Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

Classification of Soybean Leaf Disease from Environment effect Using Fine Tuning Transfer Learning

[1]Santosh Kumar Sahu, [2]Manish Pandey, [3]Kanu Geete

[1]Research Scholar, Department of Computer Science, MANIT, Bhopal (M.P.), INDIA
[2]Assistant Professor, Department of Computer Science, MANIT, Bhopal (M.P.), INDIA
[3]Research Scholar, Department of Computer Science, MANIT, Bhopal (M.P.), INDIA

Abstract
Deep learning Algorithms which involve training a Convolutional Neural Networks, have shownmajordevelopment in solving image classification, object detection, forecasting, segmentation problems and many computer vision problems. Soybean crop is widely cultivated in the world, thus identifying Soybean leaves and its disease can be an important work in deep learning and agriculture area. We have taken soybean dataset for training our deep learning model. The dataset contains 321 images under eight different classes of soybean leaf diseases. The dataset contains small amount of images which is a challenging task to train Deep CNN from scratch. Aiming this problem, transfer learning is used in addition to increasing the dataset size with the help of data augmentation methods. It is observed that dataaugmentation operations generalize the pre-trained CNN models and also increase the classification accuracy. We have compared the performance of six pre-trained deep CNNs with the traditional machine learning. The performance of this work is compared using Accuracy, Precision, Recall, Specificity, F1 score and MCC with the assistance of confusion matrix so as to seek out the best-suited model.Experimental results show that the finetuned ResNet50 network has achieved maximum accuracy among Alexnet, Resnet18, VGG16, VGG19 and GoogleNet i.e. 93% which is better than previous methods.
Keywords: Soybean leaf diseases, pre-Trained Convolutional neural network, Image augmentation, Deep Learning.

1 INTRODUCTION

Agriculture nowadays has become a lot more than just providing food to the ever-growing populations. Soybean is a rich source of various vitamins, minerals, high in protein, and a decent source of both carbs and fat [2]. It has an important place in the world's oilseed cultivation. The production of soybean in India involves two major states namely Madhya Pradesh and Maharashtra. These two states produce 89% of the complete soybean creation in the nation [3].

Diseases related to soybean crops and plantations have also increased which has led researchers to study the effect, detection, and prevention of such diseases. Recognition of plant leaf diseases plays an important role in improving the performance of recent agribusiness as plant diseases can cause production and economic losses in the agriculture field. There are many causes of plant diseases like insect attacks, weather, and environmental conditions, and a lack of knowledge among the farmers about the disease.Small and medium farmers generally fail to identify anomalies to take preventive measures. Studies related to the detection of plant diseases have been increasing rapidly and there can be many ways to detect plant pathologies. Some plant diseases can be detected visually by looking at visible symptoms occurring in several parts of the plant such as leaves, stems, and fruits, etc. Our study is dedicated to detecting diseases related to soybean leaves by Deep Learning (DL) approach [1]. Many studies are aimed at identifying diseases by the means of image processing techniques [4]. These methods generally include the steps of

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

image segmentation followed by extraction of characteristics and then at last the classification of these diseases into one of the predefined disease classes [5]. Along these methods, the methods related to noise reduction in images also helps in achieving better classification accuracy. Several methods have been proposed to identify plants leaf diseases by color, shape, vein, and texture features using classical machine learning techniques like ANNs, Decision Trees, k-nearest neighbors (KNN) algorithm, K-means, and Support Vector Machines (SVMs) [6, 8].

With the rise of deep learning methods, convolutional neural networks (CNN) have been used for detecting plant diseases [9]. CNNs are deep neural networks wherein a minimum of one among the layers of convolution operations is employed rather than a general matrix operation. The training strategies in CNNs are often either from scratch, i.e. creates your network, according to the need of datasets and their size and training for tunning hyperparameters, or with transfer learning [10]. Under transfer learning, various CNN models are introduced since 2012 as AlexNet [11], ResNet [12], VGG [13], googLeNet [14] etc. Transfer learning is usually performed through the utilization of pre-Trained CNN models. A pre-trained CNN is an already trained model on a huge amount of images and labels such as ImageNet dataset that contain 1.2 million images in 1000 classes.With the help of transfer learning, we can adapt the learning feature and apply it to a similar new dataset even when the training dataset is limited.

There have been different ways to use, transfer learning for the dataset either by training from scratch or freezing the learned weights from pre-trained CNN and fine-tuned the hyper-parameters according to the specific problem. In this work, the focus was on fine-tuning hyperparameters and evaluation of pre-Trained deep convolutional neural network for image-based soybean plant leaf disease classification. In the deep learning area, various deep architectures have been proposed with their significant improvement and their application drastically depends on the complexity of real-time problems.

Section II discussion about related work of soybean leaf disease classification using digital image processing with machine learning and deep learning approaches, Section III explains the Data Preparation and Methods. Section IV explains deep convolutional neural networks and transfer learning. Section V emphasizes the experimental analysis of the algorithm on various Pre-Trained CNN. In Section VI results and discussion and finally in Section VII Conclusion and Future Scope of this work are discussed.

