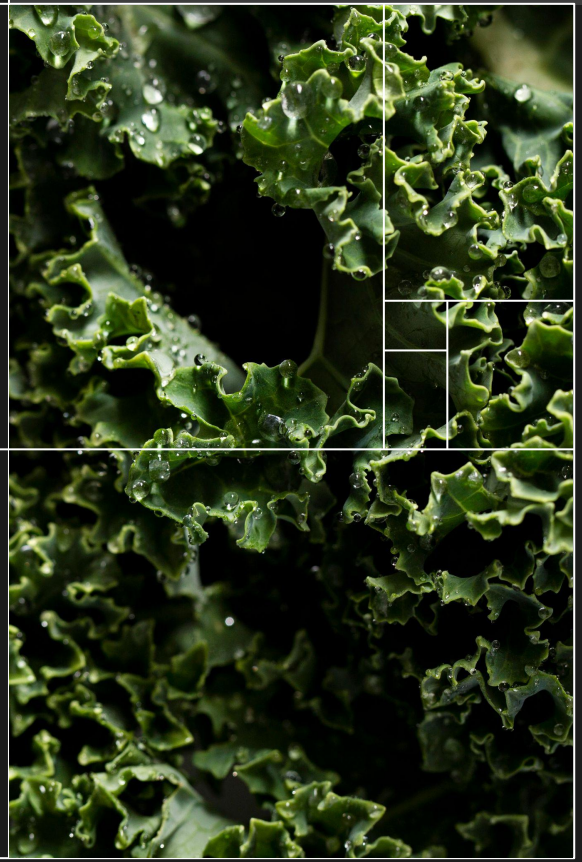
Project GreenEye



A Computer Vision project on
Preliminary Detection on Diseased Vegetables

Prepared by: Nicholas Yuen

Table of contents

01

Background & Problem Statement

Background

- Global trends threatening food security
- Singapore Food Agency's goal: 30% nutritional value by 2030 (30 by 30)
    - Limitations:
        - Limited land space
        - Sustainability/Environmental well-being
        - Limited manpower

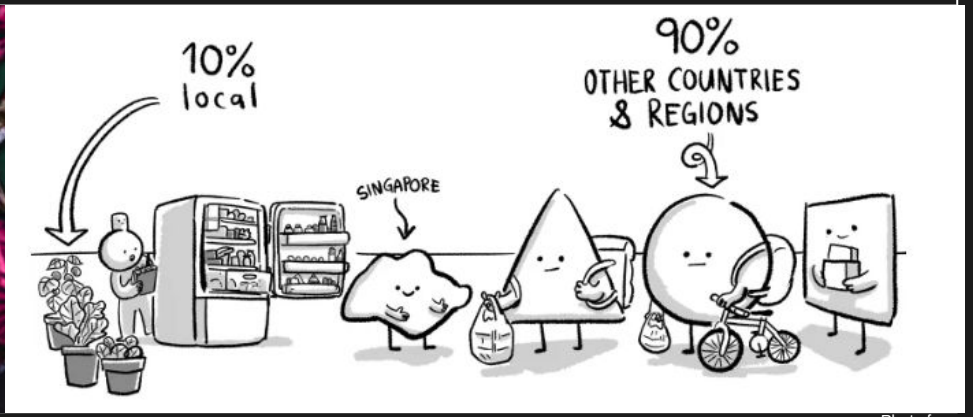


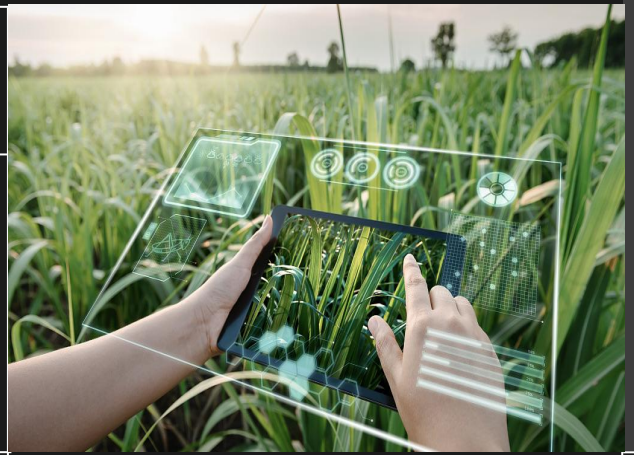
Photo from https://thewokesalaryman.com/2020/05/09/30-over-30-a-financial-goal-to-start-working-towards-by-the-time-youre-30/

