Journal of Multi Disciplinary Engineering Technologies, Vol. 13, Issue 1

Transfer and Residual Learning for Plant Disease Detection

Alka Leekha , Shruti Bhardwaj# , Sarthak

Department of Information Technology,

Bharati Vidyapeeth's College of Engineering, New Delhi

#Corresponding Author, Email: *bhardwajshruti97@gmail.com*

***Abstract*— The study about diseases in plants is paramount to alleviate the issue of food security in all around the world. The foremost necessary step to mitigate the matter is correct as well as the timely detection of the disease in the plant. The primary and initial step in the identification of a plant disease is performing a visual inspection. This large scale problem and also shortage of professionals demand an automated and correct visual investigation technique. The recent progress in the popular fields of machine learning and computer vision, mainly with the help of methods like the use of famous convolutional neural networks (CNN) and deep learning has been successfully generated spectacular leads to the sector of both object recognition and image classification. Through this paper, we try to address to solve the matter of identifying diseases in plants using the images of plant leaves using transfer and residual learning. The Plant Village dataset is used which comprises of 54,309 images of 14 crops across 39 classes (healthy or a specific disease image). The images of leaves are of top quality which were taken manually and under acceptable lighting conditions. With the Plant Village dataset, the model is successful in achieving mean accuracy of 96% using transfer** and **residual learning.**

***Keywords*— Machine Learning, CNN**

1. Introduction

Diseases caused to plants bring big damages to agriculture crops which affect the production significantly. Early blight is an instance of disease that critically decrease production [1]. Late blight is also very harmful disease in humid climate that affects the leaves of the plants, fruits and stems [1]. Saving and protecting crops from getting affected by diseases is important in order to ensure crops quantity as well as quality [2]. Successful strategies to protect should aim to detect diseases at an early stage so as to decide the right treatment and to be started at right time and avoid its spreading. [3].

This detection can be achieved only by experts who have the knowledge along with practical experience about the causes of diseases and symptoms [1]. These experts have to study plants continuously to prevent the disease from spreading. This consistent monitoring is time-consuming and difficult for a human, therefore this brings automation of the detection and then identification of plant diseases an important task to protect plants [2]. Several studies have been devised to detect and identify diseases in plants using machine learning and processing of the image. All the proposed approaches classify diseases using images of the crops. These classifiers use features that are hand-crafted and are developed by speciality in order to find suitable information useful for classification of image. So, these classifiers face lack of automating the process due to its dependency on the hand-crafted features [3]. Also, the classifier has to be trained only using images that are labelled by experts. Again, collecting labeled images is a costly affair as it is done manually. Also, it might be possible that new diseases may occur at places where there have been no such past incidents and thus might not have any local expertise from the farming community which can help to manage them [4]. An automated system is need of the hour to detect and identify diseases in plants by using images of plant leaves or some visual appearance visible by using advancement in the domain of deep learning and computer vision provides an opportunity to enhance and expand the idea of plants or crop protection and carry forward the applications of computer vision and machine learning in the domain of agriculture [5]. The paper has been organized as follows. Section 2 explains the dataset and the image processing techniques relevant to the development of plant disease classification systems. Section 3 gives a detailed explanation of our proposed methodology and section 4 gives the achieved results and section 5 presents the conclusion.

2. TECHNIQUES AND DATASET

2.1 Transfer Learning

Transfer learning is basically reusing a pre-trained model on some new problem. It is a very popular approach of solving deep learning problems because it enables us to train deep neural networks with comparatively less data. So, it proves to be very useful since most of the real-world problems do not have a large dataset to train our complex models [6]. In transfer learning, a new but related problem is solved using the knowledge from a previously trained machine learning model. For instance, if we create a simple classifier for predicting if a given image is of some plant or not, then we could use the knowledge of this model to identify other objects like car etc. [7]. Using transfer learning, we are able to transfer the weights that are learned by the network in Task A to a new Task B. In this, instead of starting the learning from scratch, we can start from a model with pre-trained weights that have been learned by solving a similar problem [8]. For example, in techniques like computer vision, Neural Networks try to discover edges in the beginning layers, shape features in the middle layer and a few task-specific features in the finishing layers. Using transfer learning, we retain the first and middle layers and re-train the end classifier.

