Corn Leaf Disease Classification

A Convolutional Neural Network Analysis

By Clement Chen

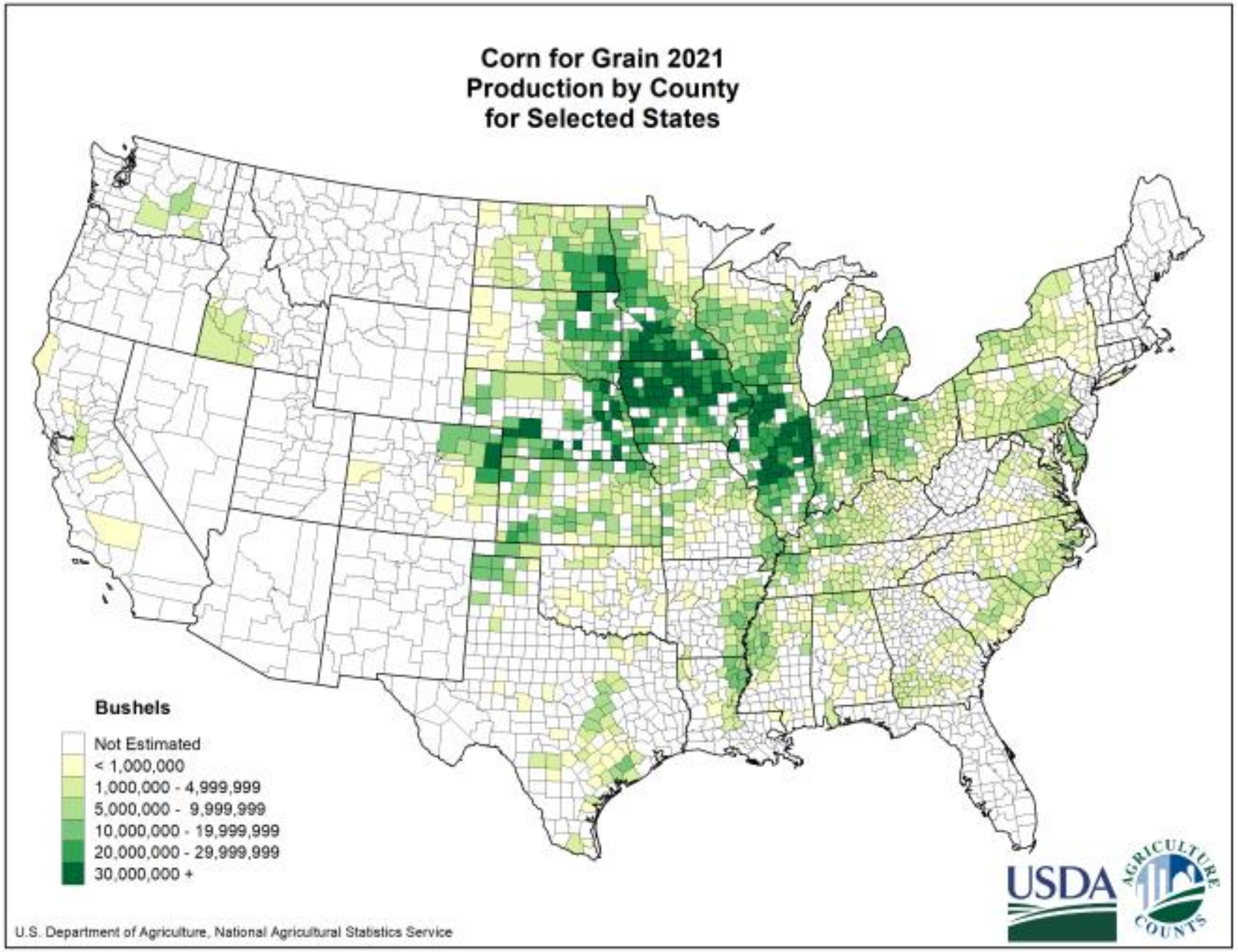
The Problem:

- Corn leaf diseases are diverse and prevalent causing millions in losses each year
- Current way of identifying corn leaf disease is through slow laboratory testing
- Human identification can be prone to errors

Can we develop a machine learning algorithm that can predict corn leaf disease?

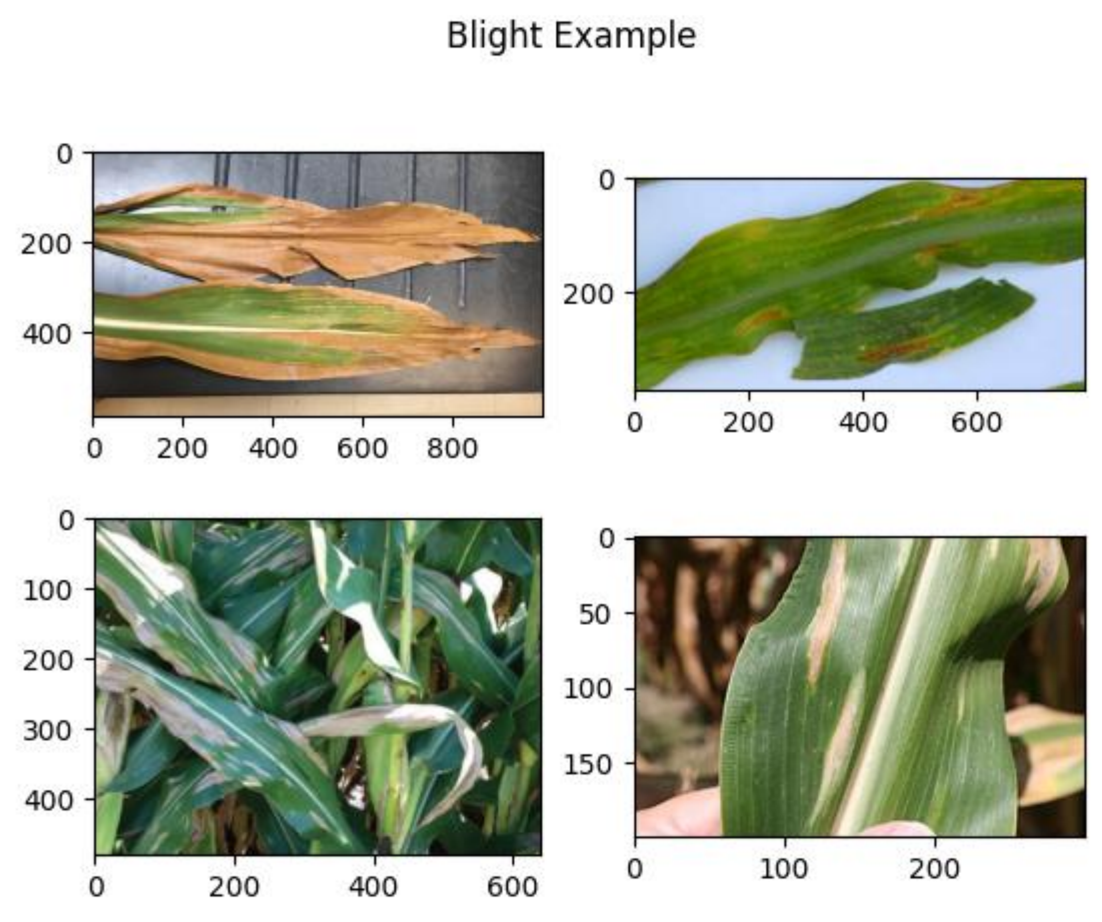
Who Would Find This Analysis Useful?

