# Classifying Vegetable Images with ResNet34

Group 4: Cooper Atkins and Ange Olson

# Agenda

- Introduction
- Description of Dataset
- Models and Experimental Setup
- Results
- Model Interpretation
- Summary and Conclusion
- References

# Introduction

**Problem:** What happens if you aren’t able to determine what vegetable you’re holding or looking at due to physical limitations?

**Possible Solutions:** App that tells you what item you’re holding or looking at via picture or hovering the camera over the item

**Use Cases:**

- Checking you’ve grabbed the right item in a fridge
- Knowing what item you’re looking at in a grocery store
- Pre-determining which vegetable is being weighed to avoid manually punching in item code

# Description of Dataset

**Dataset:** Vegetable Image Dataset
**From:** Kaggle - DCNN-Based_Vegetable_Image_Classification_Using_Transfer_Learning_A_Comparative_Study
**Data Collection:** All images were collected by study originators from vegetable farms and markets

## Categories of Vegetables

- Beans
- Bitter Gourd
- Bottle Gourd
- Brinjal
- Broccoli
- Cabbage
- Capsicum
- Carrot
- Cauliflower
- Cucumber
- Papaya
- Potato
- Pumpkin
- Radish
- Tomato

## Breakdown of Data

- Train = 1,000 each
- Validation = 200 each
- Test = 200 each

- Total Data = 1,400 each
  - 21,000 Images Total

# Examples of Images in Training Data

Bean Image

Cucumber Image

Capsicum Image

Potato Image

Cauliflower Image

Cabbage Image

Broccoli Image

Radish Image

Tomato Image

Brinjal Image

Papaya Image

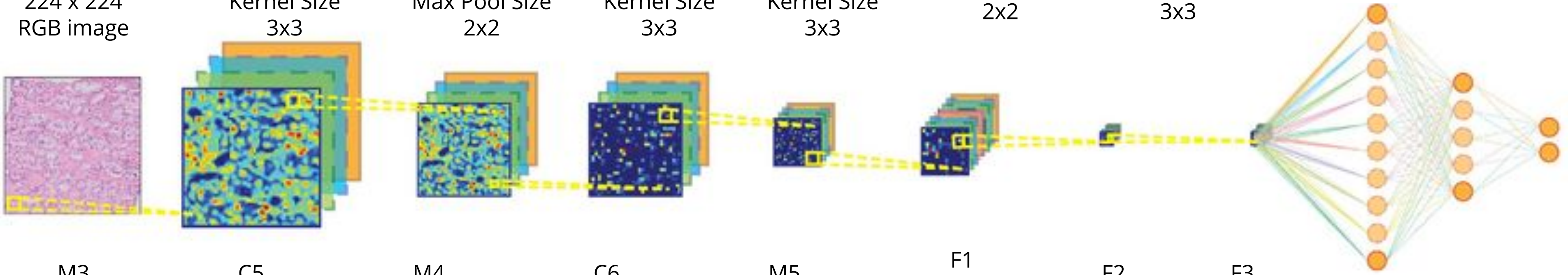
Bottle_Gourd Image

Carrot Image

Bitter Gourd Image

Pumpkin Image

# Model: Convolution (Benchmark)



Input
224 x 224
RGB image

C1
Kernel Size
3x3

M1
Max Pool Size
2x2

C2
Kernel Size
3x3

C3
Kernel Size
3x3

M2
Avg Pool Size
2x2

C4
Kernel Size
3x3

M3
Avg Pool Size
2x2

C5
Kernel Size
3x3

M4
Avg Pool Size
2x2

C6
Kernel Size
3x3

M5
Avg Pool Size
2x2

F1
ReLu
Batchnorm
800 to 512

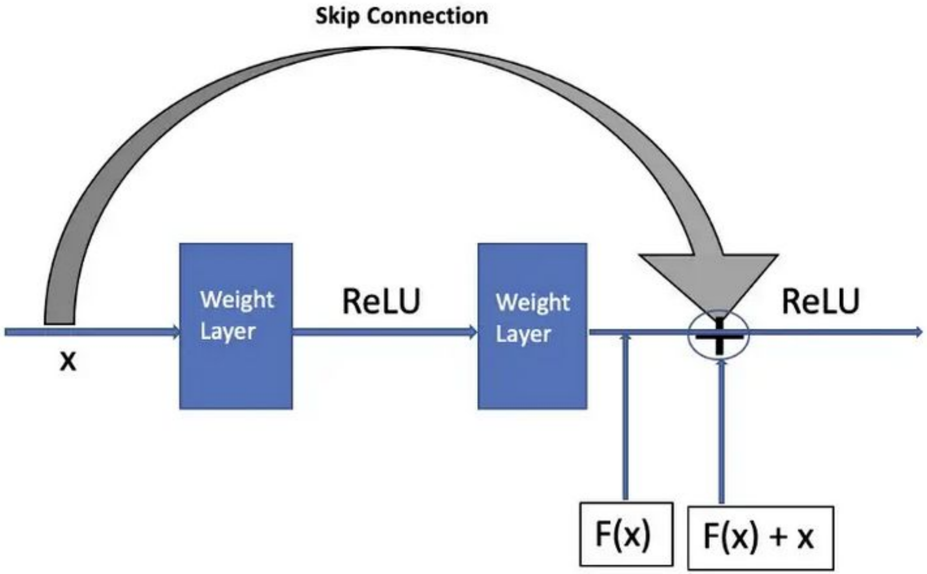
F2
ReLu
512 to 256

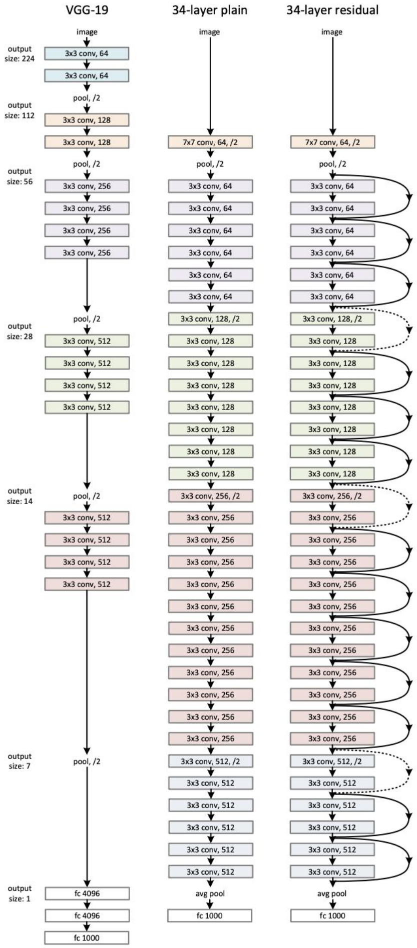
F3
ReLu
256 to 15

F4
Softmax

Image is example CNN, not a direct map of actual CNN built as described in words

