Identifying Plant-diseases using Deep Convolutional Neural Networks

Desai Sunny[1][0000-0002-3525-4874] (✉), Rikin Nayak[2], Ronakkumar Patel[3]

[1] Charotar University of Science & Technology, Changa, Gujarat, India
d16ce165@charusat.edu.in
[2] Assistant Professor, V.T. Patel Department of E & C, Chandubhai S. Patel Institute of Technology, Charotar University of Science & Technology, Changa, Gujarat, India
rikinnayak.ec@charusat.ac.in
[3] Assistant Professor, U & P U. Department of Computer Engg, Chandubhai S. Patel Institute of Technology, Charotar University of Science & Technology, Changa, Gujarat, India
ronakpatel.ce@charusat.ac.in

Abstract. In this paper, we have discussed the design of the system that associates the Deep Convolutional Neural Network that can estimate the identity of the disease from the symptoms. Identifying the disease from plants and discovering the possibility that plant is either infected or not, will decrease the likelihood of risk due to such infection by taking appropriate steps against it. Proposed CNN is trained and build with higher precision and accuracy that associate the automatic detection of the disease from the plant leaves in preference of experienced human inspection. Designing the pure CNN that can identify the healthy plant species and infected plants with an accuracy of the 99% and which can avoid the significant loss of farmers. Proposed CNN includes the multiple layers that are trained intensely to identify the convoluted features of the images. The composition of the CNN model is done over the 35,000 training images with testing set from the same distribution with 4400 images. Detailed results are discussed in the paper.

Keywords: Remote Sensing · Deep Learning Models · Feature Extraction Image Classification · Disease Identification

1 Introduction

Due to the insects and weeds, every year a momentous supply of the crop is wasted, which precisely impact the cost and demand of the food as it is the most extensive part of the food chain. Most supply for the population of the country, the food is grown by small landholding farmers [1]. 20-35% of crops of the low landholding farmers are damaged every year due to lack of knowledge and lack of experts in the rural area. As the country's maximal food supplies depend on the small landholding farmers, this issue must be solved to prevent the loss accumulated every year and to reach the demand of the market. The damages due to the insects and weeds can be avoided if identified during the earlier stages. So, the necessary steps must be carried out, and that can improve the conditions of the plants so the overall loss can be reduced.

Due to the full availability of the computational resources, commodity hardware and the vast variety of algorithms for processing each kind of data. In the last few decades, there are many breakthroughs, and lots of research and technologies have emerged. Through research in the field of AI, various algorithms are developed that are easily implemented using the various frameworks. These research and technology in the field of AI cover the different Machine Learning and Deep Learning algorithms. These algorithms are executed with such precision that the results sometimes surpass human intuitions and predictions. For the field of Agriculture and crop disease prevention and identification, image datasets are available at various sources and are easily accessible. As this is the pressing issue, and it has a direct impact on human life, it must be solved. In the wide variety of algorithms, Neural Networks has outperformed and achieve many breakthroughs and became the most popularly used techniques for the classification. The Convolutional Neural Network, which is the most widely used technique for image classification. In ILSVRC [2] proposed CNN achieved the ground-breaking results which emphasise on the use of it on a large scale in various fields including the Agriculture.

Implementation of CNN involves various image pre-processing operations, which reduces the complex procedures for the Neural Network to identify the disease and species of the plant leaves. Using the CNN, identification of the dis-ease can be made at the earlier stages, which is used to prevent a large number of damages to the crops. Using computer vision, we can primarily improve the techniques and results of agriculture.

For humans, it is easy to identify the images and shape. Humans do have this learning so that they can easily find out the differences between the images, classify images and classify person with full accuracy. However, the same task for the Computer is tedious to perform. So, the Computer uses different methods for understanding the images, their content and try to identify the patterns among them. All this thing has been possible in the past two decades due to efficient computation power, due to research and new technologies. Now, computers have an artificial mind and ability like humans to classify the image, generating new images, understanding the words, and so on. All these things are possible thanks to the Internet that makes us available to a large amount of data and computational power due to cloud computing.

To make the Computer able to classify the images, they are trained with millions of the images and uses machine learning and deep learning algorithms. For categorising the images, often Convolutional Neural Networks are used to find the pattern among the images. This is made available through the Neural Network which has the multiple layers to process the data and predicts the result, In CNN images, are given as the input and then images are treated with the number of convolutional, pooling and activation layers. At the end of the network, there are some dense layers which receive the feature-map from the previous layer. The last layer contains the neurons which are equal to the number of classes that a neural network is introduced with, to predict the output.