2 Related Work

S. Shrivastava in [8], presented a fully automated disease detection and disease severity level estimation approach based on digital image sensing and processing. Create own dataset of soybean leaves from the agriculture filed in guna district of M.P., India. The dataset consists of six types of disease, all the images were converted using YCbCr color space model then apply morphological and thresholding to extract leaf area from the cluttered scene and then extract only the infected area from the segmented leaf. Statistical calculations for disease severity levels are namely, disease severity index, disease level parameter and infected per region. Pires in [15] presented a method to detect soybean diseases using bag of visual words and local descriptors. Dataset were contains 1200 scanned images of four classes namely healthy leaf, rust tan, mildew, and rust RB.Mohanty in [9], used the PlantVillage dataset that contains fourteen crop species and twenty-six diseases. Classification of images based on two pre-trained CNN namely, AlexNet and googLeNet and performed fine-tuning based on three categories of the original dataset namely, color dataset, segmented dataset (leaf area extracted from background), and gray dataset, then performed different division of training and testing like 80-20% 60-40%, 50-50%, 40-60% and 20-80% respectively where the testing accuracy achieved was 99.35%, but when testing model

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

accuracy on new download images from the Internet, the model's accuracy was approximately 32% which is not an acceptable accuracy for practical use.
Lee SH [16], used deep CNN with different datasets and different combination of training and validation sets were applied to tuning model. In 2016 Grinblat et al. used an end-to-end CNN model which was trained using a binary mask that was extracted using leaf veins [17]. Jeon and Rhee applied pre-trained GoogLeNet architecture using transfer learning. During training process the leaf images were given as input and the activation values of different layers were stored as features. Pre-trained models, in general, give more promising results than CNN models which are trained from scratch as they need a very huge amount of data to achieve high accuracy.

From the literature review, digital image processing and color filtering techniques, machine learning, and deep learning techniques are very useful for soybean leaf disease classification. The objective of this work is to generalize the CNN model and improve the classification accuracy of the pre-trained CNN network using various data augmentation techniques and fine-tuning approach for the adopted soybean leaf disease dataset. The remaining section of the paper is organized as following sections.

3Data Preparation and Methods

Here we discuss about the employed dataset and the preprocessing steps taken before the model is trained on the dataset.

3.1 Dataset preparation

In this work, a publicly available soybean plant foliar infection images of eight different classes obtained from Digipathos dataset is used from Brazilian Agricultural Research Agency (Embrapa) [18]. From this repository, the soybean crop along with eight disease images are considered for experimental purposes. All soybean leaf images have been labeled by experienced phytopathologists, thus providing reliable data for training in the developed algorithms.

All Images of this data set are of higher dimension. To make calculation fast we first resized to 320×400 dimensions.The original dataset containing 321 images of 8 different soybean plant foliar infection images with disease labels was used for training and testing pre-trained CNNs. In figure 1 four random samples with their classes are presented. Two versions of dataset are used for training and validating the pre-trained CNN models for parameter fine tuning and evaluation. These datasets are, an original dataset and other one is an augmented dataset. In table 1 we describe both datasets with individual number of images to each class.The original dataset contain very less number of images which is not suitable for training the deep neural network. The deep neural network may not identify many of the features because of availability of relatively lesser number of images which may cause the over fitting problem. In the table 3, it is clearly shown that the accuracy of the original dataset is not satisfactory for a good Disease Classification model because the images in each class is not sufficient to train the pre-trained CNN model which may result in Overfitting [19]. In order to minimize the Overfitting problem and improve the models' accuracy the pre-trained CNN requires large number of images for each class in the dataset [10].

Table 1 Soybean leaf class images

Class Label	Num. of images in the original dataset	Num. of images after data augmentation
Bacterial_blight	51	357
Copper_Phytotoxicity	23	161
Downy_mildew	33	231

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

Powdery_mildew	74	518
Septoria_Brown_Spot	20	140
Soybean_Mosaic	23	161
Soybean_Rust	65	455
Target_spot	32	224
Total Images	321	2247

In this work we increase the training set images by using data augmentation techniques. This dataset contains all the images of similar background and under the common environment which is shown in the Figure 1. In a real environment plant leaf having different background environment and different weather conditions like brightness, darkness, image capture with different angles, scaling factors, zooming affect the images and many more conditions that impact on the plant leaf image pixel values intensity at that time and it may change the image quality. So in this work, we apply digital image processing techniques for image augmentation for handling such condition to generalize the model classification capability and increasing the dataset size to reduce the over fitting problem during training process by adding some distorted images into the training set.

Procedure to image augmentation: before image augmentation operation we first resize the entire images to 1/10 time of actual size of each images using built-in OpenCV resize function using INTER_CUBIC interpolation operation.

In order to increase training data size we used some popular image augmentation techniques likes random flipping, scaling, horizontal and vertical rotation, translation, and insert Gaussian noise [20]. In addition we create synthetic images with darkness and brightness lighting effects on the leaf. We write a python script that used HLS color space model for creating darkness and brightness effects the procedure is as follows:

1. Converting image (RGB color space) to HLS color space [7].
2. Extract channel 1 and changing the pixel values of “Lightness” channel 1 by adding a random coefficient resulting in the brightness of an image that can be changed.

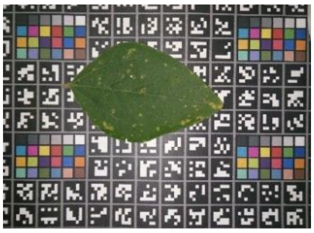
*im_hls[:, :,1] = im_hls[:, :,1] * (np.random.uniform()+ 0.5)*

3. Converting the image back to RGB gives the same image with enhanced lighting effects.

With the help of above mention data augmentation techniques we enlarge the original data set of 2247 images.

(a) Bacterial Blight

(b)Target Spot



(c) Downy Mildew

(d) Soybean Rust

Figure 1. Sample Images from four classes with their disease name

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

4 Deep Convolutional Neural Networks and Transfer Learning

ACNN contains deep neural networks algorithms[1] which is nowadays most commonly applied for images classifications and computer vision areas. It consists of combinations of convolutional layers, activation functions and pooling layer where as each convolutional layer cosists ofsequence of filters which are called convolutional kernels/filters. These filters slide over the image to generate a stack of filtered images. These images then can be fed to another layer known as pooling layer where pooling operations are performed on each individual feature mapped to reduce the spatial size of the these images in order to reduce computations. Convolutional and pooling layers can be stacked interchangeably or in any order dependingupon the network architecture [11]. The output from the last convolutional or pooling layer is flattened and passed to the fully connected layer also called as the output layer. The final output layer which is called as a classification layer is mapped to the total number of classes. Another form of transfer learning could be is utilizing trivial prior information using some other training algorithms for better convergence of the loss function to improve the accuracy on test data [12, 13].