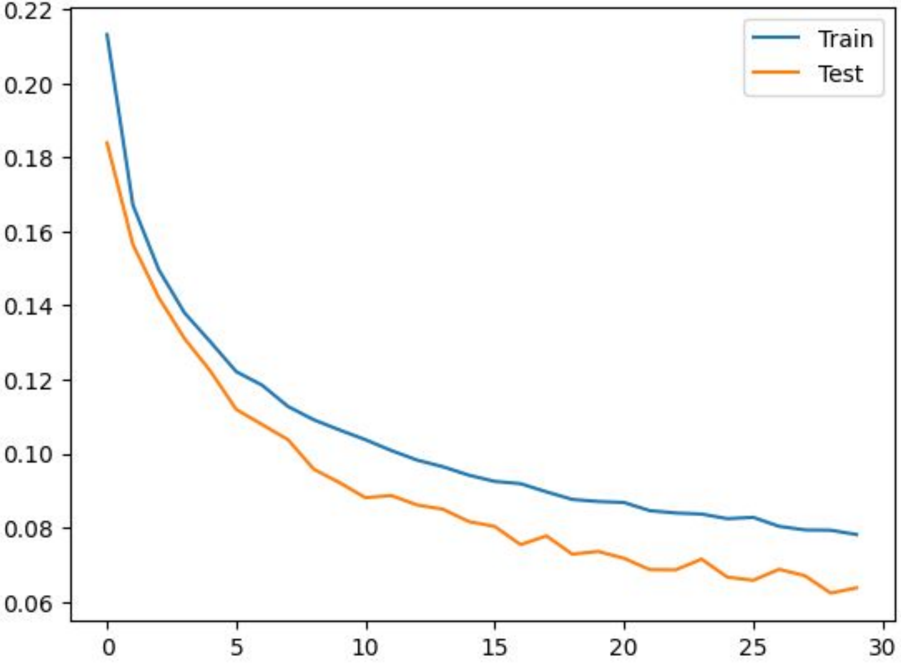
# Primary Model: ResNet34



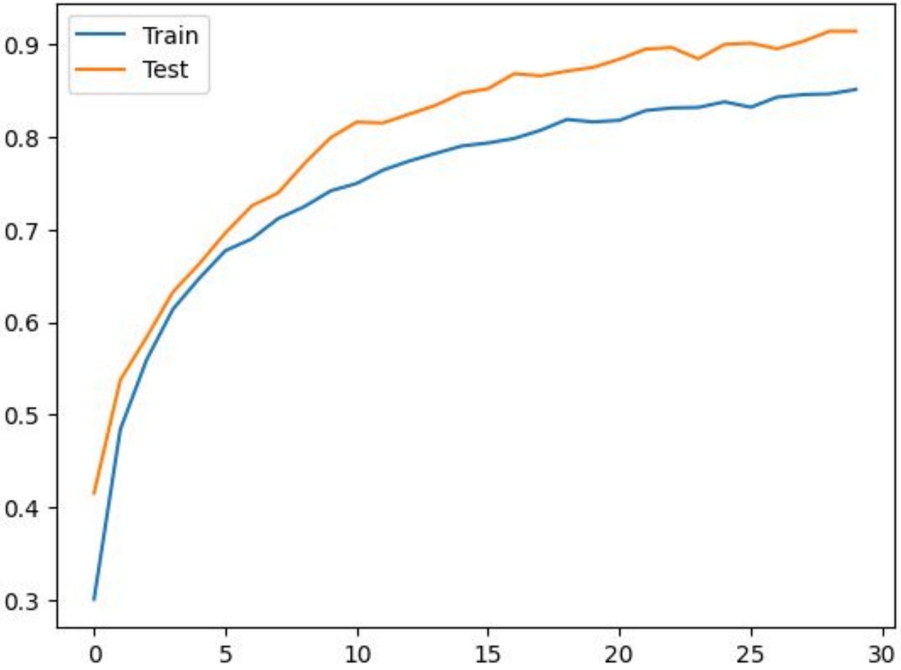


# Benchmark Results:

**Train vs. Validation Loss**

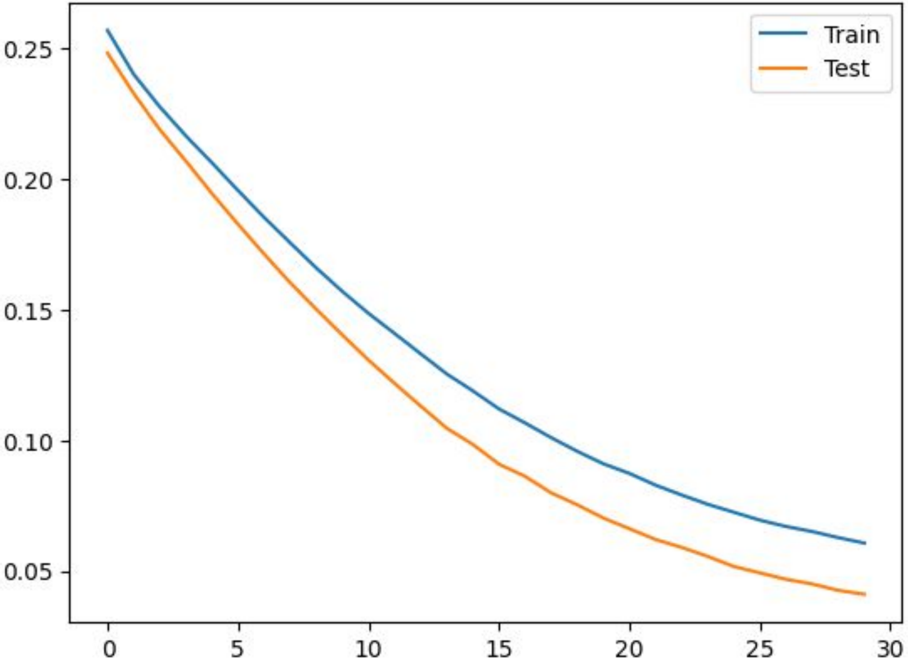


**Train vs. Validation Accuracy**

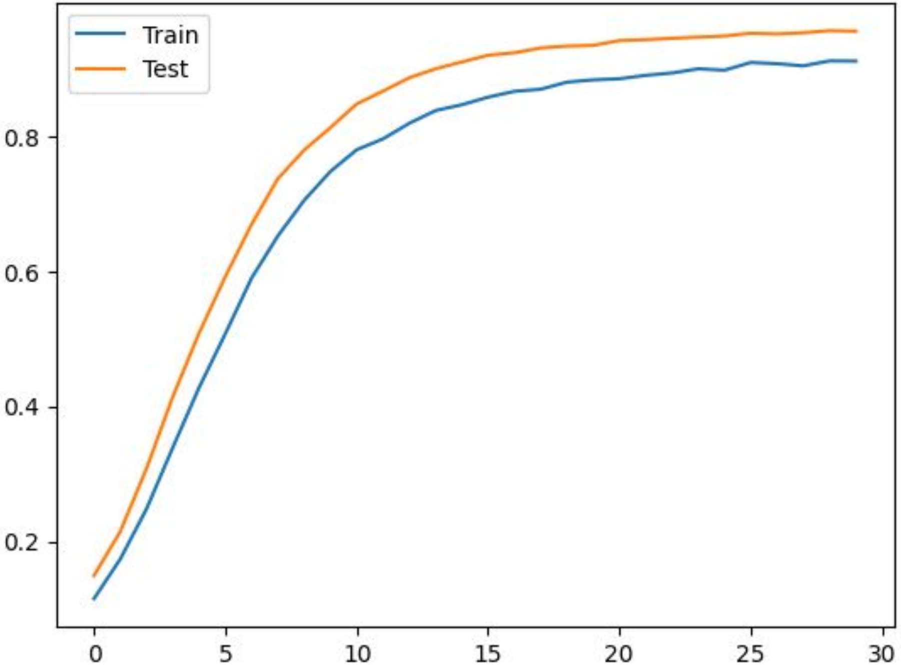


# Primary Model Results:

**Train vs. Validation Loss**



**Train vs. Validation Accuracy**



# Post-Hoc Analysis on Testing - Benchmark Model

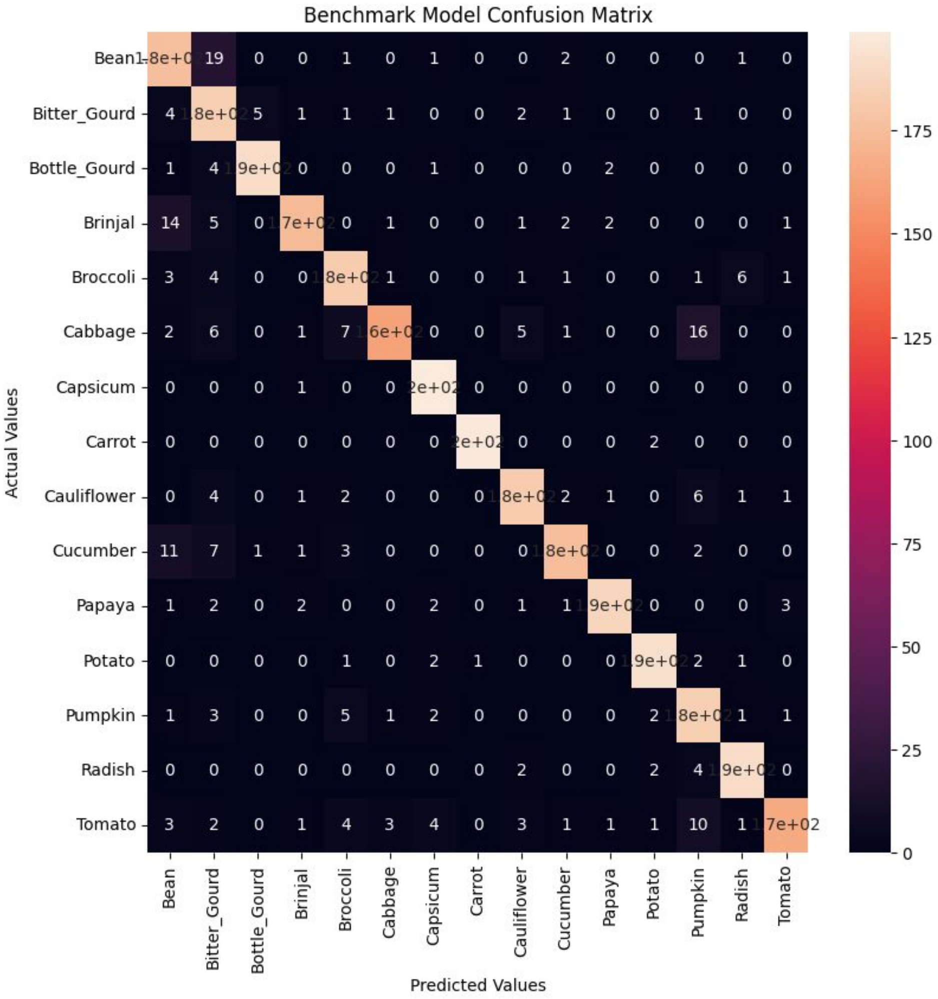
| Class | Precision | Recall | F1-Score | Class | Precision | Recall | F1-Score |
|---|---|---|---|---|---|---|---|
| **Bean** | 0.81 | 0.88 | 0.85 | **Cauliflower** | 0.92 | 0.91 | 0.92 |
| **Bitter Gourd** | 0.77 | 0.92 | 0.84 | **Cucumber** | 0.94 | 0.88 | 0.91 |
| **Bottle Gourd** | 0.97 | 0.96 | 0.96 | **Papaya** | 0.97 | 0.94 | 0.95 |
| **Brinjal** | 0.96 | 0.87 | 0.91 | **Potato** | 0.96 | 0.96 | 0.96 |
| **Broccoli** | 0.88 | 0.91 | 0.90 | **Pumpkin** | 0.81 | 0.92 | 0.86 |
| **Cabbage** | 0.96 | 0.81 | 0.88 | **Radish** | 0.95 | 0.96 | 0.95 |
