

DOG BREED IMAGE CLASSIFICATION

-Chris Burger



Contents

- 1. Problem Statement**
- 2. EDA/Preprocessing**
- 3. Modeling**
- 4. Examples**
- 5. Conclusion and the Future**



Problem Statement

This classifier was made to identify which dog breed is shown in a given picture.

-A classification model to answer the question:

“What kind of Dog IS this?!”



R.I.P. Cooper Michael Jackson Burger



Dataset information, Pre-Processing and Exploratory Analysis

The dataset used is the 'Stanford Dogs Dataset for Fine-Grained Visual Categorization' found [here](#).

20,580 total images of **120** different Dog Breeds (~170 per breed. A Good size of data to work with)

- Reformatted the images as they were different sizes, attempting the dimensions:

28 x 28, 299 x 299 and finally 180 x 180

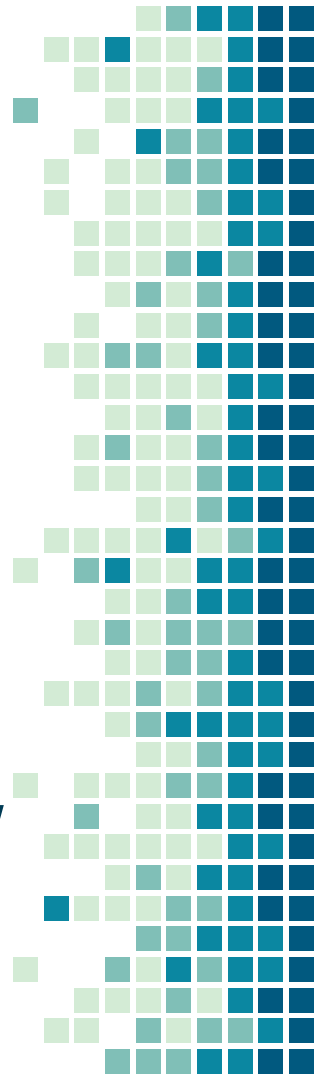
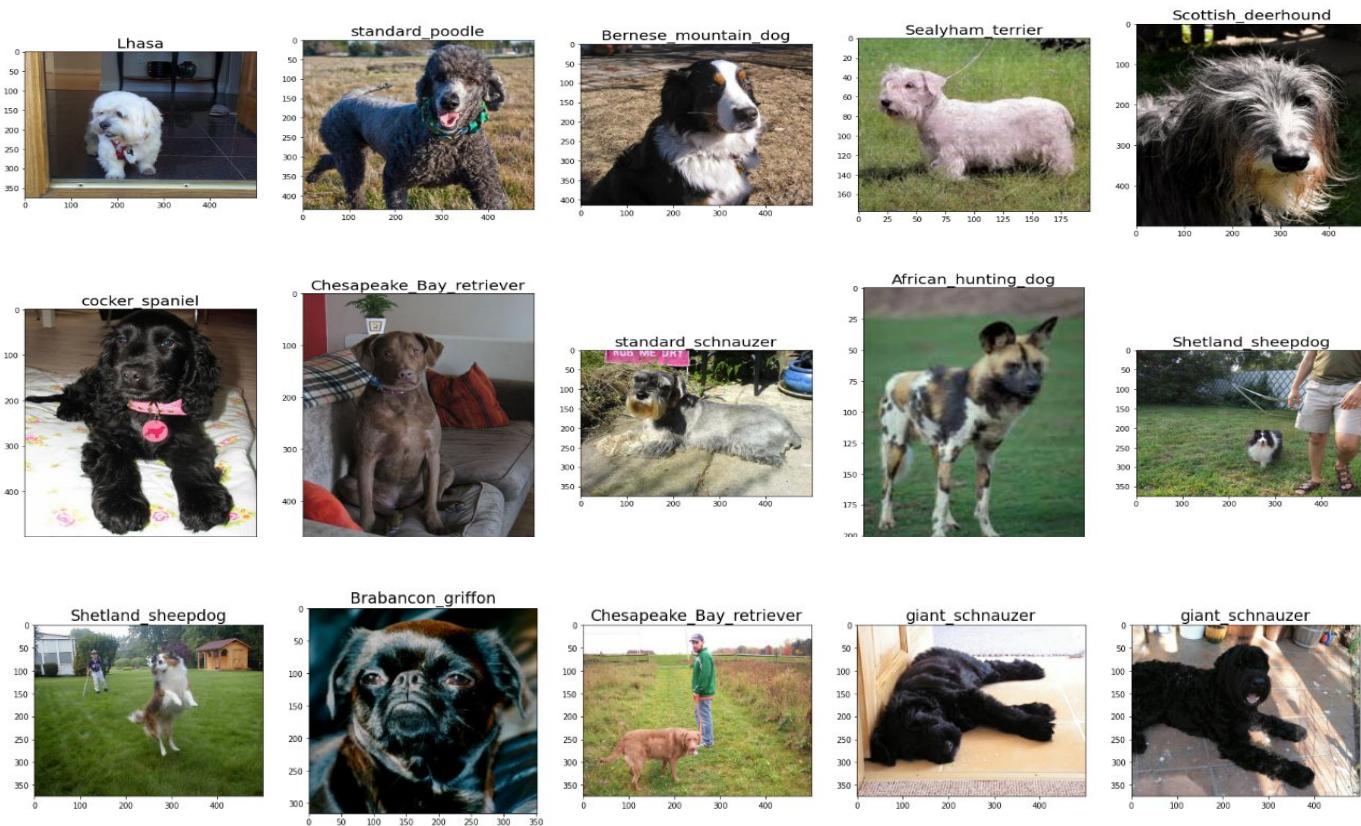