High level features are extracted as input image passes through the series of convolutional and pooling layers. Feature dimensions can be reduced depending on the kernel size. However, for better representation of input image features, the numbersof feature maps are increased as we move towards fully connected layer. The last layer of the network is also called as softmax layer as it contains softmax activation function. CNN network can have many architectures depending upon the ordering of convolutional and pooling layers. Nowadays very popular CNN architectures include AlexNet, ResNet18, ResNet 50, VGG 16, VGG 19, GoogLeNet and Inception V3.

4.1 AlexNet

AlexNet was the revolution for solving image classification problems in the computer vision domain and it was won the ImageNet_Large_Scale Visual Recognition Challenge2012 (ILSVRC) by outperforming many machine learning. AlexNet consists of eight layers including five convolutional layers and pooling layers and three fully connected layers as shown within the figure 2 [11]. Convolutions and pooling operations in the first layer are performed with Local Response Normalization (LRN) where filters of size 11x11 are used. Size of the kernel (filter) used for max pooling operations are 3x3 with stride of 2. Similarly second layer convolutional has 5x5 kernal size and 3x3 kernal are used for third, fourth and fifth layers. In the top of stack two fully connected (FC) layers are used along with dropout layer followed by a Softmax layer.

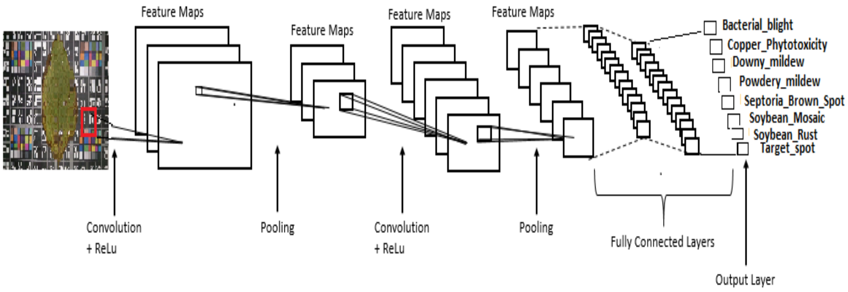


Figure2. Complete CNN architecture for Soybean leaf disease classification procedure

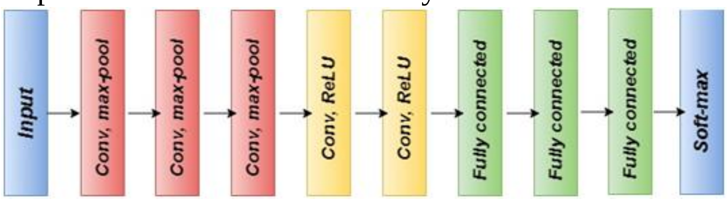


Figure3. AlexNet Layers

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

4.2 ResNet

ResNet or residual network architecture was proposed by Kaiming He and it was the winner of ILSVRC 2015 [12]. The motive was to design deep networks to avoid vanishing gradient problem. ResNet architectures are simple feed-forward neural network with residuals connections having different number of layers such as 18, 50, 152, 1202 etc. For example ResNet18 is 18 layers deep and ResNet50 has 49 convolutional layers and one fully connected layer. The main characteristic of ResNet architectures is skip connections, i.e. instead of learning features it tries to learn residuals which can be understood as subtraction of feature learned from input of that layer. In figure 4 the block diagram of skip connections or shortcut connections shown, the skip connection is directly connect the input of ith layer to some $(i+x)^{th}$ layer.

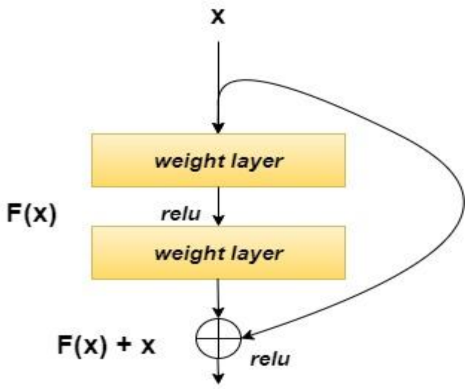


Figure4. Residual block

4.3 GoogLeNet

GoogLeNet architecture or inception V1 module was the winner of ILSVRC 2014 image classification challenge. It consists of twenty-two layers which go deeper in parallel paths with receptive fields of various sizes [13]. The amount of parameters were reduced from 60 million (AlexNet) to 4 million. This architecture uses convolution filter size of 1x1 and instead of Max Pooling used global average pooling, which allows it to travel deeper into the layers.Convolutions and max pooling of size 1x1, 3x3 and 5x5 are performed parallel at input. The output of all these arethen stacked to produce the final output. Filters of different sizes are used for better handling of objects at multiple scales.

4.4 VGG

VGG (Visual Geometry Group) is another very popular sequential pre-trained convolutional neural network.VGG-16, a kind of VGG architecture consists of sixteen layers which include thirteen convolutional layers with filter size of 3x3 with a stride and a padding of size one pixel [17]. These thirteen convolutional layers of VGG-16 are divided into five groups and every group features a max pooling Layer and finally three fully connected Layers. VGG-19 is an extension of VGG-16 architecture with sixteen combinationsof convolutional Layers, max poolingLayer and three fully connectedLayers [17]. The major drawback of VGG network is that it is comparatively slow to train with larger number of network architecture weights. Among all the VGG-E modes, VGG-19 is the most computationally expensive.