- Farmers
- Food and Beverage Corporations
- Energy Industry



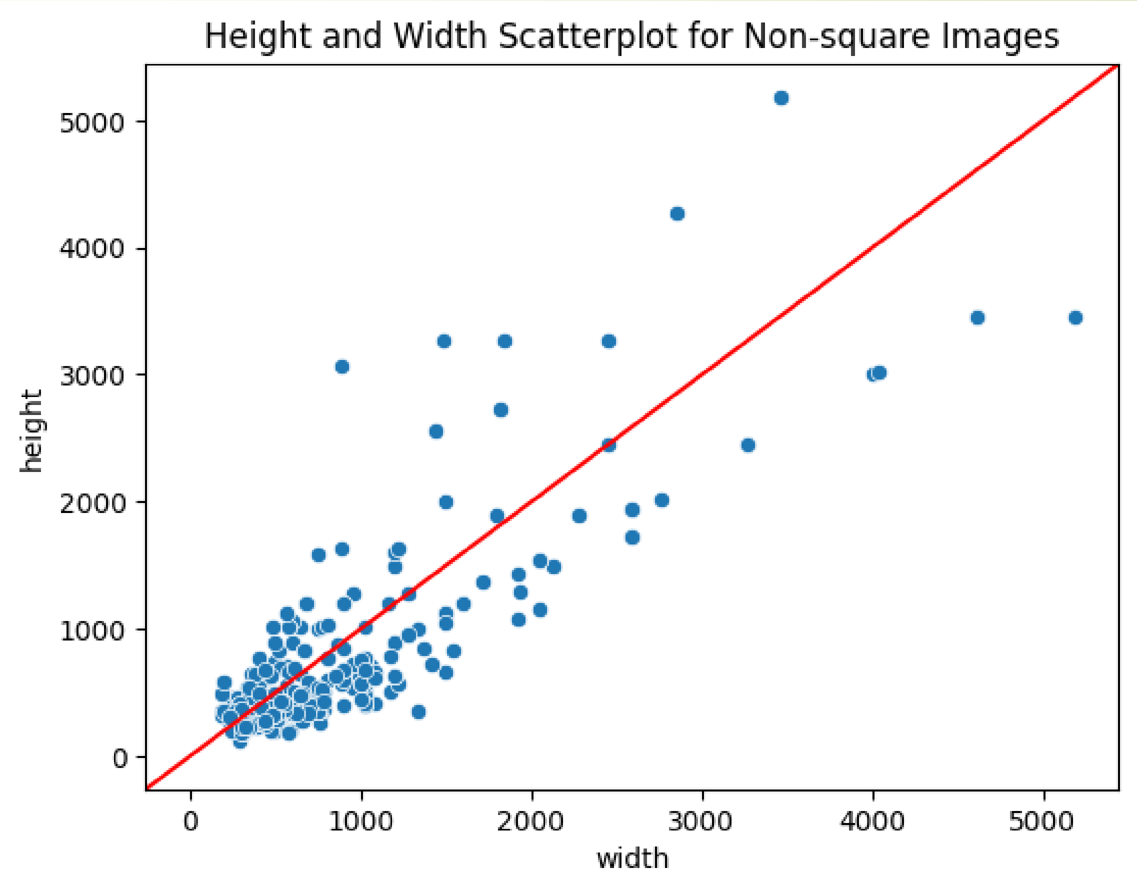
Corn Leaf Dataset

- 4188 RGB JPEG images of various dimensions
- Source: PlantDoc and PlantVillage datasets
- 4 classes: Blight, Common Rust, Gray Leaf Spot, and Healthy with 1146,1306, 574, and 1162 instances respectively
- The images have varying backgrounds, lightning, and can part of a leaf or contain multiple plants. Some images are partially occluded.



