# Training an Image Classification Convolutional Neural Net to Detect Plant Disease

##### **Singh.bhu@husky.neu.edu**

INFO 7390

***Abstract*-**

***Index Terms***- In this final project, I aim to a implement the Paper “**Using Deep Learning for Image-Based Plant Disease Detection**” [1] through Resnet-152 model [2]. The model was trained on a public dataset of 54,306 images of diseased and healthy plants containing 14 crop species and 26 diseases. It was able to predict the disease with high accuracy of approx. 96% on the test dataset. A much more diverse set of training data is needed to improve the general accuracy. In this paper, we will get introduced to the problem, then we will discuss CNN and different models based on it. Later on, we will discuss the whole process of model implementation and Finally, we will discuss results and conclusions.

1. Introduction

According to the food and Agriculture organization of the United Nations (UN), transboundary plant pests and disease effect food crops, causing significant losses to farmers and threatening food security. Crop diseases [3] are a major threat to food security, but their rapid identification still remains a difficult task in many parts of the world due to lack of the required infrastructure. Plant

diseases are not only a threat to food security at the global scale, but can also have disastrous consequences for smallholder farmers whose livelihoods depend on healthy crops. In the developing world, more than 80 percent of the agricultural production is generated by smallholder farmers [4], and reports

of yield loss of more than 50% Various efforts have been developed to prevent crop loss due to diseases. Historical approaches of widespread application of pesticides have in the past decade increasingly been supplemented by integrated pest management (IPM) approaches []. Independent of the approach, identifying a disease correctly when it first appears is a crucial step for efficient disease management. Historically, disease identification has been supported by agricultural extension organizations or other institutions such as local plant clinics. In more recent times, such efforts have additionally been supported by providing

information for disease diagnosis online, leveraging the increasing to pests and diseases are common [5]. Furthermore, the largest fraction of hungry people

(50%) live in smallholder farming households [6], making smallholder farmers a group that’s particularly vulnerable to pathogen-derived disruptions in food supply. Here, we demonstrate the technical feasibility using a deep learning approach utilizing 54,306 images of 14 crop species with 26 diseases (or healthy) made openly available through the project PlantVillage[9]. An example of each crop - disease pair can be seen in Figure 1. Computer vision, and object recognition in particular, has made tremendous advances in the past few years. The PASCAL VOC Challenge[10], and more recently the Large

Scale Visual Recognition Challenge (ILSVRC)[11] based on the

ImageNet dataset[12] have been widely used as benchmarks for

numerous visualization-related problems in computer vision, including object classification. In 2012, a large, deep convolutional neural network achieved a top-5 error of 16.4% for

the classification of images into 1,000 possible categories[13].

In the following three years, various advances in deep convolutional neural networks lowered the error rate to 3.57% [13][14] [15] [16] [17]. While training large neural networks can be very time-consuming, the trained models can classify images

very quickly, which makes them also suitable for consumer applications on smartphones.

1. RELATED WORKS

Recent years, there has been many breakthroughs in computer vision especially in image classification. Deeper neural networks, and different other strategies have been put forward.

2.1 ALEXNET[16]

In the ILSVRC [11] 2012, Krizhevsky et al. from University of Toronto designed a network later called AlexNet by which CNN is first used in image classification on large dataset. It got 16% percent top-5 error rate on the ImageNet dataset which is really a huge improvement over the results from ILSVRC 2010 and 2011. AlexNet is quite simple compared to modern CNNs. It contains 5 convolutional layers followed by 3 fully connected layers. Max-pooling is used to reduce the number of parameters and Dropout is used to prevent overfitting.

2.2 Vgg-16[18]

VGG model was introduced with the idea that deeper CNNs get better performance in image classification.

All convolution layers in VGG-16 are 3x3 filters and 1 strides with same padding. As the network gets deeper, the number of channels rises while the width and height of the volume reduces.

2.3 GOOGLENET[13]

Inception network is complex due to the fact that it uses different sizes of convolution layers for the input. In order to fit the dimensions, same padding is used. To reduce the number of parameters, 1x1 convolution layer is applied before each 3x3 and 5x5 convolution layers, which saves a lot of computation power without causing much performance issue.



**Fig. 1.** Example of leaf images from the PlantVillage dataset, representing every

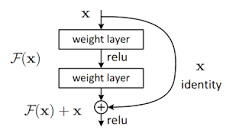
crop-disease pair used. 1) Apple Scab, Venturia inaequalis 2) Apple Black Rot,