5 Experimental Analysis

Here we discuss about conducted experiments on various pre-trained CNN. Hyper parameters, tuning and results explanation and their comparison with previous work is given below.The adopted pre-trained CNN models are AlexNet, VGG16, VGG19, ResNet 18, ResNet50, and GoogeLeNet each model require the following steps:

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

1. Load the dataset and split it into train, validation and test set.
2. Preprocess the images and normalize.
3. Apply data-augmentation operations to increase dataset size.
4. Resize input images according to the input layer of pre-trained CNN model.
5. Train the model on training set with the help of parameter and hyper-parameter tuning.
6. Validate the training accuracy by the comparing the test set.
7. Various pre-trained CNN models are being evaluated using confusion matrix and various accuracy parameters.

The dataset having images of higher dimensions to training the model we need to be resized them according to the network input layer size of the pre-trained CNN model. The input layer sizes of various pre-Trained CNNs are 227 x 227 for AlexNet, 224 x 224 for ResNet18 and ResNet50, VGG16, VGG19 and googLeNet. Then after preprocessing the input images to reduce noise from the images andfurther normalizing the input images for the model, it converges more quickly while given better generalization performance on test data. Data augmentation is playing major roles in the field of deep learning and especially in the case of less amount of dataset size andimbalance dataset. Various methods are used in this works which are already discussed in the section III.

Fine tuning of parameters is greatly affected by various hyper-parameters such as batch size, preprocessing steps, noise, etc. The pre-trainned CNN models considered in this paper are AlexNet, ResNet18, ResNet50, VGG16, VGG19, and GoogLeNet. It is observed that bigger networks like Google Net and VGG19 shows slightly lower accuracy then ResNet50. However, it also performed better than AlexNet which is a relatively small network. Since the dataset is not very large, the larger networks like GoogLeNet which have more than 1.3 million parameters in number may not converge to optimal minima in the loss space [23]. Still, the available data are utilized to present significant results.

The dataset contains 2247 images containing leaves with similar kind of backgrounds which are classified under 8 labels. The dataset is further extended with various augmentation techniques. Data augmentation is a well known regularization technique which reduces the chances of overfitting of the model on training data by slightly varying the sample distribution of the dataset. Data augmentation can improve the overall performance with a slight overhead of execution time [20]. The comparative result presented in this paper supports the above statements.

The experiments are performed in the matlab2018b using an nVidia GPU Quadro K4200 Windows 10 Operating system. The system has 8 GB RAM which is suitable for the implemented algorithm. Before start training the dataset set is divide into training set, validation set and testing set. Learning process over the training-set by the model, validation set is used to fine-tuning the hyper parameters and test set is used to evaluate the classification accuracy of the models. To evaluate the influence of training and testing performance we split dataset into three fashions for training – testing data (80%-10%-10%, 70%-20%-10%, and 60%-20%-20%). It is observed that the best combination of training-validation-testing was 70%-20%-10%. It is observed that pre-Trained CNNs pose certain benefits which are incidental in the results. Transfer learning is applied in this work by deleting the last layer and add new classification layer and mapped it with the 8 number of disease classes.The rest of the layers of pre-Trained neuralnetworks are trained during taraining process. The graphs in the Figure 6-11 shows the entire training process in which the training loss regularly decreases on the y axis with increasing the number of epochs on the x axis on both testing and training data. The upper part of graph depicts the accuracy captured in each epoch and the lower part shows loss.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

Figure 12 shows the confusion matrix for finetuned ResNet50 CNN the datasets explained in section IV. It is observed that data segmentation also handled data imbalance to some extent. Data imbalance is major problem in a lot of areas and often handled with sophisticated methods. The proposed work can be extended in future in this direction.

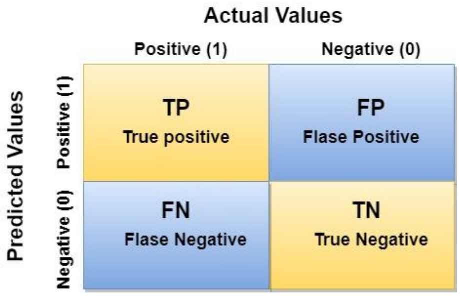
6 Results and Discussion

With Transfer Learning, less time is required to train the model and disk storage requirements to implement a model of various architectures which are substantially lower than other pre-Trained CNN models. The parameters which are calculated in order to show the analysis of the implemented technique are explained below. All the parameters are standard metrics utilized in the classification task.

Confusion matrix

Confusion matrix is employed to describe the model performanceon a set of test data that actual values are known [22]. It helps in visualising algorithm performance because it provides the summary of prediction results. It shows the errors made by the classification model and also about the kinds of errors. Table 3 shows a confusion matrix.

Table 2 Confusion matrix



Accuracy

"Accuracy or classification accuracy is simply the ratio of the total number of correct predictions (True Positive and True Negative) made by the model to the total number or predictions" [22]. Accuracy is given by the following relation as shown in equation 1. In equation TP is the true positive

$$\boldsymbol{Accuracy = {TP + TN}/{TP + TN + FP + FN}} \qquad (1)$$

Recall

"Recall or sensitivity of the model is measured as the proportion of total number of classes that have been classified correctly as positive to total number of positive classes. Recall is calculated using the equation 2" [22]. A high value of recall means there is a higher chance that our classifier recognizes the classes correctly.

$$\boldsymbol{Recall = {TP}/{TP + FN}} \qquad (2)$$

Precision

Precision is that the fraction of the entire number of classes that are classified correctly as positive to total number of positive classes as predicted by our classifier [22]. High precision guarantees that the class labeled by our classifier as positive is indeed positive. The formula for precision is given in equation 3.

$$\boldsymbol{Precision = {TP}/{TP + FP}} \qquad (3)$$

F1-score

F1 score or F-measure is given in equation 4. It gives the model performance using both precision and recall. It can be understood as the harmonic mean of precision and recall [22]. F1 score helps to make two models comparable.

$$\boldsymbol{F1\ score = {2 * Recall * Precision}/{Recall + Precision}} \qquad (4)$$

Specificity

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 2188 - 2201
Received 16 February 2021; Accepted 08 March 2021.