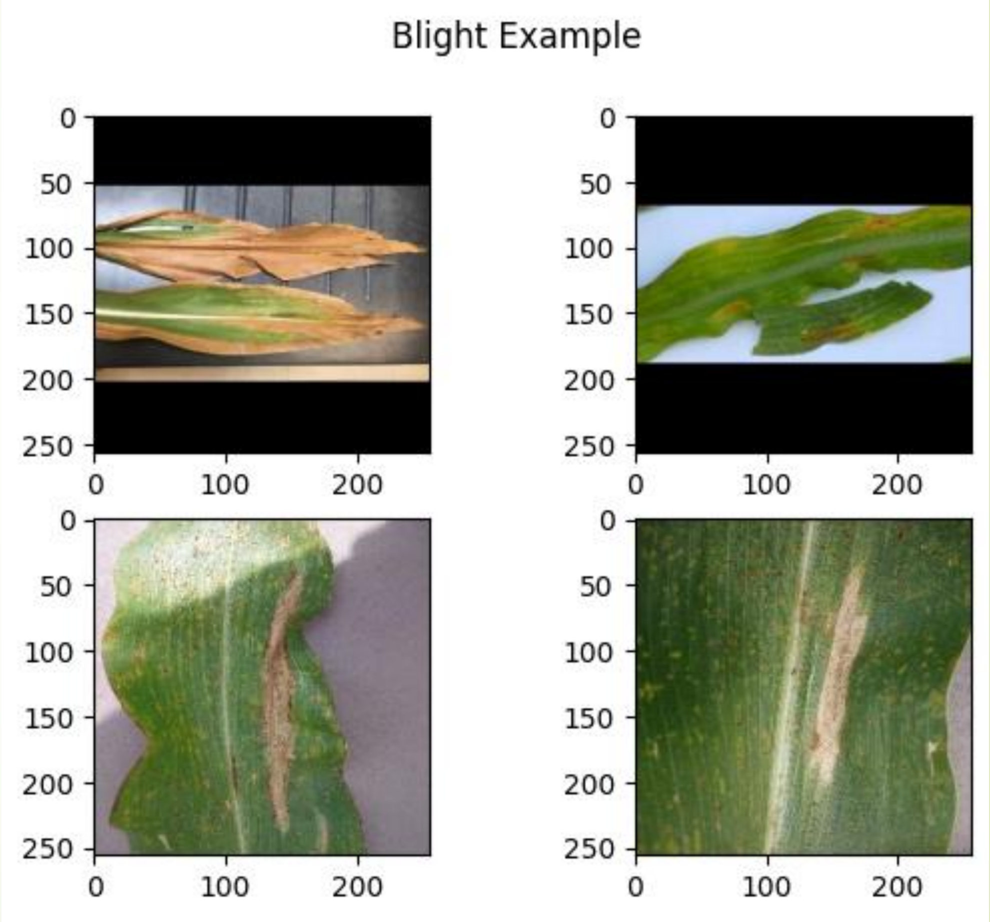
Data Wrangling and EDA

- CNNs require input images be the same
- 92% of our data has dimensions of 256 x 256
- The other 8% fall on this scatter plot.
- There are several options to resize this data:
  - Cropping
  - Stretching
  - Zero-padding



Data Wrangling and EDA

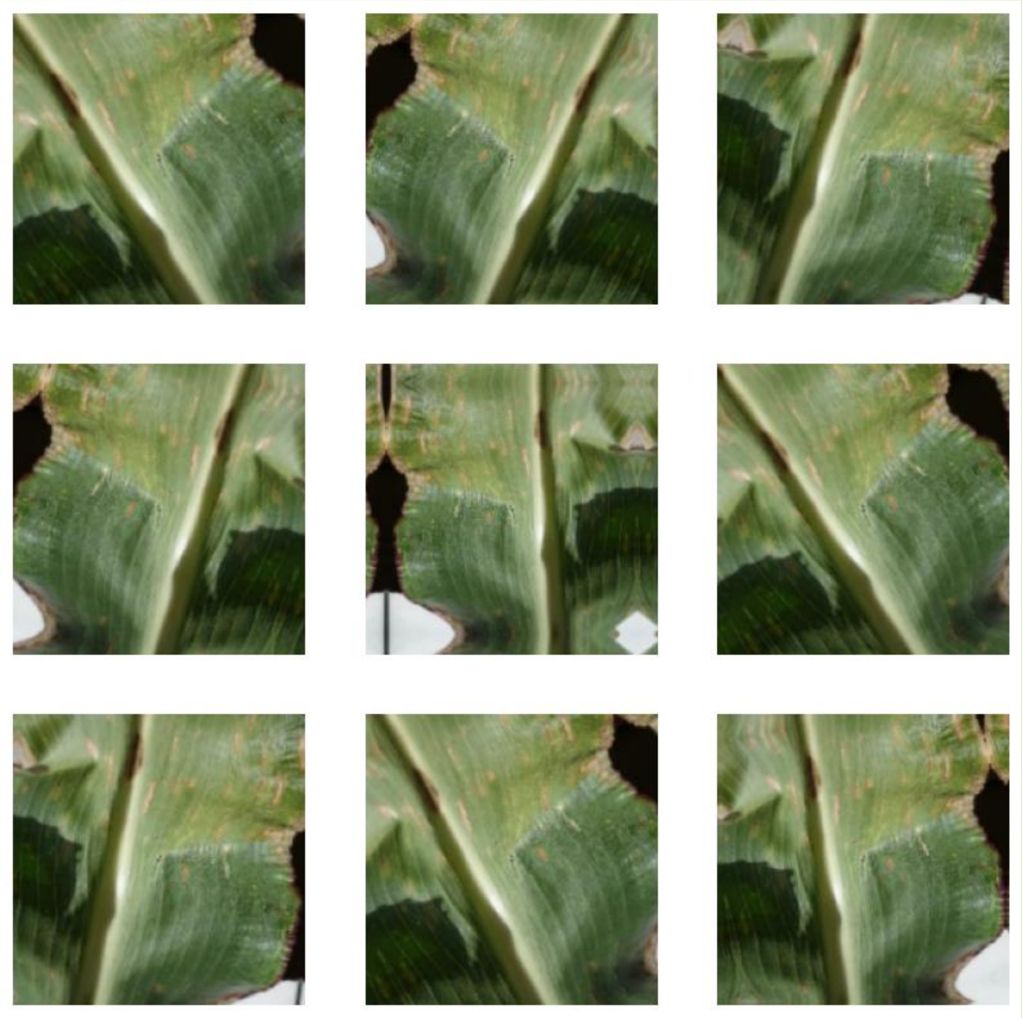
- Cropping
  - Lost pixels
  - Wide variety of image dimensions
- Stretching
  - Feature distortion due to changed aspect ratio
  - Granularity loss
- Zero-Padding
  - Maintains aspect ratio
  - Pad with ‘zero’ pixels



