



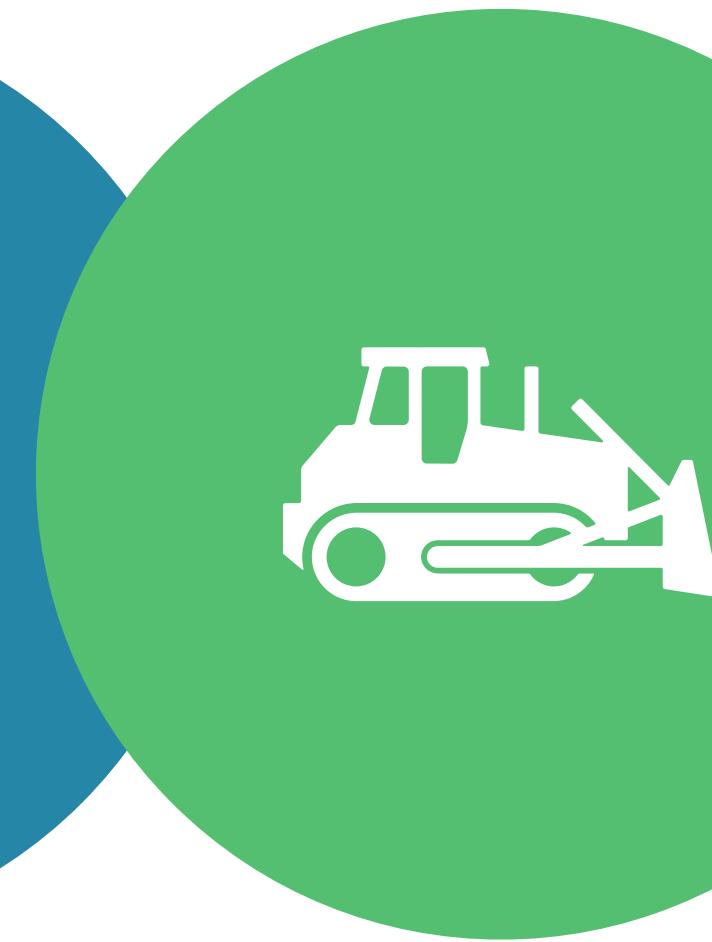
Classifying Images as Chihuahuas or Blueberry Muffins

General Assembly Capstone Project

Presentation Overview



**Problem
Statement**



**Image
Scraping**



**Image
Editing**



**Image
Classification
Neural Net**

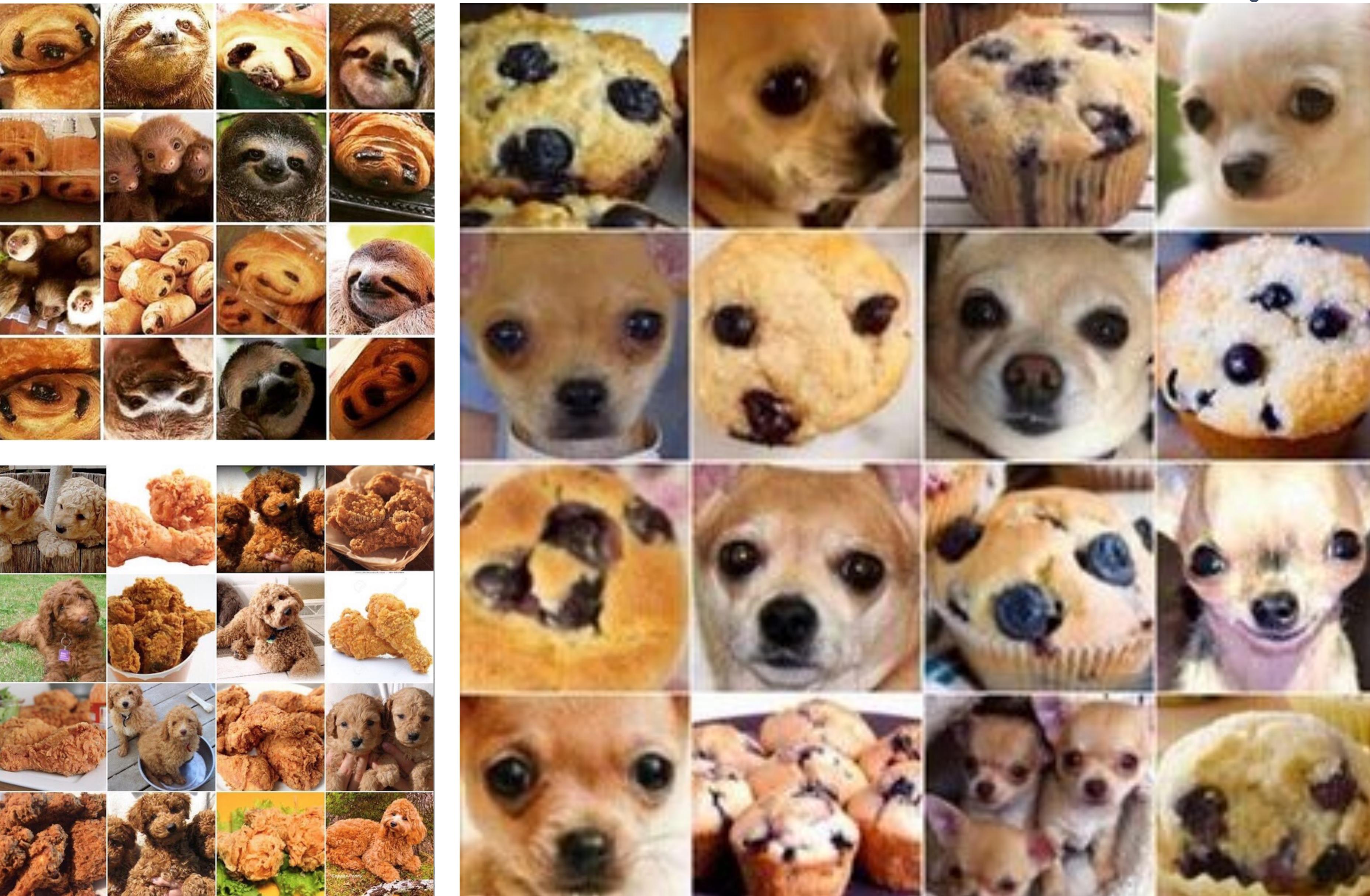


Results

Background: Meme

Problem Statement

Source: Elle Magazine 2016



Background: Image Classifiers

Problem Statement

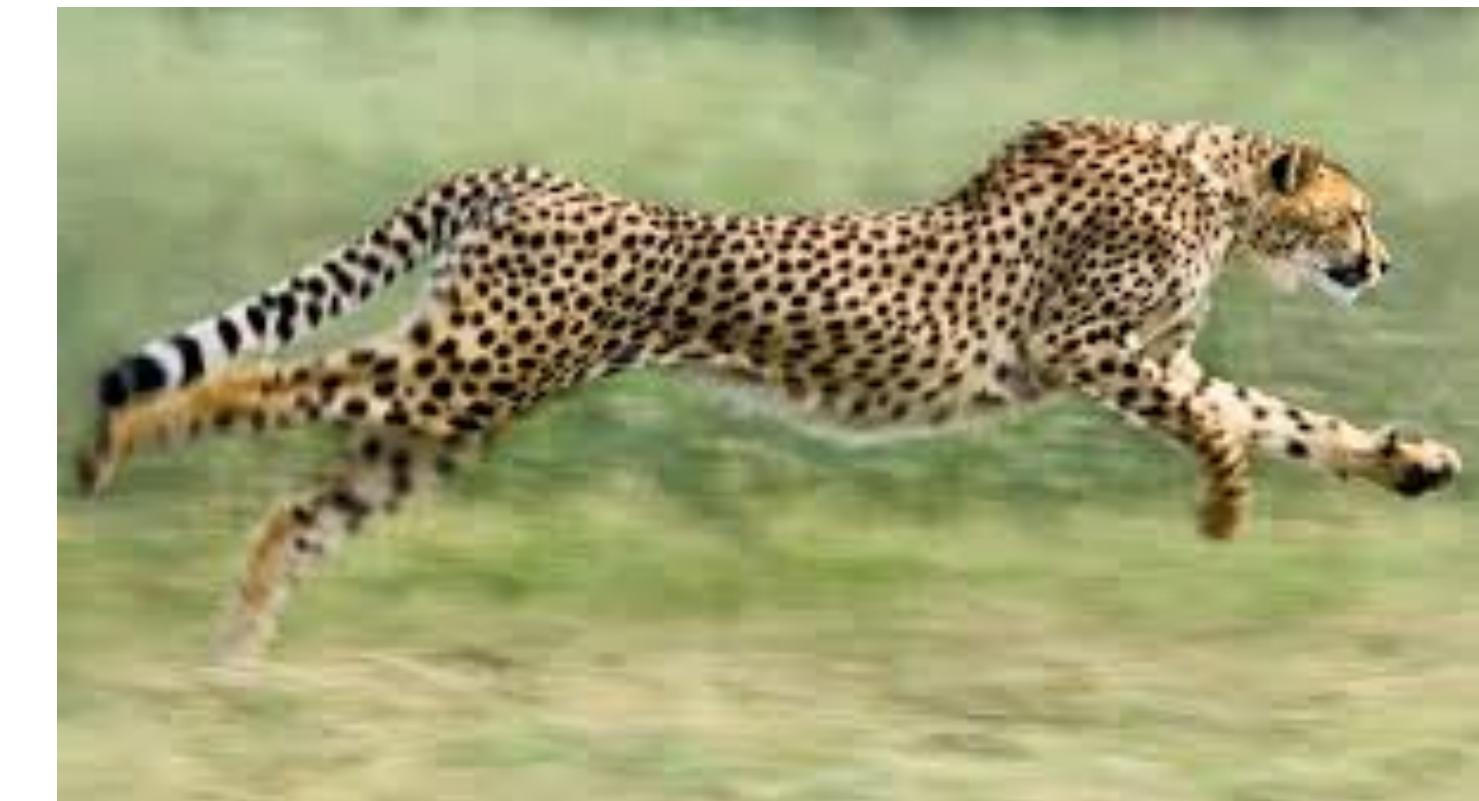


Speed

Humans



Classification Algorithms



Accuracy



Problem Statement

Problem Statement

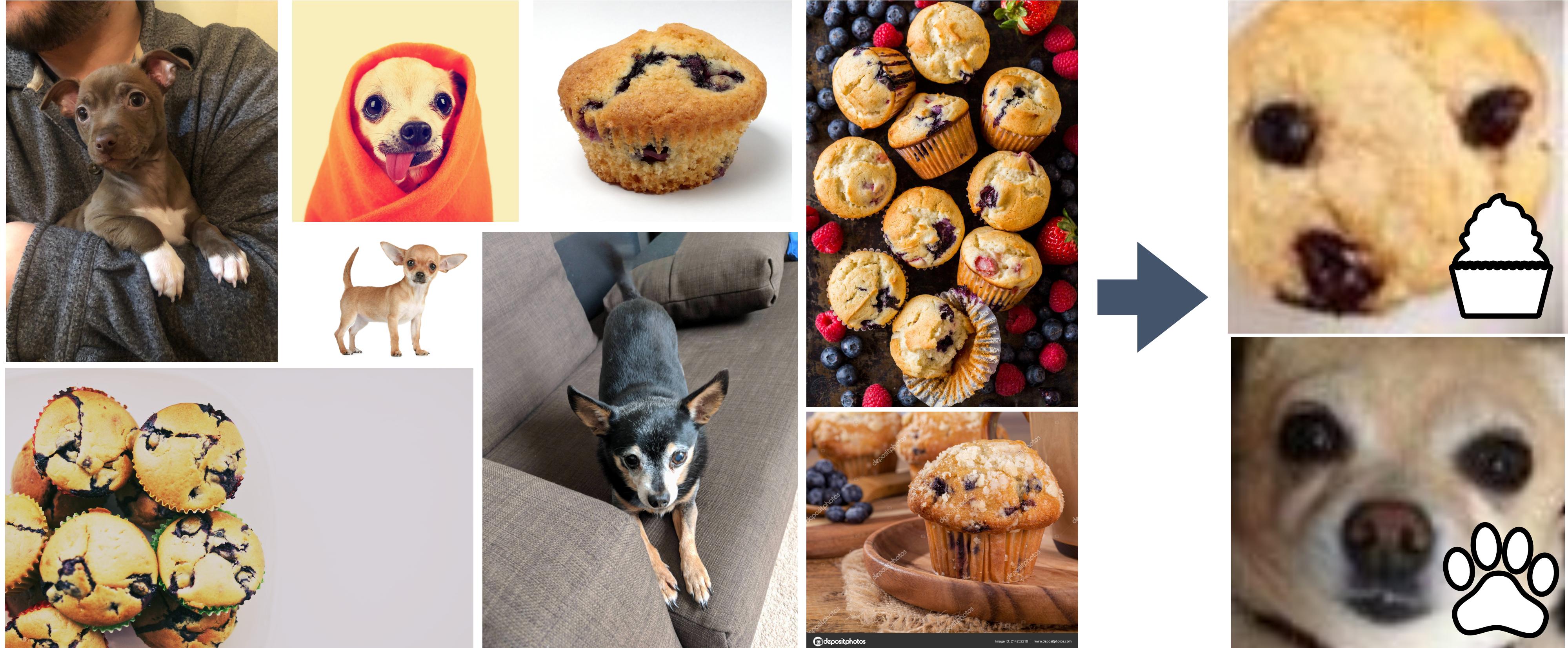
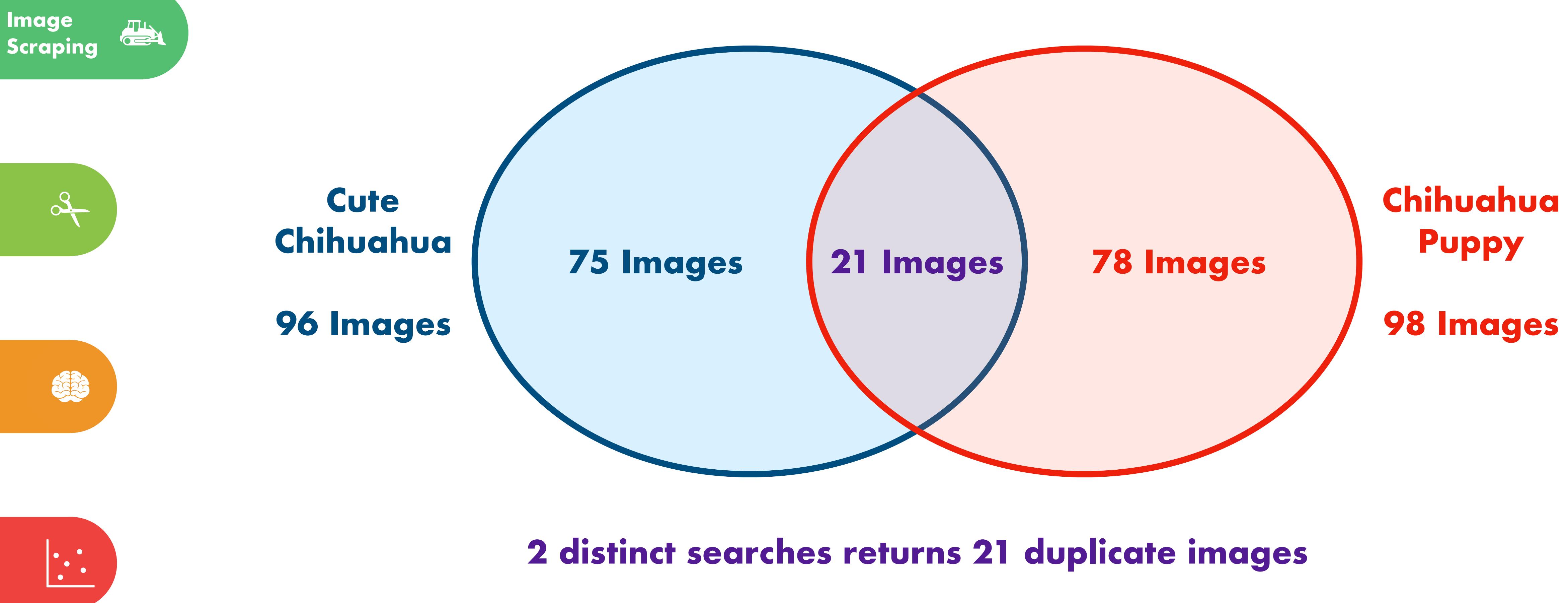


Image Scraping Process

- Google's API limits scrapes to 100 images/search
- Similar keyword searches return duplicate images



Expanding Keyword Searches



- Translating search terms to other languages using the latin alphabet sometimes expanded results