| **Capsicum** | 0.94 | 0.99 | 0.97 | **Tomato** | 0.96 | 0.83 | 0.89 |
| **Carrot** | 0.99 | 0.99 | 0.99 | | | | |

# Post-Hoc Analysis on Testing - Primary Model

| Class | Precision | Recall | F1-Score |
|---|---|---|---|
| **Bean** | 0.91 | 0.99 | 0.95 |
| **Bitter Gourd** | 0.99 | 0.93 | 0.96 |
| **Bottle Gourd** | 0.99 | 0.87 | 0.93 |
| **Brinjal** | 0.88 | 0.94 | 0.91 |
| **Broccoli** | 0.98 | 0.97 | 0.98 |
| **Cabbage** | 0.96 | 0.99 | 0.98 |
| **Capsicum** | 0.94 | 0.97 | 0.96 |
| **Carrot** | 0.98 | 0.99 | 0.98 |
| **Cauliflower** | 0.99 | 0.97 | 0.98 |
| **Cucumber** | 0.87 | 0.88 | 0.88 |
| **Papaya** | 0.92 | 0.89 | 0.91 |
| **Potato** | 0.95 | 0.97 | 0.96 |
| **Pumpkin** | 0.97 | 0.97 | 0.97 |
| **Radish** | 0.99 | 0.99 | 0.99 |
| **Tomato** | 0.98 | 0.94 | 0.96 |

# Benchmark Confusion Matrix

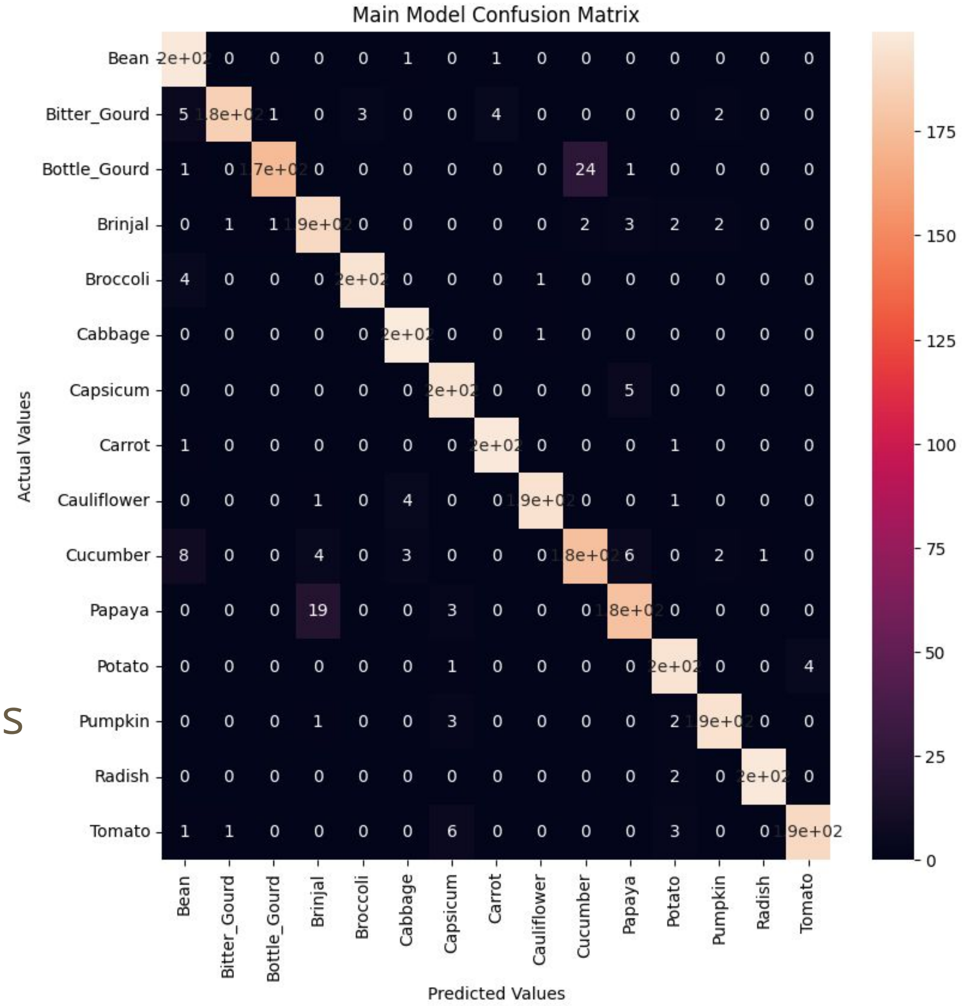
- 5 Key areas of misclassification
  - Brinjal → Bean
  - Cucumber → Bean
  - Bean → Bitter Gourd
  - Tomato → Pumpkin
  - Cabbage → Pumpkin
- General performance is a bit spotty, but overall manages to correctly classify the vast majority