Preprocessing and Modeling

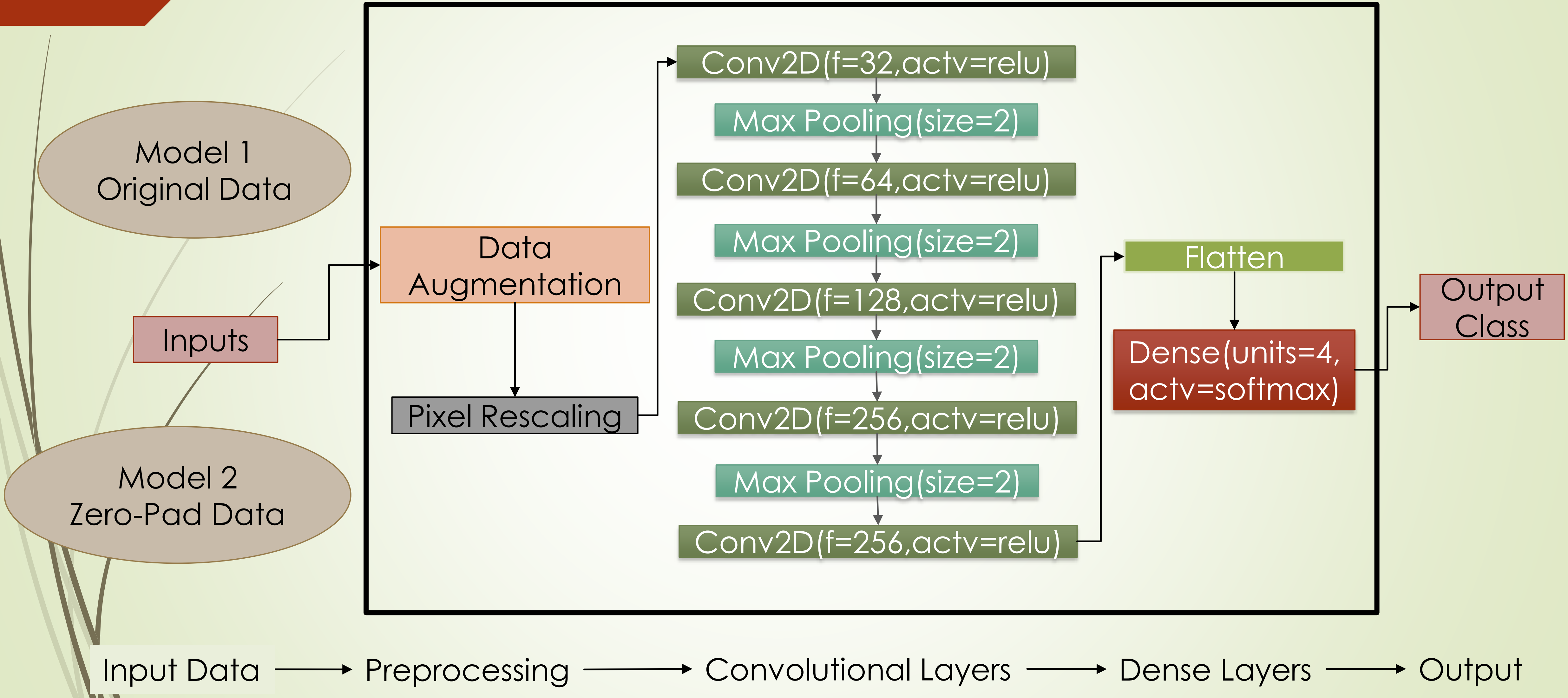
- Data Augmentation
  - Random horizontal flip
  - Image rotation +/- 10%
  - Zoom in or out +/- 20%

Data Augmentation provides a way of generating more data for the model to train on to reduce overfitting. One image can generate many other slightly different images of the same class.