2 Background and Related Work

Convolutional Neural Networks are also used for facial recognition [3-5], object recognition [6], object localisation using classification techniques [7]. In the last few years, people started concentrating on the use of CNN's to identify the diseases from the plants. Various papers show multiple ways to solve the problem on the same dataset and particularly on the tomato leaf images [8]. They make use of their model or re-trained the pre-trained model to get the best results. Designing the model from scratch can also result in better accuracy and results. Another paper [9] associated with the same dataset includes the right results by using the pre-trained models, AlexNet [10] and InceptionNet [11] with various ratios that divide the dataset into different splits and different testing techniques on the model by using the framework called Caffe [9]. Resulting F1-Score decides the variability and accuracy of the model better with dropout [12].

In this paper, the suggested CNN model is designed in the Keras [13] as it out-performs against the VGG-16 [14] and MobileNet [15] model for plant village dataset [16]. Training the network exhaustively, that makes it correctly understand the features of the images. The network trained is the final version of several models with train-test experiments. Suggested model at first developed without regularisation, hence affect the accuracy of the model to converge very slow after some iterations. Introduction with regularisation removes the issue of small convergence and generalises the network and led its performance towards the high accuracy. Different testing techniques are applied to test the accuracy of the model. To resist model from overfitting cross-validation and K-Fold validation test [17] are performed. To check the test accuracy of the model is tested against the entire dataset using the K-Fold validation [17], testing on images from outside the dataset and grayscale images [9]. In the next section, the Proposed method is discussed, followed by a dataset description and results of the algorithm.

3 Proposed Method

Designing of the Neural Network requires a significant amount of training and testing on the dataset. The way through which the problem of agriculture can be solved includes the understanding of how each layer of Neural Network processes the feature maps [17] and the layers which will further process it for. Suggested Neural Network uses 3 x 3 filter with the same padding that makes the image as intact as it was in the layer. There is a total of 11 Convolutional Layers and 3 Dropout layers to efficiently make prediction designed using the Keras framework. The architecture of the Neural Network designed in keras with a total of 14 layers of CNN and the other three essential fully connected layers are shown in Fig. 1.

The Fig. 1 displays layer with yellow colour as Convolutional layer and the orange coloured layer as Max Pooling layer. Every second layer after Convolutional Layer except the first block contains the dropout [12]. Dropout generalises the neurons of the network. So, there will be no bias in the training of neurons. Every neuron gets the opportunity to be trained well and prevents the model from overfitting. After the Convolutional blocks, the neurons are flattened for making the decision using the dense layer. The last layer is the SoftMax layer that contains the probability of each class label. In a network, two dense layers take feature map and convert into the single network that finds weights to make accurate decisions. Initially, the image size is supplied to the network is 128 x 128 x 3.

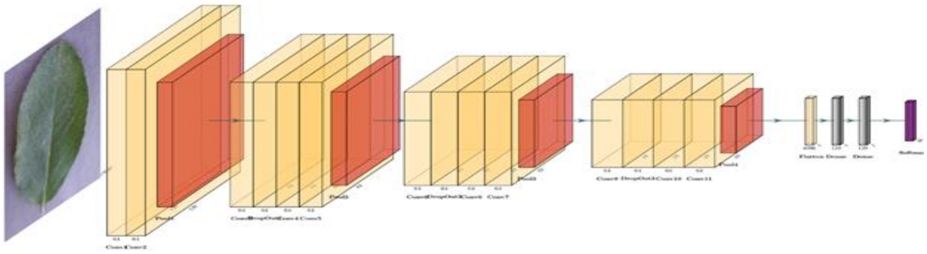


Fig. 1. Suggested Deep Convolutional Neural Network

After every Max-Pooling layer image is converted to the half-sized image, every Convolutional Layer contains the 3 x 3 filter and same padding that make the image size intact as it is. Each Convolutional Layer is normalised using the Batch-Normalization to make the training of the neural network faster. CNN does contain the padding in each layer of convolution. Training the model with 35,000 images in which, half of the images are augmented to remove the effect of over-fitting. Augmentation cannot be applied to test data.