| | Bean | Bitter_Gourd | Bottle_Gourd | Brinjal | Broccoli | Cabbage | Capsicum | Carrot | Cauliflower | Cucumber | Papaya | Potato | Pumpkin | Radish | Tomato |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bean | 1.8e+02 | 19 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 2 | 0 | 0 | 0 | 1 | 0 |
| Bitter_Gourd | 4 | 1.8e+02 | 5 | 1 | 1 | 1 | 0 | 0 | 2 | 1 | 0 | 0 | 1 | 0 | 0 |
| Bottle_Gourd | 1 | 4 | 1.9e+02 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 |
| Brinjal | 14 | 5 | 0 | 1.7e+02 | 0 | 1 | 0 | 0 | 1 | 2 | 2 | 0 | 0 | 0 | 1 |
| Broccoli | 3 | 4 | 0 | 0 | 1.8e+02 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 6 | 1 |
| Cabbage | 2 | 6 | 0 | 1 | 7 | 1.6e+02 | 0 | 0 | 5 | 1 | 0 | 0 | 16 | 0 | 0 |
| Capsicum | 0 | 0 | 0 | 1 | 0 | 0 | 2e+02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Carrot | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2e+02 | 0 | 0 | 0 | 2 | 0 | 0 | 0 |
| Cauliflower | 0 | 4 | 0 | 1 | 2 | 0 | 0 | 0 | 1.8e+02 | 2 | 1 | 0 | 6 | 1 | 1 |
| Cucumber | 11 | 7 | 1 | 1 | 3 | 0 | 0 | 0 | 0 | 1.8e+02 | 0 | 0 | 2 | 0 | 0 |
| Papaya | 1 | 2 | 0 | 2 | 0 | 0 | 2 | 0 | 1 | 1 | 1.9e+02 | 0 | 0 | 0 | 3 |
| Potato | 0 | 0 | 0 | 0 | 1 | 0 | 2 | 1 | 0 | 0 | 0 | 1.9e+02 | 2 | 1 | 0 |
| Pumpkin | 1 | 3 | 0 | 0 | 5 | 1 | 2 | 0 | 0 | 0 | 0 | 2 | 1.8e+02 | 1 | 1 |
| Radish | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 2 | 4 | 1.9e+02 | 0 |
| Tomato | 3 | 2 | 0 | 1 | 4 | 3 | 4 | 0 | 3 | 1 | 1 | 1 | 10 | 1 | 1.7e+02 |

# Primary Model Confusion Matrix

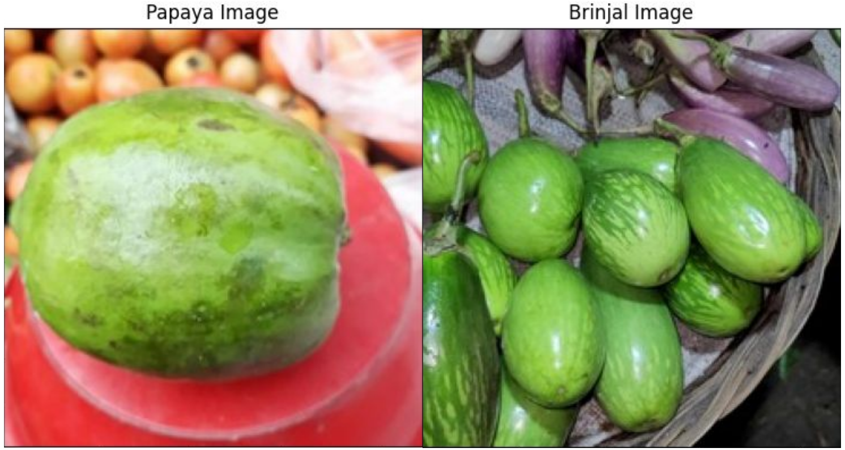
- 2 Key areas of misclassification
  - Papaya → Brinjal
  - Bottle Gourd → Cucumber
- Performance is good across the board with 2 obvious exceptions
- Improvement efforts should focus on those residuals