Image
Scraping



Bulgarian
muffin od borovnice



Igbo
acha anụnụ anụnụ muffin



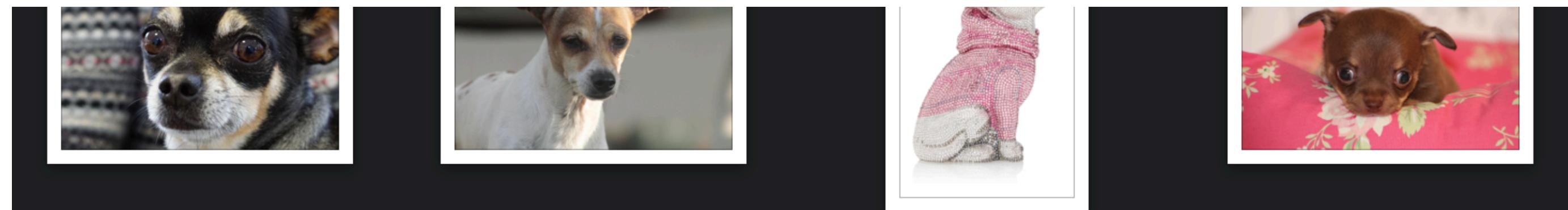
Finnish
mustikkamuffinssi

Remove Incorrect Images



- Manually scrolled through image previews to remove incorrect images

Image Scraping



3.3_cdac7cce7ed1e9f5c76fc54
7f7d84b1c.jpg

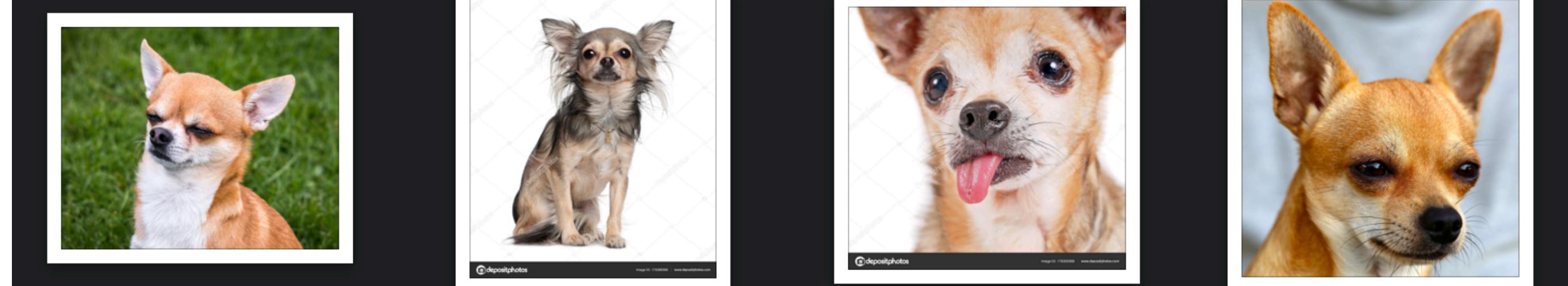
3.7s1vqcrwsa921.jpg

3.6943cb65c2c5be33ce9b127
2353bbc47.jpg

3.731435ce34774e9a2713c54
a812eca11.jpg



Chihuahua, MX

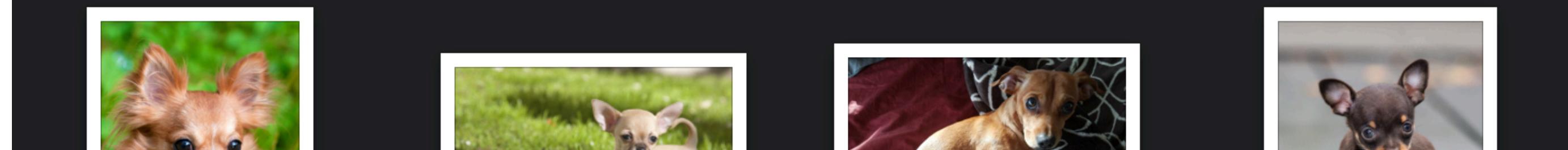


3.chihuahua-dog-chiwawa-
small-wallpaper.jpg

3.depositphotos_174389368-
stock-photo-lo...ua-2-years.jpg

3.depositphotos_179305568-
stock-photo-g...-his-tongue.jpg

3.e5ec4cfdb417e2fc063a1e86
85d021e2.jpg



1,905 items, 21.46 GB available



Image Scaling and Cropping



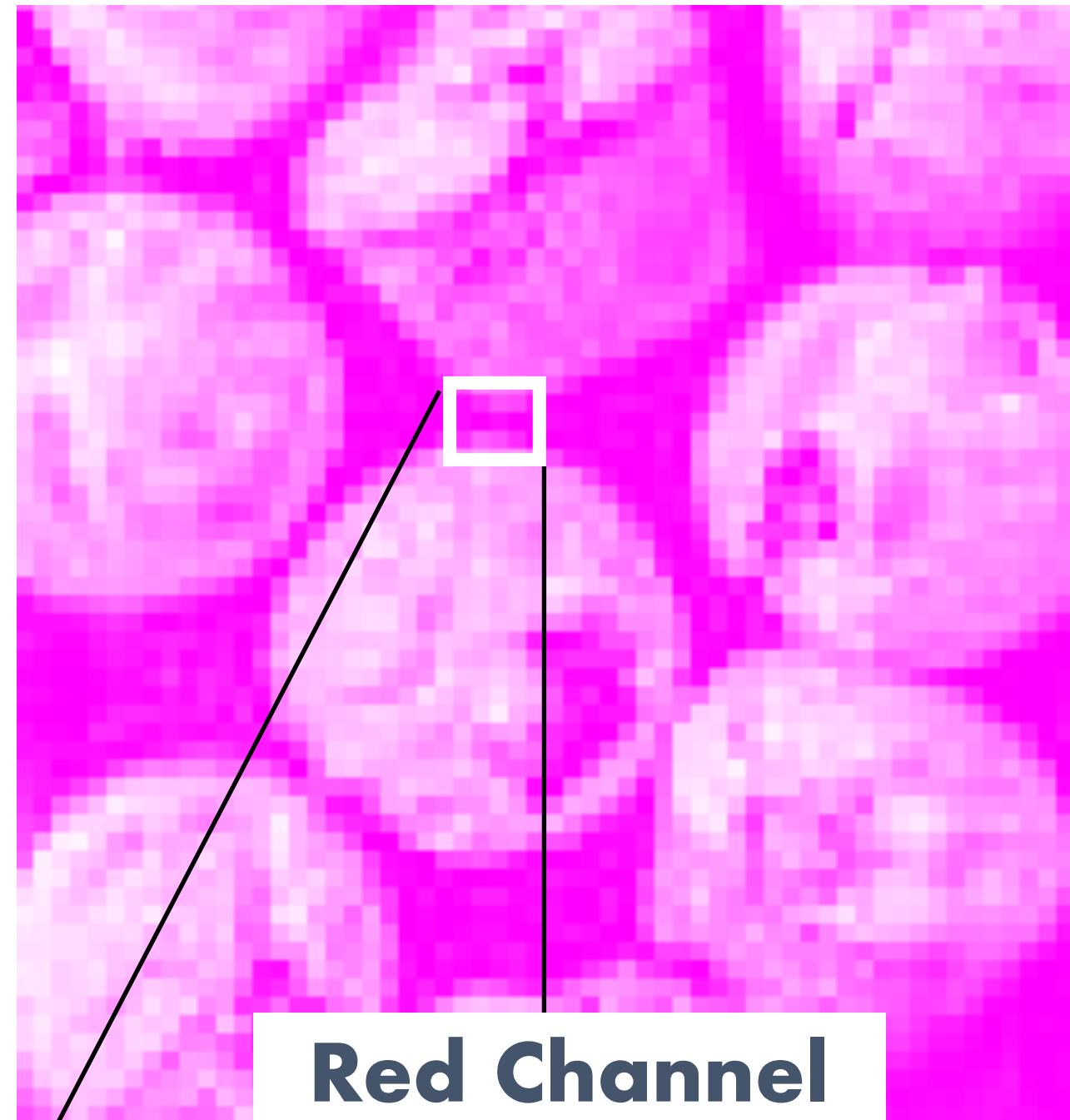
Image
Editing



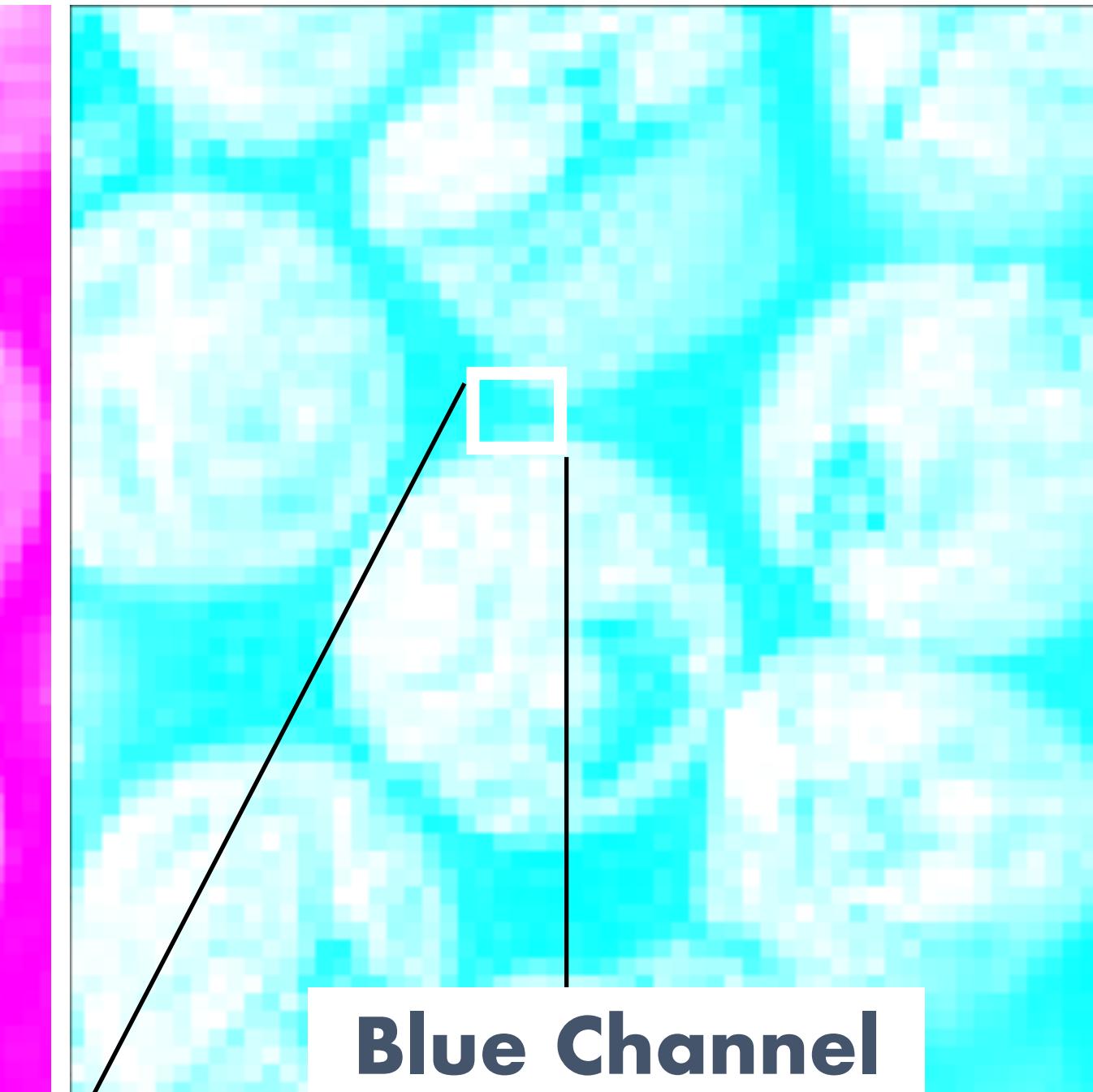
Converting to Numbers



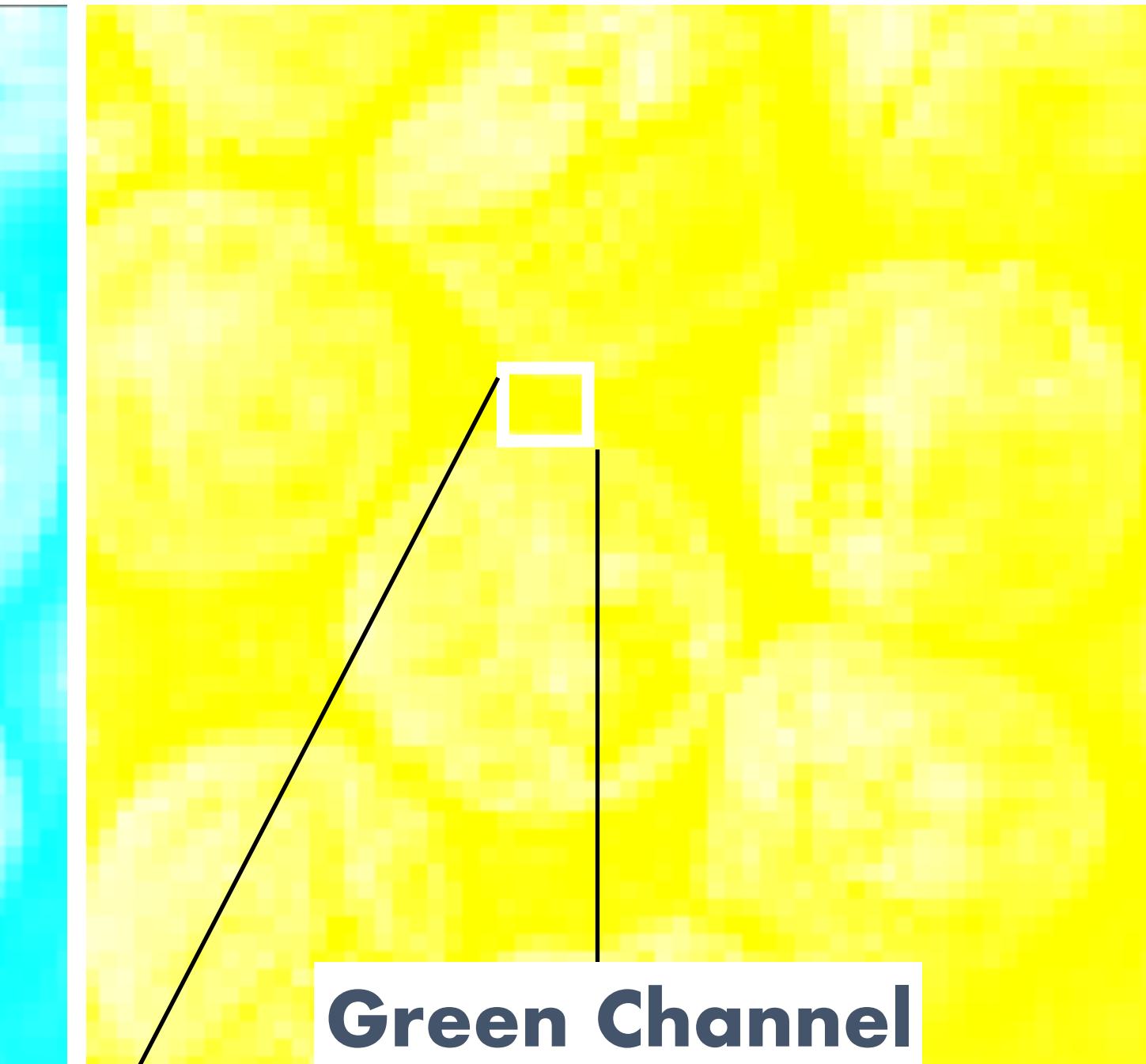
Image
Editing



0	0.3	0.5	0.7	0.8	0.6
1	0.3	0.3	0.4	0.5	0.4
2	0.2	0.2	0.3	0.3	0.2
3	0.4	0.4	0.6	0.5	0.4
4	0.9	0.8	0.9	0.8	0.8



0	0	0.2	0.4	0.5	0.3
1	0	0	0	0.2	0.1
2	0	0	0	0	0
3	0	0	0.1	0.1	0
4	0.4	0.3	0.4	0.4	0.4

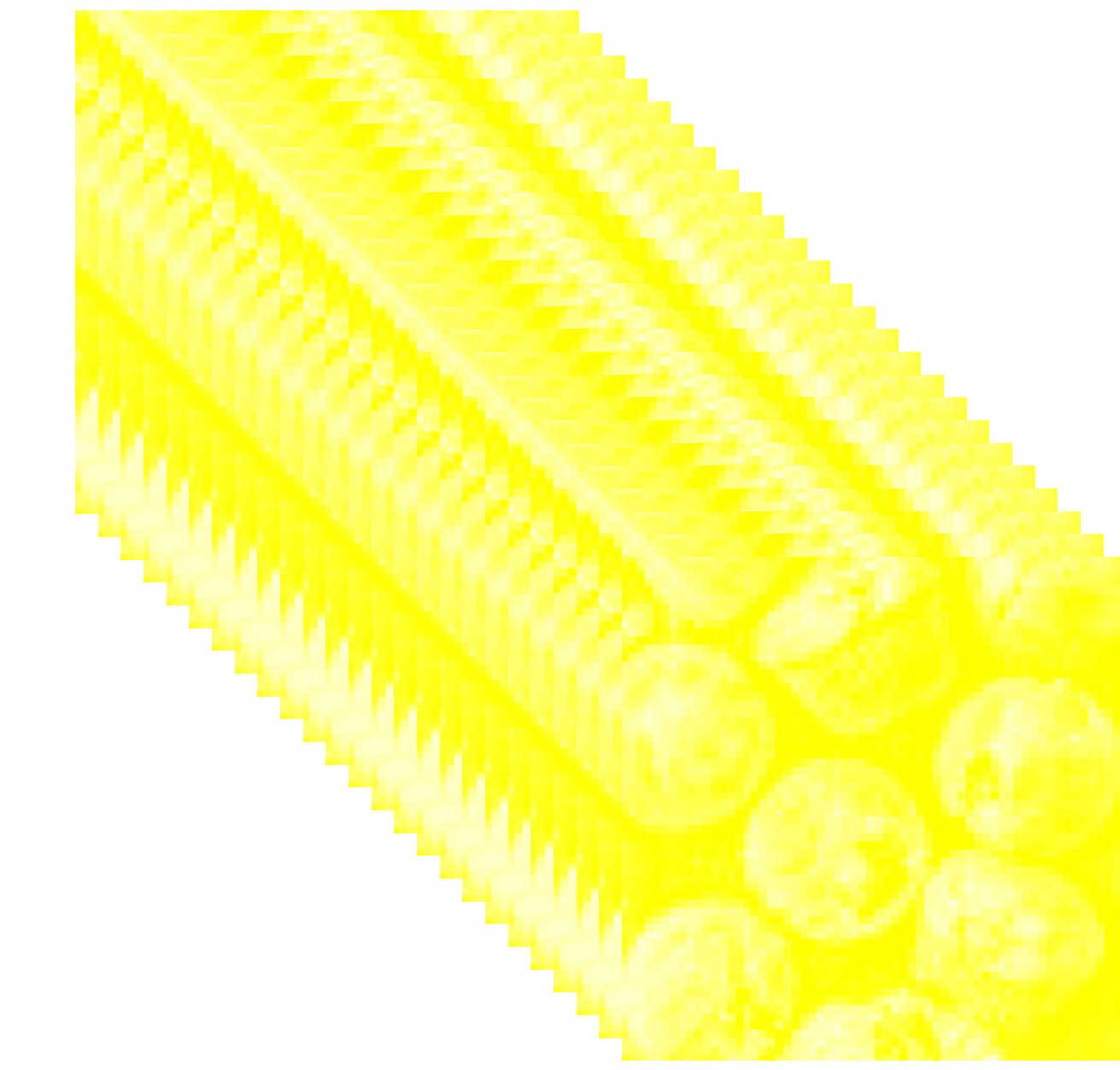
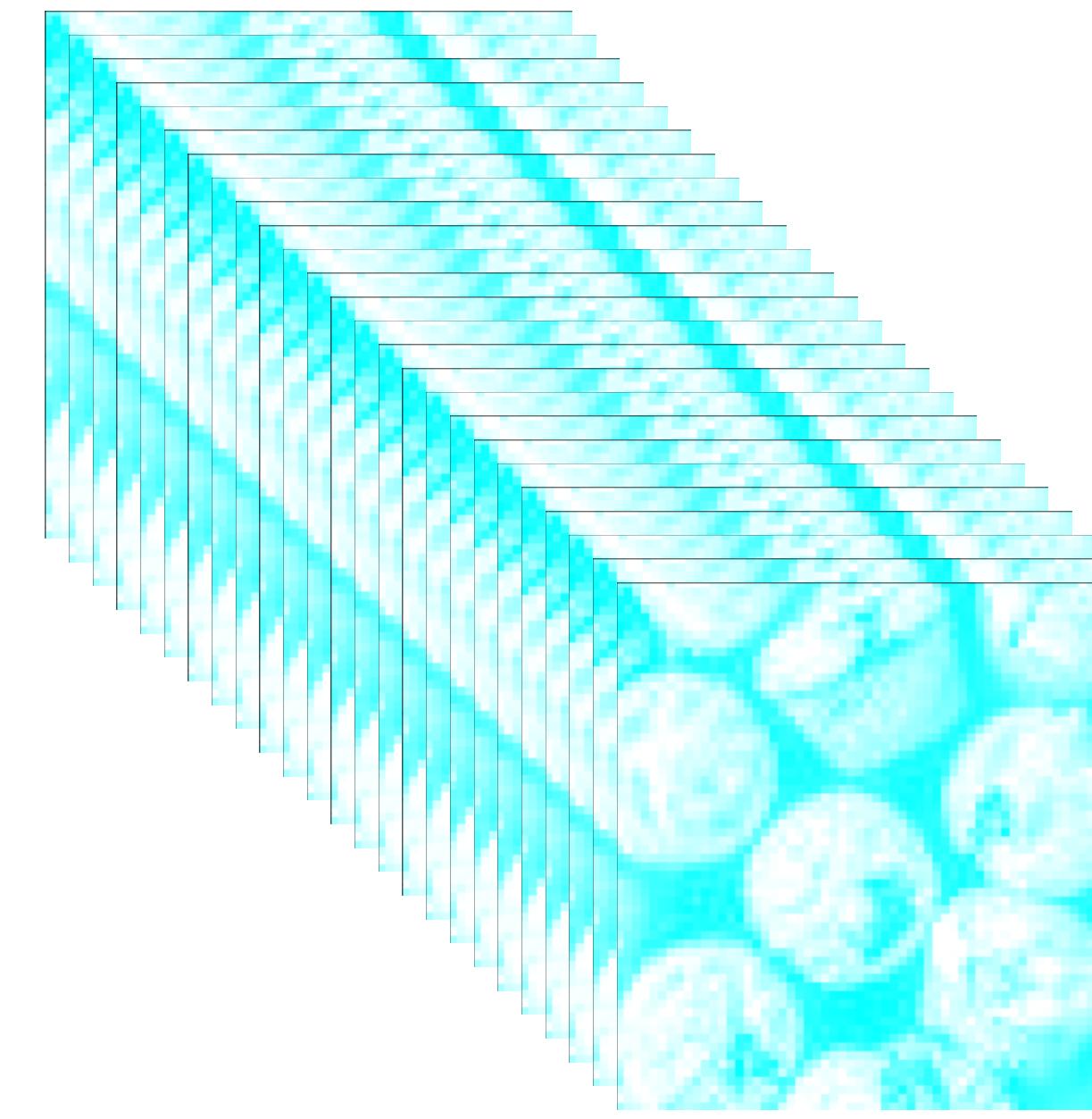
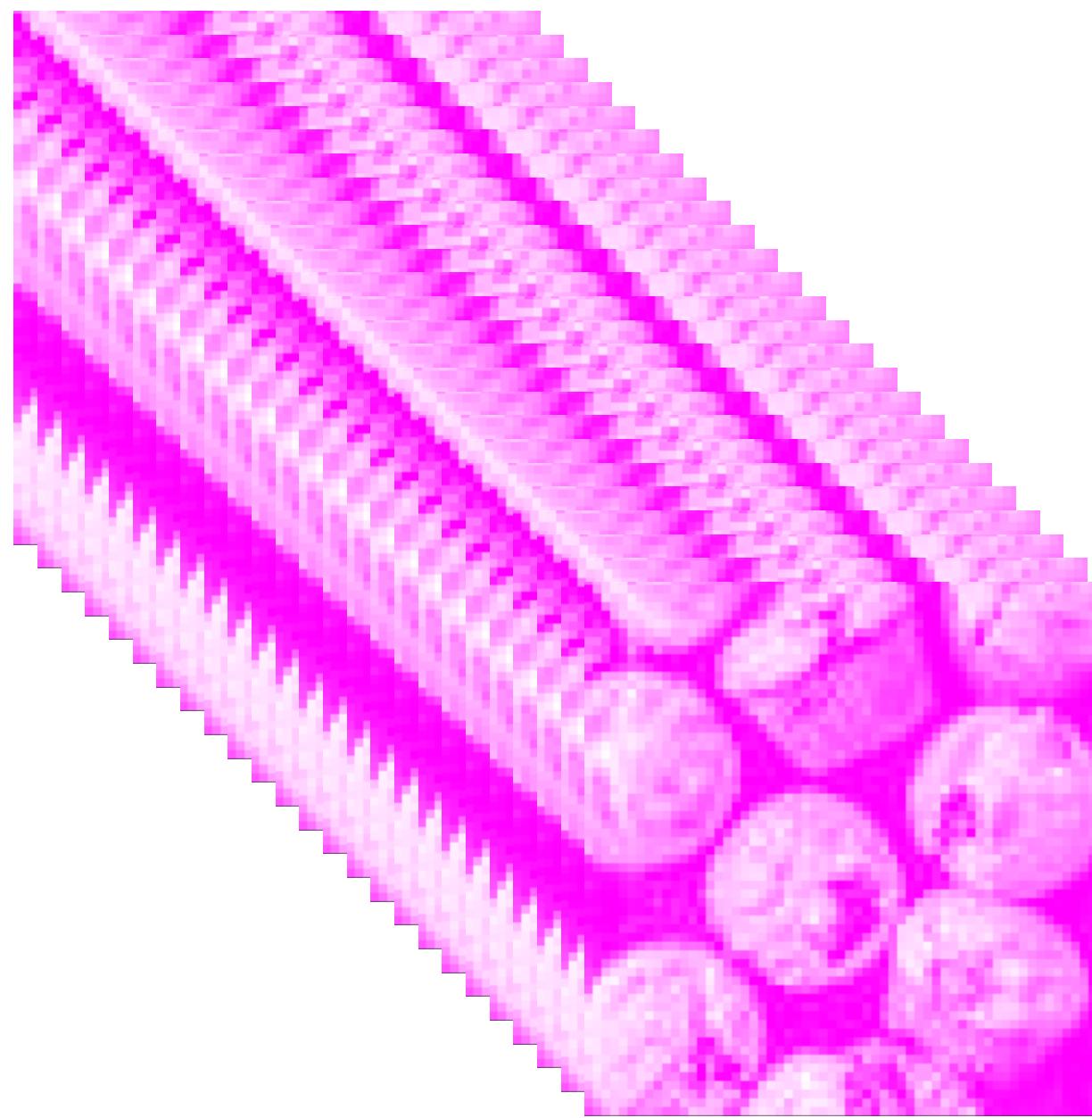


0	0.1	0.3	0.5	0.5	0.4
1	0	0.1	0.2	0.3	0.2
2	0	0	0.1	0.1	0
3	0.2	0.2	0.3	0.3	0.1
4	0.7	0.6	0.6	0.6	0.6

Saving to Array



Image
Editing



(3906, 150, 150, 3)

3906 samples, 150 rows, 150 columns, 3 colors

263,655,000 pixels

Neural Net Layers



Image
Classification
Neural Net

- Convolutional Layer
 - Pooling Layer
 - Flatten Layer
 - Dense Layer
 - Dropout Layer
-
- Image Data Generator
 - Transfer Learning

