score
C_0	Banana_ Black Sigotaka	0.91	0.9	0.9
C_1	Banana_ Healthy	0.94	0.93	0.91
C_2	Banana_ Speckle	0.89	0.89	0.89
C_3	Corn _Healthy	0.92	0.91	0.9
C_4	Corn _Gray leaf spot	0.91	0.89	0.88
C_5	Corn _lethal necrosis disease	0.92	1	0.92
C_6	Corn_ rust	0.91	0.91	0.9
C_7	Soybean_ downy mildew	0.89	0.89	0.89
C_8	Soybean_ frogeye leaf spot	0.93	0.91	0.91
C_9	Soybean_ healthy	0.91	0.89	0.9
C_10	Soybean_ septorial leaf blight	0.95	0.92	0.91
	Weighted Average	**0.92**	**0.91**	**0.91**

TABLE VI. Evaluation for Dataset3 (Lesions and Spots)

Class	Class Name	Precision	Recall	F1 Score
L_0	Cabbage_ Alternaria Leaf Spot	1	0.97	0.97
L_1	Cashew _Algae	0.96	0.86	0.92
L_2	Cashew _Angular Leaf Spot	0.95	0.94	0.93
L_3	Cashew _ Anthracnose	0.92	0.94	0.85
L_4	Cashew_ Black Mould	0.94	0.78	0.87
L_5	Cassava _Bacterial Blight	0.91	0.91	0.86
L_6	Cassava _Green Mite	0.95	0.92	0.93
L_7	Cassava_ White Leaf Spot	0.93	0.87	0.9
L_8	Citrus_ Algae	0.89	0.96	0.91
L_9	Citrus_ Canker	0.95	0.95	0.95
L_10	Citrus_ Greasy Spot	0.96	1	0.96
L_11	Citrus_ Mosaic	0.93	0.92	0.92
L_12	Citrus_ Scab	0.92	0.92	0.92
L_13	Citrus_ Sooty Mold	0.94	0.92	0.93
L_14	Citrus_ Variegated Chlorosis	0.96	0.96	0.96
L_15	Coconut_ Cylindrocladium Leaf Spot	0.91	0.91	0.91
L_16	Coconut _ Lixa Grande	0.92	0.9	0.9
L_17	Coconut Lixa Pequena	0.89	0.84	0.83
L_18	Coffee_ Bacterial Blight	0.91	0.87	0.89
L_19	Coffee_ Blister Spot	1	0.84	0.92
L_20	Coffee_ Rust	0.87	0.92	0.91
L_21	Corn_ Northern Corn Leaf Blight	0.85	0.85	0.83
L_22	Corn_ Phaeosphaeria Leaf Spot	0.93	0.86	0.89
L_23	Corn_ Physoderma Brown Spot	0.92	0.91	0.89
L_24	Corn_Southern Corn Rust	0.89	0.84	0.83
L_25	Corn_ Southern Leaf Blight	0.85	0.84	0.84
L_26	Corn _Tropical Rust	0.89	0.9	0.87
L_27	Cotton_ Areolate Mildew	0.94	0.92	0.93
L_28	Cotton _Myrothesium Leaf Spot	0.9	0.94	0.92
L_29	Dry Bean_ Anthracnose	0.9	0.89	0.89
L_30	Dry Bean_ Hedylepta Indicata	0.89	0.9	0.87
L_31	Dry Bean_ Phytotoxicity	0.96	0.96	0.96
L_32	Dry Bean_ Powdery Mildew	0.78	0.85	0.76
L_33	Dry Bean_ Rust	0.89	0.82	0.82
L_34	Dry Bean_ Target Leaf Spot	0.92	0.91	0.89
L_35	Grapevine_ Bacterial Canker	0.89	0.82	0.82
L_36	Grapevine_ Downy Mildew	0.93	0.85	0.88
L_37	Grapevine_ Powdery Mildew	0.79	0.9	0.84
L_38	Grapevine_ Rust	0.87	0.84	0.85
L_39	Passion_ Fruit Bacterial Spot	0.93	0.91	0.92
L_40	Rice_ Blast	0.89	0.89	0.86
L_41	Soybean_ Bacterial Blight	0.94	0.84	0.86
L_42	Soybean_ Brown Spot	0.84	0.89	0.86
L_43	Soybean_ Downy Mildew	0.95	0.98	0.97
L_44	Soybean_ Mosaic Virus	0.94	0.98	0.96
L_45	Soybean_ Phytotoxicity	0.93	0.97	0.78
L_46	Soybean_ Powdery Mildew	0.81	1	0.89
L_47	Soybean_ Rust	0.89	0.89	0.86
L_48	Sugarcane_ Red Stripe	0.88	0.9	0.88
L_49	Sugarcane_ Ring Spot	0.85	0.87	0.87
L_50	Sugarcane_ Rust	0.89	0.88	0.83
L_51	Wheat_ Powdery Mildew	0.84	0.89	0.86
L_52	Wheat_ Rust	0.88	0.78	0.85
	Weighted Average	**0.94**	**0.93**	**0.93**