Background (Cont'd)

- Use of hydroponic farms
- Other challenges:
  - Common diseases and pests
  - Time consuming / labour intensive for monitoring

Problem Statement

- Use of Computer Vision to monitor
  - Detection of symptoms for early damage control
- Feeds into binary classification model



Primary Objective:

- Achieve minimally 90 ~ 95% accuracy

Secondary Objective :

- Exploration of effectiveness under image with other interferences
- Difficulty in finding diseased plant images with interferences
- Correct classification % + Prediction probability

Prediction Probability

- Analyse of prediction probabilities of each image classified

0
‘High confidence’
Class: Diseased

0.5
‘Indecisive’

1
‘High confidence’
Class: Healthy

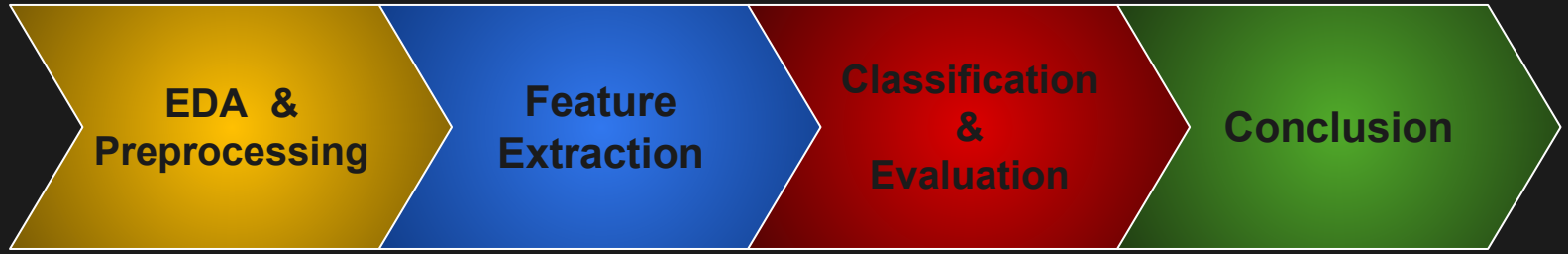
‘Confidently’ classified as diseased
Bad model

‘Confidently’ classified as healthy
Good model

Correct Classification

- $\frac{\text{Total no. classified as healthy}}{\text{Total no. of images with water droplets}}$

Workflow



- Data Acquisition & Augmentation
- Exploratory Data Analysis
- Preprocessing

- Understanding Feature Extraction
- MobileNetV2
- MobileNetV3
- MobileNetV3 + Regularization
- InceptionResNetV2