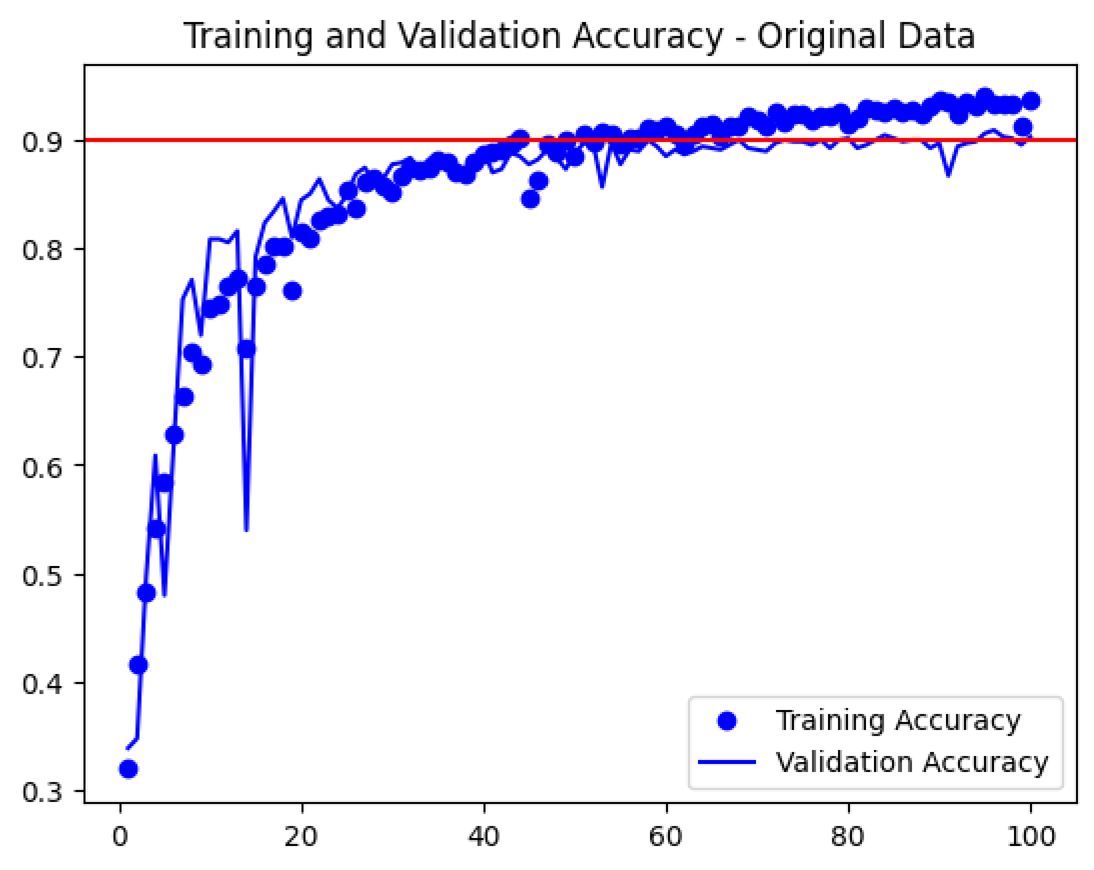


Preprocessing and Modeling

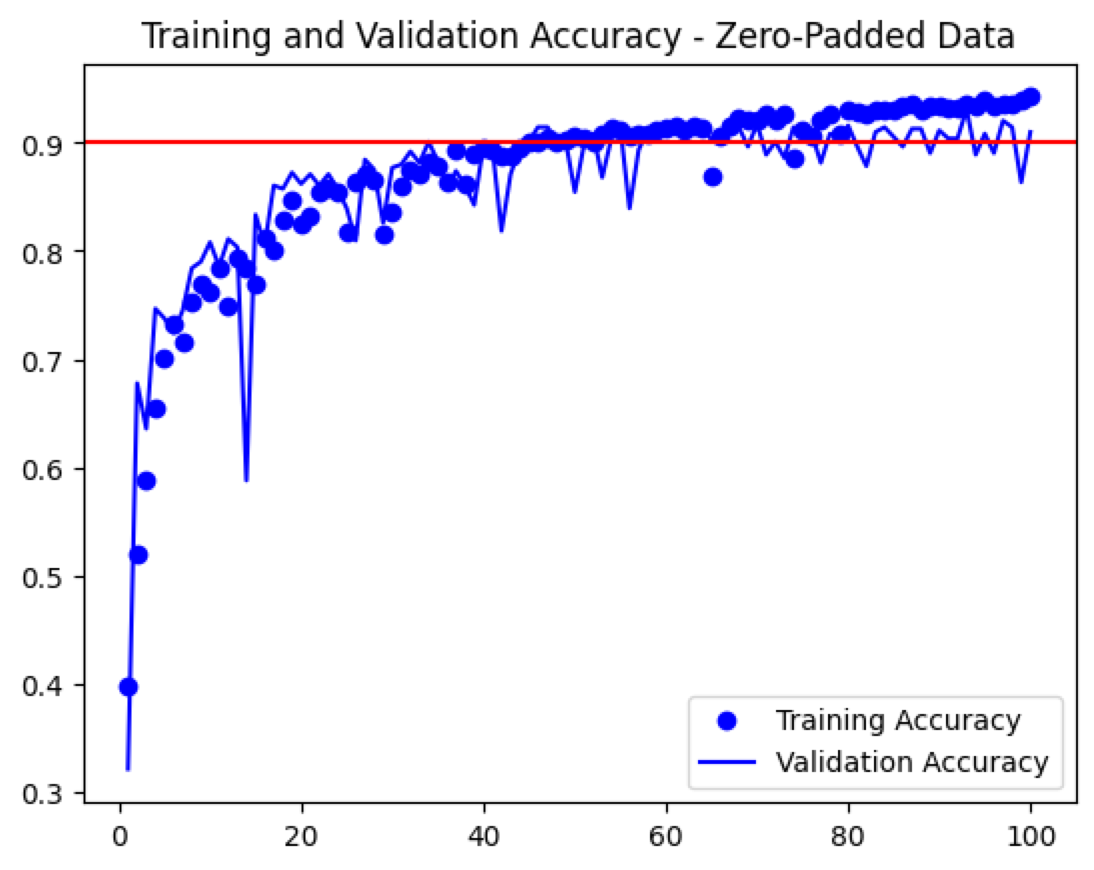
Base CNN



Preprocessing and Modeling

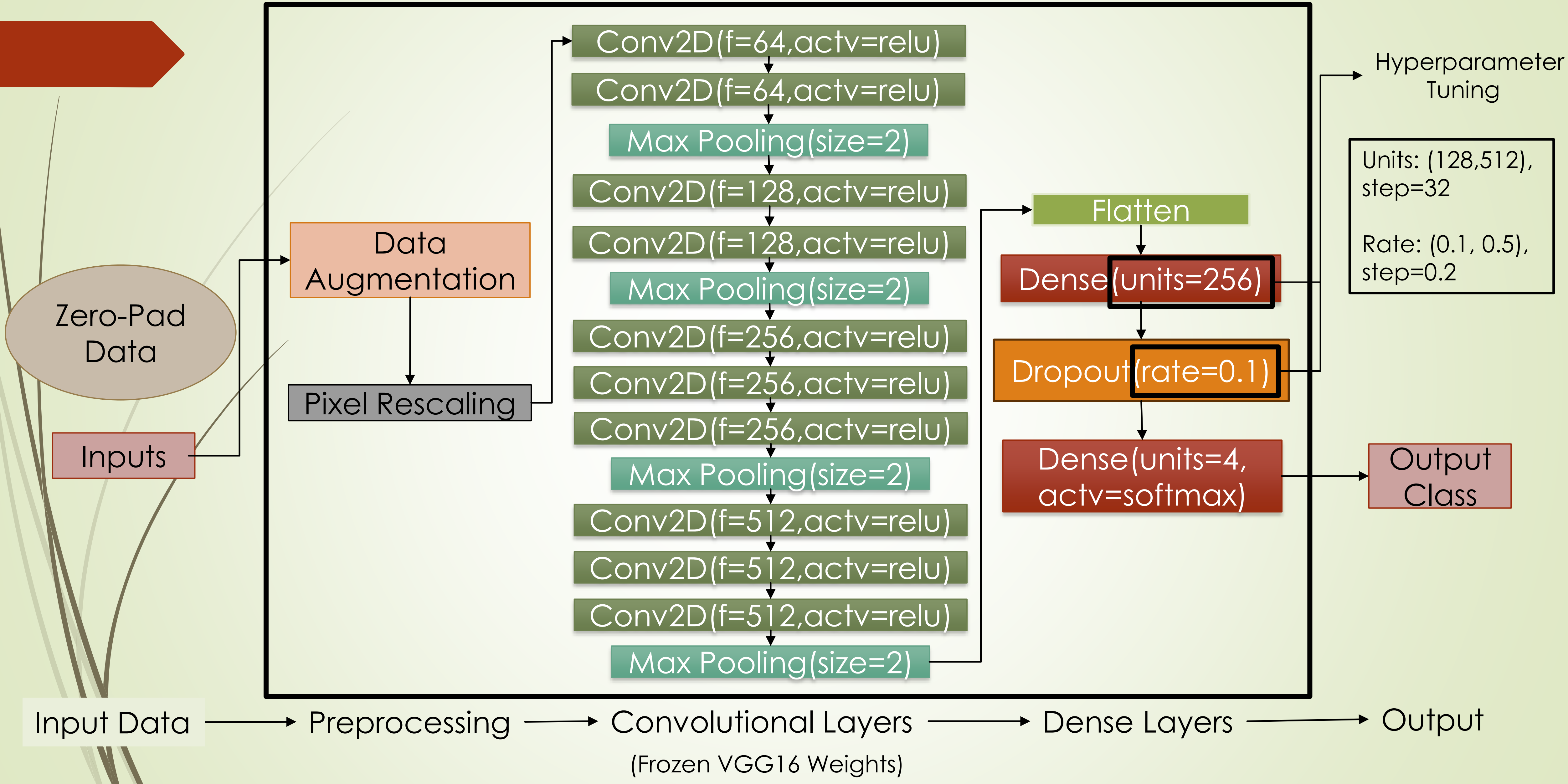