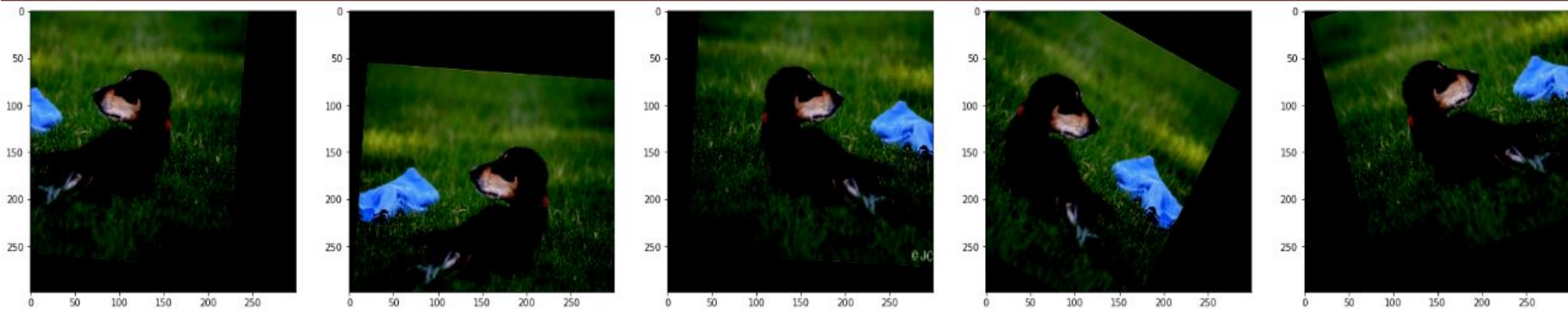
Image Examples of random breeds:



Data Augmentation

Augmenting photos, which basically rotates and flips images, was a good way of creating more data for the models to train off of. (Used for when you need more data)

Below is an example of some augmented versions of one of the photos:





MODELING

Time to create the best possible CNN
Model to determine what breeds of dog
are in these images....

Results, best parameters, etc.