- Evaluation summary
- Limitations
- Future work

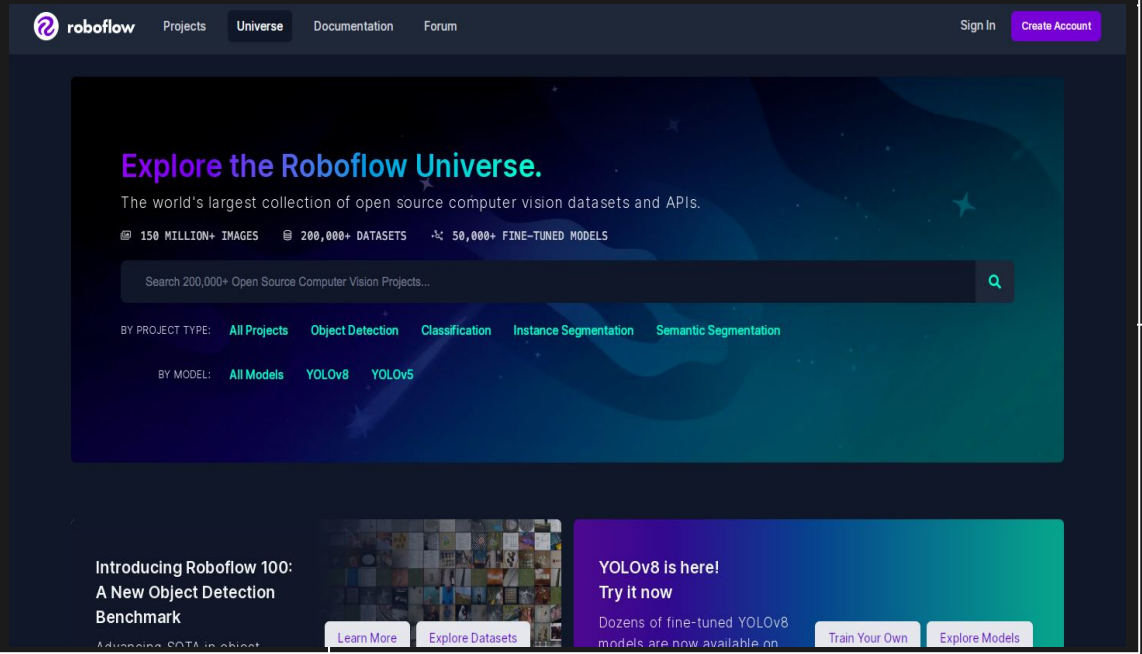
02

EDA & Preprocessing

EDA & Preprocessing → Feature Extraction → Classification & Evaluation → Conclusion

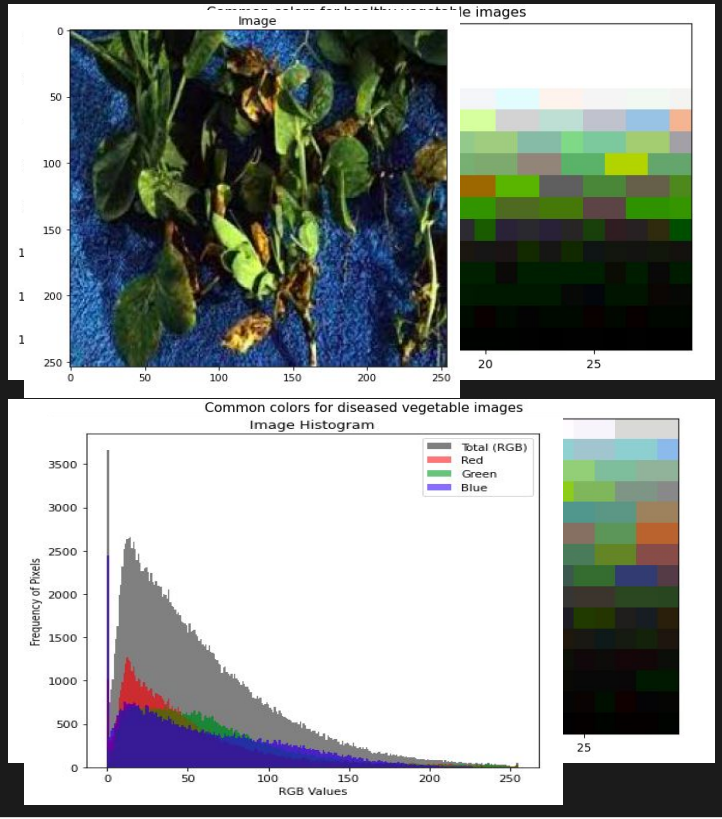
Data Acquisition & Augmentation

- Data from Roboflow
  - 468 images for training
  - 100 images for test
- Data Augmentation on images with water droplets



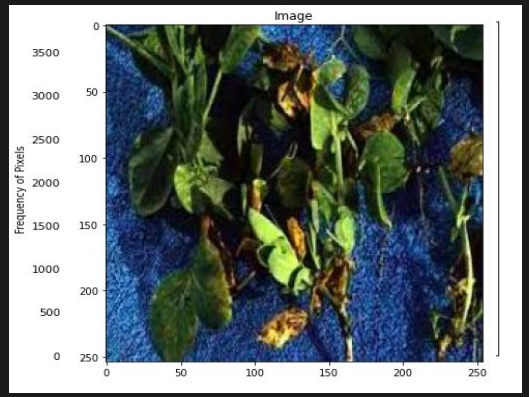
EDA

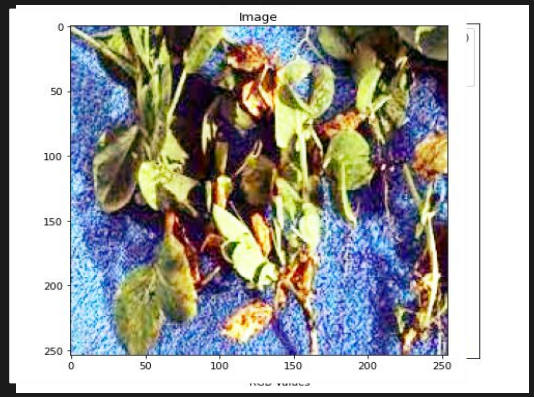
- Display of sample images
- Common occurring colors
  - Darker hues in diseased than healthy images
  - Contrast varies
- Contrast of images