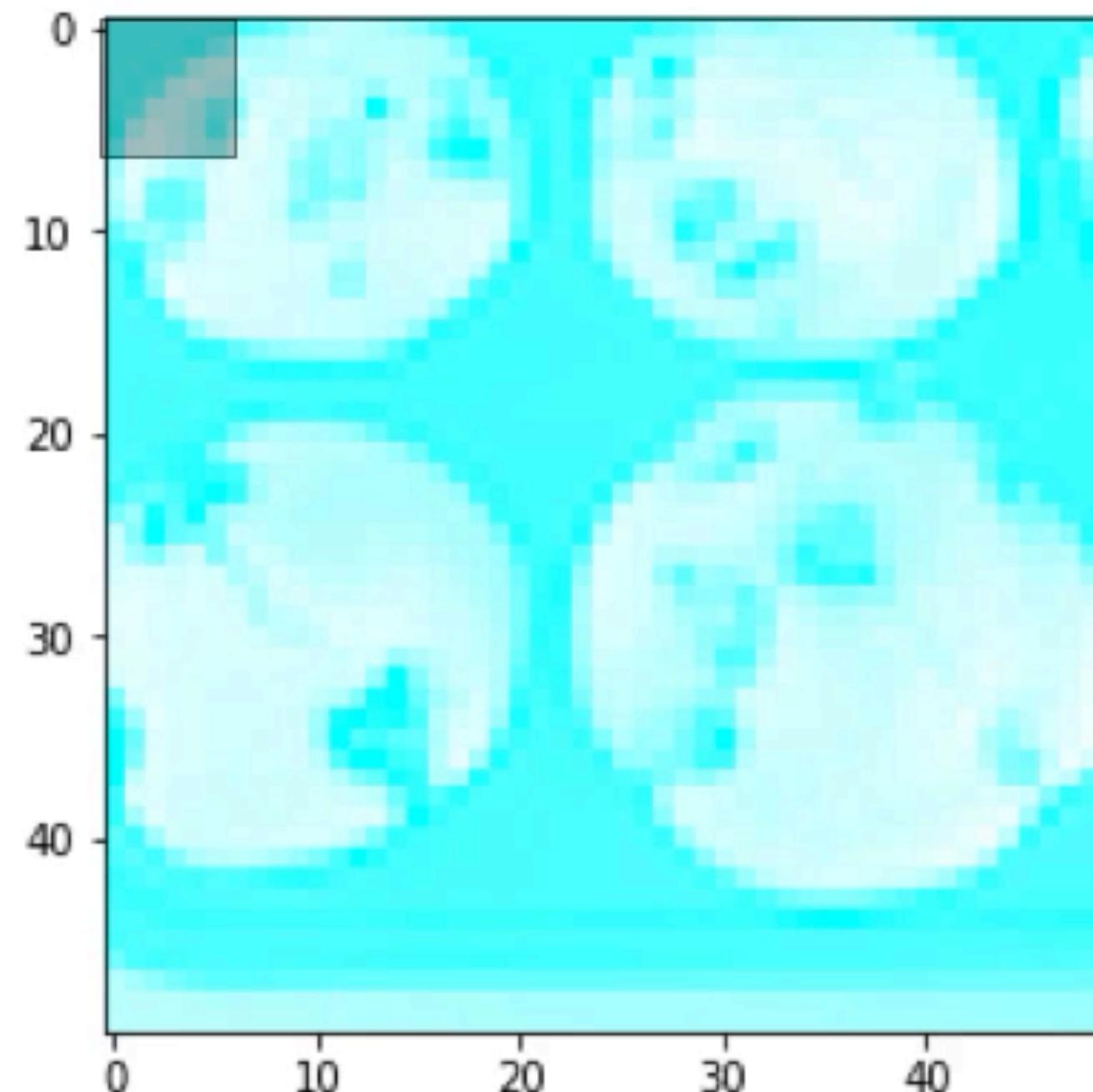


Convolutional Layer



Image
Classification
Neural Net

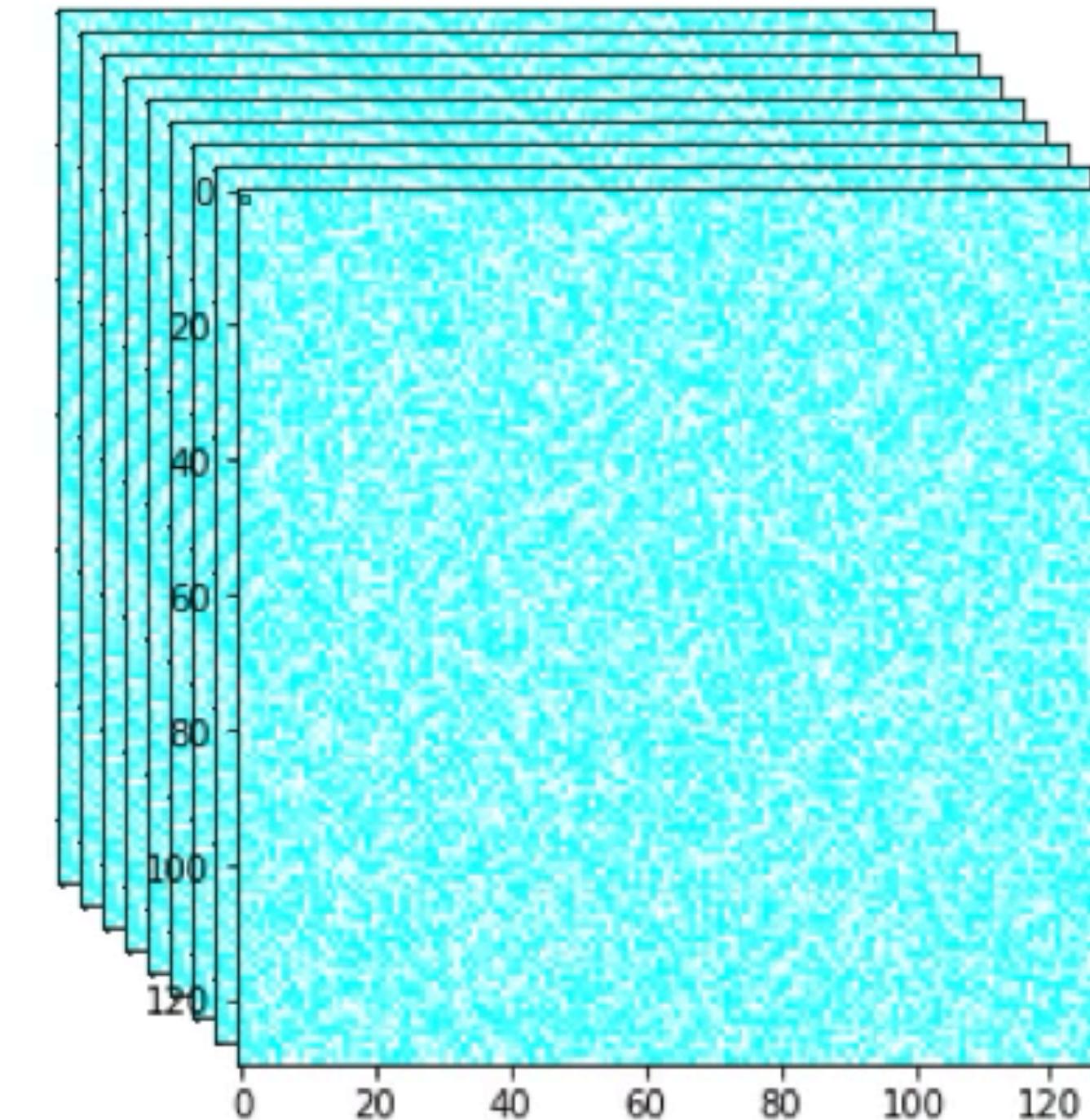
Input



Kernel size = 20x20
Number of Filters = 9



Output



(1, 150, 150, 1)
1 sample, 150 rows, 150
columns, 1 color

(1, 130, 130, 40)
1 sample, 130 rows, 130
columns, 9 filters

Repeat for all colors and images in batch

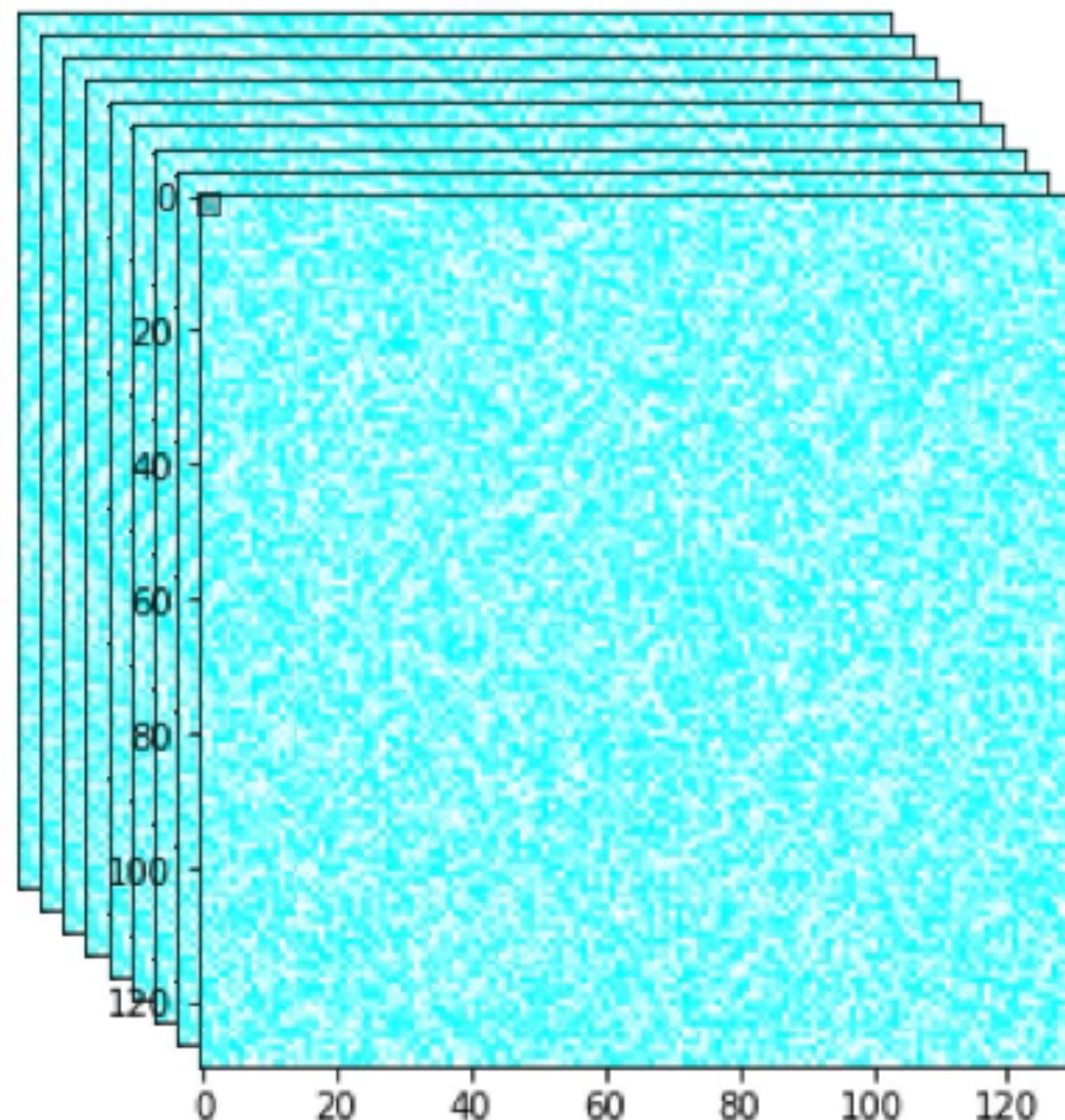
Pooling Layer



Image
Classification
Neural Net



Input



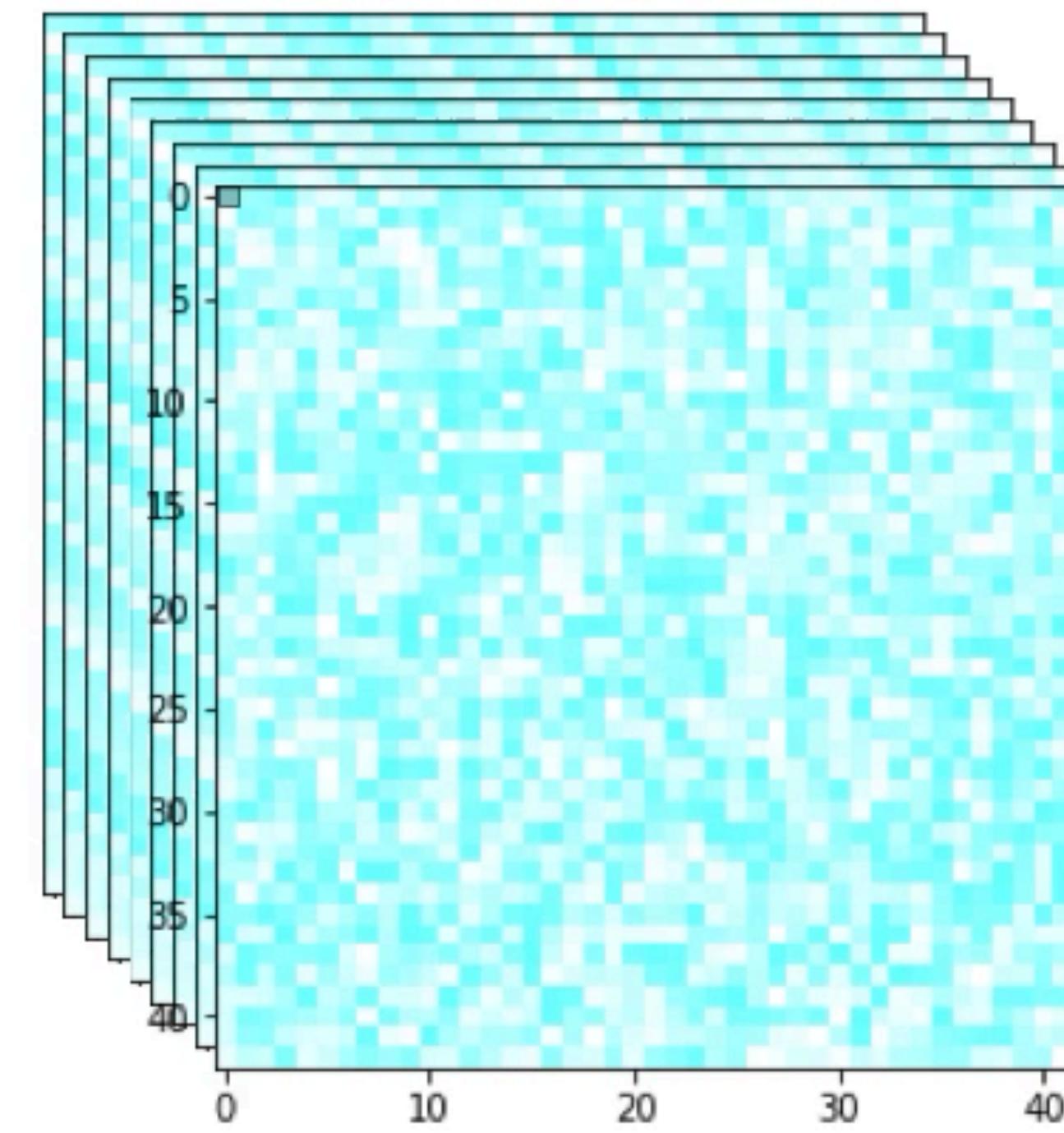
$(1, 130, 130, 40)$

1 sample, 130 rows, 130
columns, 40 filters

Type = MaxPooling
Pool Size = 3



Output



$(1, 43, 43, 40)$

1 sample, 43 rows, 43
columns, 40 filters

Repeat for all colors and images in batch
(Additional model layers not shown)

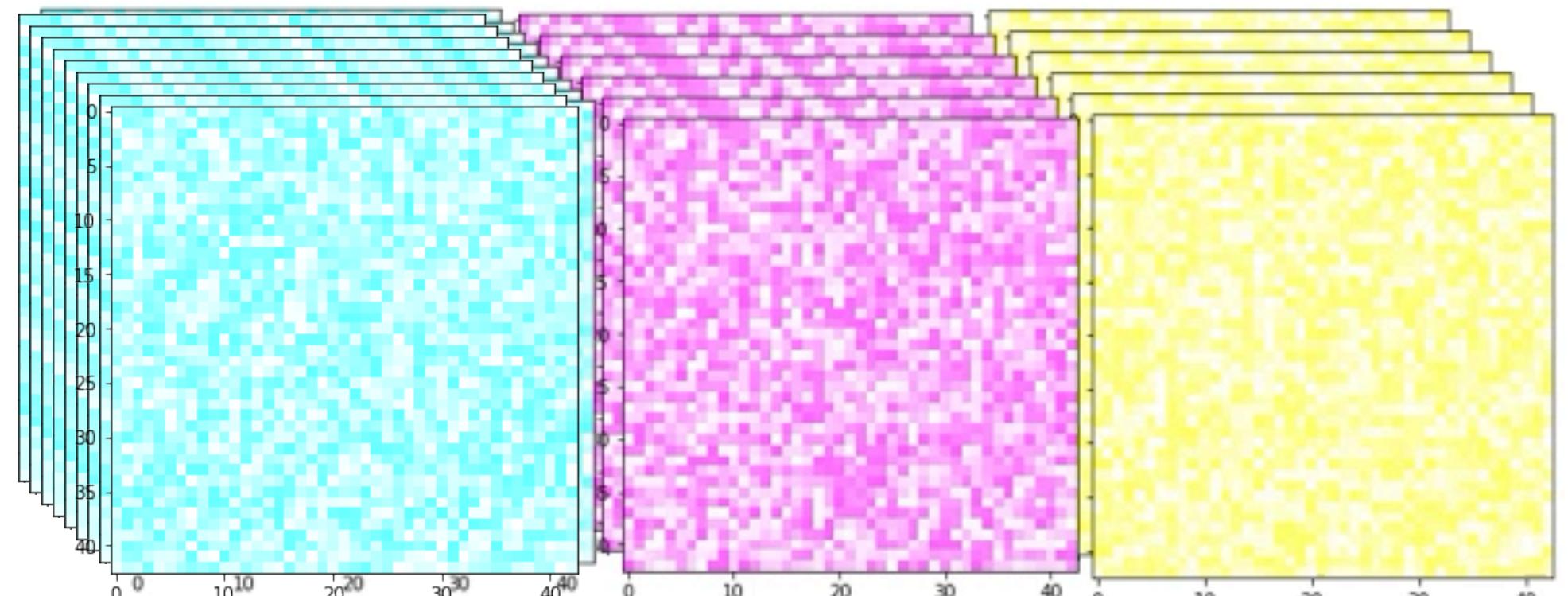
Flatten Layer



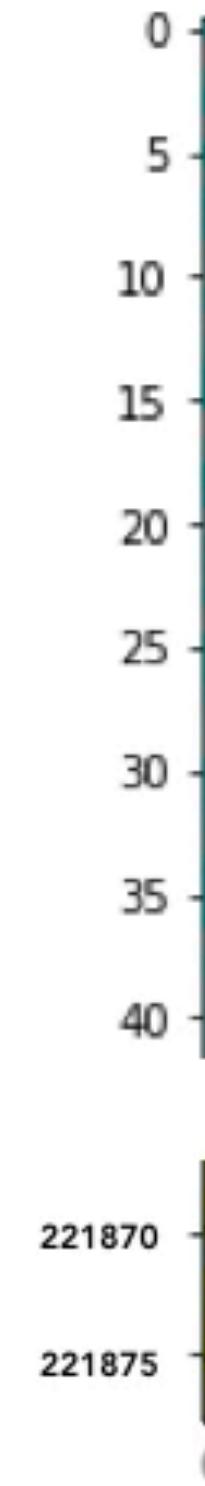
Image
Classification
Neural Net



Input



Output



(1, 43, 43, 120)
1 sample, 43 rows,
43 columns,
40 filters by 3 colors

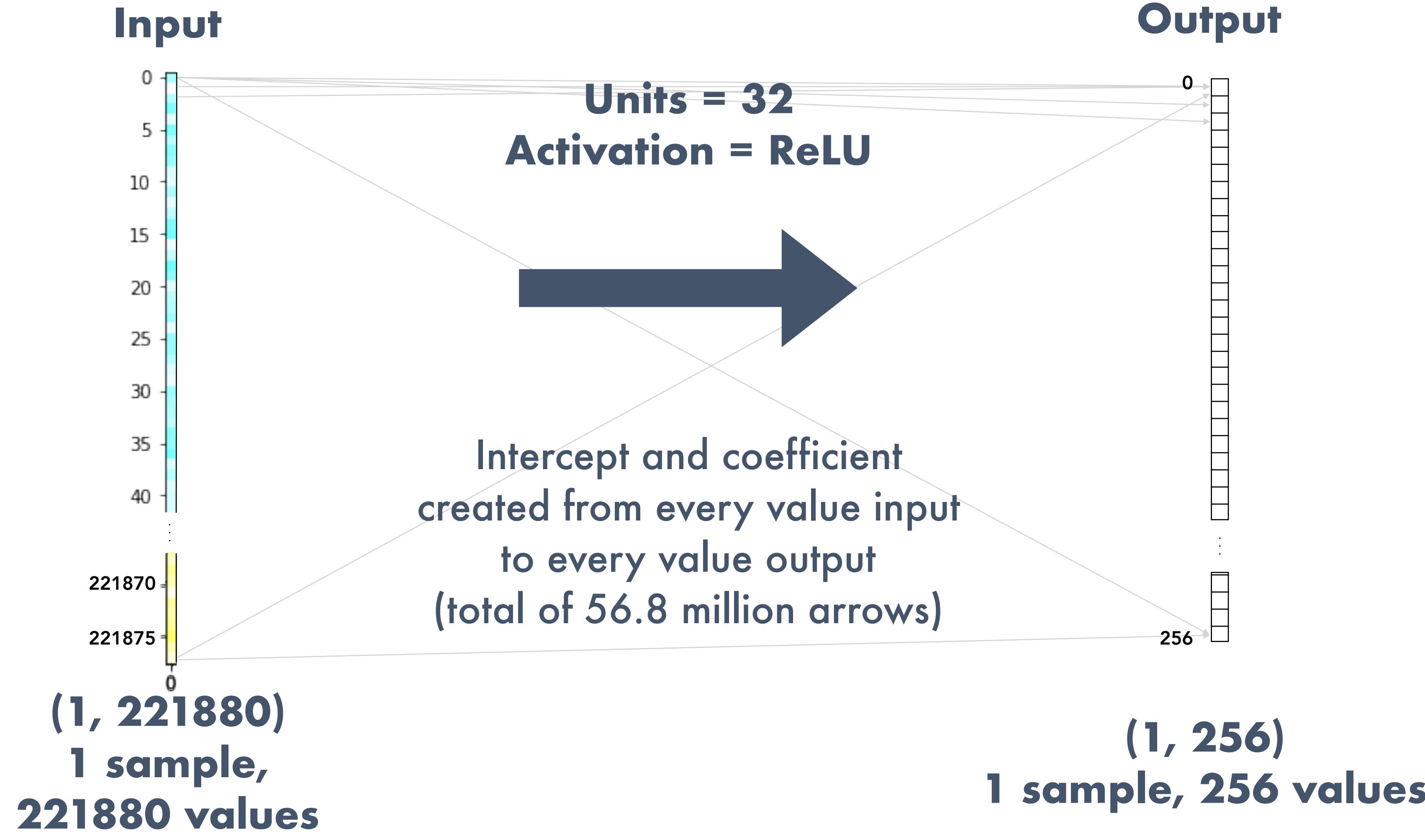
(1, 221880)
1 sample,
221,880 values

Repeat for all images in batch

Dense Layer



Image
Classification
Neural Net



Repeat for all images in batch
(Additional model layers not shown)

Final Layer



Image
Classification
Neural Net



Input

0
:
32

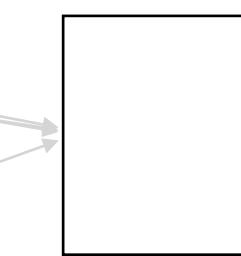
(1, 32)
1 sample,
32 values

Units = 1
Activation = Sigmoid



Intercept and coefficient
created from every value input
to every value output
(total of 32 arrows)

Output



(1, 1)
1 sample,
1 value b/t 0 and 1

≥ 0.5 indicates
Chihuahua
 < 0.5 indicates
Blueberry
Muffin

Repeat for all images in batch

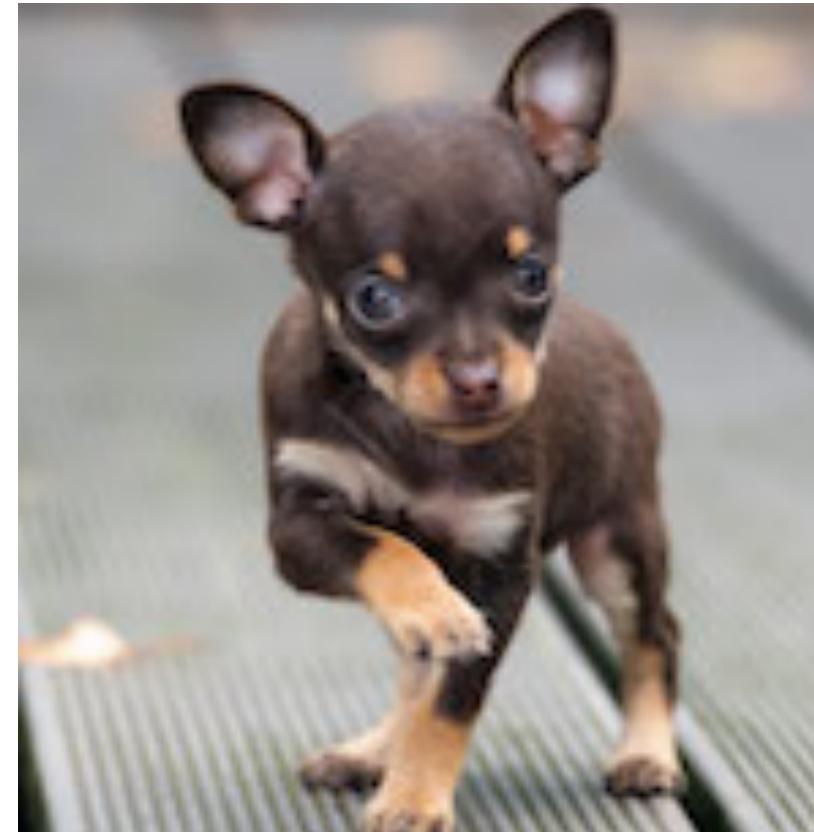
Image Data Generator



- Applies random slight image augmentation to images prior to modeling to increase training set



Input



**Rotation Range = 35
Width Shift = 0.1
Height Shift = 0.1
Brightness Range = .2
Shear Range = 0
Horizontal Flip = True
Vertical Flip = False
Fill Mode = Nearest**