Botryosphaeria obtusa 3) Apple Cedar Rust, Gymnosporangium juniperi-virginianae 4) Apple healthy 5) Blueberry healthy 6) Cherry healthy 7) Cherry Powdery Mildew, Podosphaera spp. 8) Corn Gray Leaf Spot, Cercospora zeae-maydis 9) Corn Common Rust, Puccinia sorghi 10) Corn healthy 11) Corn Northern Leaf Blight, Exserohilum turcicum 12) Grape Black Rot, Guignardia bidwellii, 13) Grape Black Measles (Esca), Phaeomoniella aleophilum, Phaeomoniella chlamydospora 14) Grape Healthy 15)Grape Leaf Blight, Pseudocercospora vitis 16) Orange Huanglongbing (Citrus Greening), Candidatus Liberibacter spp. 17) Peach Bacterial Spot, Xanthomonas campestris 18) Peach healthy 19) Bell Pepper Bacterial Spot, Xanthomonas campestris 20) Bell Pepper healthy 21) Potato Early Blight, Alternaria solani 22) Potato healthy 23) Potato Late Blight, Phytophthora infestans 24) Raspberry healthy 25) Soybean healthy

26) Squash Powdery Mildew, Erysiphe cichoracearum, Sphaerotheca fuliginea 27) Strawberry Healthy 28) Strawberry Leaf Scorch, Diplocarpon earlianum 29) Tomato Bacterial Spot, Xanthomonas campestris pv. vesicatoria 30) Tomato Early Blight, Alternaria solani 31) Tomato Late Blight, Phytophthora infestans 32) Tomato Leaf Mold, Fulvia fulva 33) Tomato Septoria Leaf Spot, Septoria lycopersici 34) Tomato Two Spotted Spider Mite, Tetranychus urticae 35) Tomato Target Spot, Corynespora cassiicola 36) Tomato Mosaic Virus 37) Tomato Yellow Leaf Curl Virus 38) Tomato healthy

2.4 RESNET[19]

It works on the idea of residual connections between layers which help to train very deep neural networks.



In the ResNet paper, Kaiming et al, indicated that maybe due to the overfitting issue , simply stacking large number of plain layers will actually result in a lower training and test accuracy which can be avoided by Resnet. The top-5 error rate on ImageNet dataset is 3.57% which surpassed the human-level performance 5.1%.

## III.DATASET

## I used the “PlanVillage dataset”. This dataset contains an open access repository of images on plant health to enable the development of mobile disease diagnostics. The dataset contains 54, 309 images. The images span 14 crop species:*Apple, Blueberry, Cherry, Grape, Orange, Peach, Bell Pepper, Potato, Raspberry, Soybean, Squash, Strawberry, and Tomato*. It contains images of 17 fundal diseases, 4 bacterial diseases, 2 mold (oomycete) diseases, 2 viral diseases, and 1 disease caused by a mite. 12 crop species also have images of healthy leaves that are not visibly affected by a disease. The dataset contains 38 classes of crop disease pairs

IV.IMPLEMENTATION

4.1 Environment

The whole project was implemented on a google colab environment. The model was trained on NVIDIA Tesla k80 gpu. The opensource PyTorch framework by Facebook was.

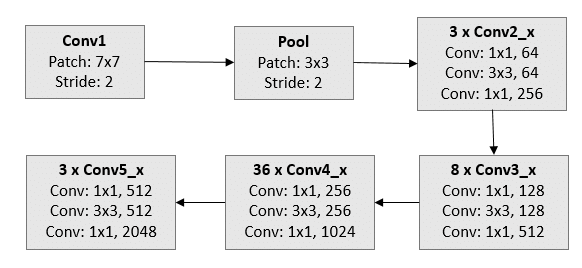
4.2 Preprocessing

The purpose of using data augmentation to our train and test dataset is to increase the number of images our model can see by applying random transformations to the images. In my case, I applied some data augmentation such random rotation, resized crop, random horizontal flip and center crop. Remember, we want our model to classify the images regardless of orientation. you'll also need to make sure the input data is resized to 224x224 pixels as required by the networks. he pre-trained networks available from 'torchvision' were trained on the ImageNet dataset where each color channel was normalized separately. For both sets you'll need to normalize the means and standard deviations of the images to what the network expects. For the means, it's ' [0.485, 0.456, 0.406]' and for the standard deviations ''[0.229, 0.224, 0.225]' , calculated from the ImageNet images. These values will shift each color channel to be centered at 0 and range from -1 to 1.



4.3 Choosing the Model

I decided to use one of the pretrained models from torchvision.models to get the image features and build and train a new feed-forward classifier using those features. The pretrained model I chose was the Microsoft’s Residual Networks architecture: Resnet-152. This is one of those models used in COCO 2015 competitions, which won the 1st place in: ImageNet classification, ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation.



The reason I used a pretrained model is because it’s a time saver process and this kind of model was trained on a large dataset to solve a problem similar to the one I wanted to solve.

After installing the pre-trained model for image classification, I removed the original classifier, then added a new one for identifying plant diseases and fine tuned the model by freezing some parameters.