Specificity or true negative rate gives us the ratio of negative classes predicted by our classification algorithm to the total number of negative classes [22].
The sum of specificity and false positive rate always equals to 1. Formula for specificity is given in equation 5.

$$\boldsymbol{Specificity} = {}^{\boldsymbol{TN}}\!/_{\boldsymbol{TN+FP}} \qquad (5)$$

MCC

MCC or Matthews correlation coefficient gives the efficiency of binary and multiclass classification. The value of MCC lies between -1 to +1 where +1 signifies a perfect prediction and -1 signifies inverse prediction. MCC is a balanced measure which can be applied even if we have classes of different sizes.

The comparative results calculated on the above parameters are given within the Table 3 and Table 4. In Table 4 pre- Trained CNNs are evaluated on the original dataset and whereas in Table 5 pre-Trained CNNs are evaluated on the augmented dataset that contains 2247 images where training set contains 1572 validation set contains 450 and test set contains 225. It is observed that ResNet50 has shown better accuracy on both the Original Dataset as well as on the Augmented Dataset. Figure 5 shows the final test accuracy on the four random sample image from the test set.

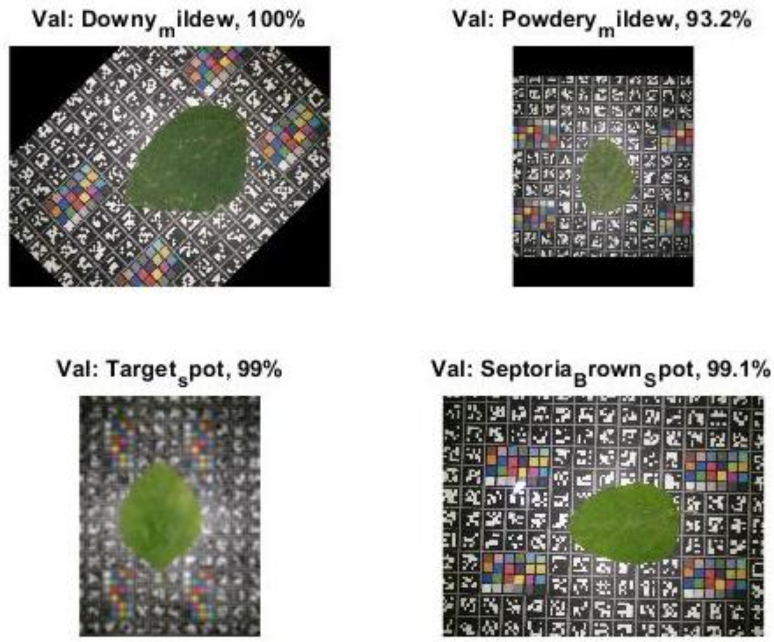


Figure 5. ResNet50 Test Results for random images

Table 3 Pre-Trained CNN Results on without (original) dataset

CNN Net	Accuracy	Precision	Recall	Specificity	F1-Score	MCC
AlexNet	0.8333	0.8167	0.7809	0.9749	0.8702	0.7643
ResNet18	0.8333	0.8423	0.753	0.9757	0.8289	0.7634
ResNet50	**0.8667**	**0.8862**	**0.8542**	**0.9795**	**0.9307**	**0.8478**
VGG16	0.8333	0.8664	0.7875	0.9753	0.8388	0.7923
VGG19	0.8462	0.8283	0.8365	0.9777	0.8822	0.8089
GoogLeNet	0.8333	0.8688	0.7917	0.9763	0.8646	0.7947

The finetuned CNNs results are presented in table 5 within the proposed column of the table 5 and therefore, the obtained results is additionally compared with Barbedo [4] obtained accuracy values 86% for soybean disease classification for an equivalent dataset and other comparison with Araujo [21] for the similar objectives and use an equivalent dataset

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 1105 - 1112
Received 16 February 2021; Accepted 08 March 2021.

achieve the simplest accuracy 75.8% using SVM classifier in Table 5. It is clear from the comparison that data augmentation can simply increase the overall performance of a machine learning model. By adding transfer learning and data augmentation, the performance is improved too much extent.

Table 4 Pre-Trained CNN Results on Augmented dataset

CNN	**Accuracy**	**Precision**	**Recall**	**Specificity**	**F1-Score**	**MCC**
AlexNet	0.9062	0.8872	0.8678	0.9802	0.8773	0.8478
ResNet18	0.9107	0.9126	0.8986	0.9867	0.9055	0.8917
ResNet50	**0.9302**	**0.9341**	**0.9267**	**0.9895**	**0.93**	**0.9103**
VGG16	0.9196	0.9239	0.9067	0.98810	0.9152	0.9026
VGG19	0.9254	0.9158	0.9178	0.9878	0.9168	0.9014
Googlenet	0.9286	0.9266	0.9006	0.9895	0.9134	0.9022

Table 5 Comparison of the proposed work with some previous works

	Proposed	**Barbedo [4]**	**Araujo[21]**
Accuracy	93%	86%	75.8%

Figure 6- 11. Pre-Trained CNN models comparison on size, parameters and training time