2.2 Residual Learning

An important question analyzed by He et al. [9] is whether simply increasing the number of layers of the neural network, hence increasing the depth of the model in order to find better results. The conclusion obtained through experiments was that on increasing the depth, accuracy may result in getting saturated and then decreases rapidly. The cause for saturation was not overfitting but the difficulty in optimizing the layers. Residual learning was proposed to solve the complication of degrading accuracy as shown in Fig. 1.

In residual learning, we can skip connections so as to bypass some convolution layers at a given instant. Each skip generates a residual block. In residual block, layers generate a residual which has to be added into an input tensor’s block. So, let input x and function of layers stacked be F(x). We add the residual to F(x), and the Rectified linear Unit (ReLU) comes out to be: $H(x) = x + F(x)$. Now, the combination of input and output layers solves our problem of degrading accuracy and is easier to optimize [9].

2.3 Dataset

The image used to train our classifier were taken from plant village project [10]. There are 54,309 images in this dataset spanning over 14 diff crops. There are images of plants suffering from 17 fungal diseases, 2 mold, 4 bacterial diseases, 1 viral and mite. There are 39 different classes labelled with the crop name and also the disease it is affected by or if it is healthy. The images of this dataset are clicked by photographers using a shoot camera and standardized digital point. All the images were intentionally taken in appropriate light in a variety of conditions so that it resembles the experience of an end user like that of an expert or a farmer. Two plant pathologists confirmed the identity of the diseases who were working with the technicians and provided the diagnosis [10].The dataset contains 39 classes of crop disease pairs which are listed below and sample is shown in Fig. 2.

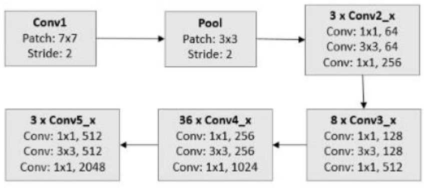


Fig. 1: Architecture of Resnet 152 [11]

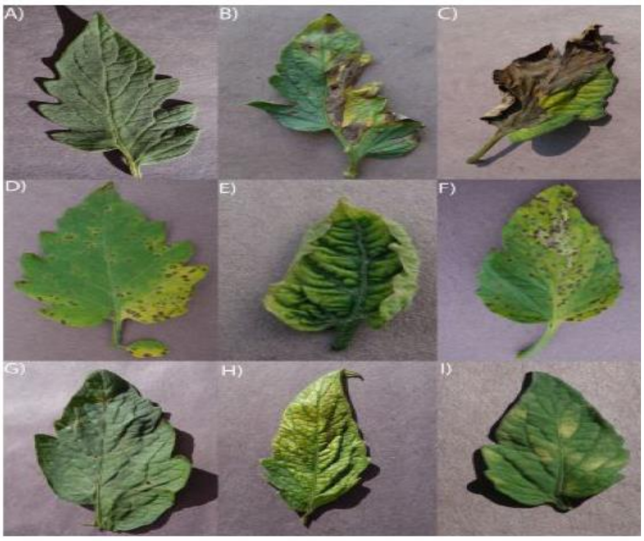


Fig 2: Images of various classes of leaves of tomato.

A) A healthy tomato leaf
B) Tomato Leaf affected by Early Blight (Alternaria solani)
C) Tomato Leaf affected by Late Blight disease (Phytophthora Infestans)
D) Septoria Leaf Spots Affected Tomato Leaf (Septoria lycopersici)
E)Tomato Leaf affected by Yellow Leaf Curl Virus(A Family Geminiviridae genus Begomovirus)
F) Tomato Leaf affected by Bacterial Spots (Xanthomonas campestris pv. vesicatoria)
G)Tomato Leaf affected by Target Spots (Corynespora cassiicola)
H)Tomato Leaf affected by Spider Mite (Tetranychus urticae)

3. PROPOSED METHODOLOGY AND EXPERIMENTAL SETUP

The process of developing the model appropriate for plant disease detection using a deep CNN model is described further in detail. The complete procedure is divided into several stages in subsections below, starting with the loading of images for classification process using resnet-152 neural networks. Fig. 3 shows the proposed methodology.

