# PLANTS DISEASE


UNIVERSITY OF TARTU
Institute of Computer Science

Aleksandr Makarov,
Nika Mikhailava


## Introduction

These days there are many challenges facing humanity in terms of food supply, especially considering the fact that there are over 8 billion of us already. Human society needs to increase food production by an estimated 70% by 2050[1] to meet growing demand. Putting aside the joke that global hunger and overpopulation have the same solution, we need we need to focus on improving harvesting.
Currently, infectious diseases reduce the potential yield by an average of 40% with many farmers and even professionals in plant biology experiencing yield losses sometimes as high as 100%[1]. Modern smartphones can assist in identification and treatment of many diseases without costly equipment or professional education. The described solution is perfect for developing countries as well as for huge agricultural companies and scientific institutions.

## Goals

- Test different convolutional neural networks and training approaches.
- Find trade off between size and accuracy of these models
- Make it feasible to run on mobile

## Dataset

Dataset[2] consists of images comprising healthy and diseased leaves of different plants. There are 38 classes overall and over 80k images in total. Numerous augmentations strategies are applied to data(rotation, shift, contrast etc.). We asked a test group(couple of friends) to identify disease of the plant, but even identifying plant type can be tricky(as you can see from example below).

Grape,
Black Rot

Strawberry,
Leaf scorch

Grape,
healthy

Strawberry,
healthy

Try me

## Methods

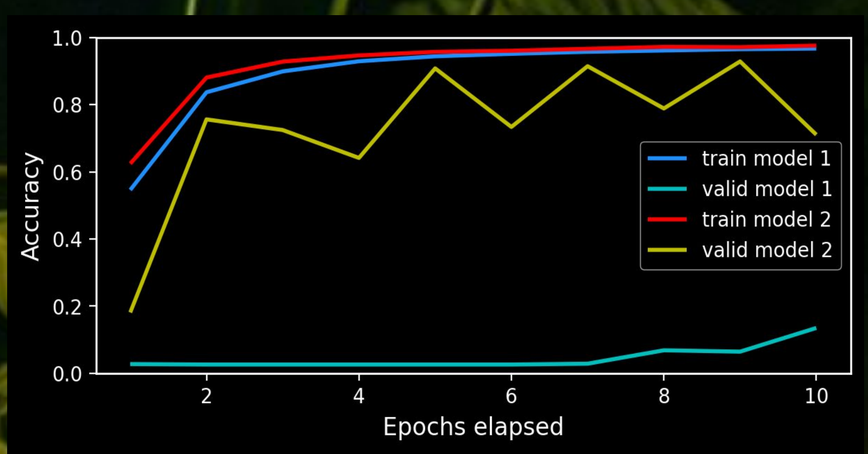
Keras & tensorflow(mortal sins) were used to train the models on GPU(CUDA) enabled devices. Two different models were used: MobileNetV2 and InceptionV3 , since they are both relatively lightweight and robust[3].
3 different approaches were utilized for each model:

i. training from scratch(randomly initialized weights)
ii. retraining the whole pretrained model
iii. training last linear layers based on pretrained network

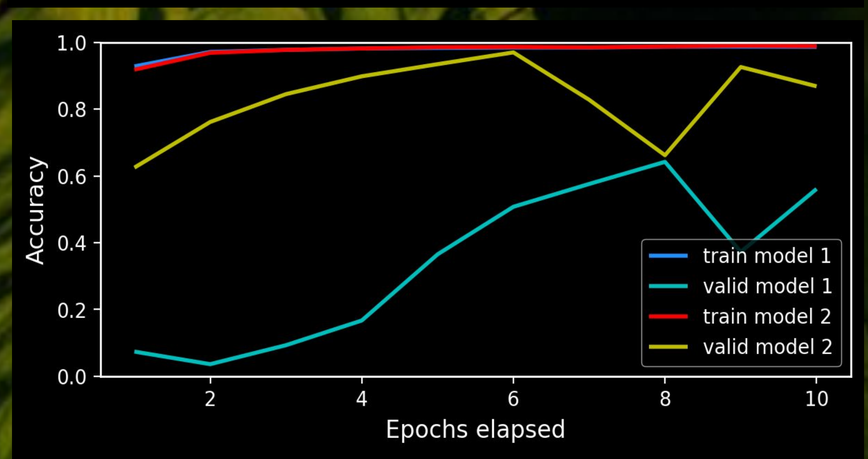
Apparently, transfer learning(training from pretrained model) is advantageous, so we also decided to check this effect. In this particular case both pretrained models were trained on imagenet dataset.

## Results

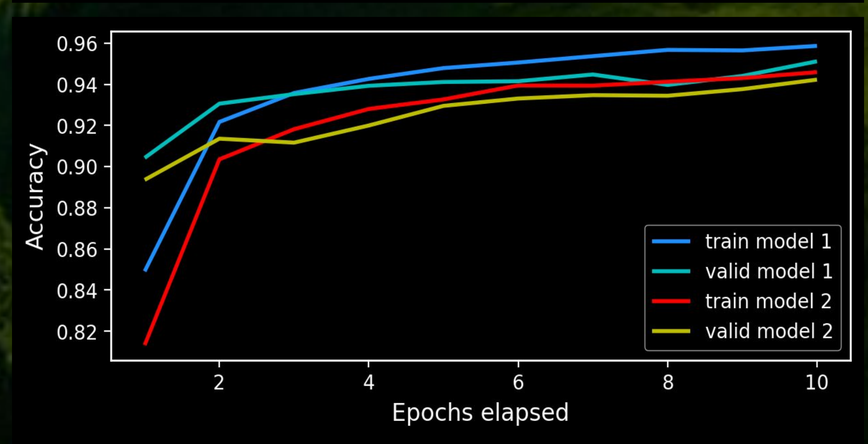
The following figures have MobileNetV2 as model 1 and InceptionV3 as model 2.




Experiment 1. Training all layers from scratch




Experiment 2. Training all layers from pretrained




Experiment 3. Training last layer from pretrained
*Range y is not [0,1]

| Model | Size(MB) | Exp. 1 | Exp. 2 | Exp. 3 |
|---|---|---|---|---|
| MobileNetV2 | 14 | 0.151 | 0.601 | **0.955** |
| InceptionV3 | 92 | 0.738 | 0.865 | **0.951** |

Table 1. Experiments 1,2,3 show accuracies for models on test set after 10 epochs.

## Conclusions

To sum up, tuning the last layer resulted in MobileNetV2 model that has higher accuracy on test set and also is much lighter than InceptionV3, that makes it a perfect choice for mobile applications.

## Sources:


[1] https://arxiv.org/ftp/arxiv/papers/1511/1511.08060.pdf
[2] https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset
[3] https://keras.io/api/applications/
[4] https://en.wikipedia.org/wiki/Transfer_learning


*Image was generated and upscaled by neural.love
Prompt: a beautiful dusty delightful painting of the plants and palms by Alfred Kubin and Giovanni Battista Piranesi, Sotto In Su, 9:16, unsplash contest winner