| | Bean | Bitter_Gourd | Bottle_Gourd | Brinjal | Broccoli | Cabbage | Capsicum | Carrot | Cauliflower | Cucumber | Papaya | Potato | Pumpkin | Radish | Tomato |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Bean | 2e+02 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Bitter_Gourd | 5 | 1.8e+02 | 1 | 0 | 3 | 0 | 0 | 4 | 0 | 0 | 0 | 0 | 2 | 0 | 0 |
| Bottle_Gourd | 1 | 0 | 1.7e+02 | 0 | 0 | 0 | 0 | 0 | 0 | 24 | 1 | 0 | 0 | 0 | 0 |
| Brinjal | 0 | 1 | 1 | 1.9e+02 | 0 | 0 | 0 | 0 | 0 | 2 | 3 | 2 | 2 | 0 | 0 |
| Broccoli | 4 | 0 | 0 | 0 | 2e+02 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Cabbage | 0 | 0 | 0 | 0 | 0 | 2e+02 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Capsicum | 0 | 0 | 0 | 0 | 0 | 0 | 2e+02 | 0 | 0 | 0 | 5 | 0 | 0 | 0 | 0 |
| Carrot | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 2e+02 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Cauliflower | 0 | 0 | 0 | 1 | 0 | 4 | 0 | 0 | 1.9e+02 | 0 | 0 | 1 | 0 | 0 | 0 |
| Cucumber | 8 | 0 | 0 | 4 | 0 | 3 | 0 | 0 | 0 | 1.8e+02 | 6 | 0 | 2 | 1 | 0 |
| Papaya | 0 | 0 | 0 | 19 | 0 | 0 | 3 | 0 | 0 | 0 | 1.8e+02 | 0 | 0 | 0 | 0 |
| Potato | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 2e+02 | 0 | 0 | 4 |
| Pumpkin | 0 | 0 | 0 | 1 | 0 | 0 | 3 | 0 | 0 | 0 | 0 | 2 | 1.9e+02 | 0 | 0 |
| Radish | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 2e+02 | 0 |
| Tomato | 1 | 1 | 0 | 0 | 0 | 0 | 6 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 1.9e+02 |