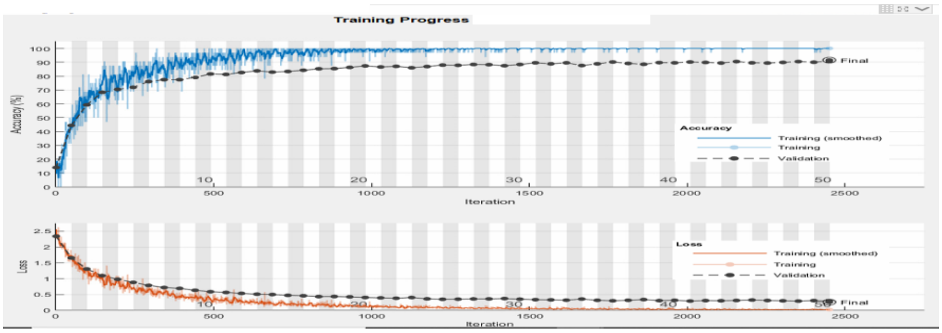


Figure 6: Finetuned AlexNet

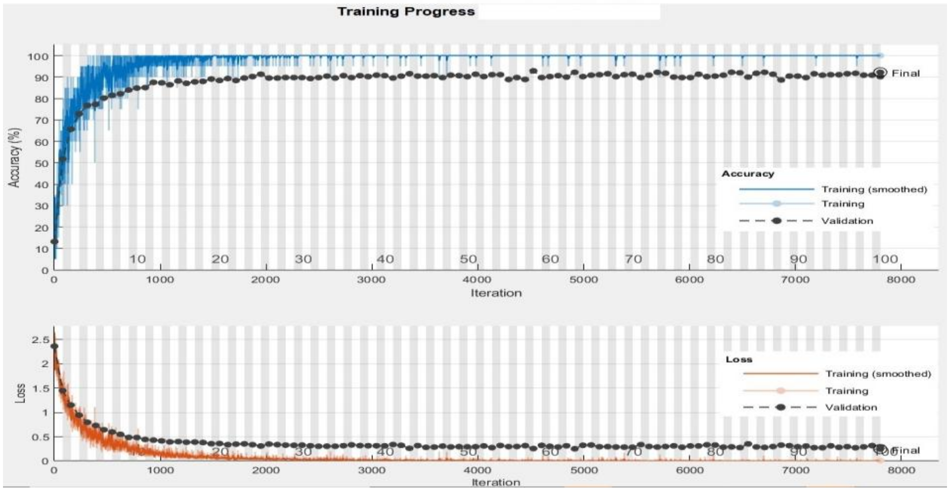


Figure 7: Finetuned ResNet 18

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 1105 - 1112
Received 16 February 2021; Accepted 08 March 2021.