This model is trained with two different optimisation Algorithms that are Ad-am and RMSprop. Adam algorithm is the composition of RMSprop + SGD (Scholastic Gradient Descent) which perform very well on the CNN and for the training, there is no need of learning rate decay in the case of the Adam Optimization Algorithms. While with RMSprop Optimization Algorithm learning rate decay is included to get better results in a smaller number of epochs. By setting proper hyper-parameters, we can get better accuracy in terms of train and test results. Inclusion of the batch-normalisation and dropout prevent the model from being biased and overfitting — a different version of the model with some of the pre-trained model like VGG-16. MobileNet is used for primary use. Due to more requirement of the resources, flexible way to deal with the issue of training time is to design own model. Developing the model at the prior stages does not guarantee the high accuracy; it requires changing several hyper-parameters. At previous, situation data is trained on the model with only batch-normalisation, only dropout, with both and with batch-normalisation, dropout as well as the augmentation. Applying augmentation allows the model to train against a versatile situation. Data Augmentation includes the preprocessing of images and makes available in the form which is different from the original data. Data Preprocessing and increase the size of the dataset will improve the invariance of CNN. Invariance is the ability of the CNN to classify the objects when their orientation is changed or in the different form from the image. Data preprocessing involves the scaling, rotation, and translation of the image using the method of finding the eigenvalue and eigenvectors in Linear Algebra.

Introducing the augmentation of data reduces chances of the model from overfitting and make model robust. The proposed model includes the dropout in each convolutional block except the first block that makes a model train by ignoring some neurons which are chosen randomly. Dropout does not work during the time of testing and evaluating the model. It is also one of the methods to prevent the model from overfitting but rather increase the training time of the model.

The proposed system involves testing of the model with every new feature added in the design of the model and change in the hyper-parameter. Table 3 provides clarification about the model with essential elements, and it is the prediction accuracy. Designing the system involves the two approaches either use pre-trained models or trained own model from scratch. During the design of the system pre-trained model like VGG-16[14] and MobileNet [15] was used as both have pros and cons. The system on further integrated on specific hardware environment that is very constrained. VGG-16 model [14] is too heavy according to the requirements, and MobileNet [15] does not provide good results with dataset distribution. Proposed model on prior stages tested and then using the trial-error method tested every time on the tested against ground truth.

3.1 Data Description

For training, CNN's dataset from the platform named "crowdai" is used, which does contain the distribution of the images as the 21,000 training images and 32,000 testing images [16]. The dataset does contain the plant leaf images, which are either infected or either in good form for the identification of the species. From the training images, the split with the ration 8:2 is applied so that cross-validation is included for the evaluation of the model and checking the validation accuracy for different diseases. For the equal distribution of the dataset, the method for an arrangement of the random number method named seeding is used. With the same seed value, uniform distribution of the dataset takes place each time on every machine and split. The dataset contains the 26 infected species of the plants and 12 healthy species of the plant leaf. For better accuracy and result to increase is to augment the dataset, which typically involves applying various image processing operations and methods for using the filter to increase the dataset size. Availability of extensive data can improve the network to get better predictions. Augmentation of the images involves removing the background details, which makes the model to precisely predict the results for the test data. Here, the test images are not pass through any operations of augmentation. The dataset is captured with high-resolution cameras and resized for processing it for further classification.

Table 1. Sample Images of the Dataset

 **Fig.2.1.**Apple Scab	 **Fig.2.2**.Apple Black Rot	 **Fig.2.3.**Apple Cedar Rust	 **Fig.2.4.**Apple Healthy	 **Fig.2.5.** Blueberry Healthy

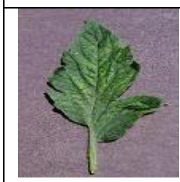
 **Fig.2.6.** Cherry Powdery Mildew	 **Fig.2.7.** Corn Gray Leaf Spot	 **Fig.2.8.** Corn Common Rust	 **Fig.2.9.** Corn Healthy	 **Fig.2.10.** Corn Northern Leaf Blight
 **Fig.2.11.** Grape Black Measles	 **Fig.2.12.** Grape Healthy	 **Fig.2.13.** Grape Leaf Blight	 **Fig.2.14.** Orange Citrus Greening	 **Fig.2.15.** Peach Bacterial Spot
 **Fig.2.16.** Pepper Bacterial Spot	 **Fig.2.17.** Pepper Healthy	 **Fig.2.18.** Potato Late Blight	 **Fig.2.19.** Potato Early Blight	 **Fig.2.20.** Potato Healthy
 **Fig.2.21.** Soybean Healthy	 **Fig.2.22.** Squash Powdery Mildew	 **Fig.2.23.** Straw-berry Healthy	 **Fig.2.24.** Straw-berry Leaf Scorch	 **Fig.2.25.** Tomato Bacterial Spot
 **Fig.2.26.** Tomato Late Blight	 **Fig.2.27.** Tomato Leaf Mold	 **Fig.2.28.** Tomato Septoria	 **Fig.2.29.** Tomato 2-spotted Spider Mite	 **Fig.2.30.** Tomato Target Spot
 **Fig.2.31.** Tomato Yellow Leaf Curl Virus	 **Fig.2.32.** Tomato Healthy	 **Fig.2.33.** Cherry Healthy	 **Fig.2.34.** Grape Black Rot	 **Fig.2.35.** Peach Healthy