# Pattern of Misclassification on Primary Model

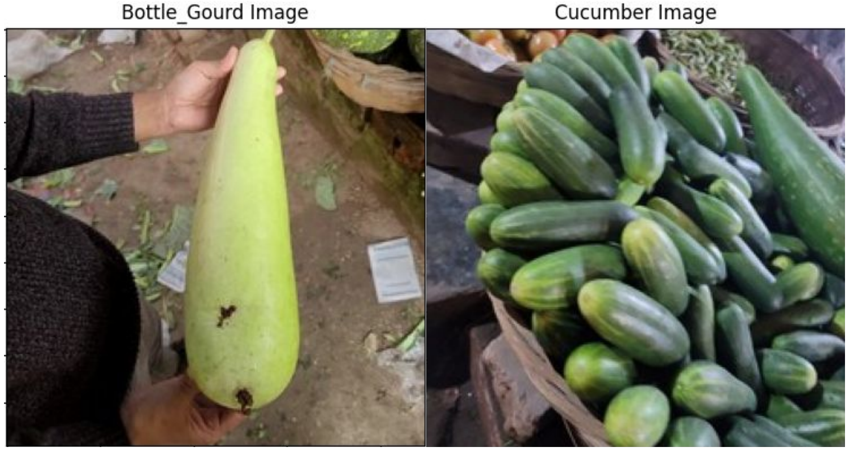
## Papaya → Brinjal



Papaya Image

Brinjal Image

## Bottle Gourd → Cucumber



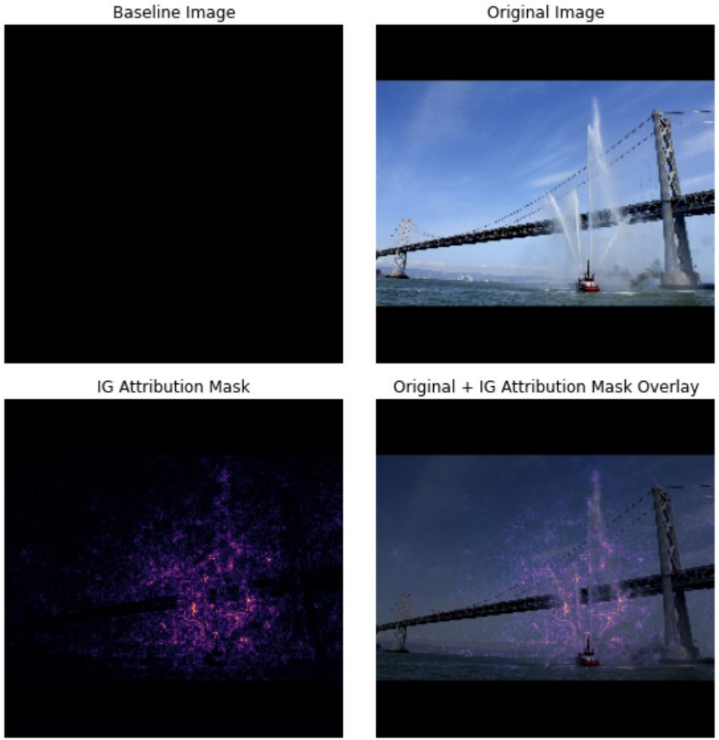
Bottle_Gourd Image

Cucumber Image

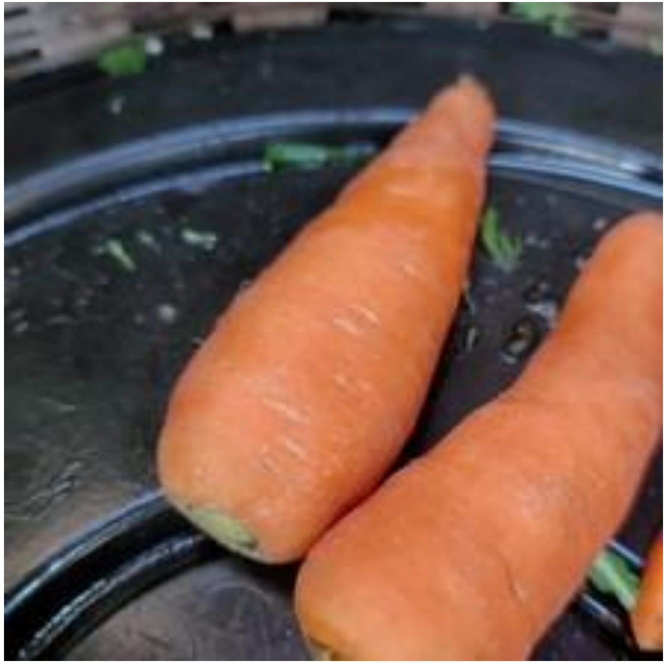
# Model Interpretation: Integrated Gradient

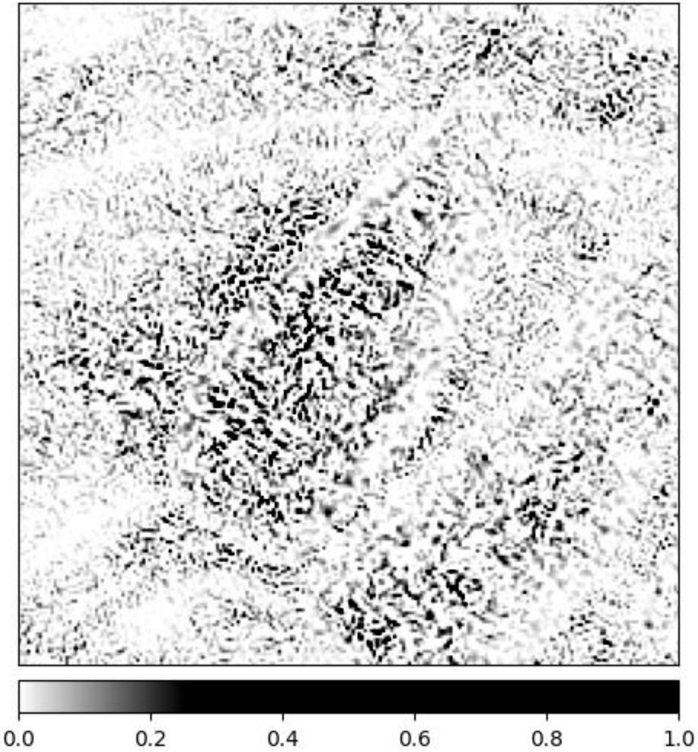
**Goal:** use Gradient calculation/approximation to attribute increases or decreases to the probability of belonging to a class by feature

$$IntegratedGrads_i(x) ::= (x_i - x'_i) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i}$$