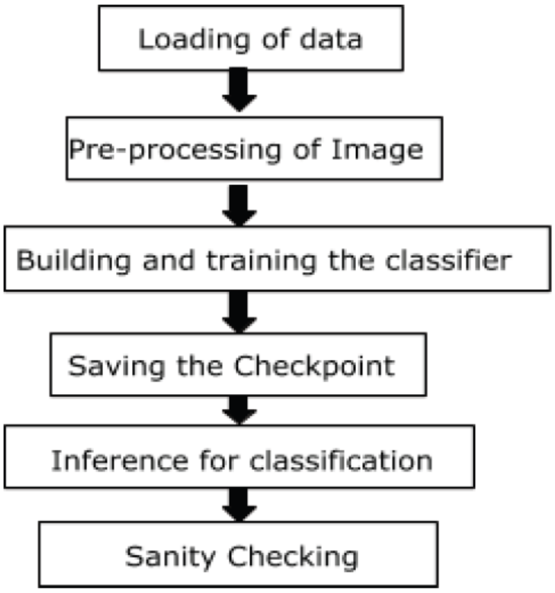


Fig 3: Proposed Methodology

3.1 Loading of Data

The data is loaded using torchvision. The dataset is split into two parts, training and validation. For the training, transformations are applied such as cropping, random scaling, and flipping. This helps the network to generalize and lead to better performance. As a pre-trained network, Resnet - 152 is used, the input data is resized to 224x224 pixels as required by the networks. The validation set is used to measure the model's performance on data it hasn't been tested on. For this scaling or rotation transformations are not performed, but the images are resized to the appropriate size.

The pre-trained networks from torchvision are trained on the ImageNet dataset where each color channel is normalized separately. For both sets normalization of the means and standard deviations of the images are needed to what the network expects. For the means, it is [0.485, 0.456, 0.406] and for the standard deviations [0.229, 0.224, 0.225]
, which are calculated from the ImageNet images. These values will shift each color channel to range from -1 to 1 and to be centered at 0.The number of threads used are 4 and the batch_size is 32.

3.2 Building and training the classifier

After the data is ready, the classifier is built and classified. Resnet - 152 pre-trained model from torchvision.models is used to get the image features. A new feed-forward classifier is built and trained using those features. A new, untrained feed-forward network is defined as a classifier, using ReLU activations. Input_size matches should match the in_features of the pretrained model.

```
# Creating the classifier
ordered dictionary first classifier =
nn.Sequential(OrderedDict
([('fc1',
nn.Linear(2048, 512)),
('relu'
,nn.ReLU()),
('fc2',
nn.Linear(512,39)),('output',
nn.LogSoftmax(dim=1))
]))
```

Adam optimizer is used with a learning rate of 0.001 and decay of LR Scheduler by a factor of 0.1 every 5 epochs.

```
# Adam optimizer with a
learning rate
optimizer_adam =
optim.Adam(model.fc.parameter
s(), lr=0.001)
# Decay LR scheduler by a 0.1
factor for every 5 epochs
exp_scheduler_lr =
lr_scheduler.StepLR(optimizer
_adam, step_size= 5,
gamma=0.1)
```

The results of accuracy achieved after each epoch are given below in the Table 1. The model is run on the training dataset and then on the validation or the testing dataset. The total number of epochs used for training is 10.

	TRAIN		VALIDATION	
	LOSS	ACCURACY	LOSS	ACCURACY
EPOCH 1/10	0.7507	0.7814	0.2986	0.9041
EPOCH 2/10	0.4504	0.8588	0.2464	0.9171
EPOCH 3/10	0.4043	0.8739	0.2179	0.9288
EPOCH 4/10	0.3696	0.8823	0.235	0.9231
EPOCH 5/10	0.3521	0.8879	0.1862	0.9374
EPOCH 6/10	0.2488	0.92	0.1356	0.9538
EPOCH 7/10	0.2262	0.9266	0.1223	0.9585
EPOCH 8/10	0.2302	0.9251	0.1187	0.9605
EPOCH 9/10	0.2183	0.9283	0.1211	0.9606
EPOCH 10/10	0.2149	0.9305	0.1131	0.9629
TRAINING COMPLETE IN 181m 26s				
BEST VALID ACCURACY : 0.962923				

