IMPROVING CURRENT LIMITATIONS OF DEEP LEARNING BASED PLANT DISEASE IDENTIFICATION

A PREPRINT

Minh-Thai Nguyen
Department of Electrical Engineering
The Cooper Union
New York, NY 10003
nguyen8@cooper.edu

Dhvanil Shah
Department of Electrical Engineering
The Cooper Union
New York, NY 10003
shah3@cooper.edu

October 4, 2020

ABSTRACT

Although there has been much success in using deep neural networks to classify plant diseases in a laboratory setting, there has not been as much attention on applying the plant disease classification task to the field. We introduce a two-stage architecture with the goal of creating a lightweight plant disease classifier for plants in a realistic field or home environment. The first stage is a leaf detector that bounds the leaves in the image. The second stage is the plant disease classifier that classifies each bounded leaf image. Using this architecture we achieved a 26.00% accuracy on images of diseased plants in diverse environments.

***K*eywords** Plant Diseases · Object Detection · Disease Classification

1 Introduction

In recent years, there have been major contributions in the field of plant disease detection[1][2]; however, there seems to be a general deficit of attempts to classify plant diseases in the field. This is mainly due to the fact that the most readily available dataset in plant disease detection is the PlantVillage Dataset, which is limited in the fact that all of the images are taken in a laboratory setting with consistent lighting and a gray background. This makes the images extremely different from the images that we would see from a person's phone when taking a picture of a diseased leaf in their home or in a field. When applying plant disease detection on a mobile device, it is also rare that the diseased leaf will be the only diseased leaf present in the picture. This implies that some sort of leaf detection must be needed to successfully apply the model to the field. To tackle these problems, we employed a two-stage architecture for classifying diseased plant leaves in the field. The first stage consists of a leaf detector trained to identify leaves in a picture. The output of this detection is then fed into the second stage that classifies the leaf's disease. The second stage is trained on the data from the PlantVillage Dataset. In addition to this, we added the constraint of condensing our architecture as much as possible so that inference may realistically be run on a mobile device. We believe that this architecture will maximize the effectiveness of the diseased plant leaf data that we do have, while mitigating the data limitation of not having images of diseased leaves in the wild.

2 Related Works

Our work is very closely related to the works of the authors of [1][3]. In [1]. In [1], the authors made a classifier that identified 13 different plant diseases as well as a healthy leaf and a background image, for a total of 15 classes. Their classifier scored an accuracy of 96.3% on their validation set. They did not use the PlantVillage Dataset, but rather scraped what they deemed as suitable images from Google images using a script. In addition to this, a few of their classes tied multiple crops to the same disease. For instance, they grouped pear, cherry, and peach into one class for porosity and they grouped all healthy leaves into one class. Their results might be a little misleading because of the way they formed their validation set. To form their dataset, they augmented their original images to produce around 10 times as many images. They then separated this dataset into a training and validation set, instead of separating the original images and only augmenting the training set. It could be very likely that simple augmentations of the same image could be present in both the validation and training set, leading to a misleading validation set accuracy.

In their more recent paper, the authors of [3] proposed a new model architecture specifically meant to classify plant diseases in the wild that involved a two-stage classification method, similar to our method. The first stage was responsible for detecting plant leaves by species, while the second stage was responsible for classifying those plant leaves. Their classifier attained an accuracy of 93.67% on a new dataset that they created. Their new dataset consisted of 79,265 new leaf images that were taken in various weather conditions, at different angles, at different times of day, and with different backgrounds all over the world. Each of these leaves were hand bounded and labeled by different experts in a total of over 300 labor hours. We tried receiving access to their dataset; however, they did not respond in time.

3 Two Stage Network for Disease Detection

In order to address the limitations of the current work, namely it's inability to inference well on plant disease images in the wild using just the PlantVillage dataset, we propose a two stage network architecture consisting of a leaf detector followed by a disease classifier. The proposed method focuses on several important stages of developing a plant disease detection model including the introduction of a new dataset, augmentation methods, analyzing different classification schemes, and image processing.

3.1 Dataset

The dataset that we were primarily focused on was the PlantVillage Dataset[4], which consists of over 50,000 images of 26 types of diseased and non-diseased leaves from 14 different crops. The images came in color, grayscale and segmented, as shown in Figure 1.

(a) Color (b) Grayscale (c) Segmented

Figure 1: An example of the data from the PlantVillage Dataset. The leaf shown in the pictures is an apple leaf with apple scab

In order to try to classify plant diseases in the field, we needed to create a dataset of our own that consists of images from each of the 38 classes in the field. This would serve as our validation set for final model. To do this we used a plant disease database called the American Phytopathological Society (APS) Image Database[5], which has peer-reviewed images of plant diseases in different environments from various APS publications. Our target number of images was 3 to 5 images per class. If there was not enough data for a class, we would try to pull the best images that we could from Google images. In the end, we had a validation set that consisted of over 150 images. An important note is that many of these images contained multiple diseased leaves of the same class in the same image.

3.2 Leaf Detector

Deep neural network-based object detection algorithms were explored as a solution for successful plant disease detection methods in situations where there are multiple leaves on the same image. These leaves were assumed to be from the same plant species. The object detection dataset was created using a randomly chosen images of leaves from the web.