 **Fig.2.36.** Raspberry Healthy	 **Fig.2.37.** Tomato early Blight	 **Fig.2.38.** Tomato Mosaic Virus		

3.2 Evaluation Method

Before the final model, there are several models trained and checked against the ground truth. Design of the model without applying the data-augmentation do contain the 17,520 images for the training of the model and 4300 images for the testing model. The first model gets 89% accuracy on training data. Validation is the most preferred technique to evaluate the model. For validation techniques like cross-validation during training time and K-Fold validation testing is performed where the average model accuracy for K-Fold validation [17] is 94.42 and F1-Score is 93.52. After adding the Batch-Normalization, there is a slight improvement in the test predictions. These were a slow increment in the accuracy as reach near to the final model that has the accuracy of 98.7% and F1-score 98.7. Adding methods and tweaking hyper-parameters largely contribute to the development of the Final model that is 99% accurate on test results.

$$Recall = \frac{True\ positive}{True\ Postive + False\ Negative} \tag{1}$$

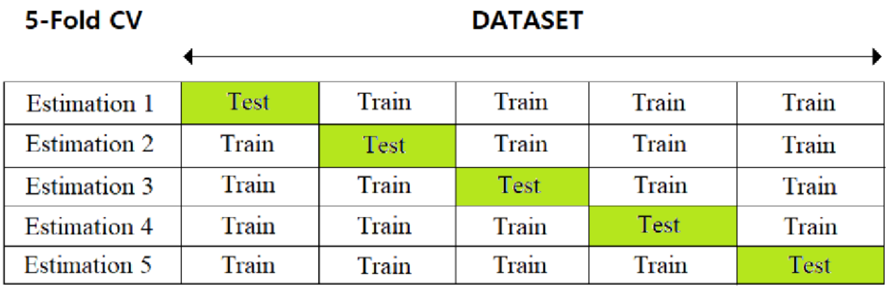
$$Precision = \frac{True\ positive}{True\ Postive + False\ Positive} \tag{2}$$

$$F1\text{- }Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{3}$$

Here Table 2 contains the detailed results of the model with and without features and data augmentation was applied. So, the total number of images were 35,361 for training and 4400 images for the testing of the model accuracy.

4 Results

As the dataset contains 38 classes, some of the classes do not provide enough images for the classification. Table 2 includes the detailed results of the K-Fold validation testing in which every split of the data is tested against the remaining training data with ration 80-20% where the value of K = 5. Here data augmentation [18] is not included, which further make it possible to use the data augmentation that makes significant addition in the testing accuracy of the model. All five models trained using 5-Fold Cross Validation contains Batch Normalization and Dropout [19].



Estimation 1	Test	Train	Train	Train	Train
Estimation 2	Train	Test	Train	Train	Train
Estimation 3	Train	Train	Test	Train	Train
Estimation 4	Train	Train	Train	Test	Train
Estimation 5	Train	Train	Train	Train	Test

Fig. 3. 5-Fold Cross-Validation Method

Table 2. 5-Fold Cross Validation on Corresponding Dataset

Name	Accuracy	F1-Score
Fold-1	95.85%	95.16
Fold-2	93.56%	93.24
Fold-3	91.85%	90.79
Fold-4	95.24%	93.01
Fold-5	95.60%	95.44

After the problem is solved using augmentation techniques. This issue is handled either manually or automatically. Keras provides techniques for augmentation of image dataset that makes the model to learn weights from each class. Table 3 displays the result that is tested against the highest achieving model as the technique and proper understanding of hyper-parameters [20] make the network more precise to predict the results. The main aim throughout the design of the model is to surpass the F1-Score of 99.34 which highest on the dataset which is solved at the time of competition on the platform of "crowdai" [21].