This could be due to the real field surroundings, which have varied illumination conditions including partial shadows on the leaves, presence of multiple other objects like, fingers, shoes, hand, etc. along with the leaves in the image. In Dataset 3, the images are of lone lesions and spots, thus having very localized areas of the disease symptoms. Few crops like dry bean powdery mildew, grapevine powdery mildew, or crops like sugarcane rust, wheat rust, soybean rust, dry bean rust and coffee rust have relatively similar disease symptoms and can therefore affect the overall performance leading to misclassifications.

C. Comparison with State of Art Architectures

To further access the potential of the suggested model, it was compared with the other popular CNN architectures like GoogleNet, VGG16, VGG19, ResNet, etc. Fig. 3 shows the comparison of the suggested architecture with the state of art architectures.

As illustrated in Fig. 3, the proposed model gives the maximum accuracy and is succeeded by VGG19, ResNet50, VGG16 and GoogleNet. The proposed model gives a notable increase of 5% on Datbase1 and Database3. It can also be noted that the proposed model outperforms other models on Datbase2 as well, with an increased classification accuracy of 4%. The results prove that the suggested model performs better on all three datasets.

D. *Effect of Model Architecture on Model Efficiency*

This subsection demonstrates the impact of the model architecture on the model performance. The experimental results for database 1 are used for the analysis. Table VII shows four model structures for Convolutional and max-pooling layers (M1, M2, M3, and M4) tested for experimental comparison and the classification results for each model. The structure of the M1 model has two convolutional and max-pooling layers.

The overall accuracy and precision achieved with this model are 99.74% and 0.95, respectively. It is clear from Table VII that as the number of layers is increased the corresponding accuracy and precision increases till we reach a level where there is no further improvement. It has to be noted that the increase in the number of layers makes the model complex and deeper thus increasing the model performance. However, it also increases the computational cost and may also lead to over-fitting. Thus, selecting the number of layers while designing CNN is very critical.

It is evident from Table VII that the maximum accuracy and precision are achieved for the M3 model that has four convolutional and four max-pooling layers, and hence this model is selected as the final model.

The number of filters used for each of the four convolutional layers is 32, 64, 128 and 192, respectively. The number of filters is less initially as they capture the low level features required for differentiating the complex objects in the image while the number increases with the layers to capture more global features.

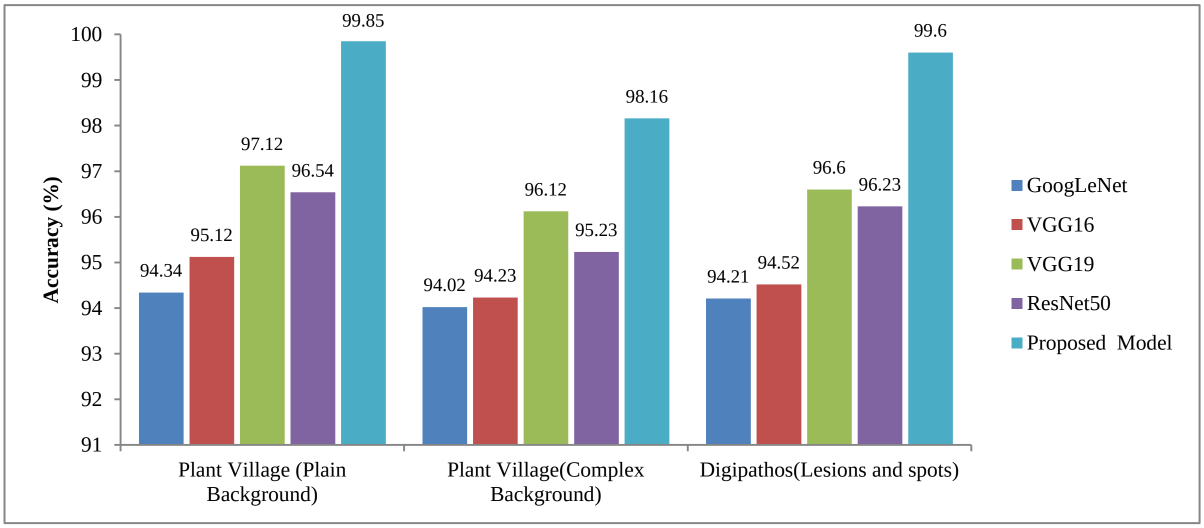

Fig. 3. Comparison of Proposed Architecture with the State of Art Models.