Table 1 : Loss and accuracy after each Epoch

3.3 Saving the checkpoint

After training the network, the model is saved to be loaded later for making predictions. The mapping of classes to indices is also saved which is attached to the model as an attribute for making inferences easier later on.

image datasets: image_datasets['train'].class_to_idx

3.4 Inference for classification

An image is passed into the network and the class is predicted of the plant disease in the image. The top 5 most likely classes are returned along with the probabilities. This is done by calculating the class probabilities and then finding the largest values. To get the top 5 largest values in a tensor x.topk(k) is used. This method returns both the highest k probabilities and the indices of those probabilities corresponding to the classes. These indices are converted to the actual class labels using the following code.

class_to_idx.
print
(predict('PlantVillage/val/Bl
ueberry___healthy/06eacfab-fb
39-40e0-bbce-927bc98fa2ac___R
S_HL 2663.JPG', loaded_model))
(array([9.9871051e-01, 4.5131423e-04,
4.0855847e-04, 2.2177442e-04, 6.1075392e-
05],
dtype=float32), ['Blueberry___healthy',
'Apple___Apple_scab', 'Apple___healthy',
'Peach___Bacterial_spot',
'Apple___Black_rot'])

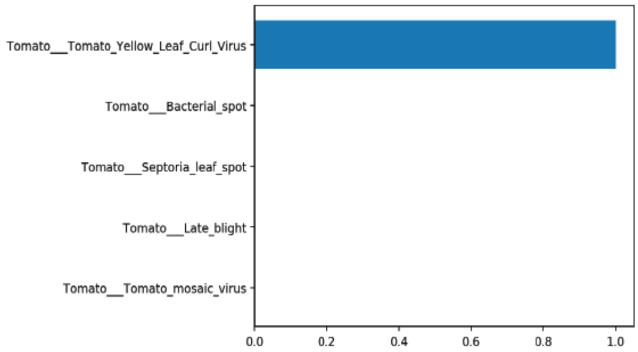
3.5 Sanity Checking

Matplotlib is used to plot the probabilities for the top 5 classes as a bar graph, along with the input image.The class integer encoding is converted to actual plant diseases names.To show a PyTorch tensor as an image, the `imshow` function described below is used.

```
def imshow(image, ax=None, title=None):
if ax is None:
fig, ax = plt.subplots()
image = image.numpy().transpose((1, 2, 0))
mean = np.array([0.485, 0.456, 0.406])
std = np.array([0.229, 0.224, 0.225])
image = std * image + mean
image = np.clip(image, 0, 1)
ax.imshow(image)
return ax
```

4. RESULTS

Through our study, we achieve an accuracy of 96.29% using Resnet - 152 pre-trained model. It was able to score 93.05% accuracy in the training phrase under 10 epochs with the learning rate of 0.001. Our model is able to achieve an accuracy of 96.29% for testing data. Our Resnet-based detector is able to successfully recognize different categories of diseases in various plant leaves. Test images displayed along with the top 5 classes in Fig. 4 and Fig. 5 shows the classification result.