10 Epochs:

299x299:

85.6% train & 88.6%
testing accuracies

180x180:

76.3% training and
77.3% testing
accuracies

Layers:

Inception V3

**Global Average
Pooling 2D**

Flatten

Dropout

**3 Dense layers (relu,
relu and softmax for
final layer to pick the
breed)**

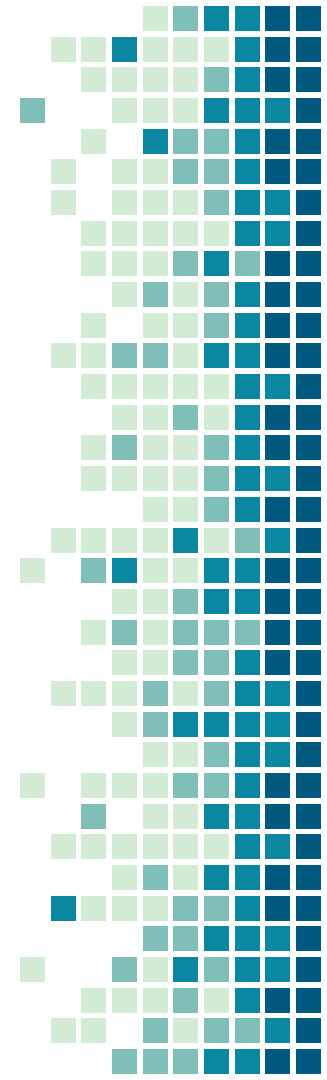
20 Epochs

299x299:

94.2% training &
92.7% testing
accuracies.

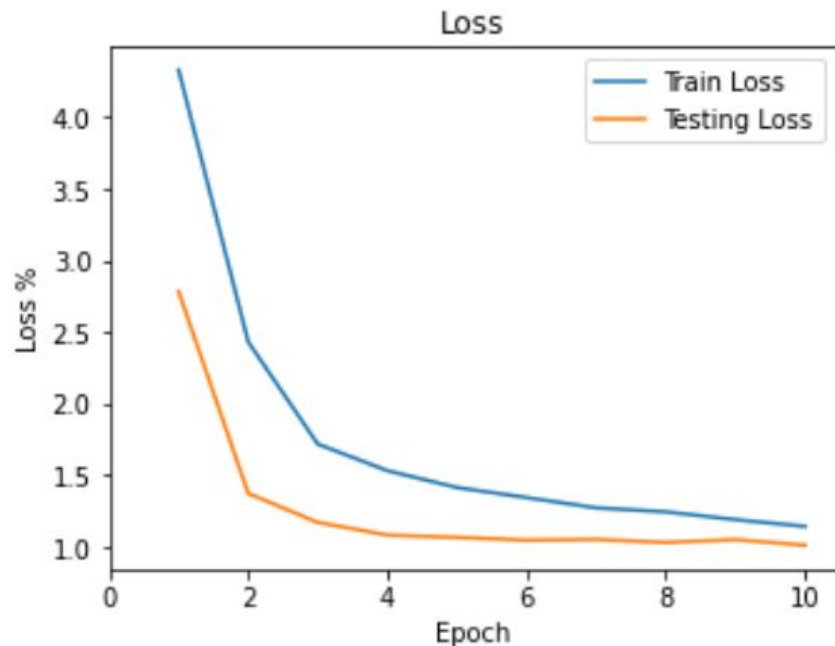
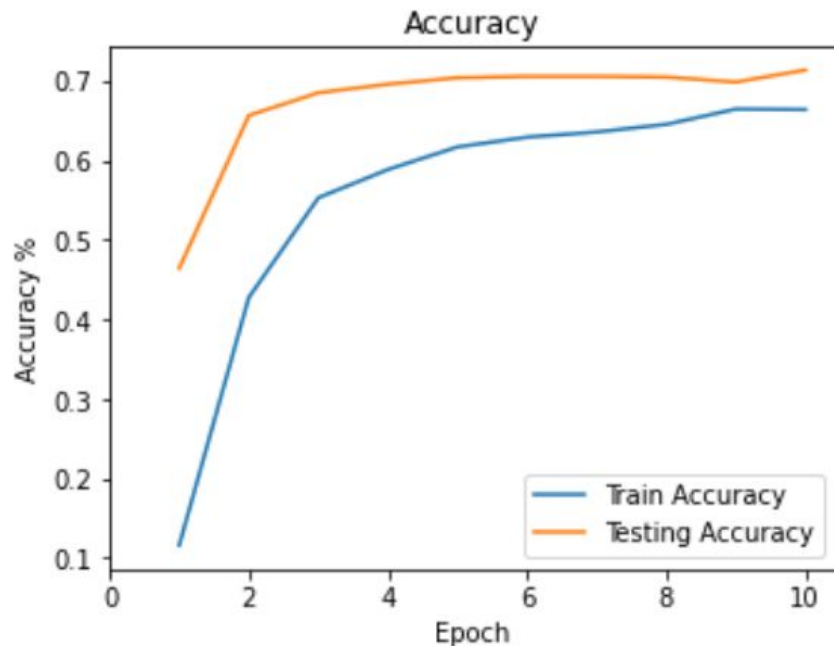
180x180:

81.92% training and
76.61 testing
accuracies

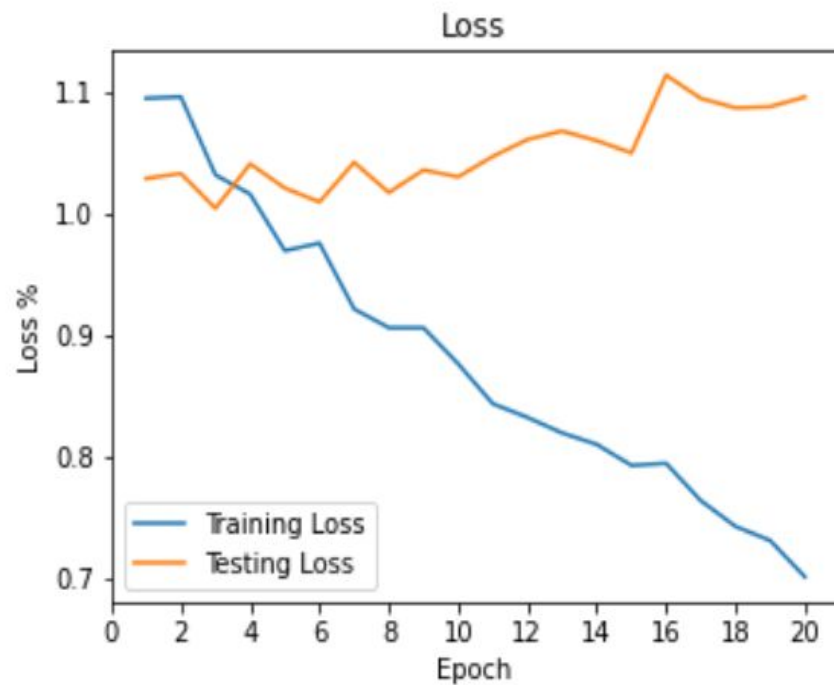
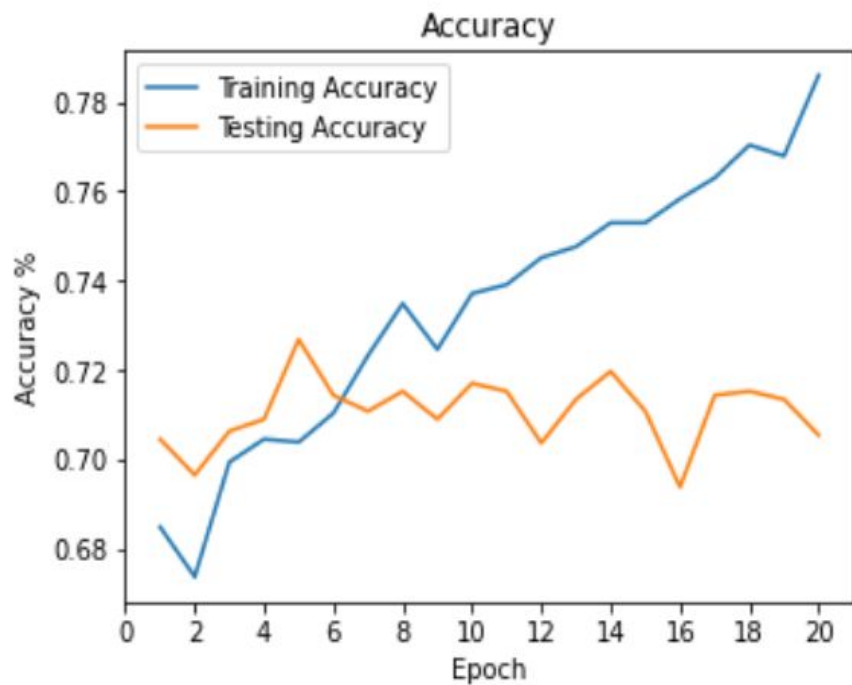