Outputs

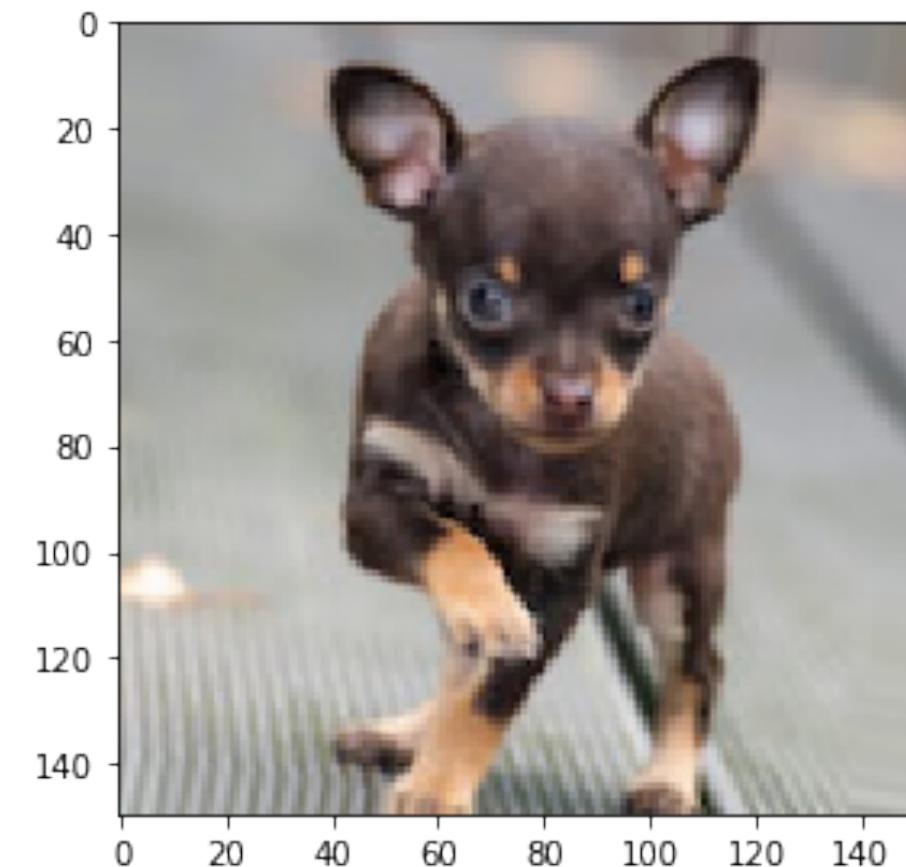
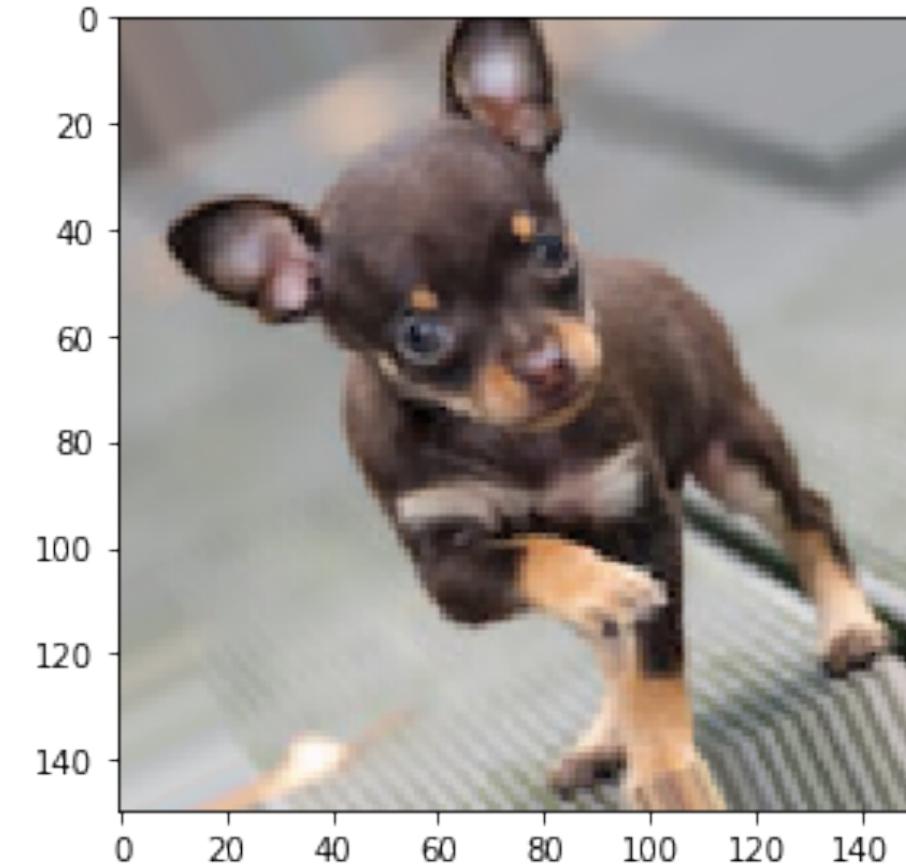
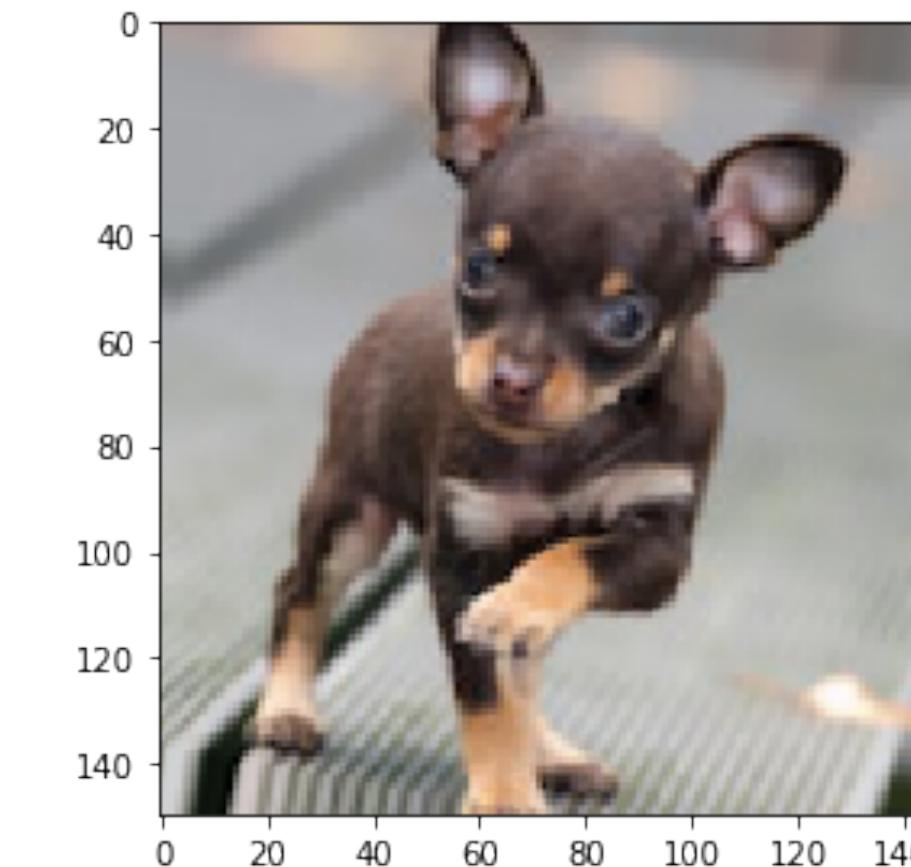
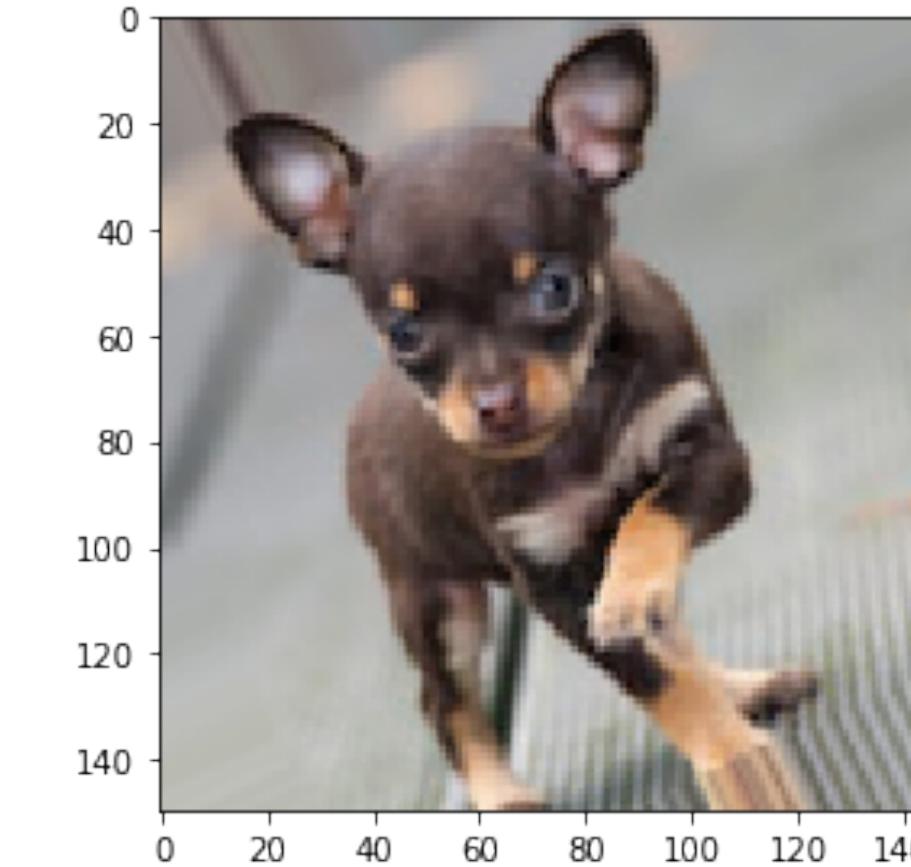


Image
Classification
Neural Net



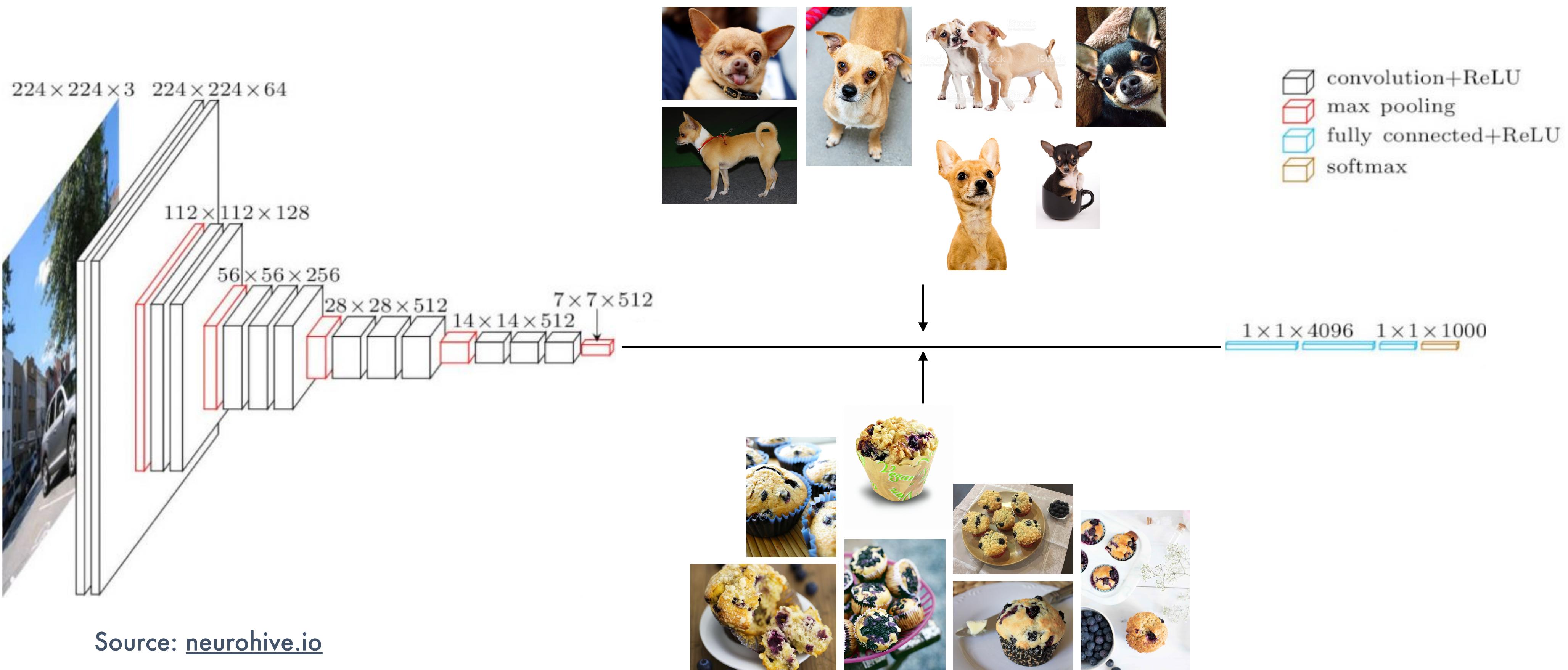
Transfer Learning



- VGG16 was trained on over 14 million images belonging to 1000 classes
- Last layers were retrained on my image set



Image
Classification
Neural Net



Blueberry Muffin Accuracy



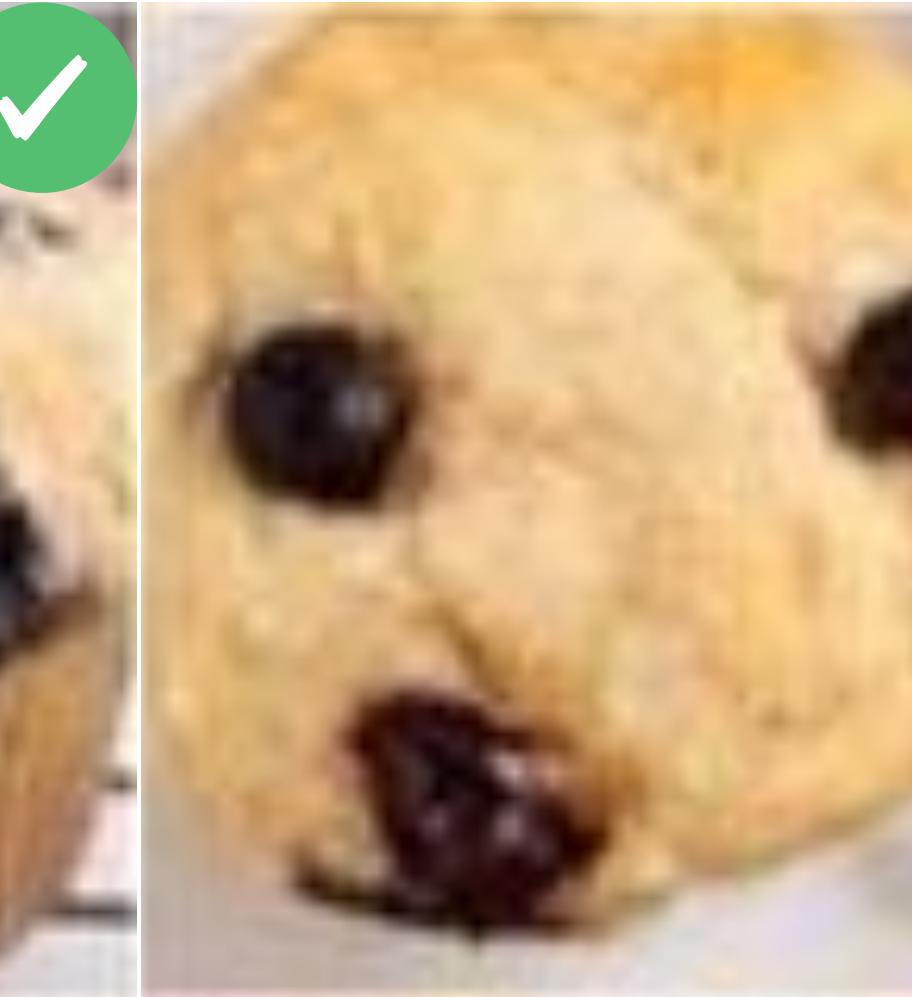
Results



99% Muffin



99% Muffin



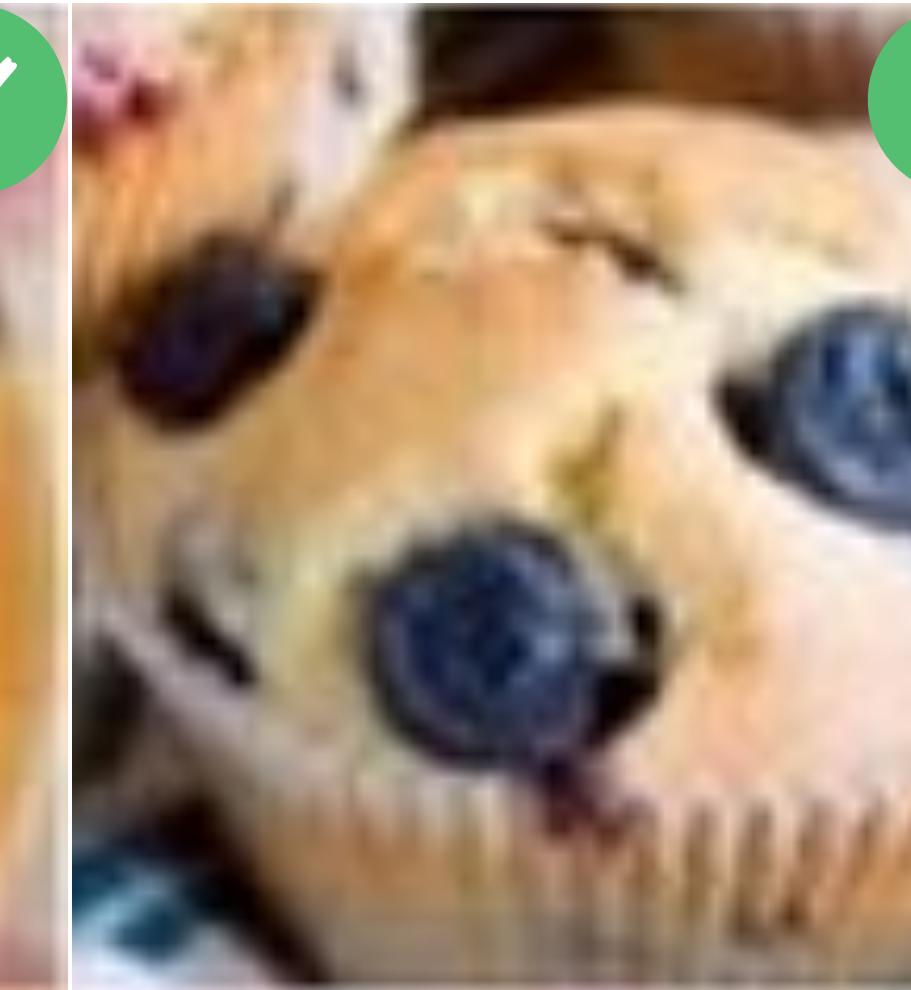
93% Muffin



99% Muffin



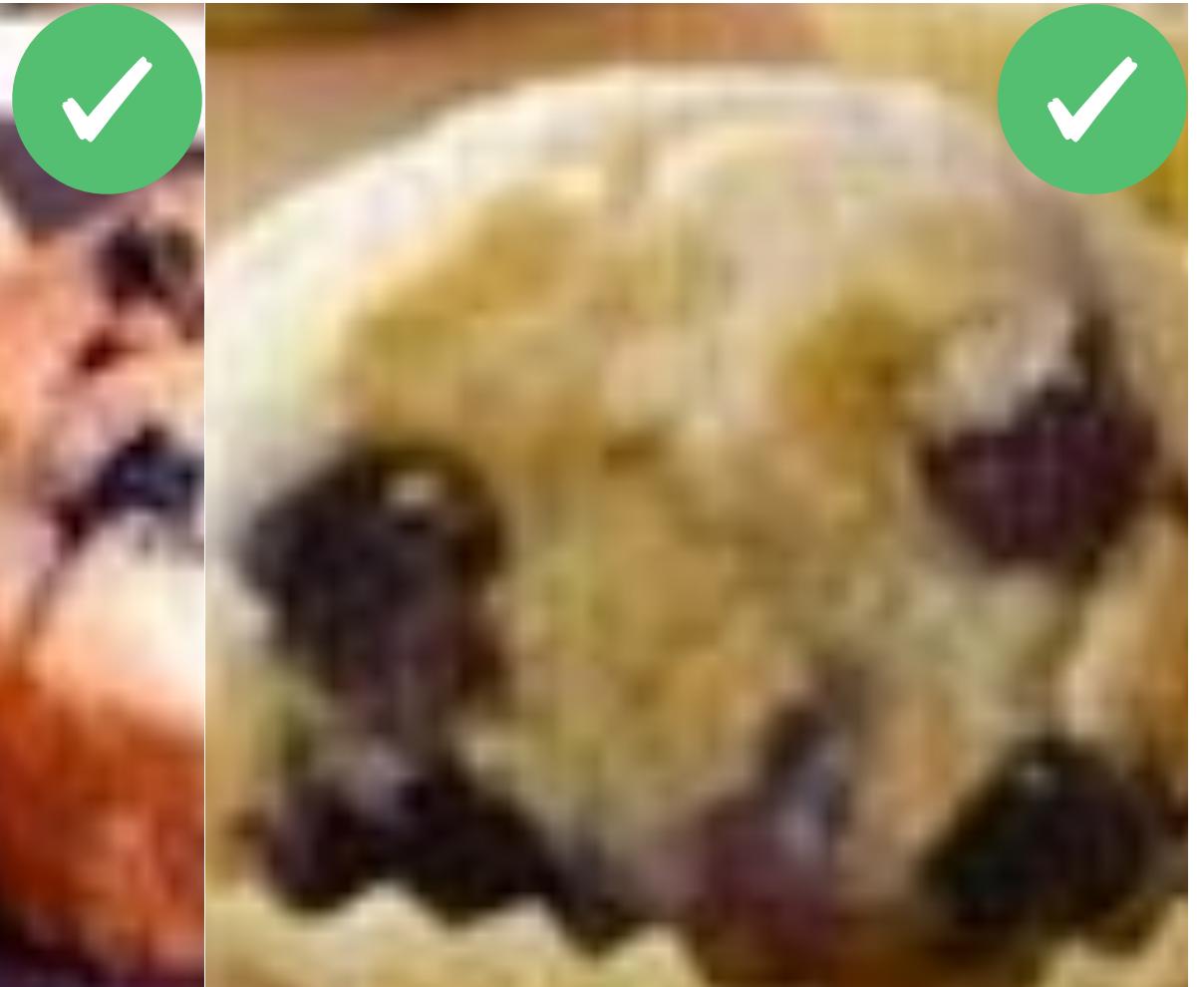
92% Muffin



99% Muffin



99% Muffin

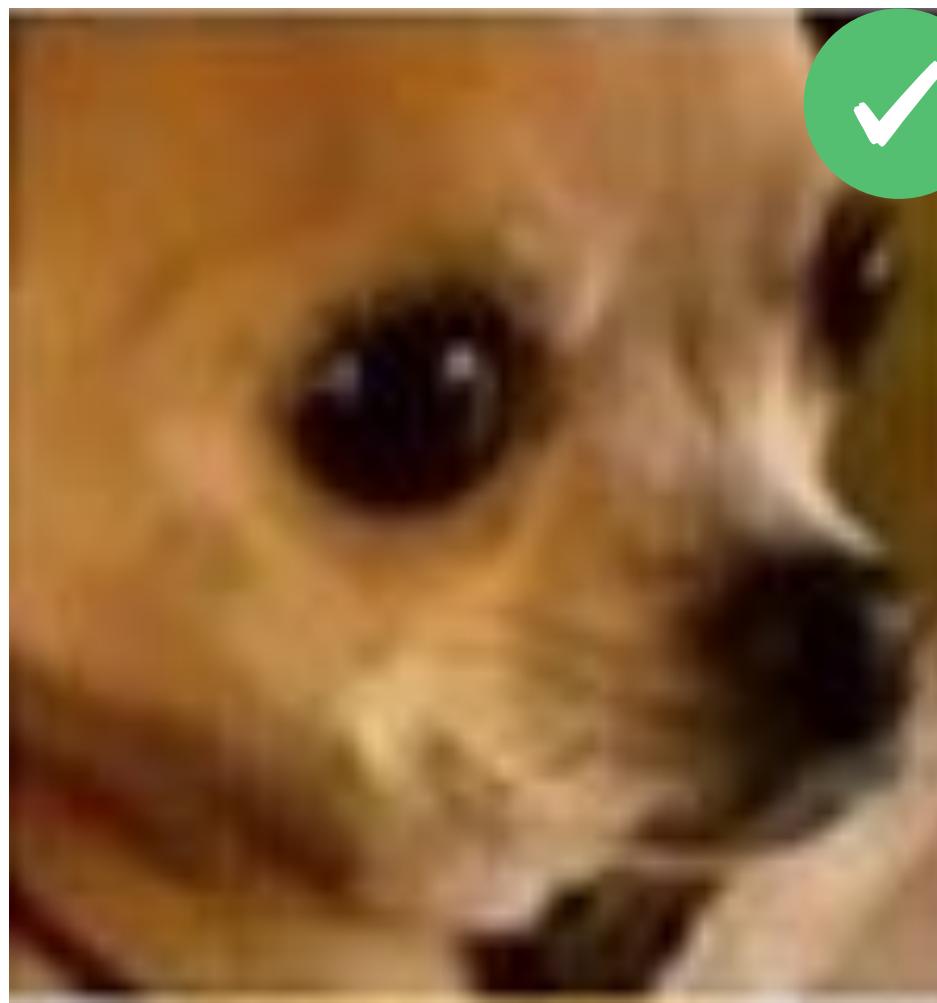
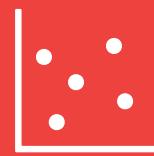