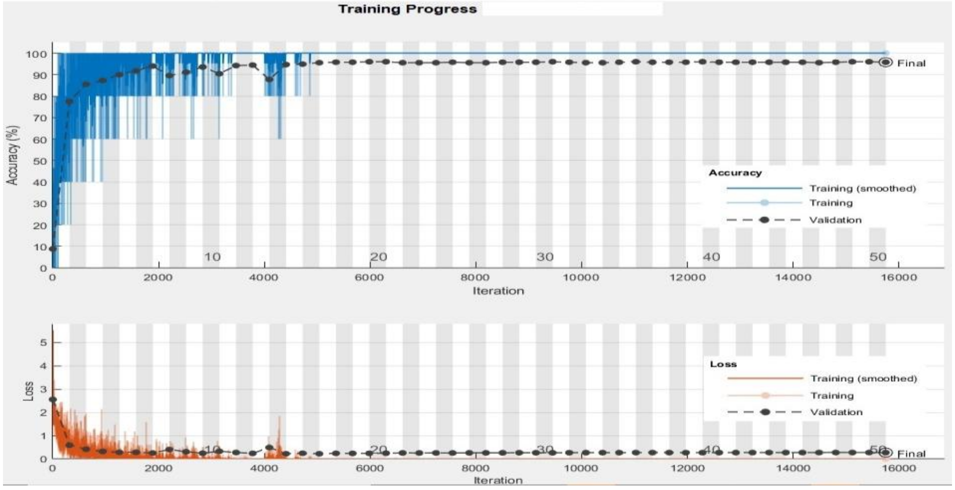


Figure 8: Finetuned ResNet 50

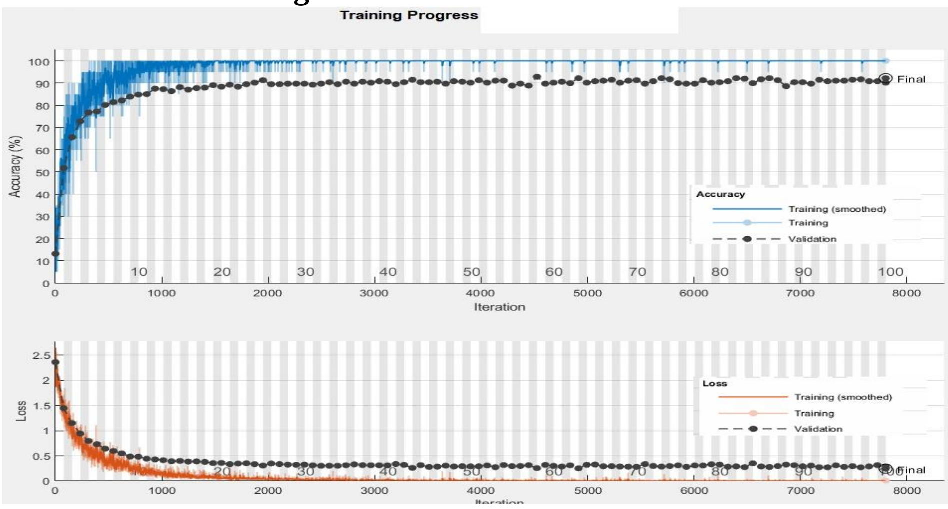


Figure 9: Finetuned VGG 16

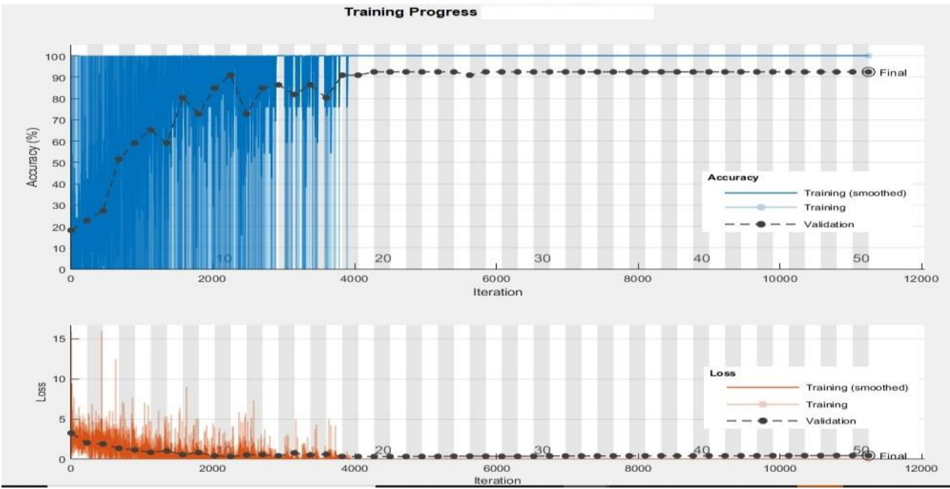


Figure 10: Finetuned VGG 19

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 1105 - 1112
Received 16 February 2021; Accepted 08 March 2021.