TABLE VII. EFFECT OF MODEL DESIGN ON MODEL PERFORMANCE

Model	**Architecture**										**Accuracy (%)**	**Weighted Precision**
	Conv1	***Max-Pool1***	***Conv2***	***Max-Pool2***	***Conv3***	***Max-Pool3***	***Conv4***	***Max-Pool4***	***Conv5***	***Max-Pool5***		
M1	✓	✓	✓	✓							99.74	0.95
M2	✓	✓	✓	✓	✓	✓					99.77	0.96
M3	✓	✓	✓	✓	✓	✓	✓	✓			99.85	0.97
M4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	99.83	0.97

E. Effect of Hyperparameters on Model Efficiency

Hyperparameters play a key role in the model architecture due to their impact on the performance of the learned model. This subsection demonstrates the impact of hyperparameters like learning rate, dropout, and type of optimizer on the model efficiency. Results of Dataset1 are used for the evaluation.

1) Impact of learning rate: The learning rate regulates the speed at which the model learns by controlling the adjustments made in the weights of the network. A lower learning rate can provide more accurate results but it takes more time to converge while a large learning rate allows fast learning but the weights might not be optimal. Therefore it is essential to select a proper learning rate for the model.

The proposed model is evaluated for different learning rates as illustrated in Fig. 4. It can be noted that the learning rate of 1e-3 shows oscillations in performance. The model performs well for the learning rate of 1e-4 and 1e-5 with the most stable performance at the learning rate of 1e-4 while it gives the lowest accuracy for the learning rate of 1e-6.

2) Impact of Dropout: Dropout helps in preventing overfitting of the model. It is a regularization approach and helps the network in learning more powerful distinguishing features. In the experimentation for selecting the best dropout value, the probabilities are varied from 0.2 to 0.8 as shown in Fig. 5. It can be noted that the test accuracy increases with the increase in the dropout values till it reaches the value of 0.5 and then the accuracy starts decreasing for further dropout values. The dropout value of 0.5 which gives the maximum accuracy of 99.85% is selected for the proposed architecture.

3) Impact of Optimizer: Optimizers improve the weight parameters to give the most accurate outcome possible by minimizing the loss function. Selecting a suitable optimizer is very important for training deep models [32]. The model performance was verified using several optimizers like SGD, Adagrad, Adadelta and Adam. Fig. 6 shows the performance of the model for these optimizers.

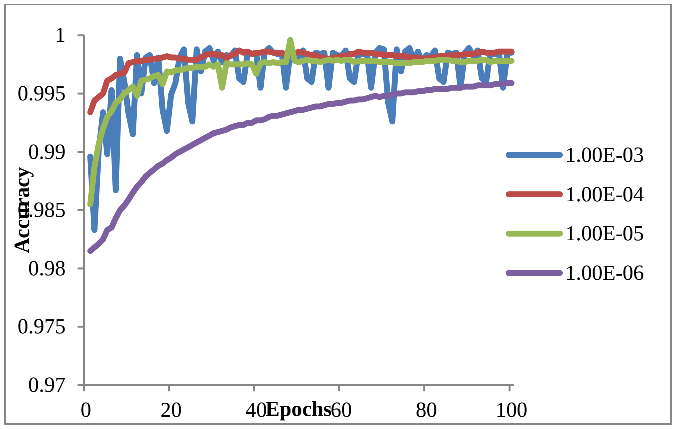

Fig. 4. Impact of Learning Rate on Model Performance.