96% Muffin

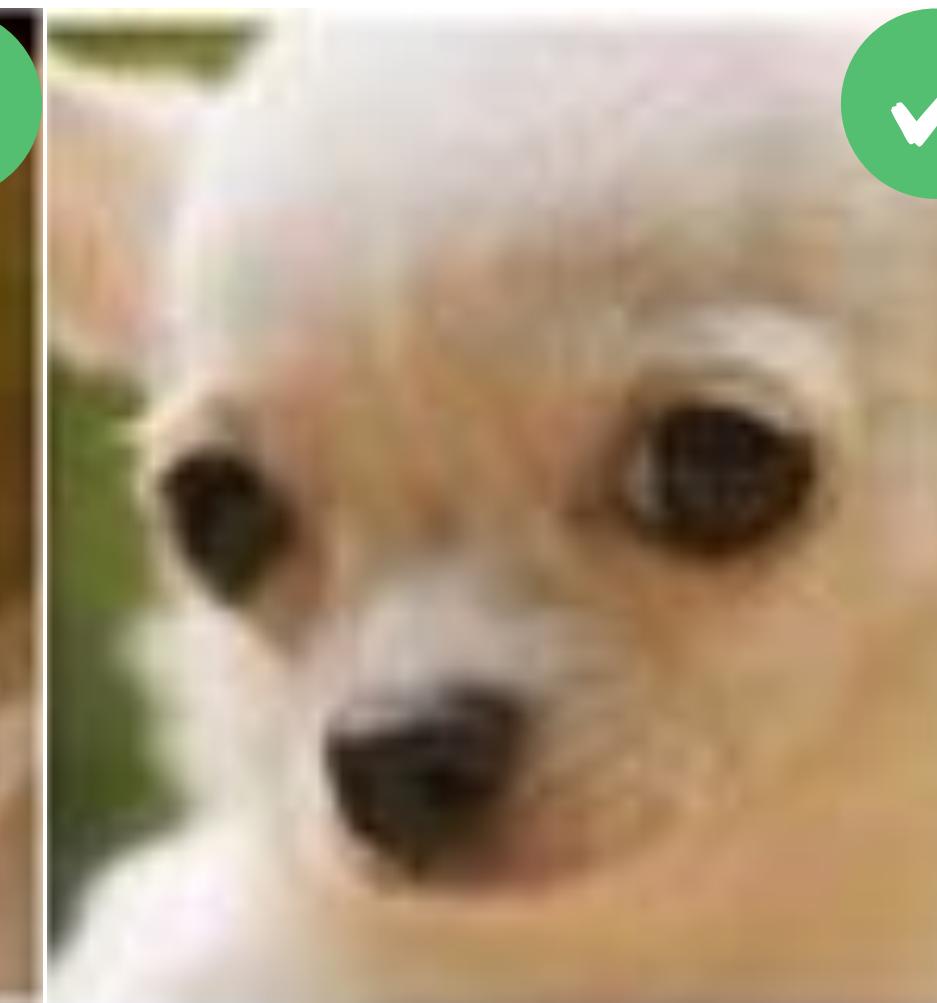
Chihuahua Accuracy



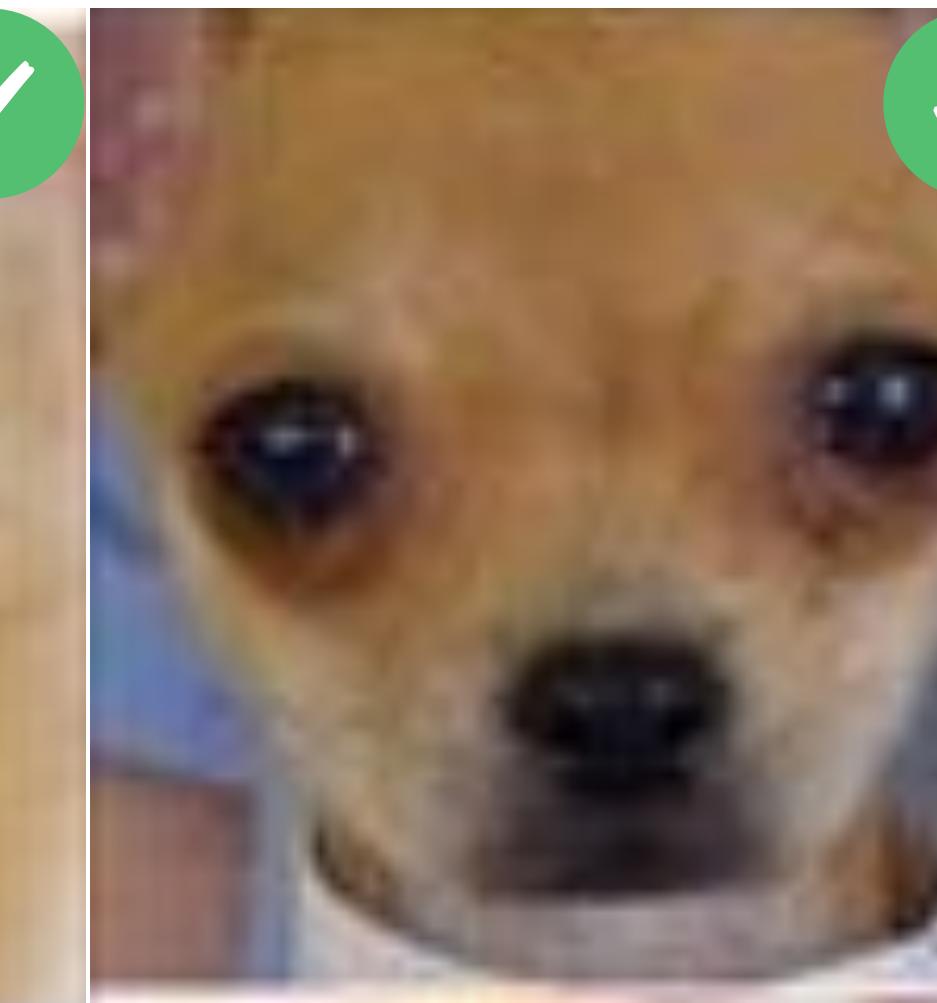
Results



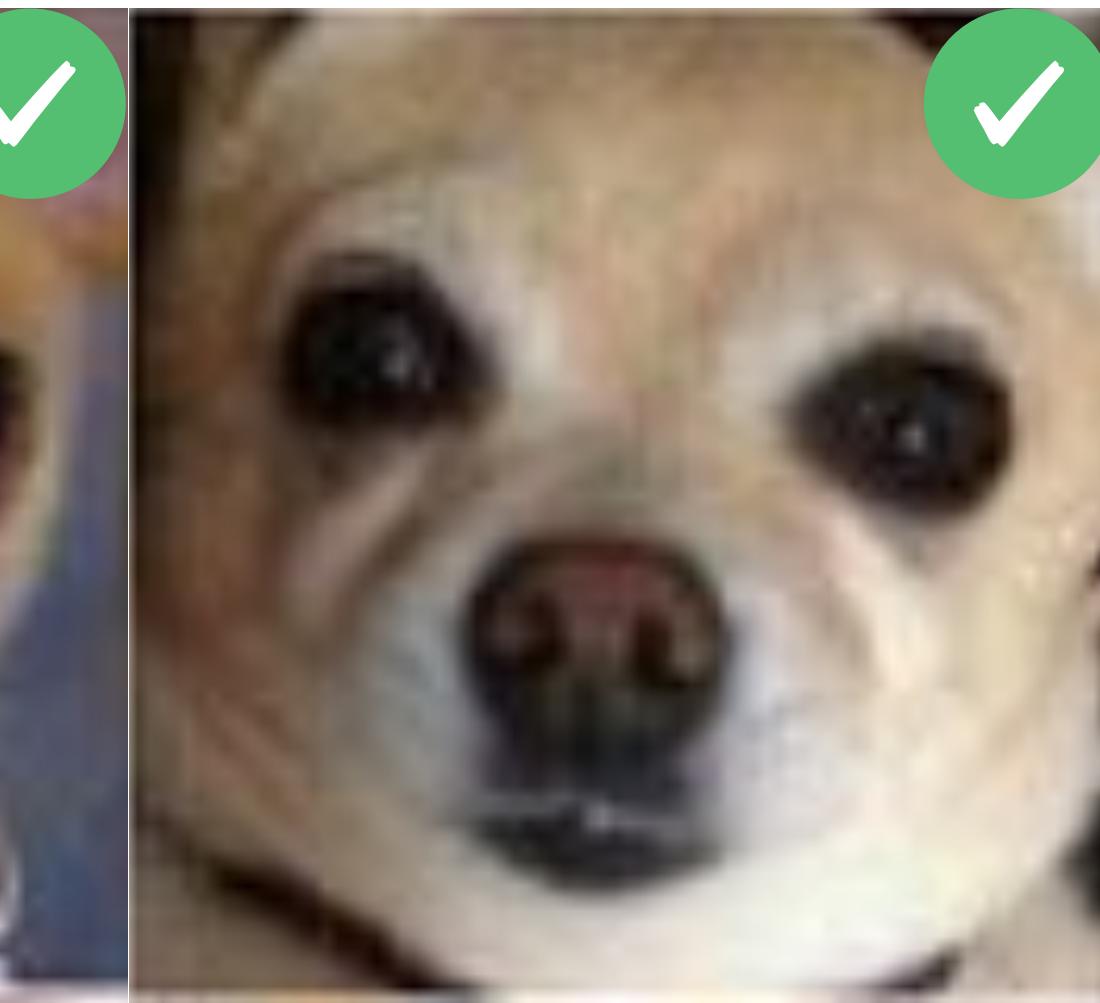
99% Chihuahua



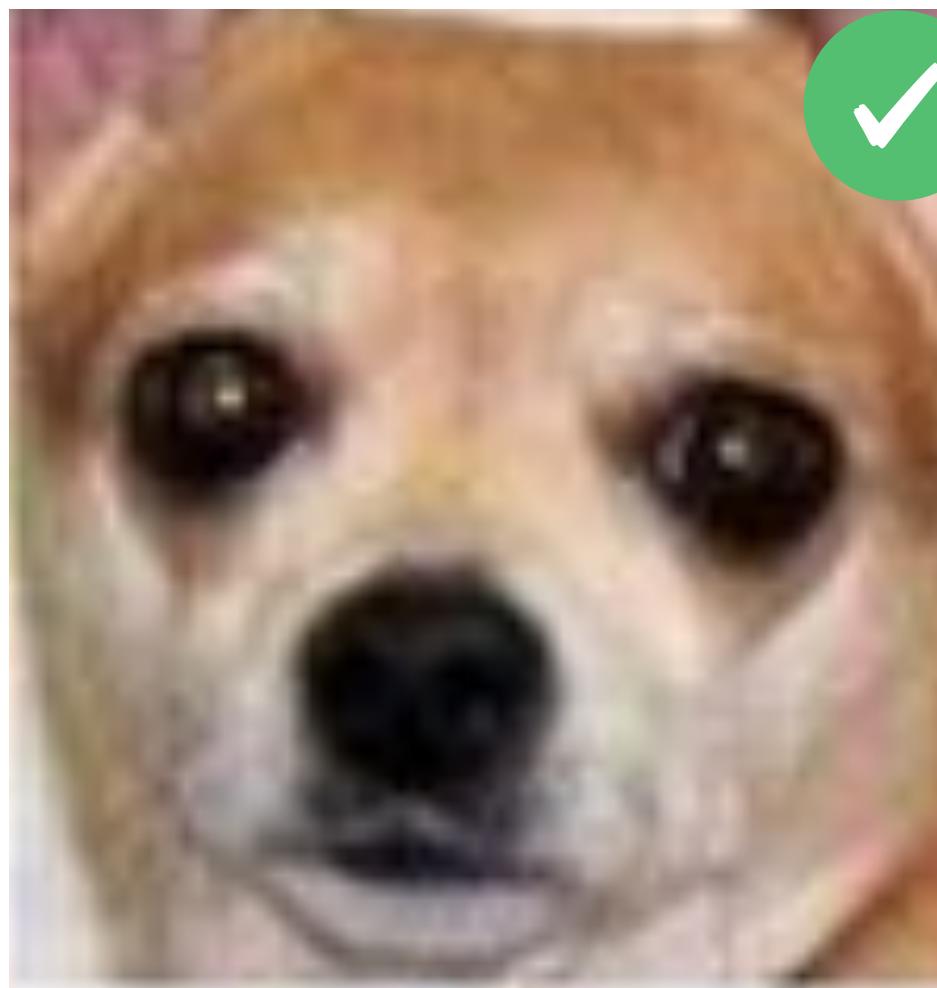
99% Chihuahua



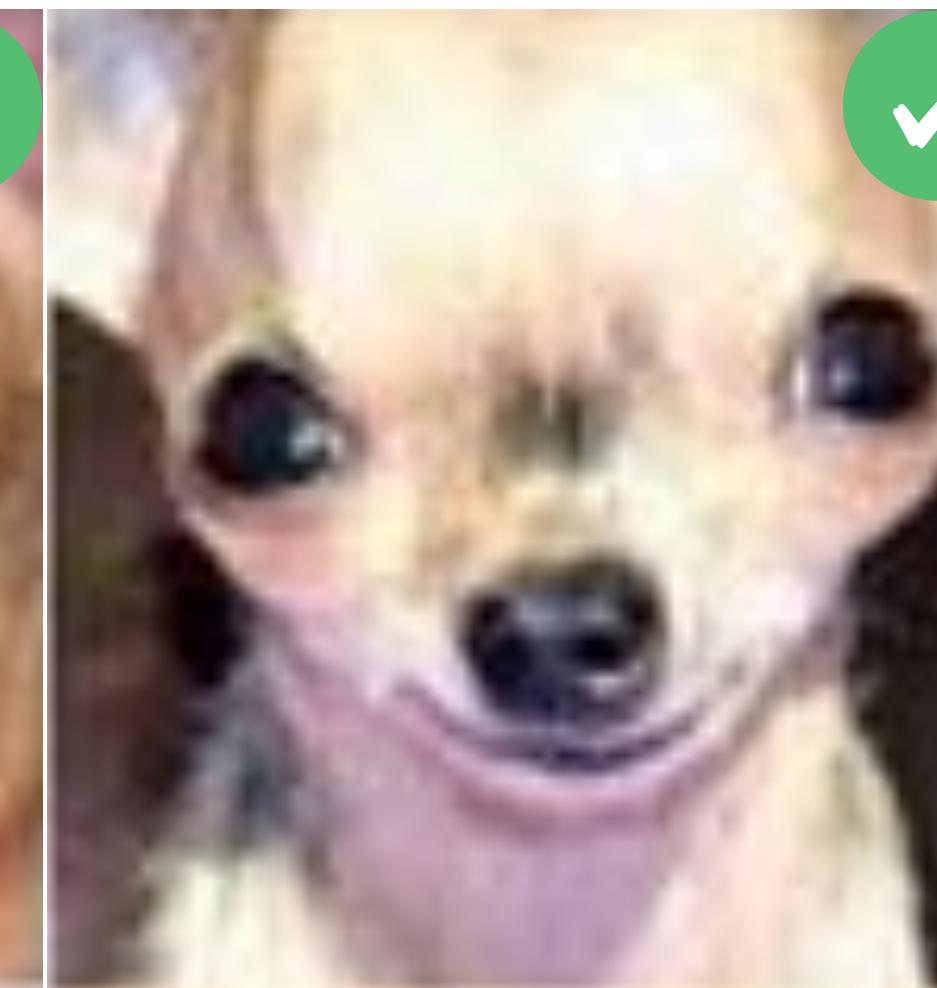
97% Chihuahua



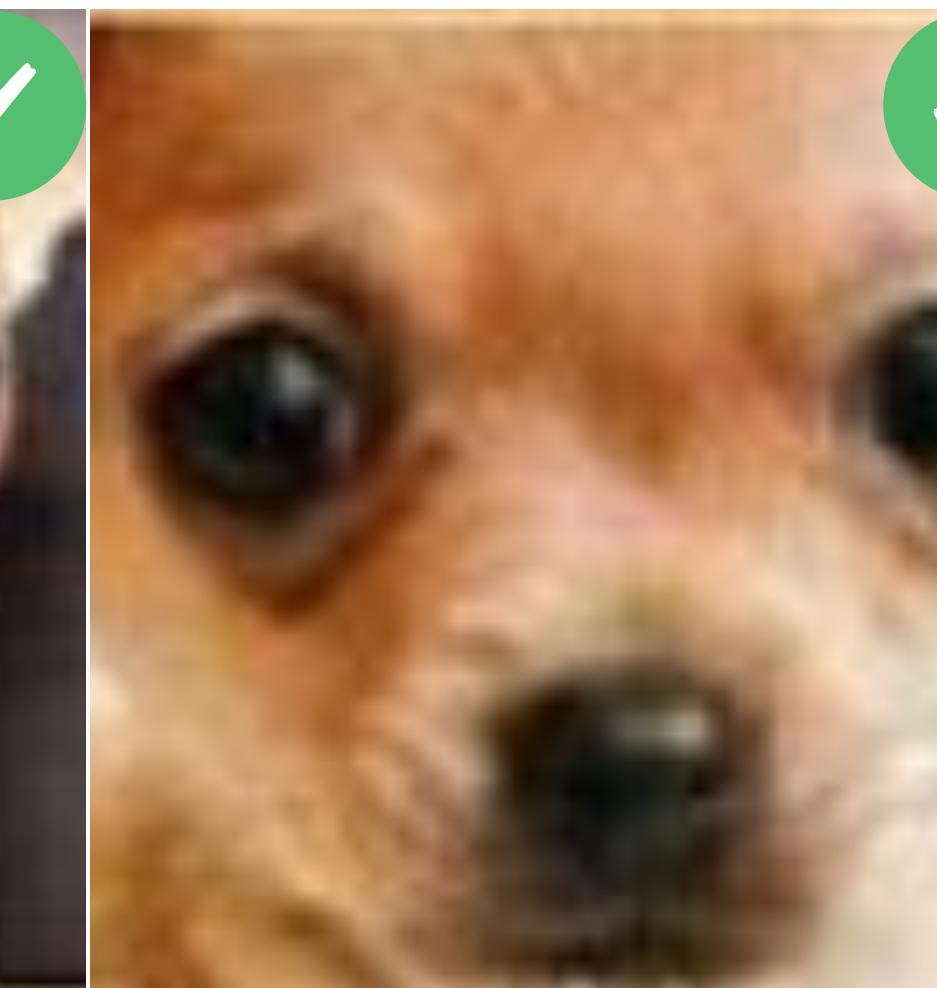
99% Chihuahua



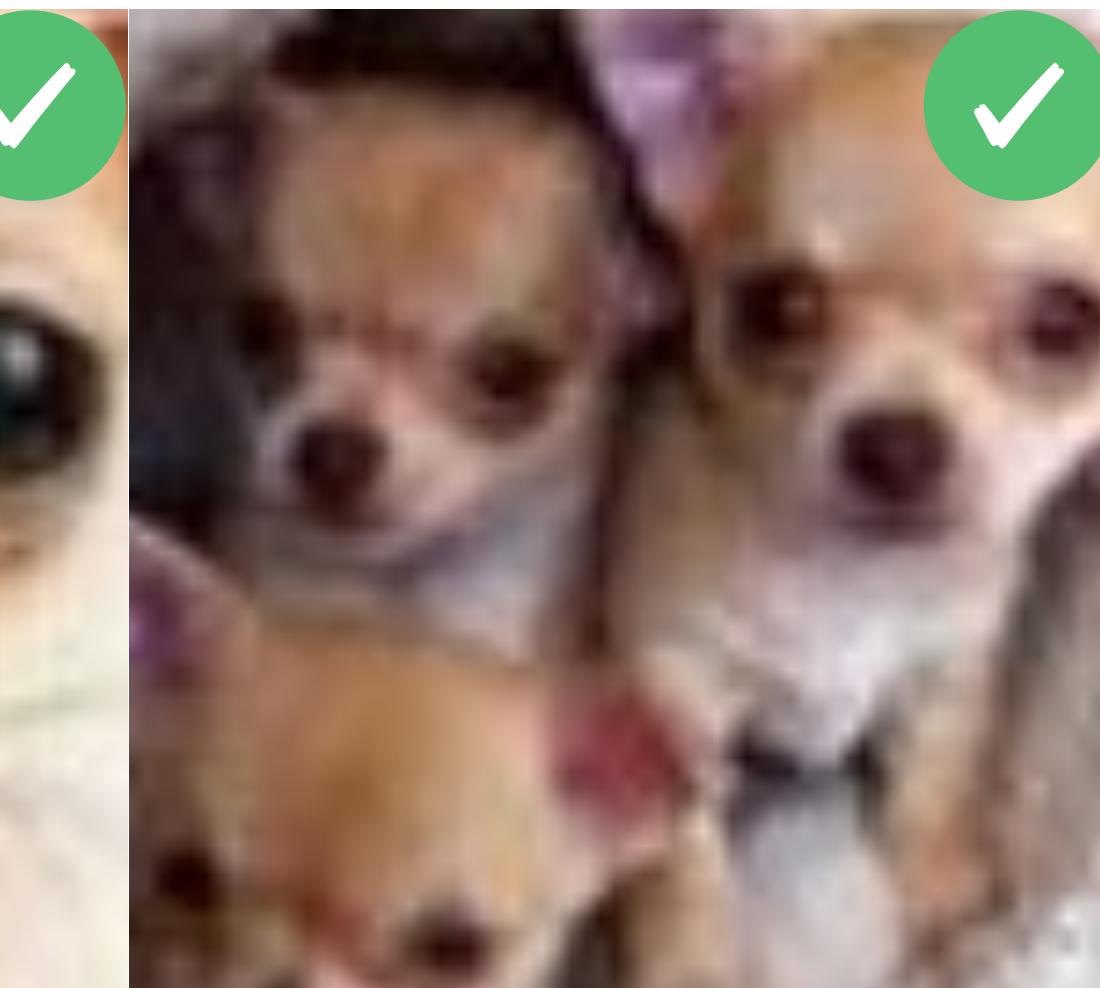
99% Chihuahua



99% Chihuahua



99% Chihuahua

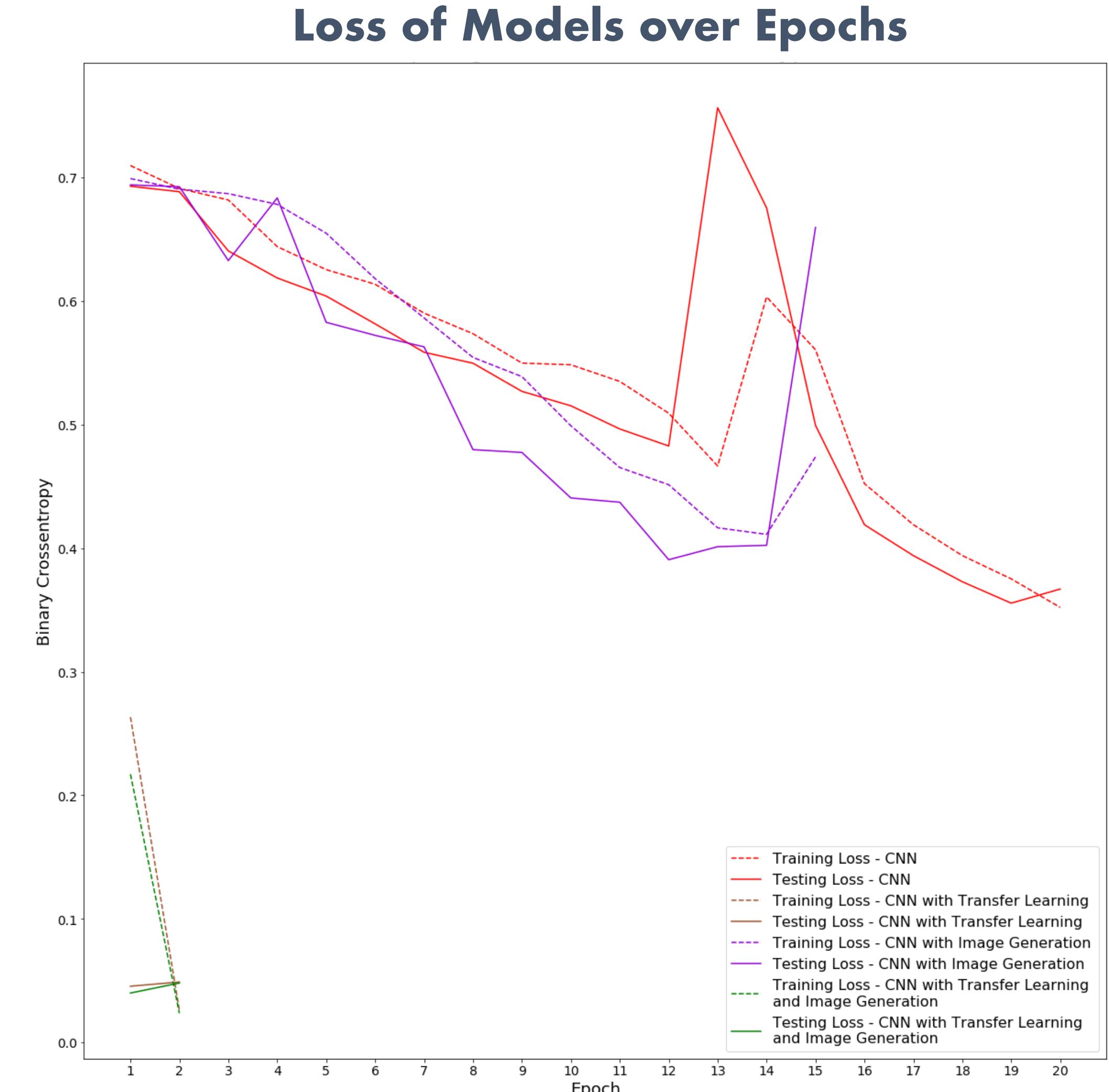
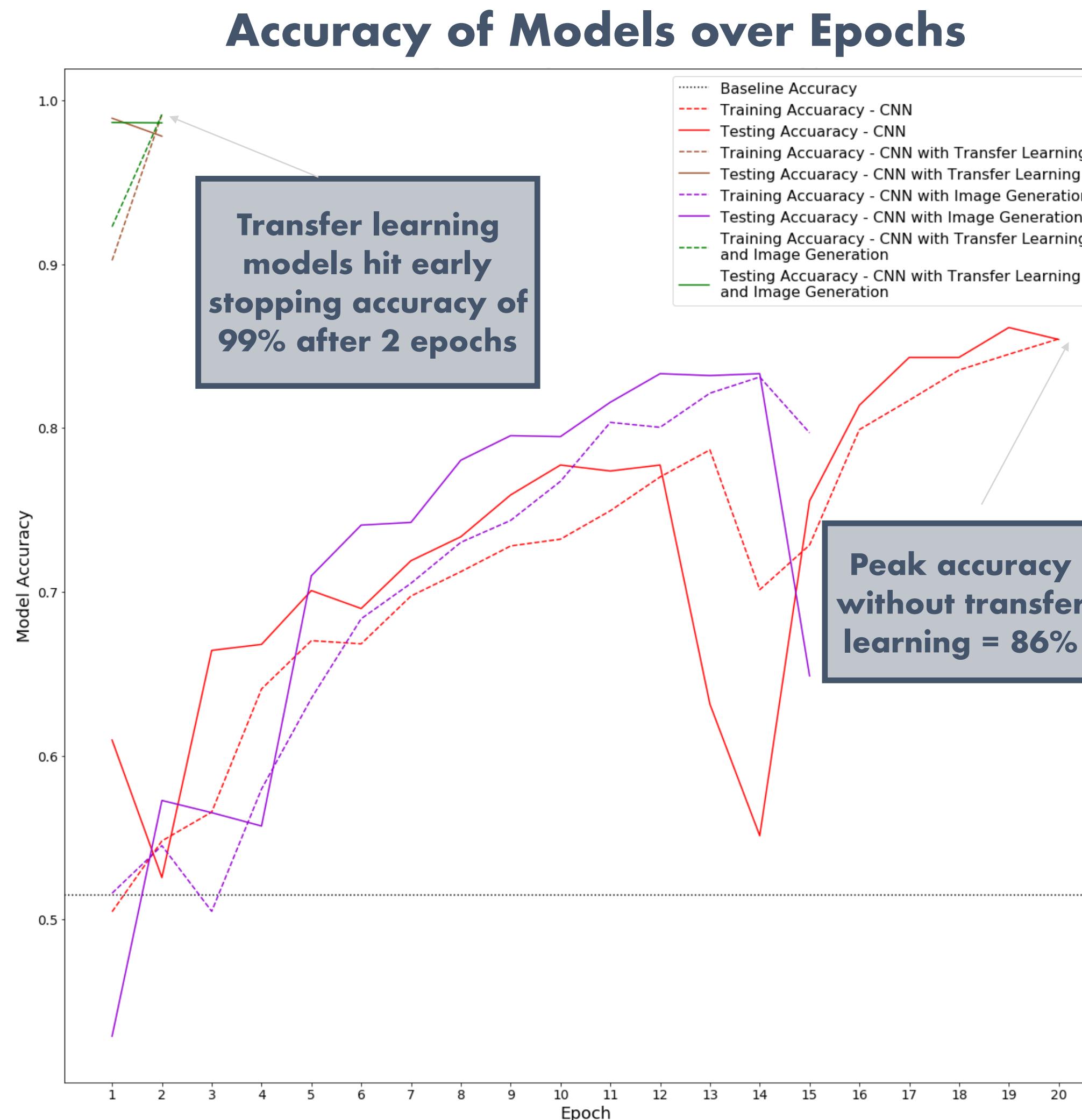