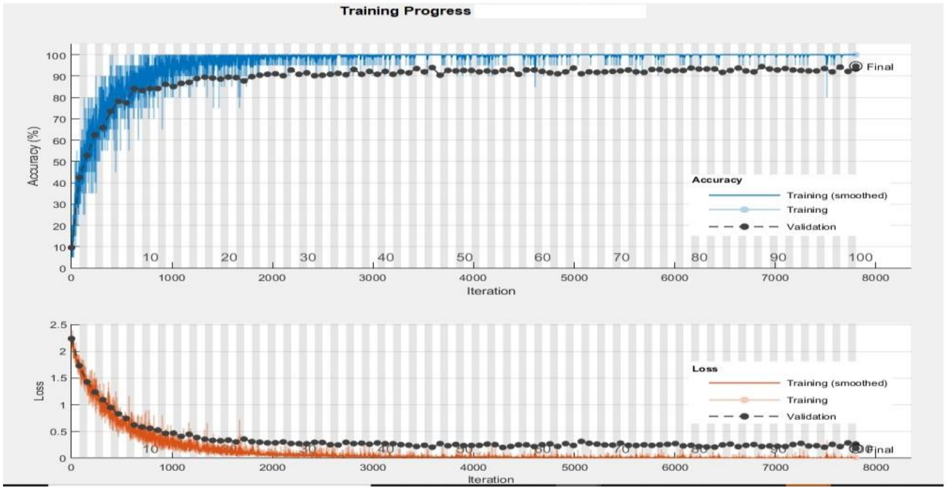


Figure 11: Finetuned GoogleNet

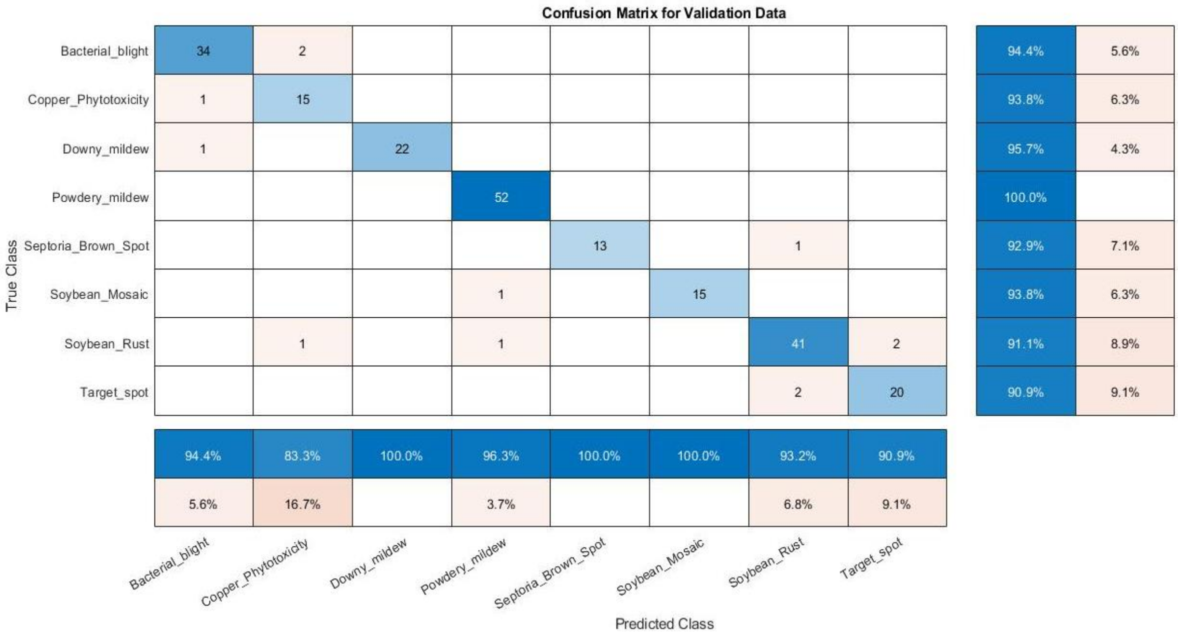


Figure 12: Confusion Matrix of Finetuned ResNet50

7Conclusions and Future Scope

This paper presented a comparative analysis on the application of the transfer learning using fine-tuned pre-Trained CNNs models for well known open dataset of the soybean leaf diseases which contain several soybean leaf diseases classes in which we took only eight of their classes namely, bacterial_Blight, powdery_Mildew, soybean_Rust, copper_Phytotoxicity, soybean_Mosaic, downy_Mildew, Septoria_Brown spot and target_Spot. In this work, we analyze that the original dataset contains less number of images in each classes that often were too less number of training samples for the CNN model. Thus,increasing the samples using data-augmentation technique improves the performance of the model. Our proposed work has been tested on the six types of finetuned pretrained CNN and found that the ResNet50 got the highest accuracy among the others and it has also shown up to 8% and 24% increment respectively within the accuracy over the previous works, as presented within the paper. Also, we have implemented our work on well known pre-trained networks in order to evaluate the efficiency in different scenarios. For future work, the proposed work may be extended for handling class imbalance problems in various domains. Also, the dependency on data augmentation for improvement may be decreased in future works as it poses some level of overhead in the training process.

Acknowledgements

We are very thankful to all the faculty members and research scholars of MANIT, Bhopal, India, for providing a lot of encouragement and support to finish this work.

Conflict of Interest

The authors declare that they have no conflict of interests in this manuscript. This manuscript has not been submitted to, nor is under review in another journal/conference or other publishers.

References

[1] Y. LeCun, Y. Bengio & G. Hinton, “Deep Learning”, Nature, vol. 521, no. 7553, pp. 436–444, 2015.
[2] https://extension.umn.edu/soybean/soybean-pest-management.
[3] A. Westphal, L. Xing, TS. Abney, RG Shane, “Sudden death syndrome Diseases of soybean”, Purdue University, 2006.
[4] J.G.A. Barbedo, L.V. Koenigkan, T.T. Santos, “Identifying multiple plant diseases using digital image processing, Biosystems Engineering”, vol. 147, p. 104-116, 2016.
[5] A. Kadir, L.E. Nugroho, A. Susanto, P.I. Santosa, “Leaf classification using shape, color and texture features”, arXiv:1401.4447, 2013.
[6] R. Calderon, J.A. Navas-Cortes, P.J. Zarco-Tejada, “Early detection and quantification of verticillium wilt in olive using hyperspectral and thermal imagery over large areas”, Remote Sensor, 7(5): 5584–610, 2015.
[7] Y. Guo, S. Han, Y. Li, C. Zhang, YuBai, “K-Nearest neighbor combined with guided filter for hyperspectral image classification”, Procedia Comput Sci, 129: 159–65, 2018.
[8] S. Shrivastava, S.K. Singh, D.S. Hooda, “Color sensing and image processing-based automatic soybean plant foliar disease severity detection and estimation”, Multimedia Tools Appl., 74 (24), 11467–11484, 2015.
[9] S.P. Mohanty, D.P. Hughes, M. Salathe, “Using deep learning for image-based plant disease detection”, Front Plant Sci, 7:1419. doi: 10.3389/fpls.2016.01419, 2016.
[10] N. Kajbakhsh, J. Y. Shin, S. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang. “Convolutional neural networks for medical image analysis: Full training or fine tuning?” IEEE Transactions on Medical Imaging. 35(5):1299-1312, 2016
[11] Alex Krizhevsky, Ilya Sutskever, & Geoffrey E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, Advances in neural information processing systems, 2012.
[12] He Kaiming, Xiangyu Zhang, Shaoqing Ren, Jian Sun, “Deep Residual Learning for Image Recognition”, arXiv:1512.03385, 2015.
[13] Karen Simonyan, Andrew Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, arXiv:1409.1556, 2015.
[14] Christian Szegedy, Wei Liu, Yangqing Jia, “Going Deeper with Convolutions”, CVPR2015 Computer Vision Foundation, 2015.

[15] R.D.L. Pires, D.N. Gonçalves, J.P.M. Orue, W.E.S. Kanashiro, "Local descriptors for soybean disease recognition", Comput. Electron. Agric. 125, 48–55, 2016.
[16] SH. Lee, CS. Chan, P. Wilkin, P. Remagnino, "Deep-plant: plant identification with convolutional neural networks", IEEE International Conference on Image Processing, p.452–6, 2015.
[17] L. G. Grinblat, LC. Uzal, LC. MG. Larese, MG. Pablo, "Deep learning for plant identification using vein morphological patterns", Computer Electron Agric, 127:418–24, 2016.
[18] https://www.digipathos-rep.cnptia.embrapa.br/.
[19] J. Shijie, W. Ping, Jia Peiyi, Hu Siping, "Research on data augmentation for image classification based on convolution neural networks", Chinese Automation Congress, pp. 4165-4170, 2017.
[20] J. Wang, L. Perez, "The effectiveness of data augmentation in image Classification using deep learning", IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8, 2017.
[21] J.M.M. Araujo, Z.M.A. Pelxoto, "A new proposal for automatic identification of multiple soybean diseases", Computers and Electronics in Agriculture, 167, 105060, 2019.
[22] A. Kulkarni, D. Chong, F.A. Batarsehm, "Foundations of data imbalance and solutions for a data democracy", At the Nexus of Artificial Intelligence, Software Development, and Knowledge Engineering, Pages 83-106, 2020.
[23] Kanu Geete, M. Pandey, "A Noise-Based Stabilizer for Convolutional Neural Networks", Journal of Statistical Computation and Simulation; 89:11, 2102-2120, 2019.