Preprocessing

- Histogram Equalization
- Amplification of yellow hues





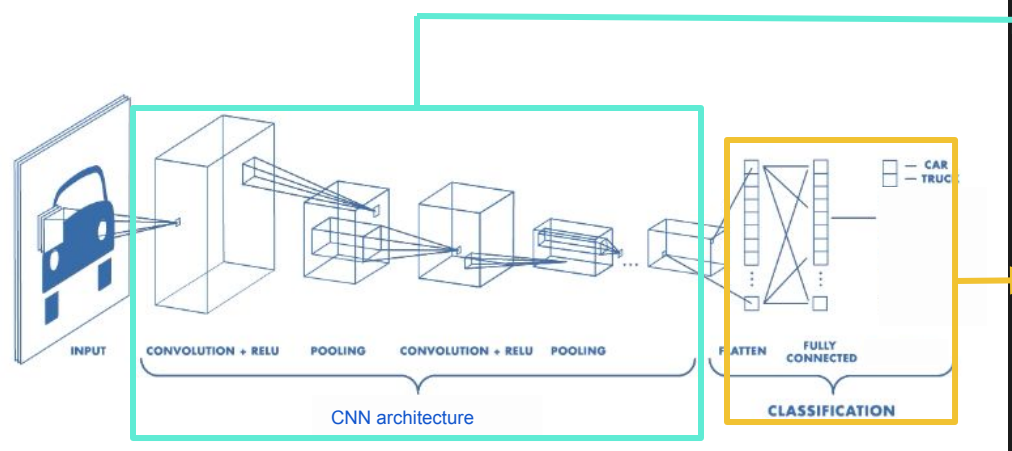
Modeling

03

EDA & Preprocessing → Feature Extraction → Classification & Evaluation → Conclusion

Understanding Feature Extraction

- Convolutional Neural Network (CNN)



Types used:

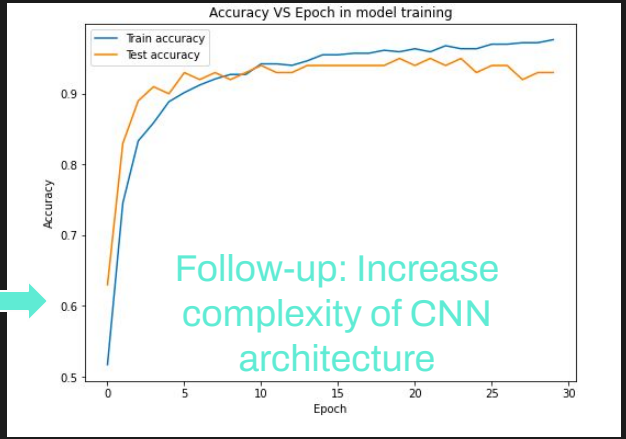
- MobileNetV2
- MobileNetV3
- InceptionResNetV2

Fully connected layer:

- 1-layer Output layer
- Sigmoid activation function

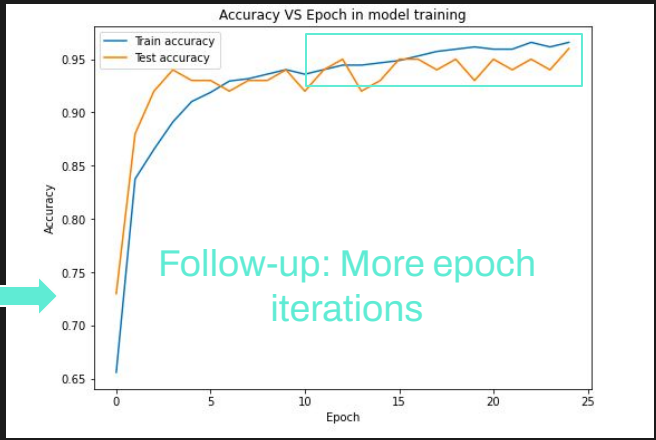
Approach 1: MobileNetV2

- Why MobileNetV2?
  - Fast convolution process using depthwise-seperable
  - Baseline choice
- Evaluation (Regular images)
  - Accuracy test score: 97%
  - Accuracy train score: 96%
- Evaluation (Water droplet images)
  - 27% correct classification
  - Majority of classification probabilities are close to zero