Results cont. – only 180 x 180 pixels

Learning Performance Over 10 Epochs

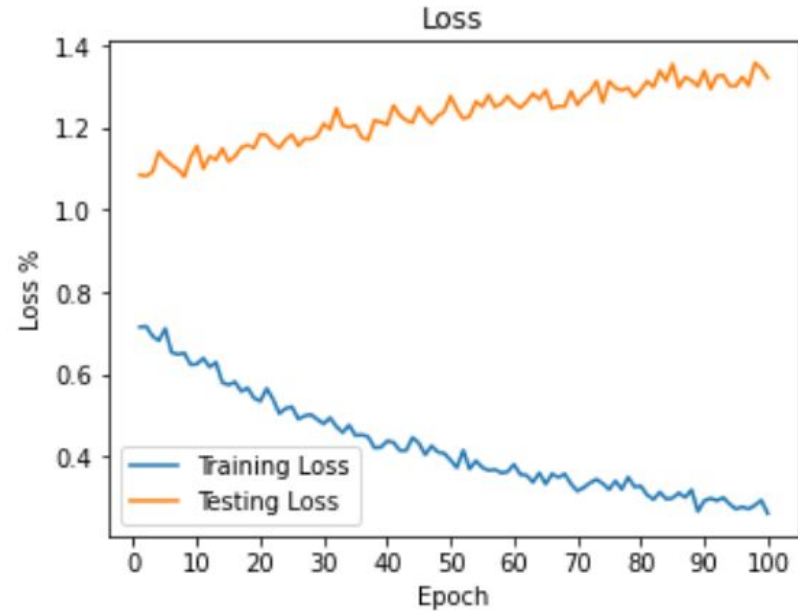
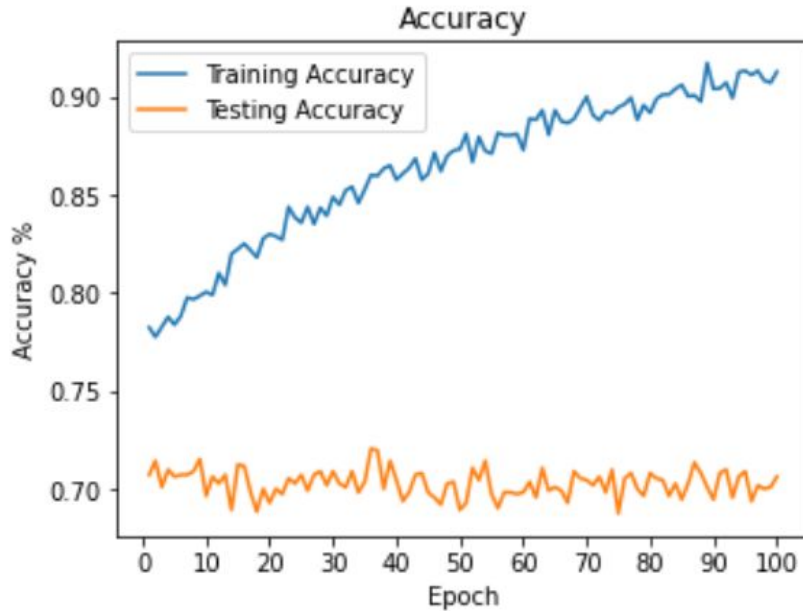