Original data reaches a validation accuracy of ~ 90% before starting to overfit around epoch 65.

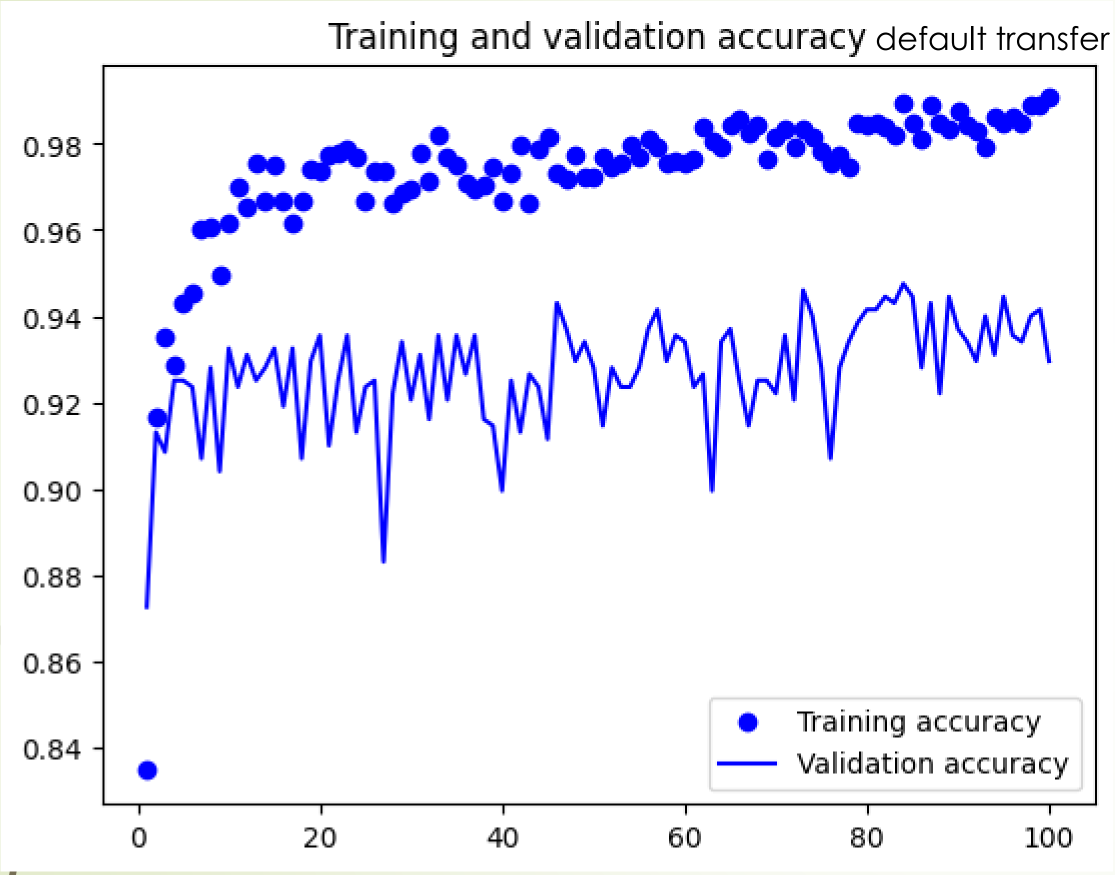


Zero-Padded data reaches a validation accuracy of ~93% before starting to overfit around epoch 80.