Table 3. Model Trained-Tested with Different Features

Model Version	Features	Epochs	Optimisation Algorithm	Accuracy	F1-Score
1	Plain Model	60	Adam	89.67%	90.06
2	Batch Normalization	60	Adam	89.92%	89.79
3	Batch Normalization + Dropout	60	Adam	99.68%	99.48
4	Batch Normalization + Dropout	60	RMSprop	99.49%	99.28

5 Conclusion

The paper presents the way to identify the diseases from the plant leaves at the earlier stages. That can help to prevent the loss and improve the quality of production of the crop in Agriculture using trending technology of Neural Networks. Use of the Deep Convolutional Neural Network designed in such a way that training with proper data can make Neural Network more efficient in terms of identification and recognition. Since our focus was on the development of the model in a modular way that number of layers and used parameters forms the network, which is highly accurate in terms of identifying diseases. The trained model can be scaled and designed by limiting the number of hardware resources. So, Final model is designed to integrate on hardware with limited configuration. The main aim while designing the Neural Network at the beginning is achieved as the model achieved excellent results in the cost-effective way and resources. **Acknowledgement**
The NVIDIA Corporation donated the Titan XP used for this research." or "We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan XP GPU used for this research.

References

1. Sladojevic, Srdjan, et al. "Deep neural networks-based recognition of plant diseases by leaf image classification." Computational intelligence and neuroscience 2016 (2016).
2. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.
3. Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.
4. Fei-Fei, Li, Rob Fergus, and Pietro Perona. "One-shot learning of object categories." IEEE transactions on pattern analysis and machine intelligence 28.4 (2006): 594-611.
5. Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.

6. Belongie, Serge, Jitendra Malik, and Jan Puzicha. "Shape matching and object recognition using shape contexts." IEEE Transactions on Pattern Analysis & Machine Intelligence 4 (2002): 509-522.
7. Lampert, Christoph H., Matthew B. Blaschko, and Thomas Hofmann. "Beyond sliding windows: Object localisation by efficient subwindow search." 2008 IEEE conference on computer vision and pattern recognition. IEEE, 2008.
8. Fuentes, Alvaro, et al. "A robust deep-learning-based detector for real-time tomato plant diseases and pest's recognition." Sensors 17.9 (2017): 2022.
9. Mohanty, Sharada P., David P. Hughes, and Marcel Salathé. "Using deep learning for image-based plant disease detection." Frontiers in plant science 7 (2016): 1419.
10. Iandola, Forrest N., et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size." arXiv preprint arXiv:1602.07360 (2016).
11. Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.
12. Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." The Journal of Machine Learning Research 15.1 (2014): 1929-1958.
13. Gulli, Antonio, and Sujit Pal. Deep Learning with Keras. Packt Publishing Ltd, 2017.
14. Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014). Wiesner-Hanks, Tyr, et al. "Image set for deep learning: field images of maize annotated with disease symptoms." BMC research notes 11.1 (2018): 440
15. Howard, Andrew G., et al. "Mobilenets: Efficient convolutional neural networks for mobile vision applications." arXiv preprint arXiv:1704.04861 (2017).
16. Hughes, David, and Marcel Salathé. "An open access repository of images on plant health to enable the development of mobile disease diagnostics." arXiv preprint arXiv:1511.08060 (2015).
17. Rodriguez, Juan D., Aritz Perez, and Jose A. Lozano. "Sensitivity analysis of k-fold cross-validation in prediction error estimation." IEEE transactions on pattern analysis and machine intelligence 32.3 (2010): 569-575.
18. Ding, Jun, et al. "Convolutional neural network with data augmentation for SAR target recognition." IEEE Geoscience and remote sensing letters 13.3 (2016): 364-368.
19. Rawat, Waseem, and Zenghui Wang. "Deep convolutional neural networks for image classification: A comprehensive review." Neural Computation 29.9 (2017): 2352-2449.
20. Montavon, Grégoire, Geneviève Orr, and Klaus-Robert Müller, eds. Neural networks: tricks of the trade. Vol. 7700. Springer, 2012.
21. Escalera, Sergio, et al. "Introduction to NIPS 2017 Competition Track." The NIPS'17 Competition: Building Intelligent Systems. Springer, Cham, 2018. 1-23.