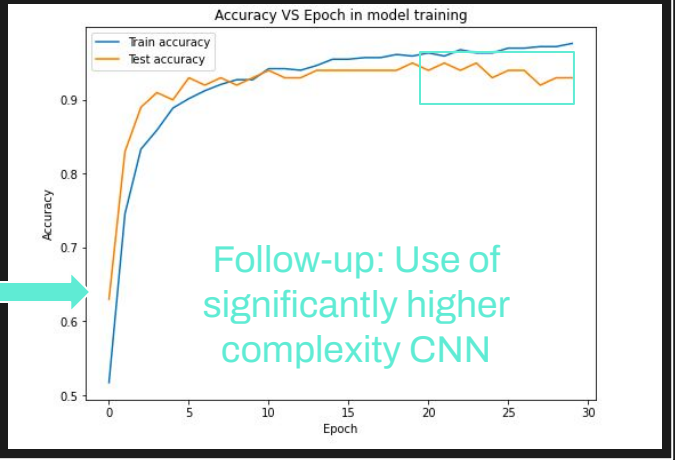
Approach 2: MobileNetV3

- Why MobileNetV3?
  - MobileNetV2 + Squeeze and Excitation + h-swish
  - More 'detailed' feature extraction
  - Faster with lower calculations
- Evaluation (Regular images)
  - 94% for both train and test
- Evaluation (Water droplet images)
  - 29% correct classification
  - Minimum probability of 0.34, mostly Centralized at 0.5



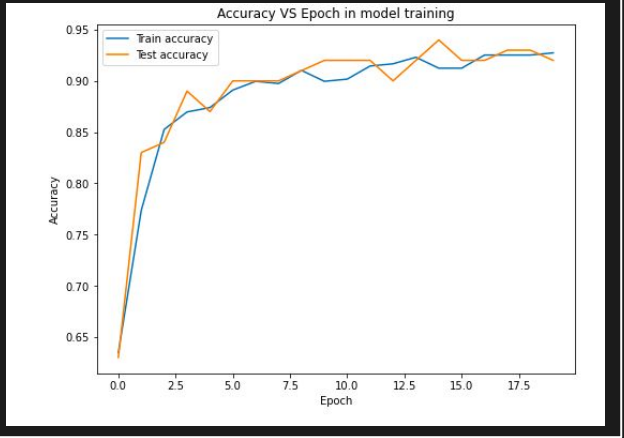
Approach 2B: MobileNetV3 + Regularization

- Why Ridge Regularization?
  - To enable higher degree of epoch iterations
  - Less aggressive type of regularization than Lasso
- Evaluation (Regular images)
  - Accuracy test score: 95%
  - Accuracy train score: 96%
- Evaluation (Water droplet images)
  - 2% correct classification
  - Minimum probability of 0.23, mostly Centralized at 0.35



Approach 3: InceptionResNetV2

- Why InceptionResNetV2?
  - Extensive feature extraction through parallel convolutional branches
  - Residual block that result in faster run time and better learning process
- Evaluation (Regular images)
  - Accuracy test score: 94%
  - Accuracy train score: 91%
- Evaluation (Water droplet images)
  - 15% correct classification
  - Substantial amount of classification probabilities are close to zero



04 Conclusion

EDA & Preprocessing → Feature Extraction → Classification & Evaluation → Conclusion

Evaluation Summary

Approach	Performance (Test accuracy on Regular images)	Performance (Water droplet images)
MobileNetV2	97%	27% and majority close to zero
MobileNetV3	94%	29% and lesser close to zero
MobileNetV3 + Regularization	95%	2% and lesser close to zero
InceptionResNetV2	94%	15% and majority close to zero

Correct classification % & proportion of prediction probabilities

Limitations

Limitations	Prospective Solutions
• Limited application for outdoor farming	• Relevant data collection
• Accuracy for night/low light images	• Use of promising preprocessing methods (eg. Learning to See in the Dark) to enhance low light images
• Potential accuracy loss on collective images of crops	• Use of Object Detection to isolate individual plant

Future works

- Use of Neural Architecture Search for more efficient exploration of CNN
- To be able to classify the various specific diseases for more efficient damage control measures.
- Extend model to detect common pest (Diamondback Moth)
- Scale app to accept live feed of images in backend

Try it out!

For mobile users

OR

greeneye-by-nicholas-yuen.streamlit.app

Thank you!

Email: nicholasyuen92@gmail.com

Github: https://git.generalassemb.ly/nichyuen