

Name: Corbin W Gjerstad
Course: Machine Learning Engineer Nanodegree
Project: Capstone Project - CNN Dog Breed Classifier

This proposal is for an image classifier, specifically one specializing in identifying dog breeds with a few odd little twists added in. At its core it's a classic computer vision problem, how to take an image and enable a computer to extract data from said image in order to make complex and meaningful decisions.

Background:

Computer vision as a field has arguably been around since the 1960s being birthed at MIT(1). Though only in the past decade has it really started to gain momentum thanks to accomplishments like AlexNet. It is so versatile that the sky really is the limit. For example in 2019 archaeologists used CV AI to discover ancient artwork in peru with by analyzing satellite imagery(2). Or look at the medical field where they are working to train AI to analyze CT scans(3). I mention these things to illustrate my point of how far reaching and amazing Computer vision is.

Problem & Solution:

As much as I wish it was, my proposed project isn't nearly as grand as some of the above examples but non the less a interesting problem and a good stepping stone into the field. To get more specific the goal of this Project is to create a CNN which can take in images of dogs or humans as input. It will identify if it is looking at a human or a dog. Then guess the breed of dog, or in the case of a human the breed the subject most closely resembles. Note only the initial classifier that distinguishes dog or human is trained against human data. The breed classifying CNN will just be fed human images after its trained on dog images for fun to see what it says.

Data Sets

To start though I need my raw data. Thankfully, as this is one of the suggested capstone projects the dataset was provided and already split between train, test and validation. As to the contents we have 8531 dog images spread across 133 different breeds. The image sets for each classification are fairly balanced, averaging 45ish images per breed classification. In addition we have 13234 images of humans looking to belong to 5794 unique individuals. I will also note the images themselves are not uniform in size or quality. Which should result in a fairly broad base with which to let the CNN learn. At least from the perspective of a student project.

Data Set links:

Dogs - <https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip>

Humans - <https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/lfw.zip>

Evaluation Metrics:

As mentioned above the training set is relatively balanced. In turn I plan to evaluate the model based on its accuracy when classifying dog breeds. The goal is for the model to distinguish between humans and dogs when ran against the test data set. Then guess the breed of a given dog with at least 60% accuracy. These numbers are on a bit of a low side, but limited hardware and time is a concern when it comes to training the network.

Benchmark Model:

When it comes to finding a benchmark model I found a variety of sources online as this looks to be a fairly classic problem as an introduction to image classification. Issue I ran into is that they often used transfer learning from an established network such as VGG. In turn I wanted to find something which one was completely from scratch like my own project and used CNNs. I did ultimately find an article on medium (reference 4 for link) going over the creation of a dog breed classifier that meets the outlined criteria. Said model only hits a 44% accuracy which is lower than my own goal but as a whole is almost the same problem. Due note that the intent of not building the breed classifier is the challenge of seeing how far I can get starting from scratch.

Project Design:

As this is one of the proposed projects and even comes with a starting Jupyter notebook the workflow is already pretty much set already. As mentioned before the breed classifier will be built from scratch, but the detection of human or dog will be different. For identifying humans I will use a OpenCV version of Face detection using Haar Cascades(5). Then to tell if an image is a dog I will use a VGG-16 model pre-trained on ImageNet. These will be used to create base functions to decide if a image is a dog or a human.

As for Image preprocessing Haar needs the images to be grayscale. And VGG-16 requires they be 3 channel 224x224 images. So I'll need to handle accordingly. For my own CNN I currently intend to go the same route as used by VGG-16 but this may change after some experimentation.

After the functions to decide Dog or Human I will feed the image into my breed classifying CNN as a final result deciding the breed of the dog or what breed a human most closely resembles.

References

- 1-
<https://hackernoon.com/a-brief-history-of-computer-vision-and-convolutional-neural-networks-8fe8aacc79f3>
- 2-<https://www.smithsonianmag.com/smart-news/ai-helps-identify-1-143-new-nazca-lines-180973621/>
- 3-<https://healthtechmagazine.net/article/2019/01/computer-vision-healthcare-what-it-can-offer-providers-perfcon>
- 4-<https://medium.com/nanonets/how-to-easily-build-a-dog-breed-image-classification-model-2fd214419cde>
- 5-https://docs.opencv.org/trunk/d7/d8b/tutorial_py_face_detection.html