Learning Performance (20 Epochs)



Finally, at 100 Epochs

Learning Performance (100 Epochs)



(Using the same params - dims of 180 x 180) 91.27%

Training and 70.63% testing. Not much improvement/overfit

Examples of some predictions & errors made:

Actual: Rhodesian_ridgeback
Predicted: Rhodesian_ridgeback
Conf: 0.9988971



Actual: English_foxhound
Predicted: Walker_hound
Conf: 0.81851655



Actual: giant_schnauzer
Predicted: giant_schnauzer
Conf: 0.99751306



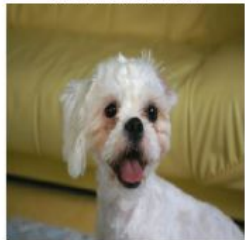
Actual: Bedlington_terrier
Predicted: Bedlington_terrier
Conf: 0.9978999



Actual: Sealyham_terrier
Predicted: Sealyham_terrier
Conf: 0.9645313



Actual: Maltese_dog
Predicted: Shih-Tzu
Conf: 0.9184978



Actual: Brabancon_griffon
Predicted: Brabancon_griffon
Conf: 0.9979175



Actual: Scottish_deerhound
Predicted: Irish_wolfhound
Conf: 0.5985619



Actual: Afghan_hound
Predicted: Afghan_hound
Conf: 0.98769623



Actual: Gordon_setter
Predicted: Gordon_setter
Conf: 0.9842406

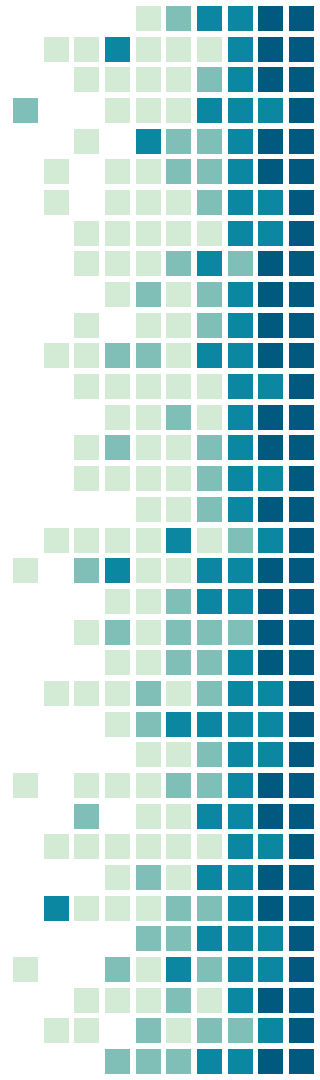


Conclusion/Future

After messing with many different models, InceptionV3 with Global Average Pooling seemed to perform the best.

In the future, I would like to (and will) continue tinker with not only the models/hyperparameters but also different ways of processing the images before sending them through the model, as all of that can have a huge impact.

Also: a Demo, as seen [HERE](#) (github user: stormy-ua)



THANK YOU FOR YOUR TIME!

Chris Burger
cmburger11@gmail.com
General Assembly

