Dog Breed Classifier Proposal

Domain Background:

Image classification is a widely used and extremely useful technique that forms the basis of the field of computer vision. Other computer vision techniques such as localization, image segmentation and object detection rely on image classification.

In practice, image classification is used in autopilot in the aviation and automotive industries, in manufacturing settings, to detect proper PPE is being worn, and in medicine to diagnose potential illness in medical imagery with greater accuracy than the human eye.

Detecting dog breeds is a popular image classification problem. There is a Kaggle competition with almost 9,000 entries, where the goal is to minimize multiclass loss[1].

Problem Statement:

The classification of dog breeds can be problematic. Images of dogs can be indistinguishable by breed to the human eye, a dog breed classifier mitigates this issue.

The objective of this project is to develop an app that can take an image as an input and determine whether or not the supplied image is a dog. If so, the app should be able to determine what pure breed of dog is shown in the image.

Datasets and Input:

Two large datasets are to be used in this project. A human dataset and a dog dataset. These datasets are provided by Udacity. The dog dataset is a subset of ImageNet[2] used as the dataset for Kaggle competition.

Dog Dataset: There are 8351 images, split into train, test and validation with 6,680, 836 and 835 images in each respectively. Each subset contains 133 different folders corresponding to the breed of dog. The breakdown of images per dog breed is not uniform.

Human Dataset: There are 13233 images, the dataset is split into 5750 folders, corresponding to different people. Once again, the breakdown of images per human is not uniform.

Solution Statement:

The proposed solution to this multiclassification problem is as follows. Obtain a pretrained Convolutional Neural Network (CNN), VGG16, to detect the presence of a dog in image. Load pre-trained Haar feature-based cascade classifier from Open CV. This will be used to detect the presence of a human face in an image.

Train dog breed classification model using transfer learning on a pre-trained CNN Resnet50. This model will take an image as an input and return an output labelling the image as one of 133 dog breeds.

An image is passed in to the application, our VGG16 CNN determines whether or not the image contains a dog, if so, we pass this image to our transfer learning model. Here the breed of dog in the image is outputted. If there is no dog present, the input image is passed to the human face detector, if a human face is detected, the app will output what breed of dog the human face looks like. Otherwise, an error is returned.

Benchmark Model:

To benchmark using Dog Breed Classification models on the Kaggle competition page is somewhat unrealistic. Multiple models have returned 0 multiclass loss. A compromise would be to say the model is expected to have an accuracy above 70%.

Evaluation Metrics:

The model should be evaluated as mentioned before using accuracy (what percentage of images in test dataset were correctly classified) and also using Multi Class Log Loss. Multi Class Log Loss, such as cross entropy loss is as important if not more important than accuracy as we are using a dataset with uneven numbers of images belonging to each class.

Project Design:

1. Import necessary datasets and libraries, split datasets into train, validation and test. Pre-process data as required for each model
2. Import pre-trained Haar classifier and use to detect humans in datasets
3. Import pre-trained VGG16 model and use to detect dogs in datasets
4. Create a CNN classifier from scratch and justify your choice of model architecture
5. Decide on pre-trained model, once trained, test accuracy and loss
6. Use models trained in previous steps to write algorithm that tests firstly for the presence of a dog, returning the breed if true. If a dog is not detected, the model will check for human features. If human features are present, then return what breed the human looks like, otherwise, return an error.

References:

1. Dog Breed Identification, Kaggle https://www.kaggle.com/c/dog-breed-identification/overview
2. ImageNet, https://www.image-net.org/
3. Resnet, https://arxiv.org/abs/1512.03385
4. Haar feature-based cascade classifier, https://github.com/opencv/opencv/tree/master/data/haarcascades