These leaves were labeled with the singular class "leaf" whether or not they were diseased or healthy or positioned in complex backgrounds. The object detection network of our choice was YOLOv3 for it's recent successes in real-time object detection [6]. Specifically, we use Tiny-YOLOv3 to make our two-stage model as small as possible.

3.3 Disease Classifier

Two popular mobile friendly deep neural networks were explored for diseases classification. The first was again a Tiny-YOLOv3 network trained on the PlantVillage Dataset [6][4]. To train our Tiny-YOLOv3 we created random bounding boxes centered at the $\frac{1}{2}x$ and $\frac{1}{2}y$, randomly ranging from half to full size of the image, around each leaf with its appropriate label. We trained our Tiny-YOLOv3 on all 38 classes of the dataset.

Our second disease classifier that we experimented with was a MobileNetV2[7] network, with the hopes of a smaller parameter size and a faster inference time due to less computations. Our MobileNetV2 was trained on the segmented leaves within the PlantVillage Dataset with harsh data augmentation. Again, we trained our MobileNetV2 on all 38 classes of the PlantVillage Dataset.

4 Experiments

Two different types of architectures were included in our experiments: one-stage and two-stage detectors. The main difference between these two strategies is that two-stage detectors primarily find candidates for object locations at the first stage by using a leaf detector model and then perform classification on each candidate object location. In dealing with multiple leaves in a singular image, the per class confidences are summed and the max is chosen as the prediction for the disease. In the two stage case where the first stage, the leaf detector stage, is not able to detect any leaves, the image is fed directly into the classifier.

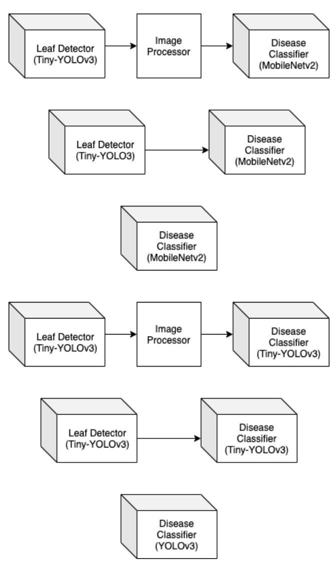

Figure 2: The following network topologies were tested. These consist of two stage networks with image processing, two stage networks without image processing, and single stage networks.

We first tested on the two stage model with image processing. The image in Figure 3 will be used as the baseline image to demonstrate our methodology using the two stage model with image processing. The result of the two stage model on the original image in Figure 3 can be seen in Figure 4.

(a) Original Image

(b) Disease Classification using Tiny-YOLOv3

Figure 3: The original image of a Grape Black Rot leaf (a) passed through a single stage Tiny-YOLOv3 disease classifier inferences an incorrect result of Strawberry Leaf Scorch (b).

(a) Leaf Detector Result

(b) Image Processor Result

(c) Disease Classifier Result

Figure 4: The original image of a Grape Black Rot leaf 3 passed through a Tiny-YOLOv3 leaf detector (b) into an image processor (c) and finally a Tiny-YOLOv3 disease classifier (d). The leaf disease is correctly classified as Grape Black Rot.

Our results from these various architectures are summarized in figure 5 below. The MobileNetv2 used in our architectures attained an accuracy of 91.11% on the PlantDisease test set. From our experimental results, the Tiny-YOLOv3 fed directly into a MobileNetv2 architecture worked the best.

Leaf Detector Model	Image Processor Included?	Disease Classifier Model	Our Dataset Validation Accuracy
None	N/A	Tiny-Yolov3	16.94%
None	N/A	MobileNetv2	25.81%
Tiny-YOLOv3	No	Tiny-YOLOv3	18.56%
Tiny-YOLOv3	Yes	Tiny-YOLOv3	13.71%
Tiny-YOLOv3	No	MobileNetv2	26.00%
Tiny-YOLOv3	Yes	MobileNetv2	21.77%

Figure 5: Results from different architectures

5 Conclusion

From the results of our experiments, it is pretty apparent that our disease classifier does not generalize well to our validation set of plants in the wild, with our best two-stage architecture achieving 26.00% accuracy. Although this accuracy is significantly better than random guessing, it is hard to say this architecture could be used to realistically diagnose plant diseases in the field reliably. The fact that there is a huge discrepancy between our architecture's PlantDisease test accuracy and our dataset's validation accuracy might be indicative of the limitations of the PlantDisease dataset when applying it to the field. This might be due to our current data augmentation scheme. A semi-supervised approach may help generalize our disease classifier to more realistic environments by creating more diverse images. However, given our very limited data with respect to the problem at hand, the results that we achieved with our two-stage architecture for classifying plant disease in the field is satisfactory and gives clear paths for further exploration.

References

[1] 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*, 2016:1–11, 2016.

[2] Sharada Prasanna Mohanty, David Hughes, and Marcel Salathe. Using deep learning for image-based plant disease detection, 2016.

[3] Marko Arsenovic, Mirjana Karanovic, Srdjan Sladojevic, Andras Anderla, and Darko Stefanovic. Solving current limitations of deep learning based approaches for plant disease detection. *Symmetry*, 11(7):939, 2019.

[4] David P. Hughes and Marcel Salathé. An open access repository of images on plant health to enable the development of mobile disease diagnostics through machine learning and crowdsourcing. *CoRR*, abs/1511.08060, 2015.

[5] The American Phytopathological Society. Aps image database, 2019.

[6] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement, 2018.

[7] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks, 2018.