4.3 Activation Function

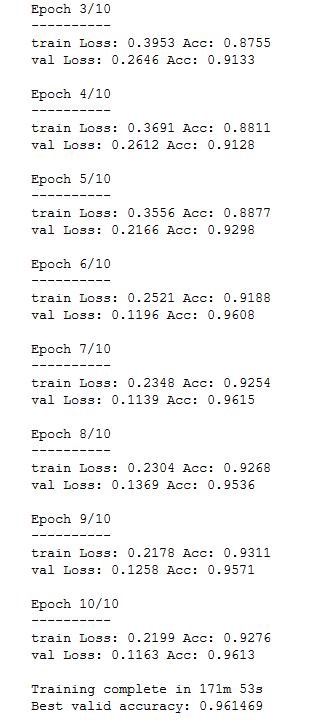
I found that Rectified Linear Unit(ReLU) would be better fit over the sigmoid function for the classifier.

4.4 Optimizer Function

The model was trained with Adam optimizer due to its computational efficiency, suitability for large data/parameters and little tuning required for it.

4.5 Number of Epochs

At first, we set our epoch to 1 to make sure our model could run normally and finally settled on 10 as optimal epochs considering the other hyperparameters.



4.6 Learning Rate

We consider a constant rate of .001 to be the optimal and gave good results.

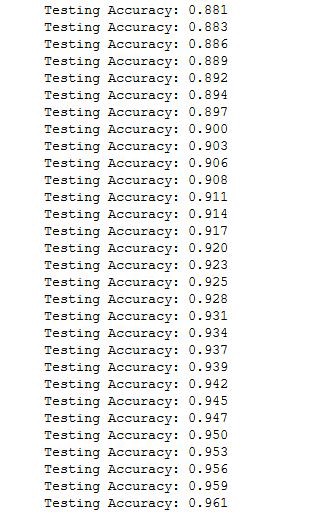
V Code

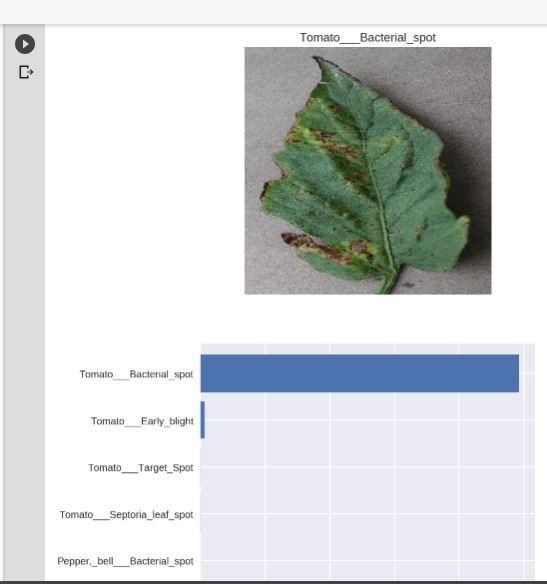
<https://github.com/singhbhupender1/plant-disease-detection-Resent-152-model>

The above is a link to GitHub repository

VI. Results

After running on the test data, the accuracy came out to be .961.





1. CONCLUSION

Although the model gives very good result on test data but the performance on actual real world data is still unconfirmed. In my experience, a better performance can be achieved with some hyperparameters tuning. Although the dataset was healthy size, but in my opinion more variation in the dataset pictures could produce good results in real world situations as well. This model can be finetuned to specific regions of the world for accurate crop disease identification. This model can be implemented in the form of a mobile application that does offline processing for use in parts of the world where internet penetration is scarce.

Acknowledgment

I would like to thank Professor Nik Bear Brown for sharing this project idea and giving guidance.

References

1. arXiv:1604.03169
2. arXiv:1512.03385
3. Strange RN, Scott PR (2005) Plant disease: a threat to global food security. Phytopathology
4. UNEP (2013) Smallholders, food security, and the environment.
5. Harvey CA et al. (2014) Extreme vulnerability of smallholder farmers to agricultural risks and climate change in madagascar. Philosophical Transactions of the Royal Society of London B: Biological Sciences 369(1639).
6. Sanchez PA, Swaminathan MS (2005) Cutting world hunger in half.
7. J. Wang, “Fundamentals of erbium-doped fiber amplifiers arrays (Periodical style—Submitted for publication),” *IEEE J. Quantum Electron.*, submitted for publication.
8. Hughes DP, Salathé M (2015) An open access repository of images on plant health to enable the development of mobile disease diagnostics. CoRR abs/1511.08060
9. Gfgf
10. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) The pascal visual object classes (voc) challenge. International journal of computer vision 88(2):303–338.Fgffg
11. Russakovsky O et al. (2015) ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) 115(3):211–252.
12. Deng J et al. (2009) Imagenet: A large-scale hierarchical image database in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. (IEEE), pp. 248–255.
13. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks in Advances in neural information processing systems. pp. 1097–1105.
14. Zeiler MD, Fergus R (2014) Visualizing and understanding convolutional networks in Computer vision–ECCV 2014. (Springer), pp. 818–833.
15. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
16. Szegedy C et al. (2015) Going deeper with convolutions in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–9.
17. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385.
18. ‎[arXiv:1502.01852](https://arxiv.org/abs/1502.01852)