99% Chihuahua

Accuracy and Loss Curves



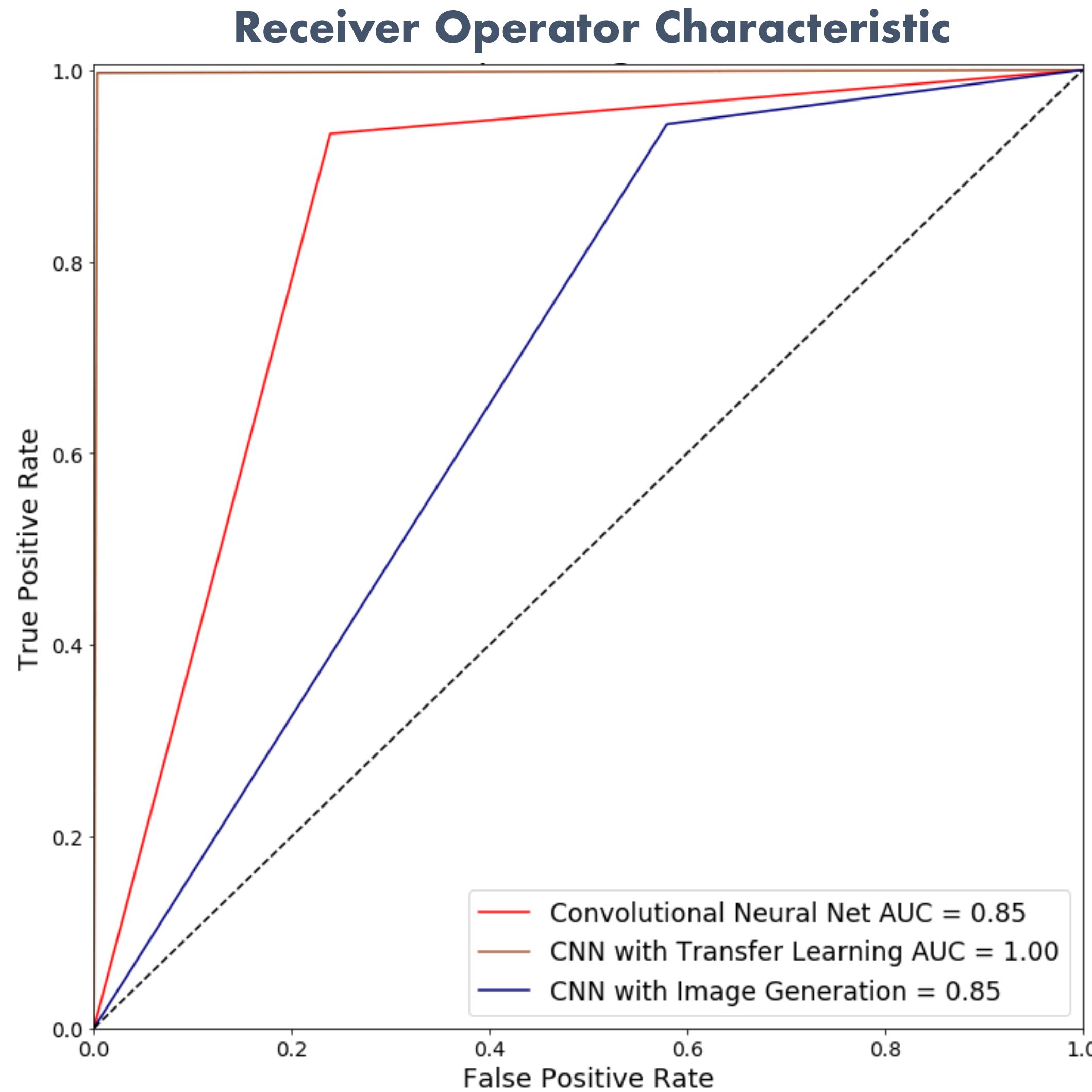
Results



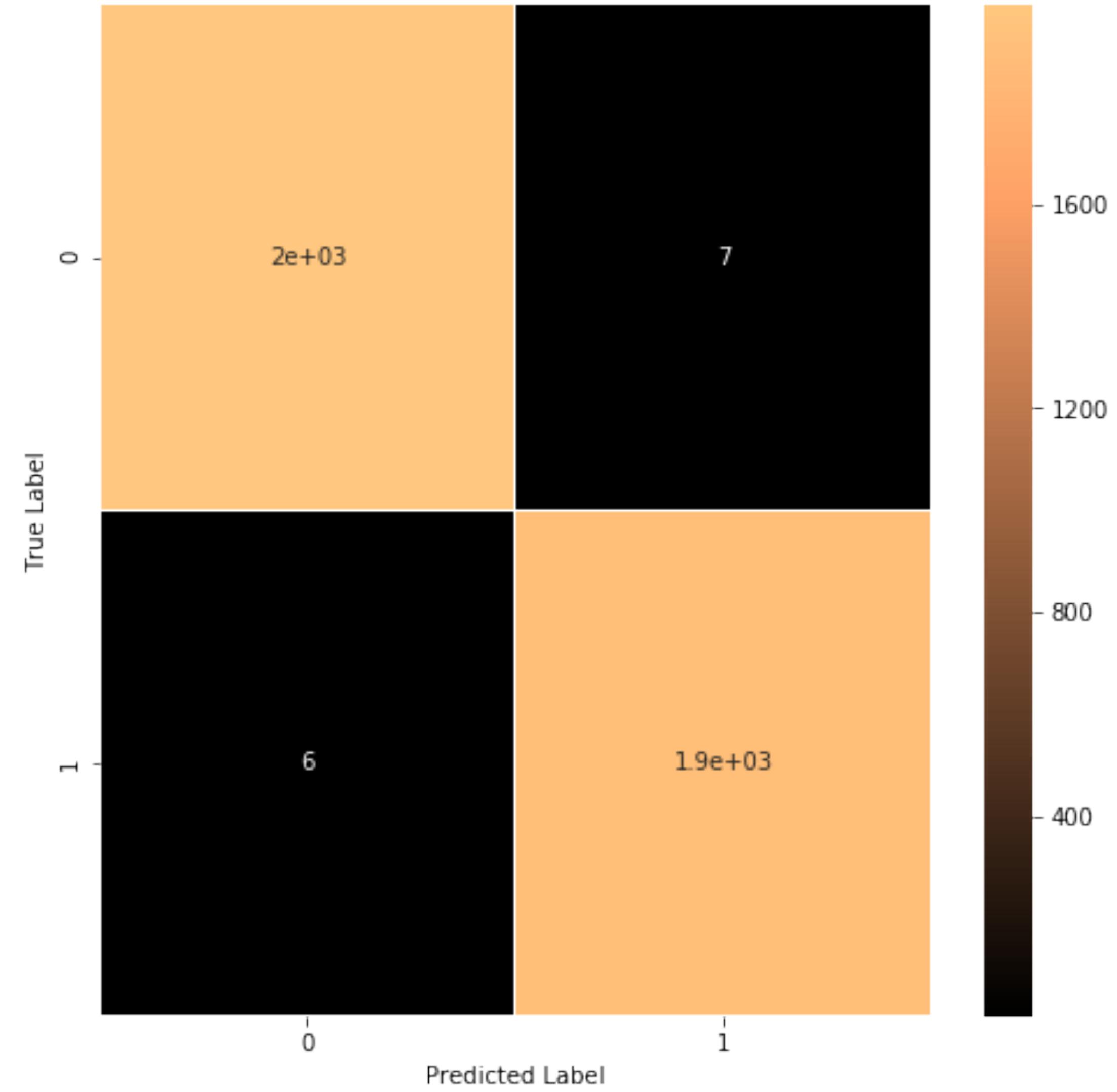
ROC and Confusion Matrix



Results



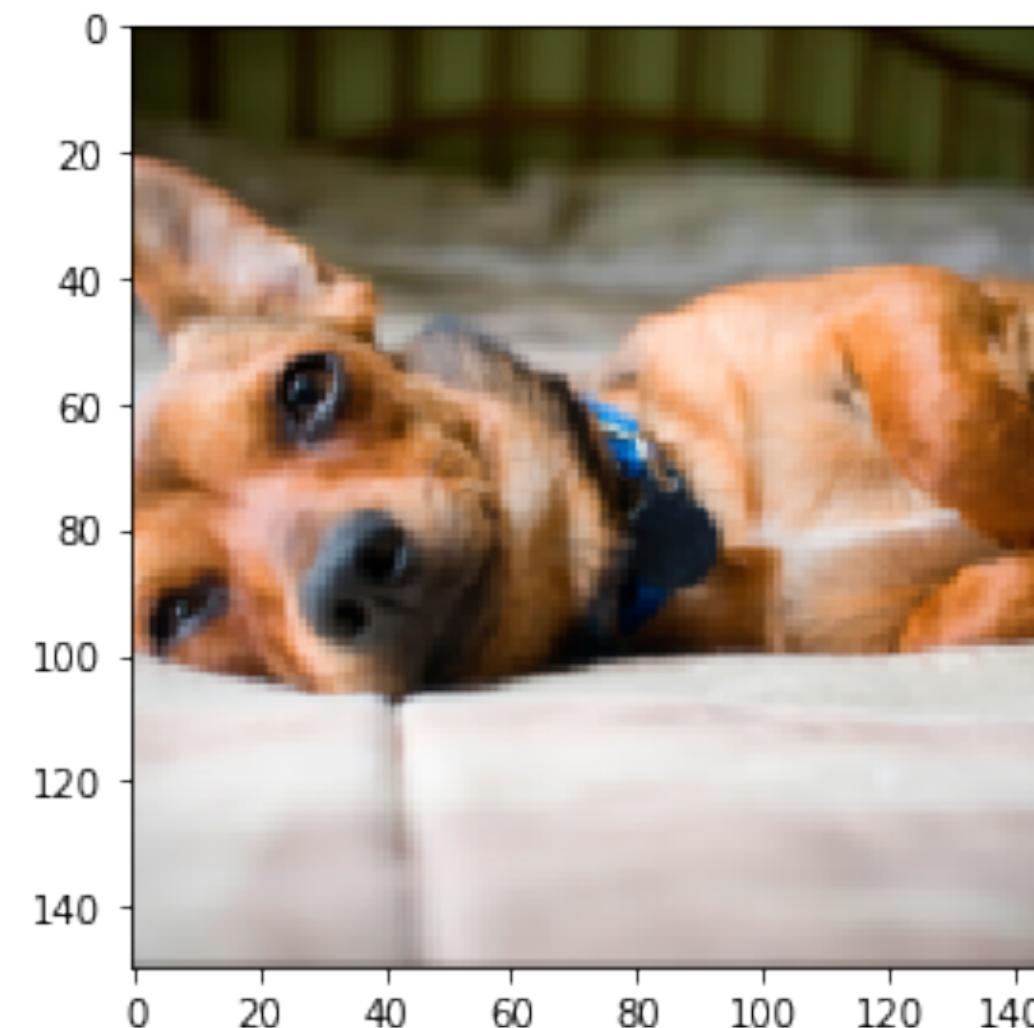
Confusion Matrix - Transfer Learning



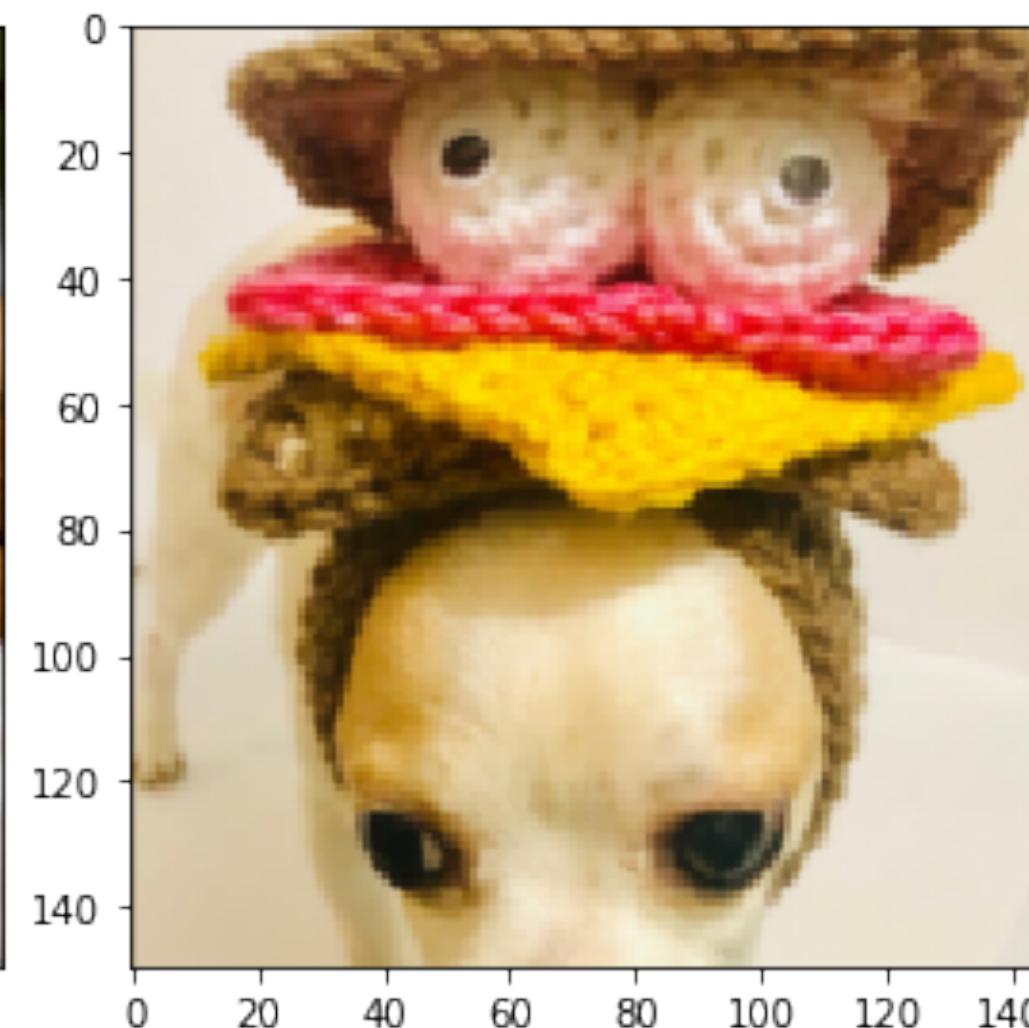
Misclassified Images



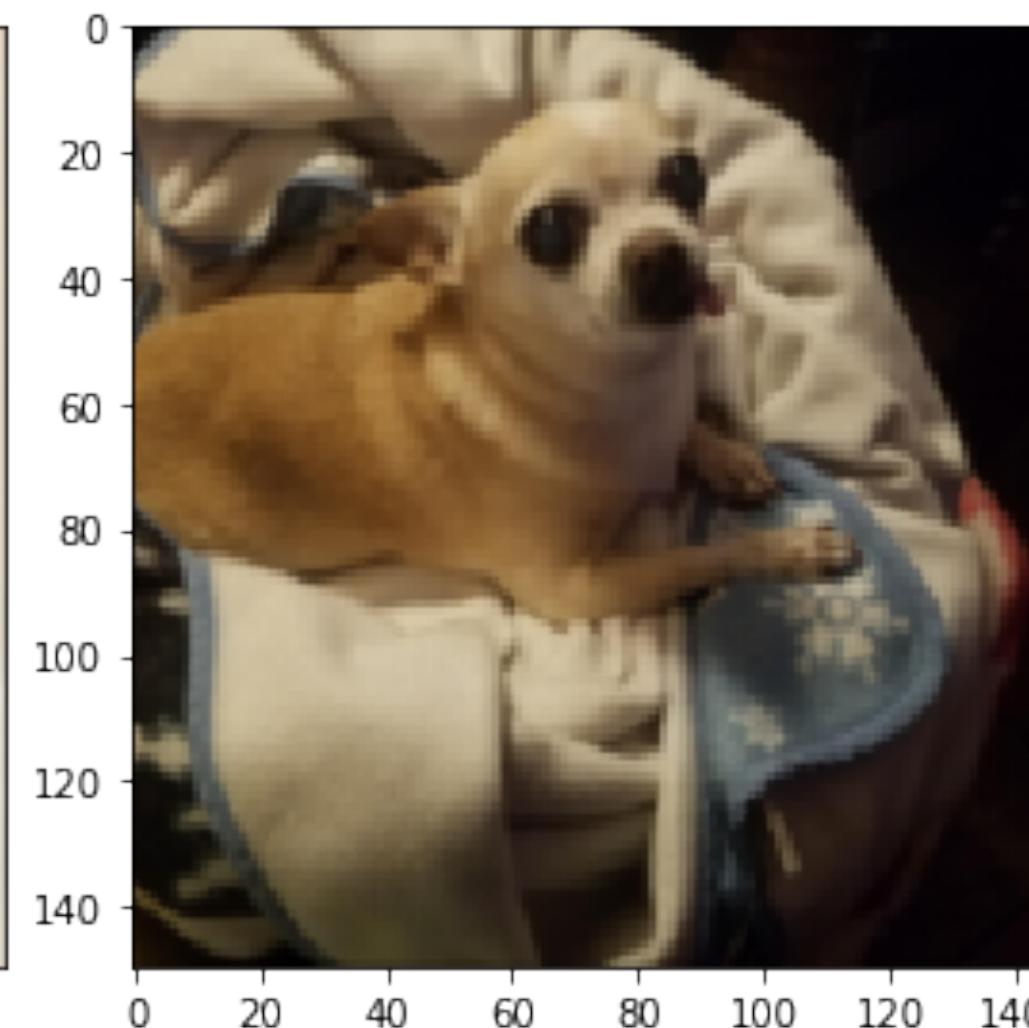
Results



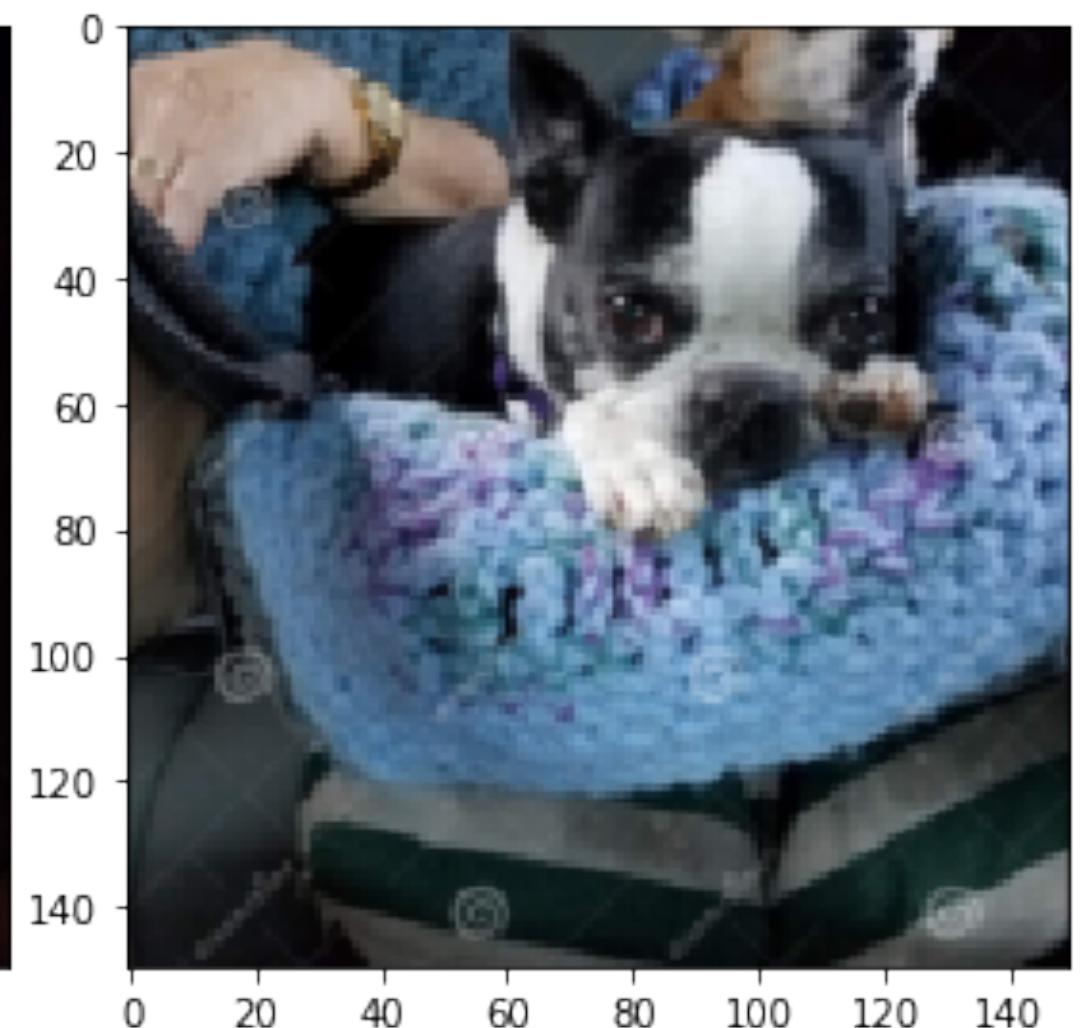
3% Chihuahua



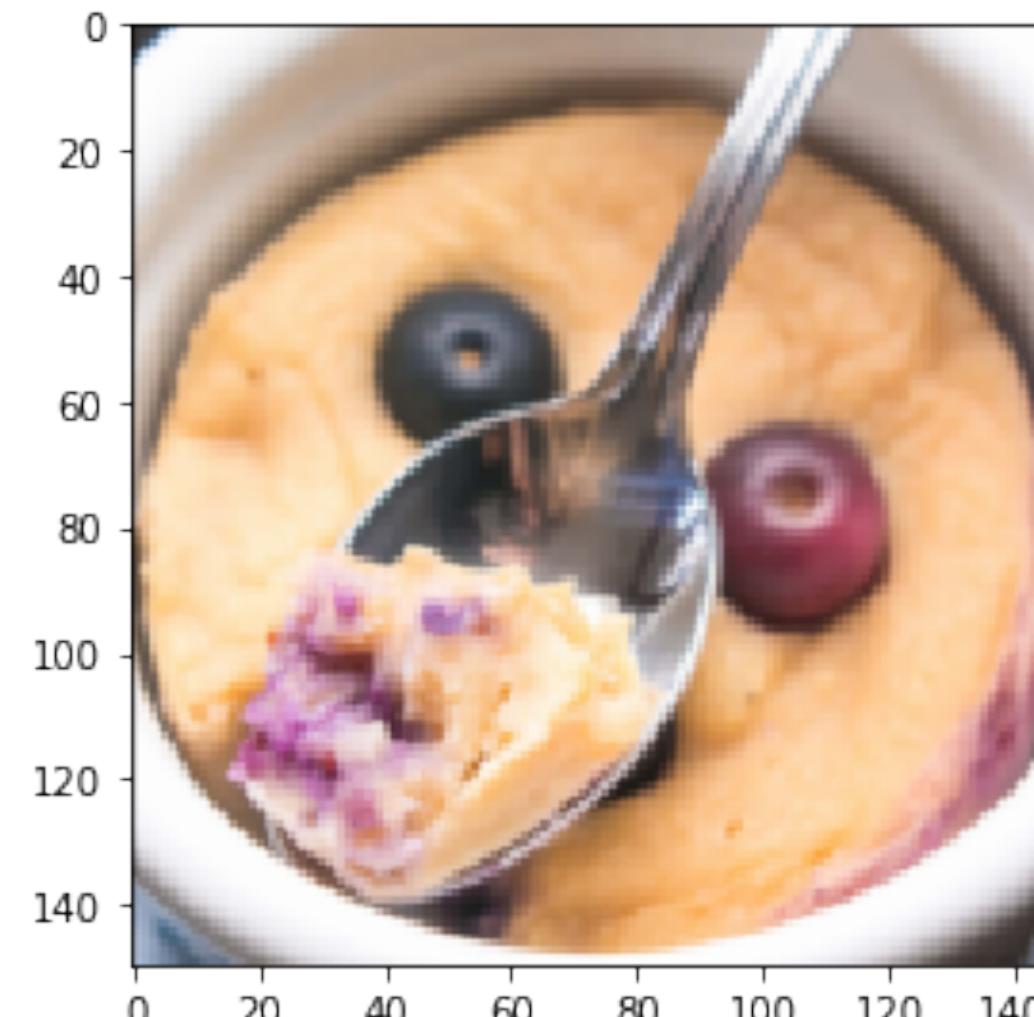
44% Chihuahua



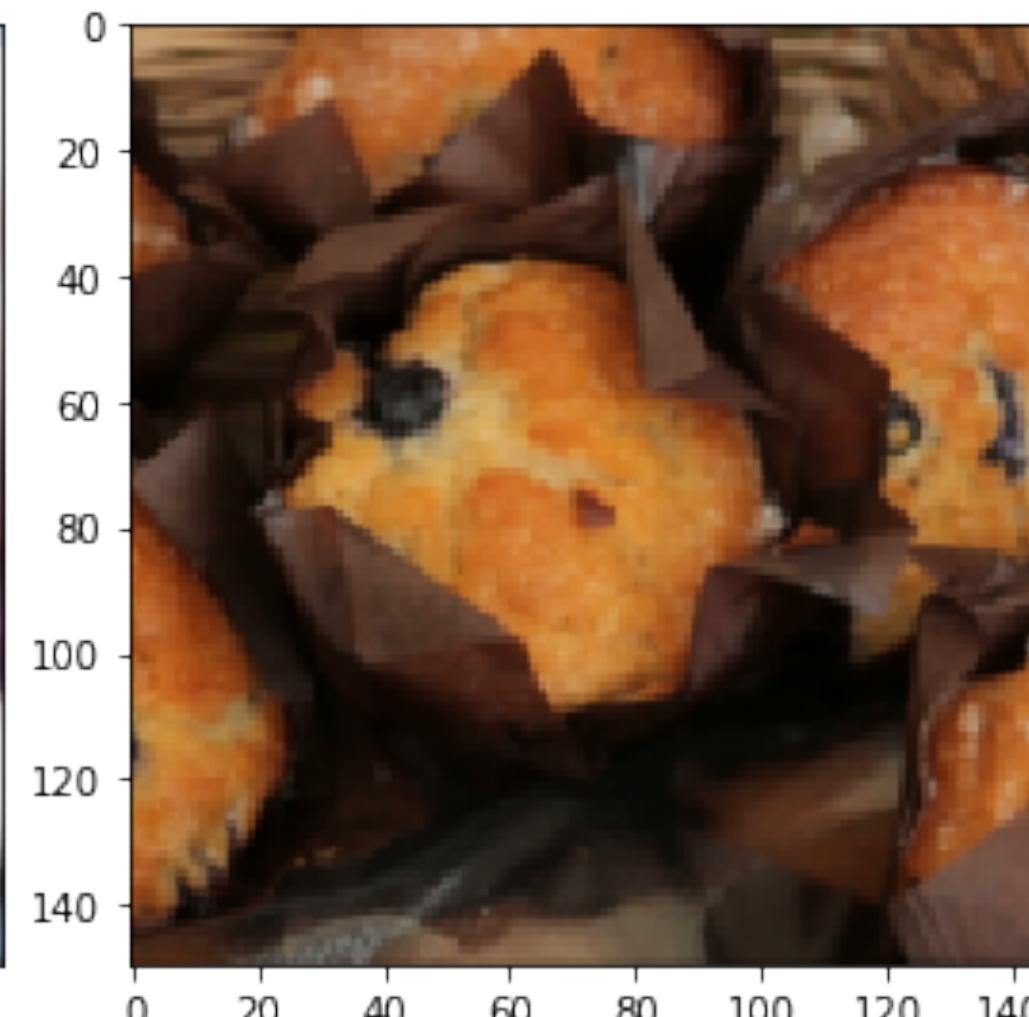