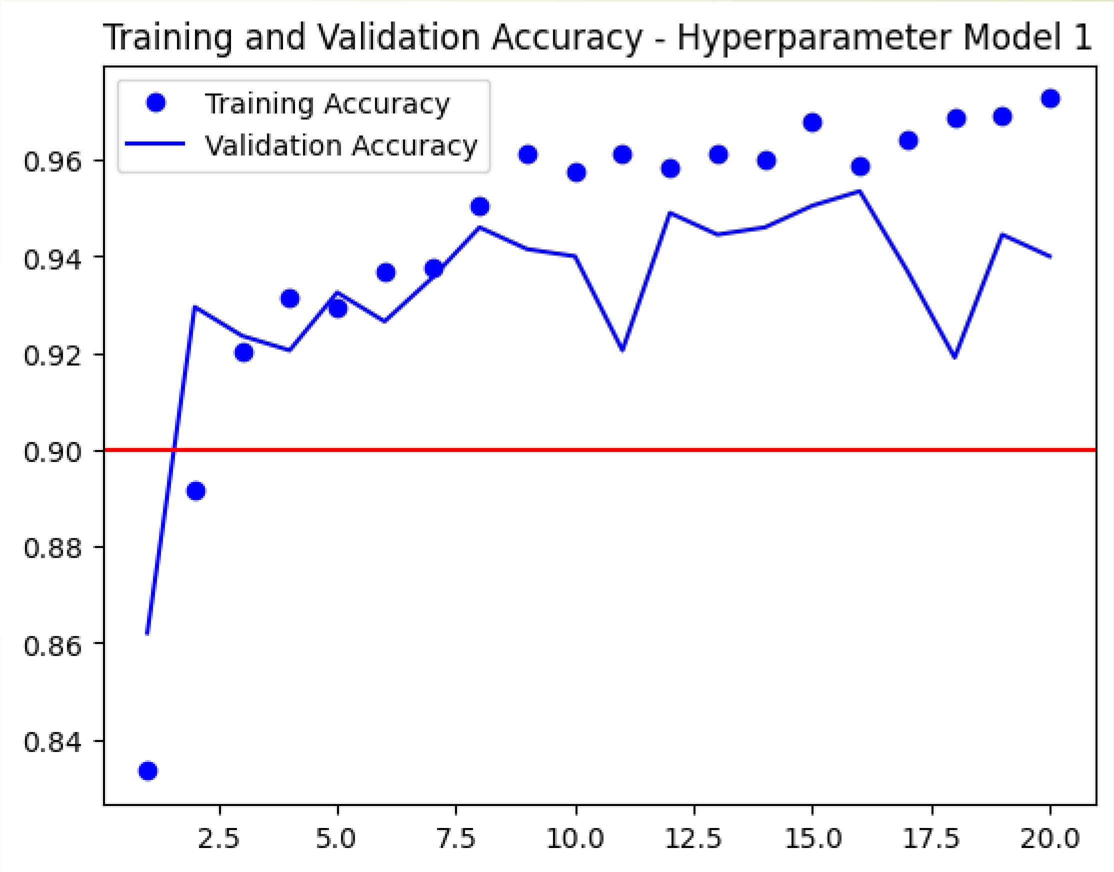
Transfer Learning CNN



Preprocessing and Modeling



Accuracy starts in the low 90s and begins overfitting immediately even with randomly initialized weights in the densely connected layers. (units =256, dropout=0.1)

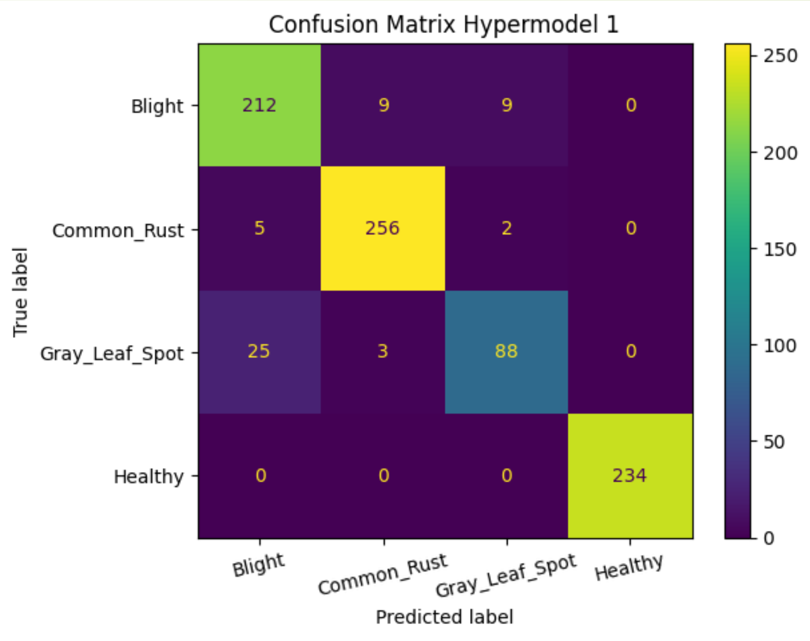


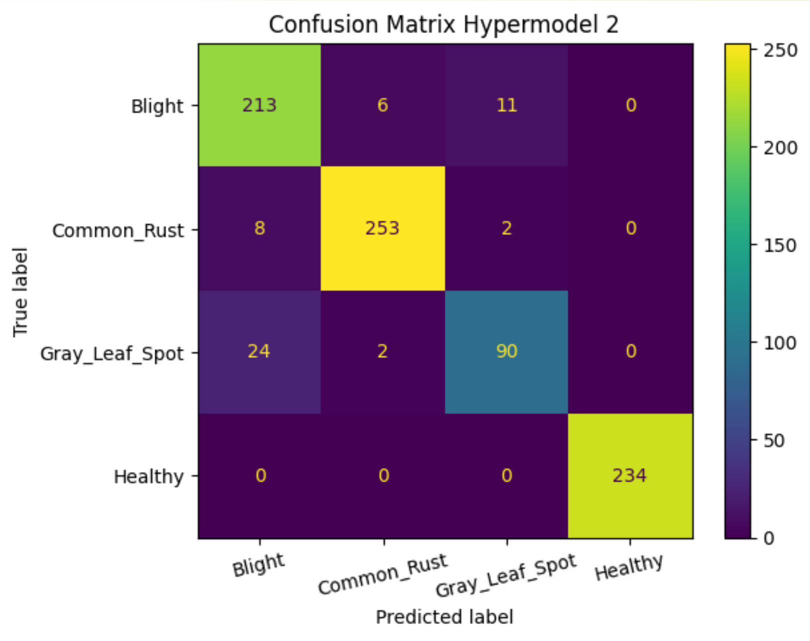
Our best performing hypermodel reaches an accuracy of ~95% at epoch 16 before beginning to overfit. (units =128, dropout=0.5)