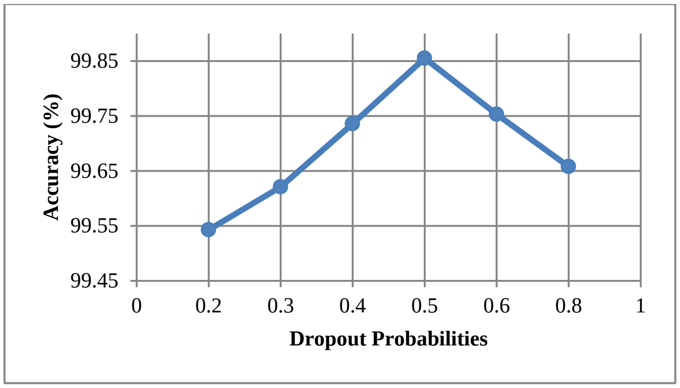

Fig. 5. Impact of Dropout Values on Model Performance.

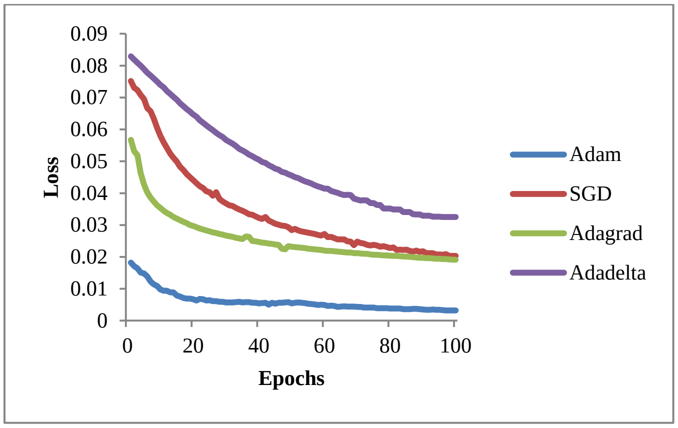

Fig. 6. Impact of Optimizer on Model Performance.

It can be seen that the loss function has a huge gap between Adam and other optimizers and that the Adam optimizer gives the minimum while the Adadelta optimizer giving the maximum loss function. In this work, Adam optimizer was used for training the model.

IV. CONCLUSION

A comprehensive multilayer convolution neural network is proposed in this paper for plant disease detection. To prove the generalization capability and efficiency of the model, three datasets were generated using Plant Village and Digipathos repository, where dataset1 consists of leaf images taken under laboratory conditions with plain background, dataset2 has real field images with complex background while dataset3 has images of lone lesions and spots. The classification accuracy for dataset1, dataset2 and dataset3 achieved is 99.85%, 97.16% and 99.6%, respectively. The model was explored to study the impact of the model architecture and hyperparameters like learning rate, dropout probability and type of optimizer on the model performance. The best hyperparameters were selected for the final optimal architecture. Furthermore, the model is also compared with the state of art techniques. The experimental result proves the superior capability of the proposed CMCNN model in handling various types of leaf images and has better classification efficiency. The obtained results suggest the beneficial use of the proposed model in a decision support system to identify diseases in several plant species for a large range of leaf images.

REFERENCES

[1] Savary, S., Ficke, A., Aubertot, J. N., & Hollier, C. (2012). Crop losses due to diseases and their implications for global food production losses and food security.pp.519-537.

[2] Barbedo, J. G. A. (2013). Digital image processing techniques for detecting, quantifying and classifying plant diseases. *SpringerPlus*, *2*(1), 660.

[3] Sankaran, S., Mishra, A., Ehsani, R., & Davis, C. (2010). A review of advanced techniques for detecting plant diseases. *Computers and Electronics in Agriculture*, *72*(1), 1-13.

[4] Phadikar, S., Sil, J., & Das, A. K. Rice diseases classification using feature selection and rule generation techniques. *Comput. Electron. Agric.* .2013; 90, 76-85.

[5] Sharif, M., Khan, M. A., Iqbal, Z., Azam, M. F., Lali, M. I. U., & Javed, M. Y. Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection. *Comput. Electron. Agric.* .2018; 150, 220-234.

[6] Singh, V., & Misra, A. K. Detection of plant leaf diseases using image segmentation and soft computing techniques. *Information processing in Agriculture*.2017; 4(1), 41-49.

[7] Camargo, A., & Smith, J. S. Image pattern classification for the identification of disease causing agents in plants. Comput. Electron. Agric. . 2009; 66(2), 121-125.

[8] A. Camargo, J.S. Smith. "An image-processing based algorithm to automatically identify plant disease visual symptoms," Biosystems Engineering, vol. 102, no.1, pp.9-21, 2009.

[9] Omrani, E., Khoshnevisan, B., Shamshirband, S., Saboohi, H., Anuar, N. B., & Nasir, M. H. N. M. Potential of radial basis function-based support vector regression for apple disease detection. Measurement.2014; 55, 512-519.