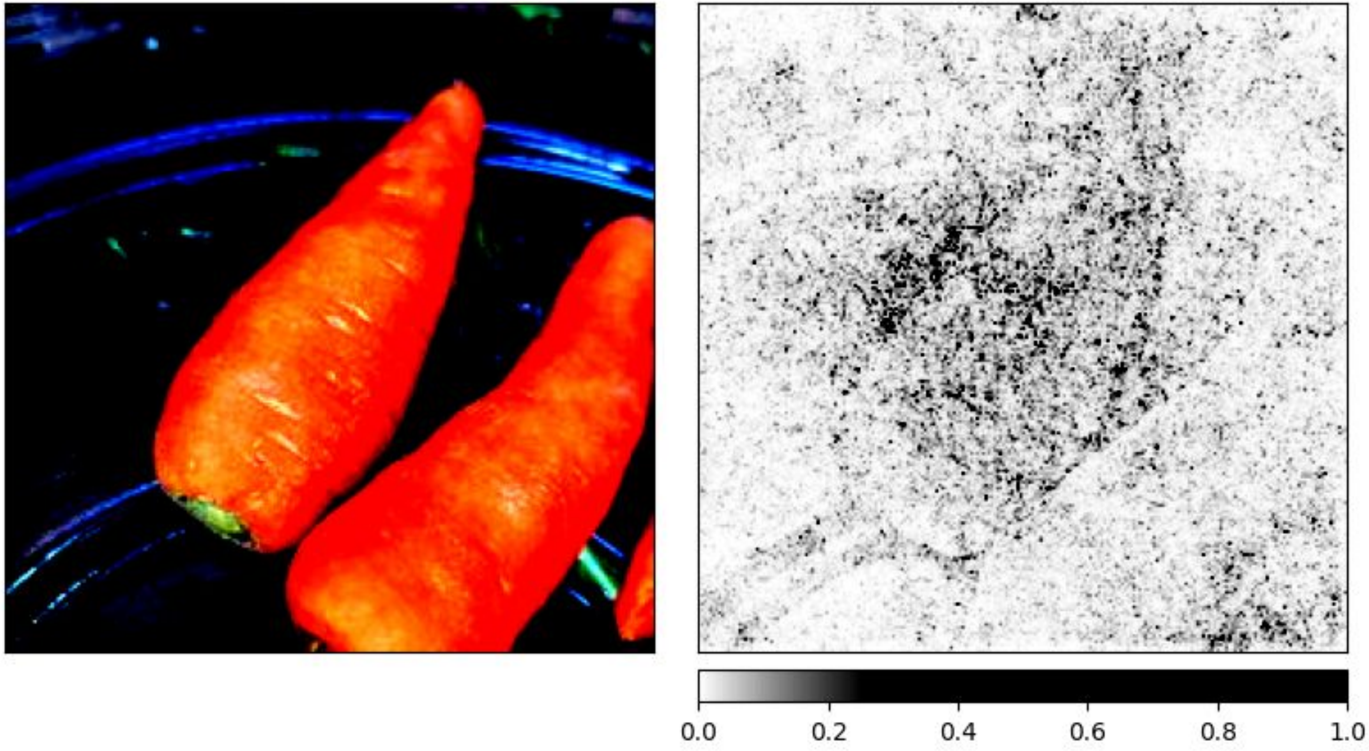


# Integrated Gradient

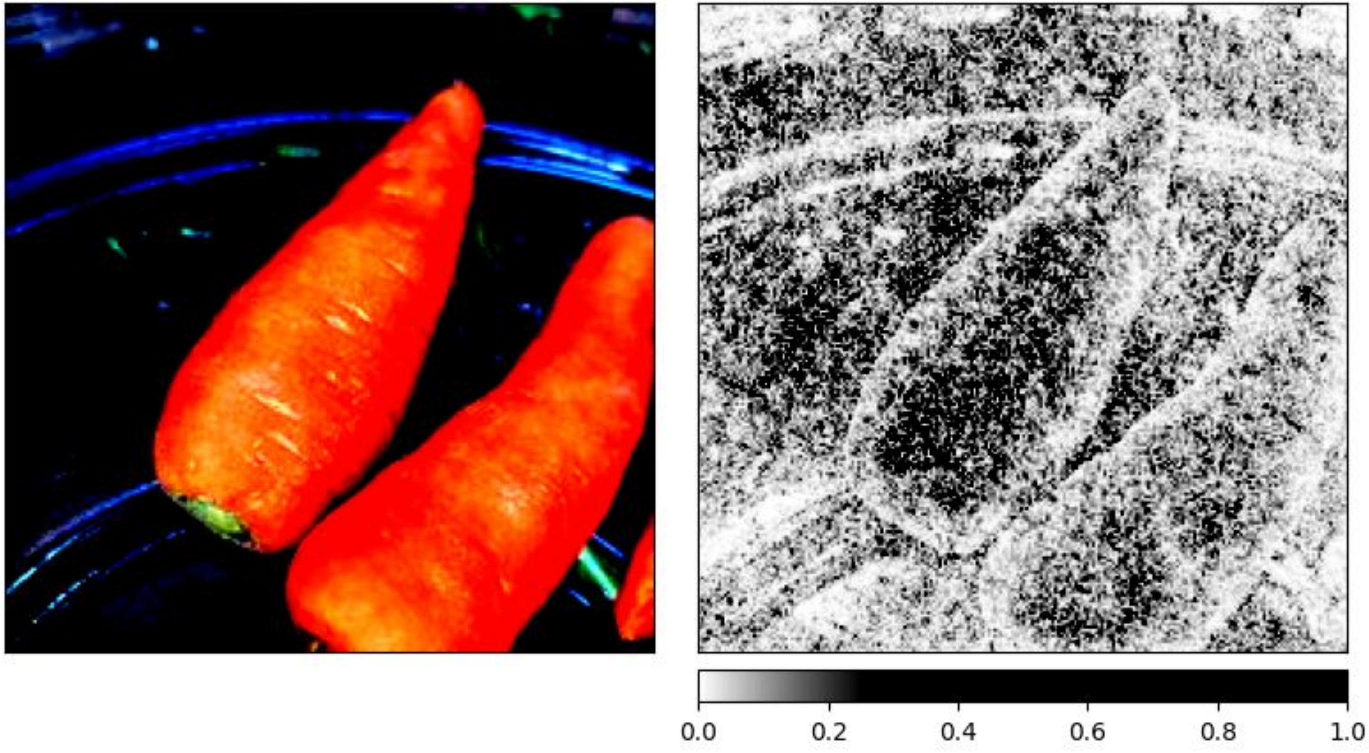




# Integrated Gradient: Noise Tunnel



# Gradient SHAP Approximation



# Conclusion + Next Steps

- **Outcome:** Transfer ResNet34 model is better than the CNN Benchmark
- **Future Testing:** Increase resources, greater diversity of samples
- **Future Testing:** Add separate out of sample images to test population
- **Future Development:** Consider further subclassifications
  - Ripe vs unripe determination
  - Multilabel Classification
- **Future Development:** Create sub-models
  - Train the model to predict a category called “bitter gourd or cucumber” then develop a separate model as a binary classifier

# Questions?

# References

- https://captum.ai/tutorials/Resnet_TorchVision_Interpret
- https://towardsdatascience.com/how-to-load-a-custom-image-dataset-on-pytorch-bf10b2c529e0
- https://www.tensorflow.org/tutorials/interpretability/integrated_gradients
- https://papers.nips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf
- https://github.com/slundberg/shap#deep-learning-example-with-gradientexplainer-tensorflowkeraspytorch-models
- https://medium.com/analytics-vidhya/understanding-resnet-architecture-869915cc2a98
- https://arxiv.org/pdf/1512.03385.pdf
- https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
- https://medium.com/@bdhuma/which-pooling-method-is-better-maxpooling-vs-minpooling-vs-average-pooling-95fb03f45a9#:~:text=Average%20pooling%20method%20smooths%20out,lighter%20pixels%20of%20the%20image.
- https://www.analyticsvidhya.com/blog/2021/06/confusion-matrix-for-multi-class-classification/