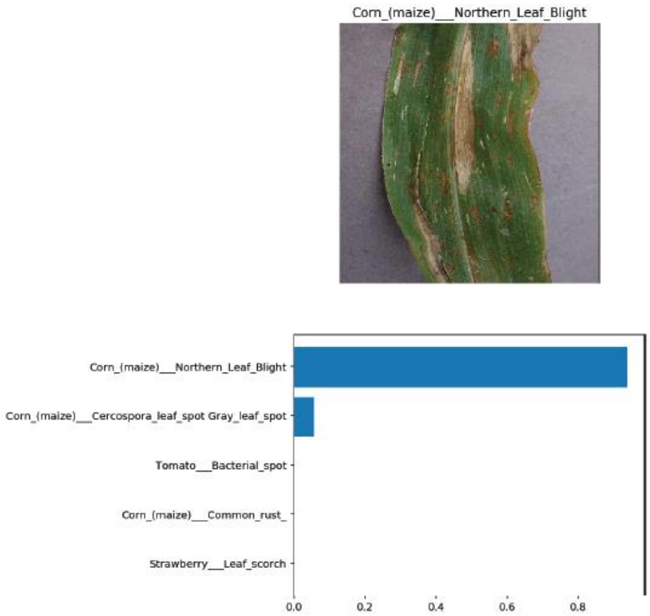


Fig 4: Successful classification of Corn leaf affected with Northern Leaf

5. CONCLUSION

Crop protection in organic agriculture is not a simple matter. It depends on a thorough knowledge of the crops grown and their likely pests, pathogens and weeds. Neural networks' accuracy can be improved by training the model for more number of epochs. Due to this, there are chances that the foremost optimum solution could get overlooked or missed because the model could end up converging. Inline to a sensible thought, a price needs to be paid for each and every computation, a model is considered to be good enough if that is able to achieve high accuracy and that too in as minimum number of epochs as it is possible. Thus, even if residual and transfer learning is successful in achieving a significant and big improvement, finding optimum parameters such as that of learning rate as well as the number of epochs used for training the model is also an equally necessary issue.

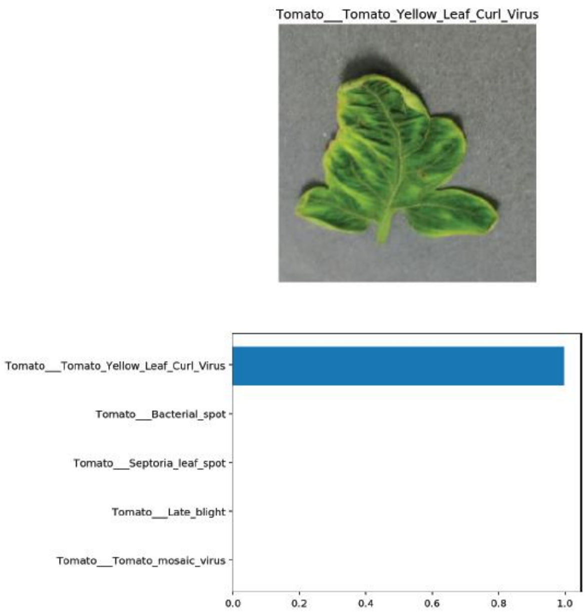


Fig 5: Successful classification of Tomato leaf affected with Yellow Leaf Curl Virus

REFERENCES

[1] Al Hiary H, Bani Ahmad S, Reyalat M, Braik M, ALRahamneh Z (2011), “Fast and Accurate Detection and Classi · cation of Plant Diseases”, International Journal of Computer Applications 17(1):31–38.

[2] Blancard D (2012), “Diagnosis of Parasitic and Non-Parasitic Diseases”, Academic Press The Netherlands

[3] Brahimi M, Boukhalfa K, Moussaoui A (2017), “Deep Learning for Tomato Diseases: Classi · cation and Symptoms Visualization.”, Applied Arti · cial Intelligence 31(4):1–17

[4] Tatem,Hay, S.I., A.J., Rogers, D.J., “Global transport networks and infectious disease spread”, Adv. Parasitol. 62, 293–343 (2006)

[5] Srdjan Sladojevic, Marko Arsenovic, Andras Anderla, Dubravko Culibrk, and Darko Stefanovic, “Deep Neural Networks Based Recognition of Plant Diseases by Leaf Image Classification,” Computational Intelligence and Neuroscience, vol. 2016, Article ID 3289801, 11 pages, 2016.

[6] Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? In Advances in neural information processing systems (pp. 3320-3328).

[7] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.

[8] Huh, M., Agrawal, P., & Efros, A. A. (2016). What makes ImageNet good for transfer learning?. arXiv preprint arXiv:1608.08614.

[9] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

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

[11] Nguyen, L. D., Lin, D., Lin, Z., & Cao, J. (2018, May). Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation. In 2018 IEEE International Symposium on Circuits and Systems (ISCAS) (pp. 1-5). IEEE.