[10] Dargan S, Kumar M, Ayyagari MR, Kumar G. A survey of deep learning and its applications: A new paradigm to machine learning. *Arch. Comput. Methods Eng.*. 2019;1-22.

[11] Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. Deep learning for computer vision: A brief review. *Comput. Intell. Neurosci.*, 2018.

[12] Kamilaris, A., & Prenafeta-Boldú, F. X. Deep learning in agriculture: A survey. *Comput. Electron. Agric.* 2018; 147, 70-90.

[13] N. Duong-Trung, L.-D. Quach, C.-N. Nguyen, "Learning Deep Transferability for Several Agricultural Classification Problems," International Journal of Advanced Computer Science and Applications, 10(1), 2019, doi:10.14569/IJACSA.2019.0100107.

[14] Lu, Y., Yi, S., Zeng, N., Liu, Y., & Zhang, Y. Identification of rice diseases using deep convolutional neural networks. *Neurocomputing*.2017; 267, 378-384.

[15] Mohanty, S. P., Hughes, D. P., & Salathé, M. Using deep learning for image-based plant disease detection. *Front. Plant Sci.*.2016; 7, 1419.

[16] Ferentinos, K. P. Deep learning models for plant disease detection and diagnosis. *Comput. Electron. Agric.* .2018; 145, 311-318.

[17] Coulibaly, S., Kamsu-Foguem, B., Kamissoko, D., & Traore, D. Deep neural networks with transfer learning in millet crop images. *Comput. Ind.*.2019; 108, 115-120.

[18] Hu, G., Wu, H., Zhang, Y., & Wan, M. A low shot learning method for tea leaf's disease identification. *Comput. Electron. Agric.* .2019; 163, 104852.

[19] Lu, J., Hu, J., Zhao, G., Mei, F., & Zhang, C. An in-field automatic wheat disease diagnosis system. *Comput. Electron. Agric.* . 2017; 142, 369-379.

[20] Too, E. C., Yujian, L., Njuki, S., & Yingchun, L. A comparative study of fine-tuning deep learning models for plant disease identification. *Comput. Electron. Agric.* .2019; 161, 272-279.

[21] 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, 2016.

[22] Iqbal, Z., Khan, M. A., Sharif, M., Shah, J. H., ur Rehman, M. H., & Javed, K.An automated detection and classification of citrus plant diseases using image processing techniques: A review. Comput. Electron. Agric. .2018; 153, 12-32.

[23] Selvaraj, M. G., Vergara, A., Ruiz, H., Safari, N., Elayabalan, S., Ocimati, W., & Blomme, G. (2019). AI-powered banana diseases and pest detection. Plant Methods, 15(1), 92.

[24] Jiang, P., Chen, Y., Liu, B., He, D., & Liang, C. (2019). Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks. IEEE Access, 7, 59069-59080

[25] Geetharamani, G., & Pandian, A. Identification of plant leaf diseases using a nine-layer deep convolutional neural network. Comput. Electr. Eng..2019; 76, 323-338.

[26] Ngugi, L. C., Abelwahab, M., & Abo-Zahhad, M. (2020). Recent Advances in Image Processing Techniques for Automated Leaf Pest and Disease Recognition-A Review. Information Processing in Agriculture.

[27] Hughes, D., & Salathé, M. An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*.2015.

[28] Barbedo, J. G. A., Koenigkan, L. V., Halfeld-Vieira, B. A., Costa, R. V., Nechet, K. L., Godoy, C. V., ... & Oliveira, S. A. S. Annotated Plant Pathology Databases for Image-Based Detection and Recognition of Diseases. *IEEE Lat. Am. Trans.*.2018; 16(6), 1749-1757.

[29] Barbedo, J. G. A. Plant disease identification from individual lesions and spots using deep learning. *Biosyst. Eng.*.2019; 180, 96-107.

[30] Barbedo, J. G. A., Koenigkan, L. V., & Santos, T. T. Identifying multiple plant diseases using digital image processing. *Biosyst. Eng.*.2016; 147, 104-116.

[31] Scherer, D., Müller, A., & Behnke, S. Evaluation of pooling operations in convolutional architectures for object recognition. *In* International conference on artificial neural networks. Springer, Berlin, Heidelberg.2010,September; pp. 92-101.

[32] Kawatra, M., Agarwal, S., & Kapur, R. (2020, October). Leaf Disease Detection using Neural Network Hybrid Models. In 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA) (pp. 225-230). IEEE.