Model Selection

Model Results

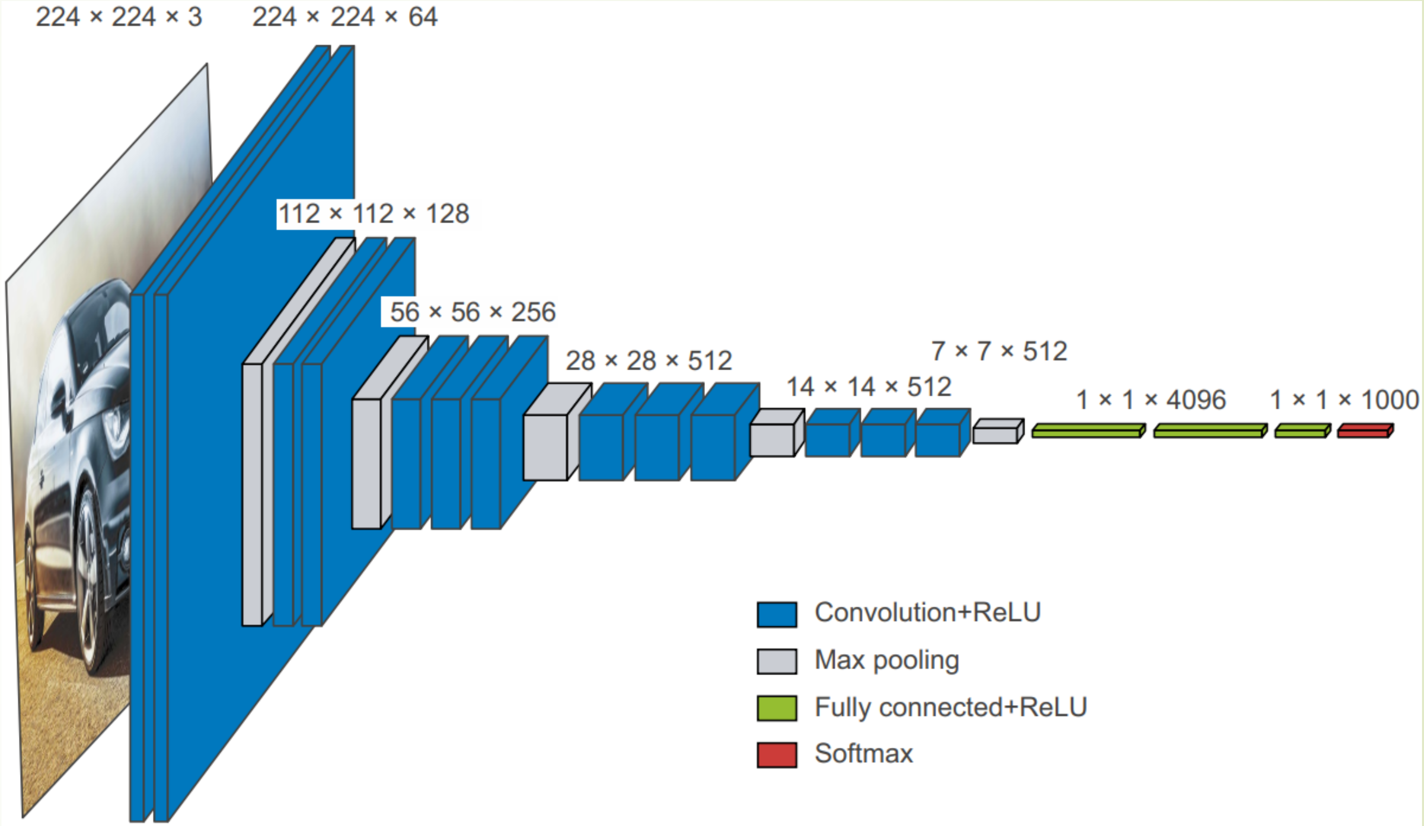
Model Name	f1 Score	Test Accuracy	Precision	Recall	Winner
Base CNN Org Data	0.892	0.891	0.895	0.891	❌
Base CNN Zero-Pad	0.900	0.902	0.902	0.902	❌
Transfer Learning Zero-Pad	0.898	0.900	0.901	0.900	❌
Hypermodel1 Zero-Pad	0.936	0.937	0.937	0.937	tied ✅
Hypermodel2 Zero-Pad	0.937	0.937	0.937	0.937	tied ✅
Hypermodel3 Zero-Pad	0.910	0.906	0.919	0.906	❌
Hypermodel4 Zero-Pad	0.846	0.864	0.879	0.864	❌
Hypermodel5 Zero-Pad	0.924	0.922	0.932	0.922	❌





Conclusions

- Zero-pad data takes longer to overfit but performs better than original data
- Transfer learning reaches high accuracy and overfits very quickly
- Hyperparameter tuning is essential for model performance
- If our classification task was a binary diseased or healthy leaf, our models would have 100% accuracy (as well as precision recall etc.)



Ideas for Future Research

Future Research

Streamline Corn Leaf Imaging

- Satellite imagery or drones
- Image segmentation

Change CNN Model Architecture

- Residual connections
- Batch normalization
- Transfer learning from other

Search A Broader Hyperparameter Space

- Number of densely connected layers
- Number of units per layer
- Optimizers such as 'adam' or 'sgd'

Acquire Better and More Training Data

- More Gray Leaf Spot class instances to balance dataset
- Make sure images are labeled correctly
- Foreground and background segmentation
- Avoid or work around corn leaf occlusion in images

Questions?