36% Chihuahua



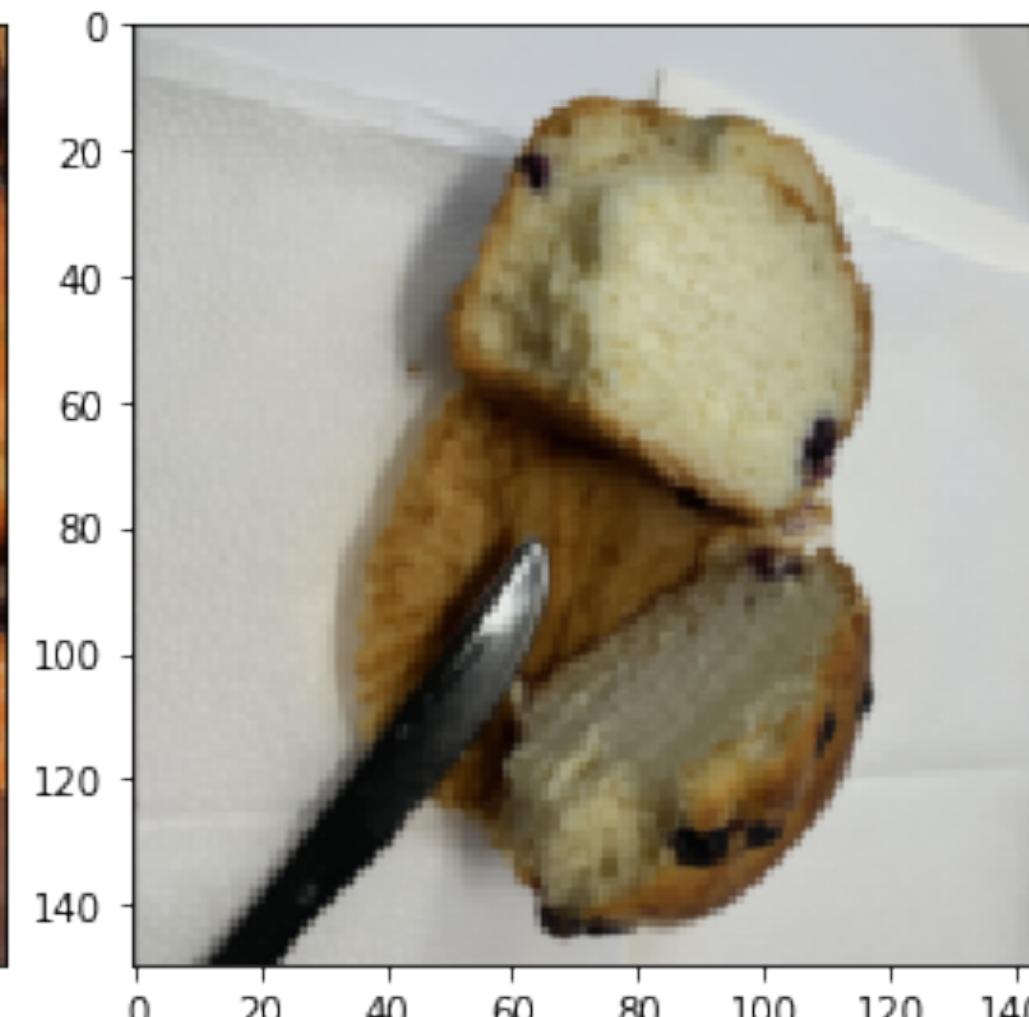
0% Chihuahua



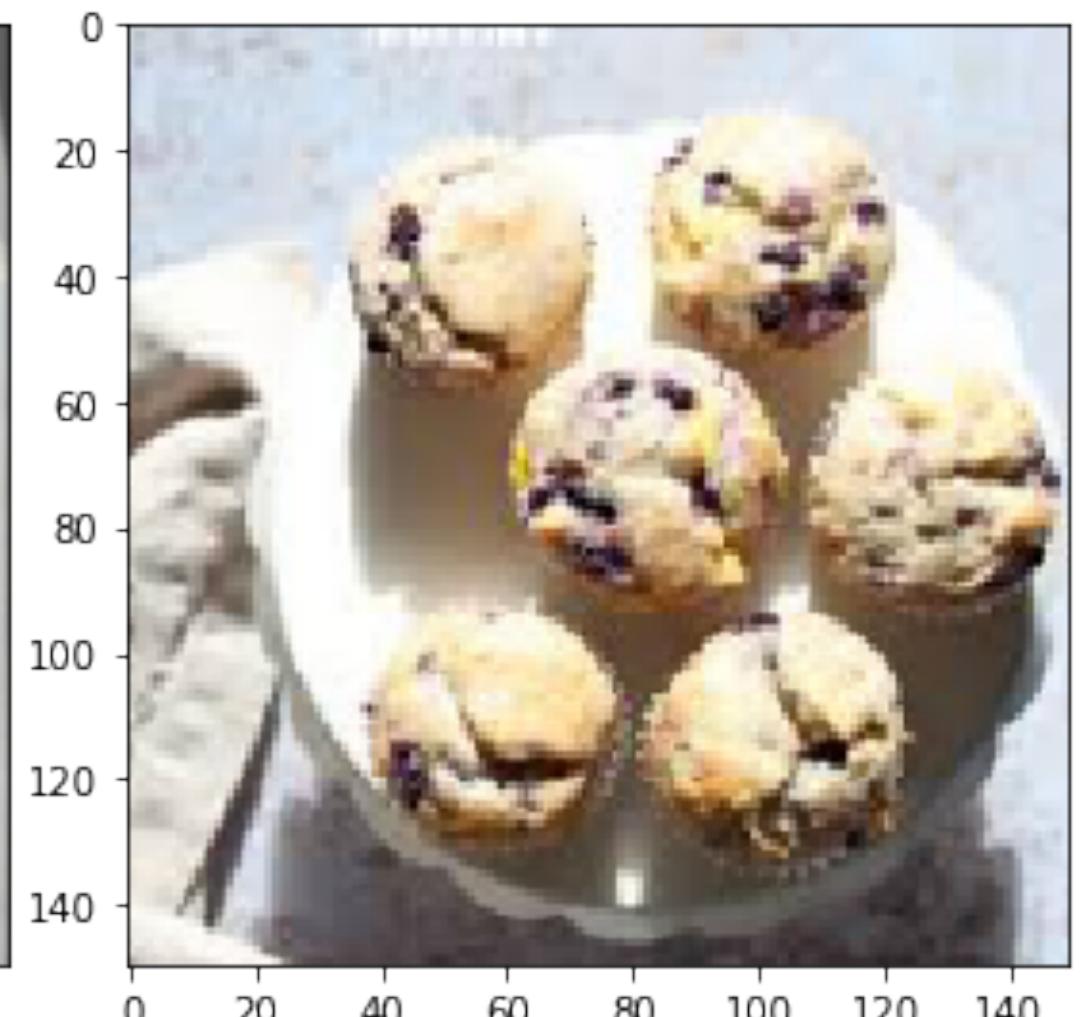
10% Muffin



40% Muffin



16% Muffin



49% Muffin

Presentation Overview



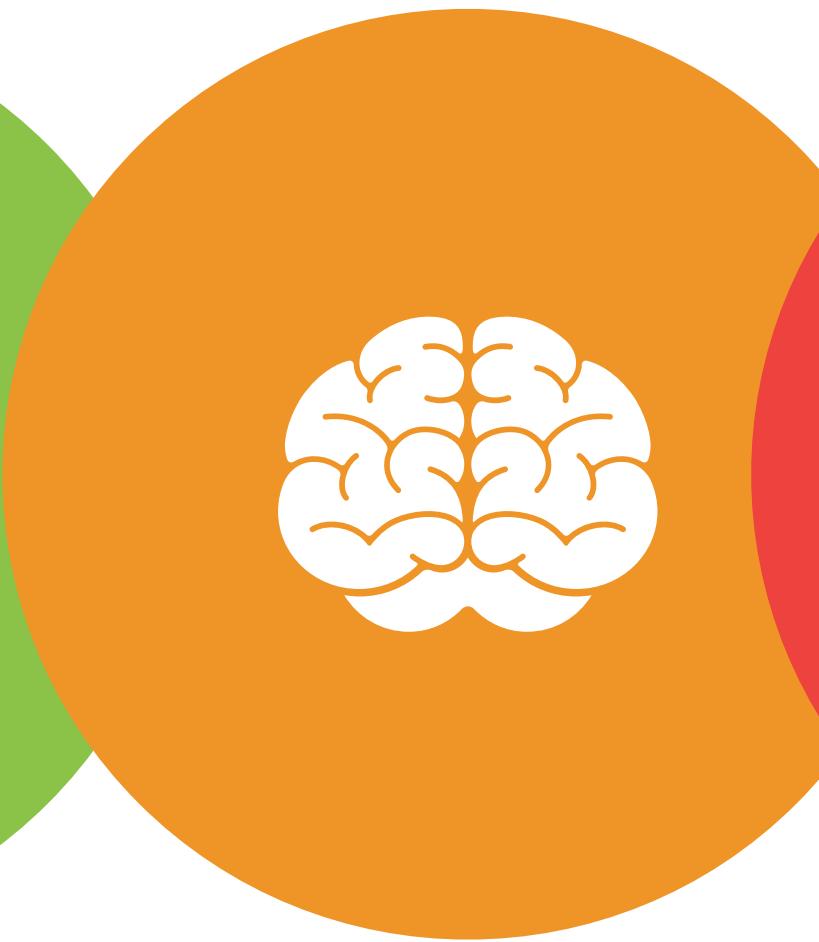
**Problem
Statement**



**Image
Scraping**



**Image
Editing**



**Image
Classification
Neural Net**



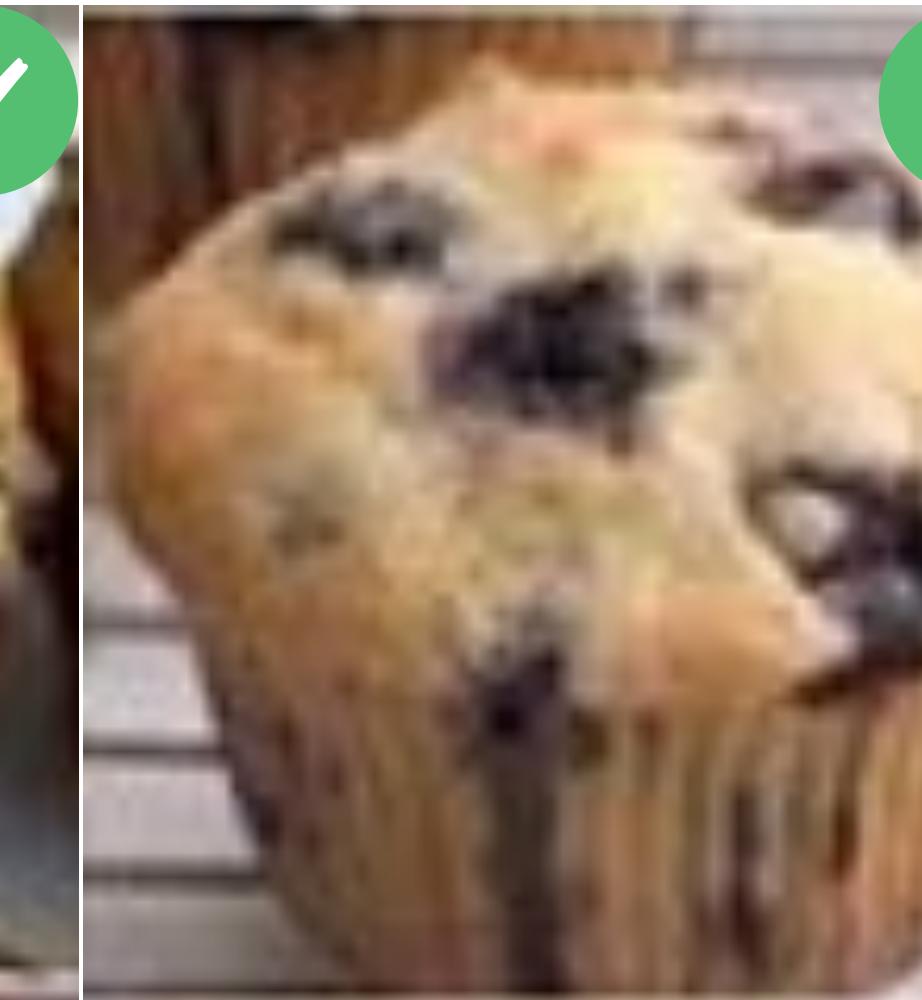
Results

Appendix

Muffin Accuracy - CNN Only



Results

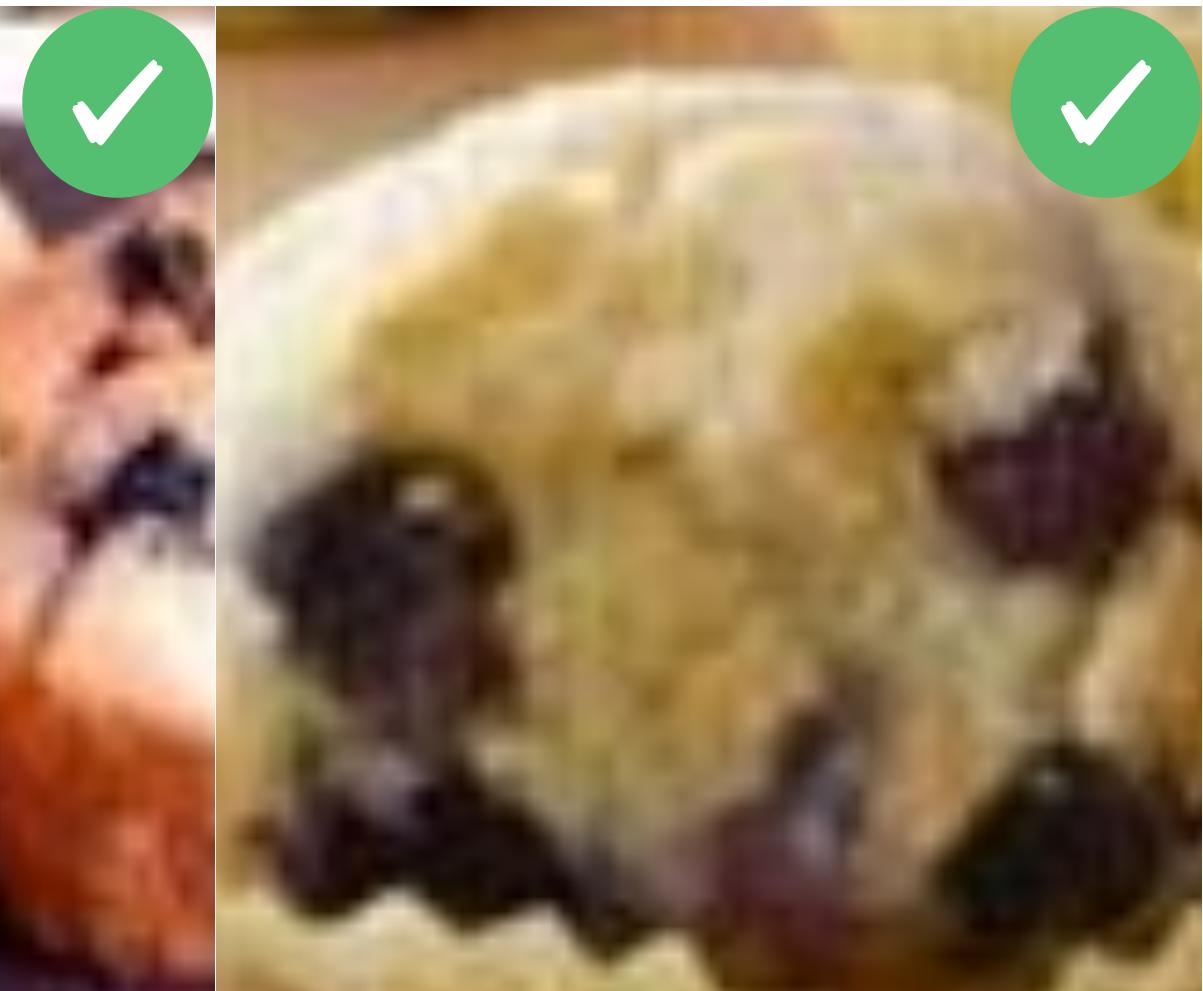
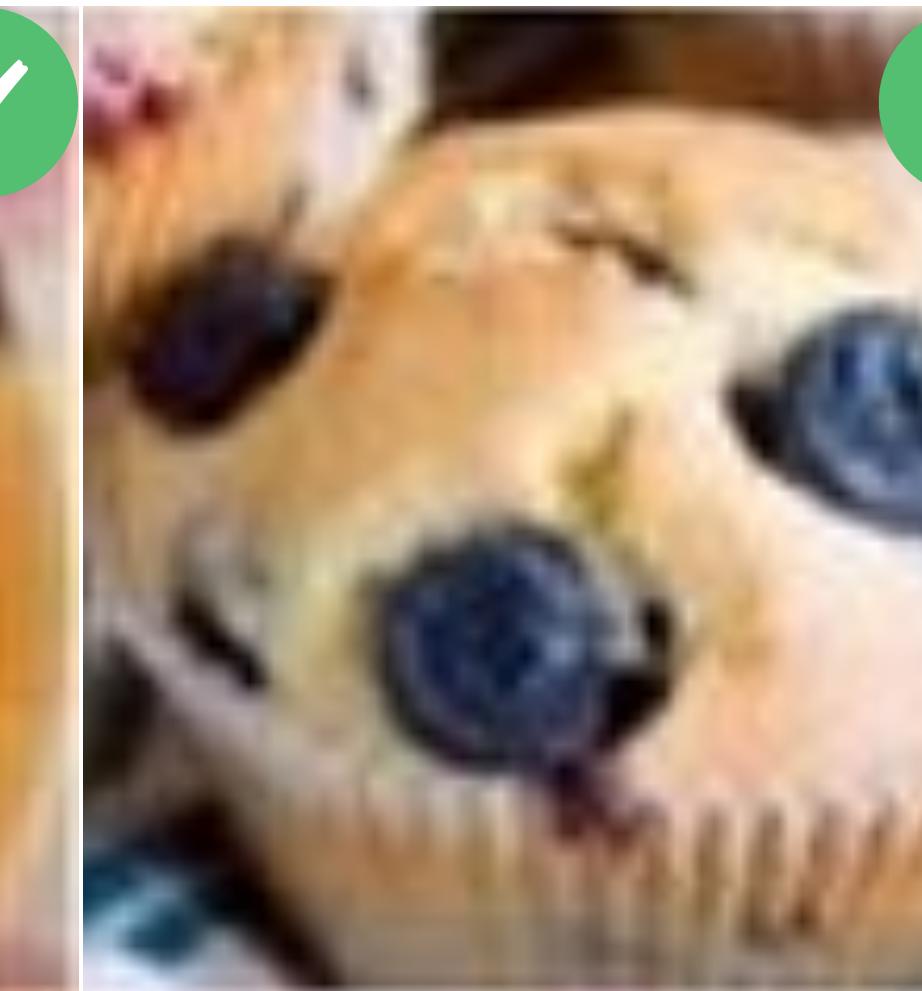


98% Muffin

88% Muffin

92% Muffin

34% Muffin



75% Muffin

82% Muffin

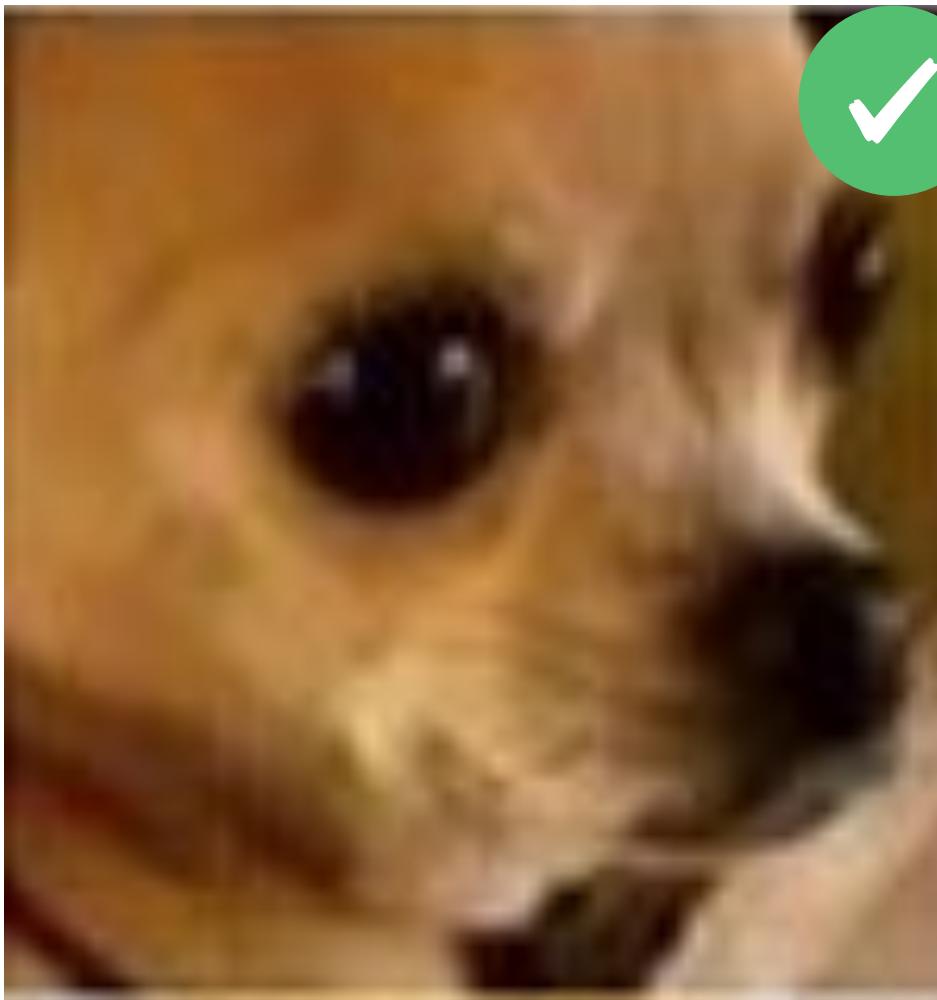
51% Muffin

91% Muffin

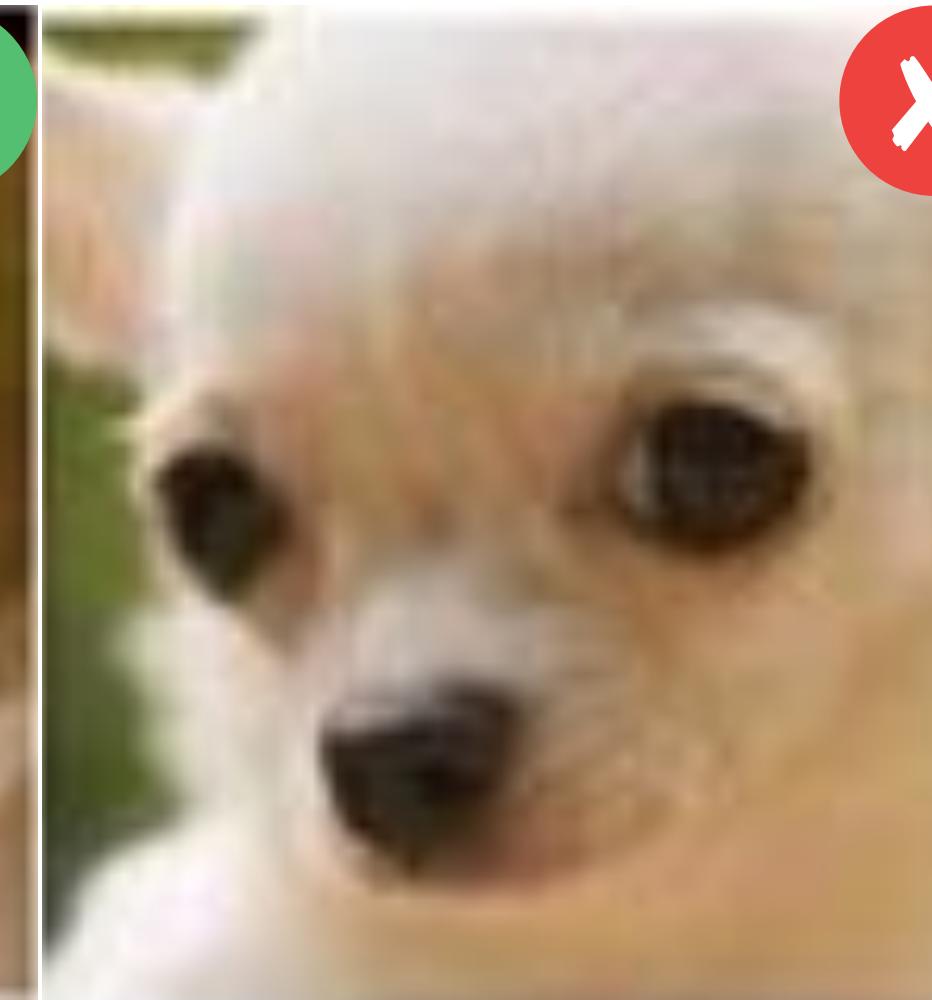
Chihuahua Accuracy - CNN Only



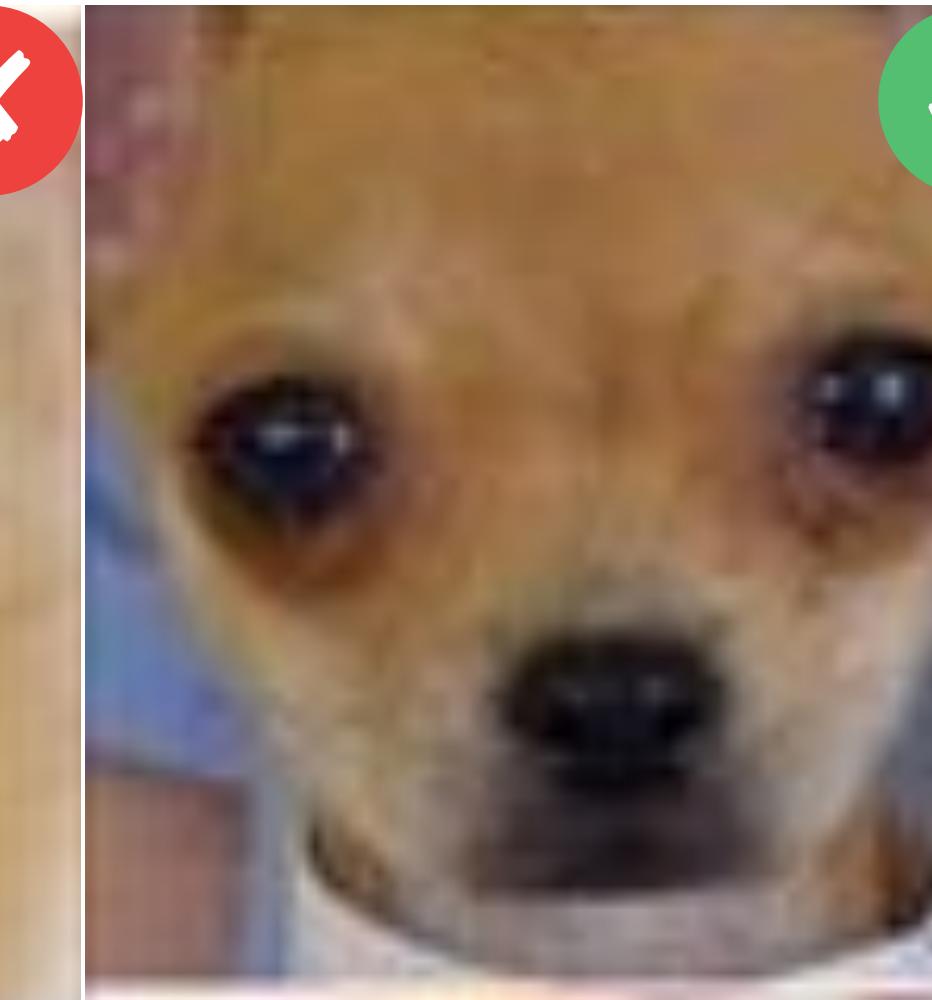
Results



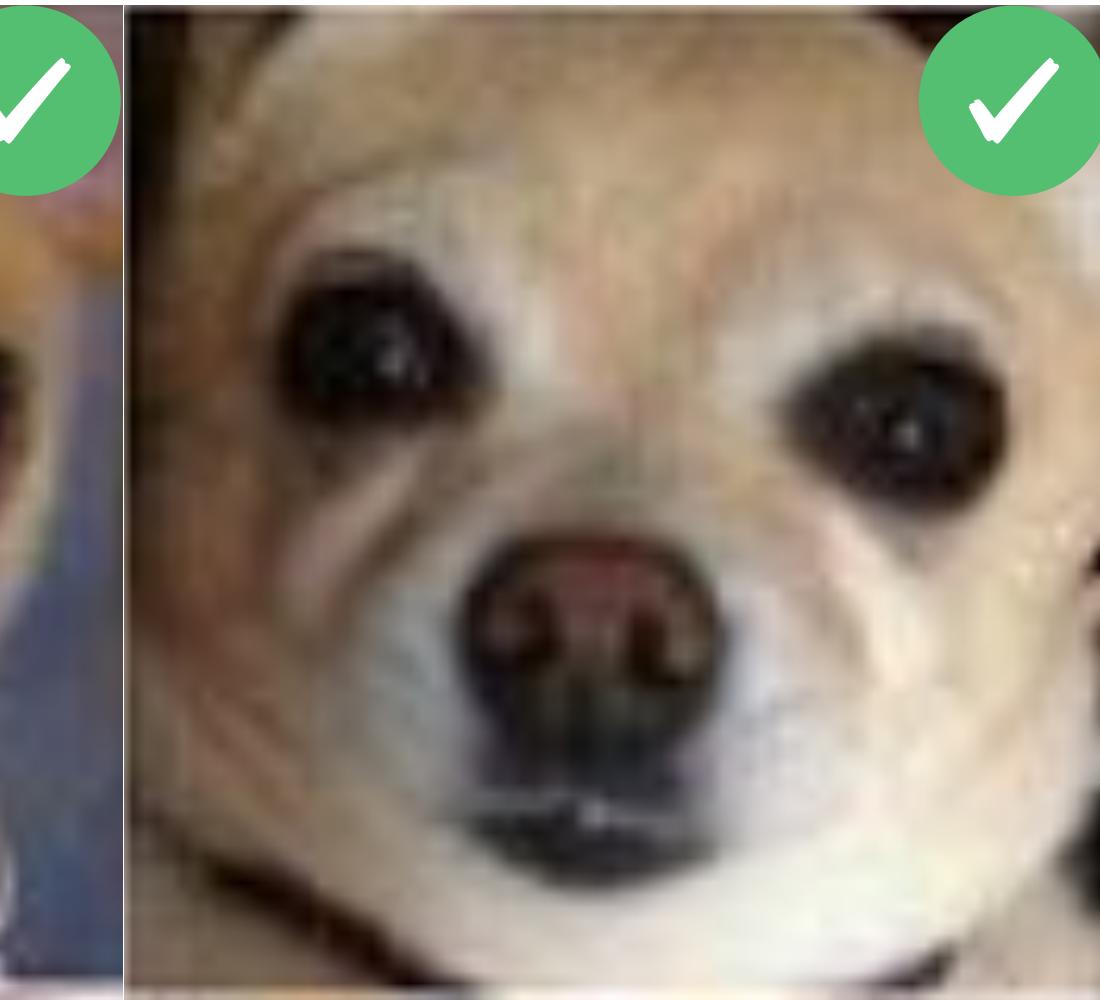
59% Chihuahua



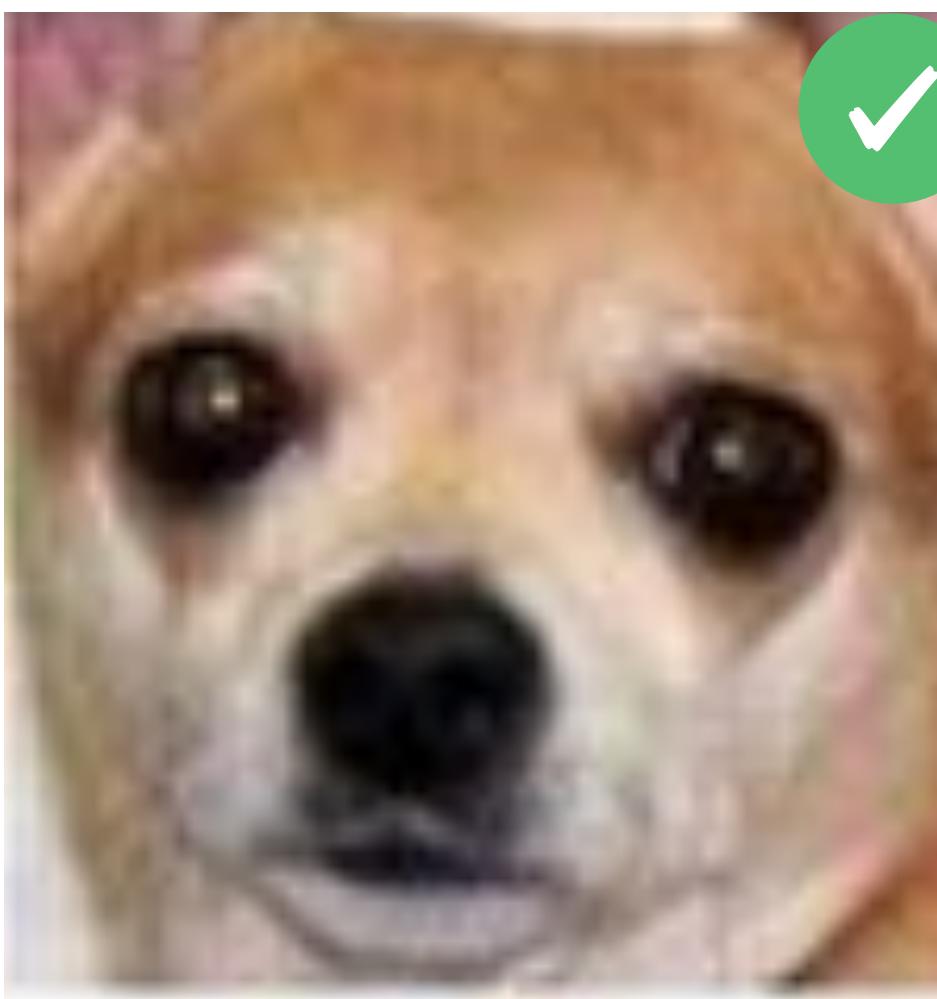
25% Chihuahua



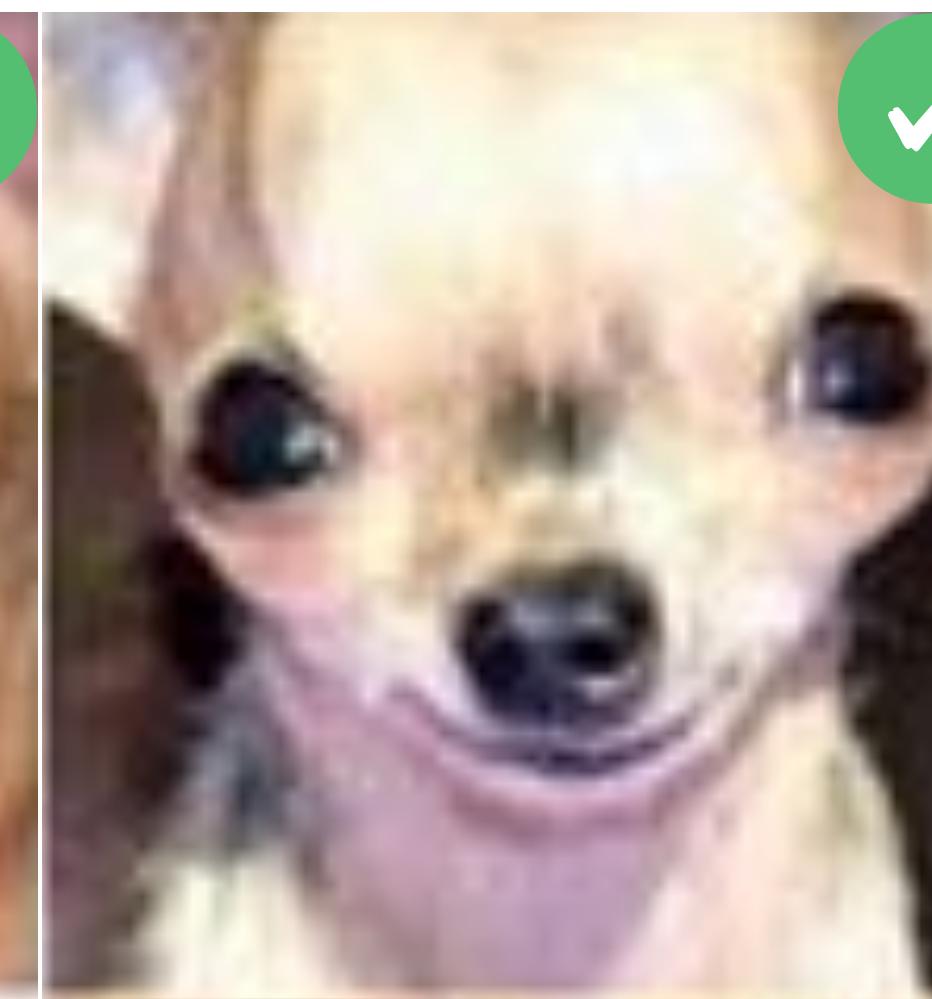
53% Chihuahua



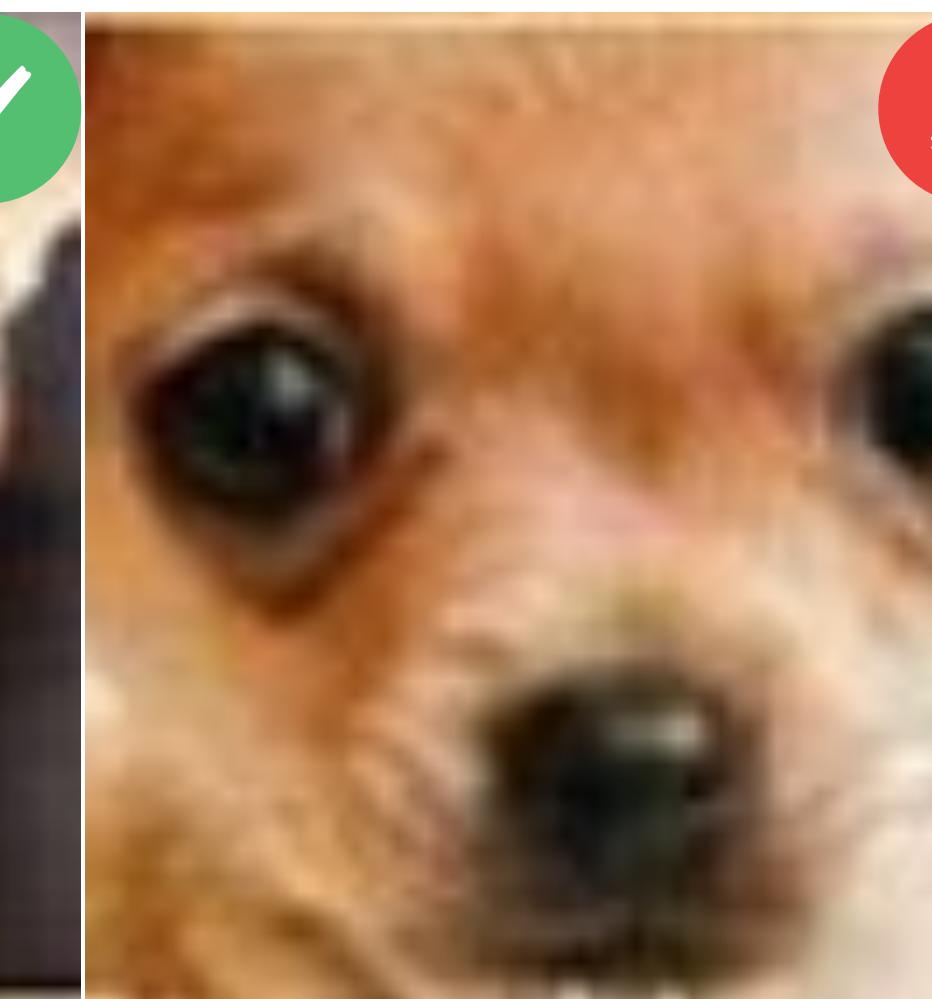
99% Chihuahua



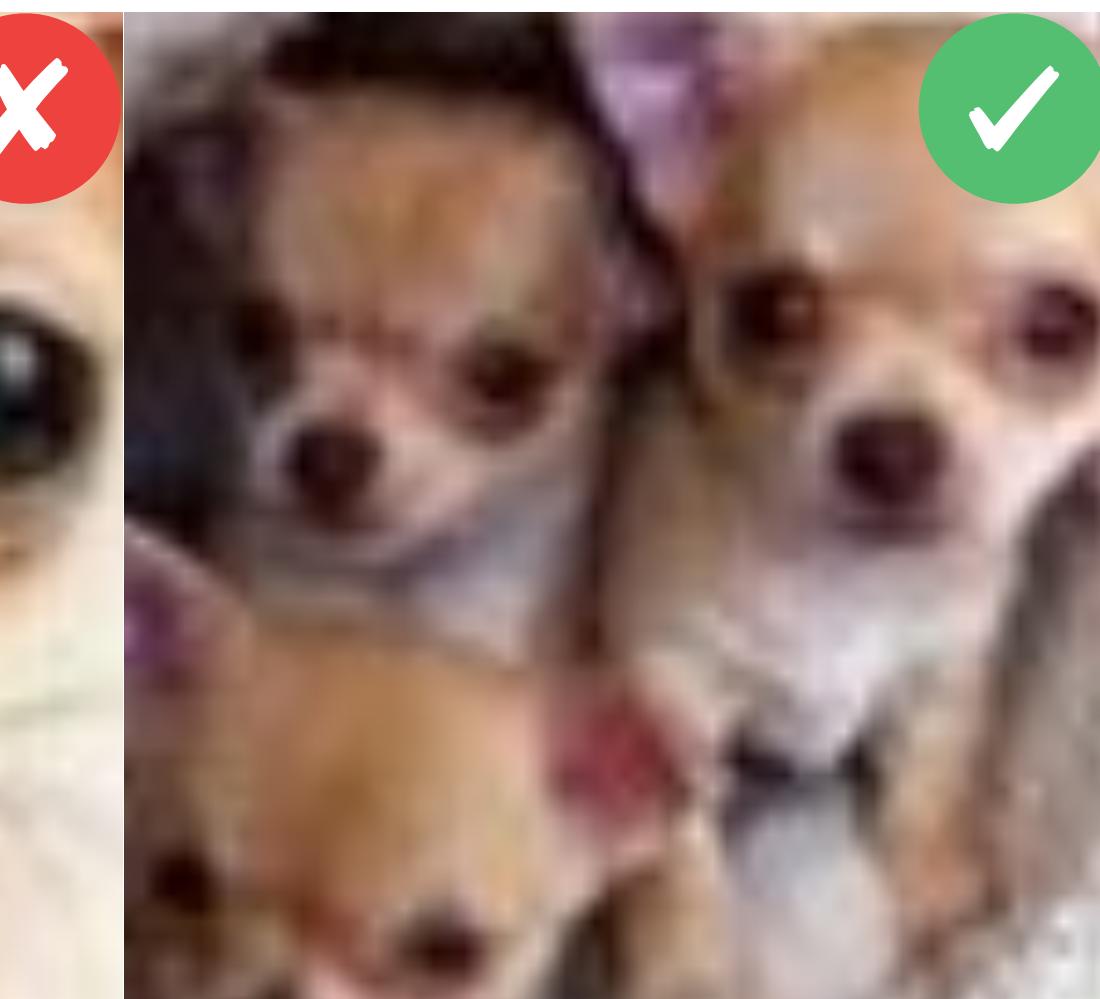
79% Chihuahua



88% Chihuahua

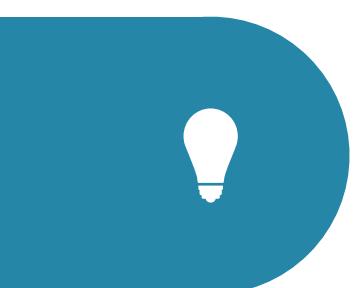


37% Chihuahua



95% Chihuahua

ROC and Confusion Matrix CNN Only



Results

