Capstone Proposal

Domain background
This project is an image classification project to identify the breed of a dog given an image of the dog. Identifying a dog's breed helps buyers find dogs with preferred characteristics, like growth size, whether they are suited to indoors or outdoors and the toes and quantity of food they eat. It can also be used for targeted advertising. Food and toys for specific breeds of dogs can be advertised to users with those dogs.
Some work has been done by researchers on classifying images of animals. This research paper for example uses SVMs and MLPs to classify animals into either predators or pets. Despite this being less complex than identifying individual breeds, a high precision was achieved without utilizing CNNs. This research paper outlines using CNNs for identifying different wild animals (elephants and cheetahs).

Problem Statement
Given an image of a dog, identify an estimate of the breed. If the image is an image of a human, identify the resembling dog breed.

Datasets and Inputs
This project will use this dog dataset (8351 images) and this human dataset (13233 images) from Udacity. The data has been split into training, test and validation data. The images are colored and their dimensions and aspect ratios are not consistent. There is also a significant variation in the animal poses as well as the lighting conditions. The training data is fairly balanced with most breeds having between 40-60 images. However, there are some breeds with 60 and 70+ images and a few breeds like Xoloitzcuintli and Plott with less than 30 images.

Solution Statement
The problem will be solved using Convolutional Neural Networks. We will train the CNN using the training data indicated above and use the trained model to predict dog breeds.

Benchmark Model
There are some existing models used for similar tasks like VGG16 and ResNet50. We will benchmark our model against these existing models.

Evaluation Metrics
Our model will be assessed by its accuracy, precision and recall.

Project Design
The project will make use of Convolutional Neural Networks.
Firstly, each image's RGB values will have to be normalized from a range of 0-255 to a range of 0-1.
I would then perform data augmentation on the data using random rotations of 10 degrees before transforming it into tensor data types and loaded in batches using PyTorch's DataLoader.

The data is already split into training, validation and test sets so I would go ahead and implement my CNN. I would start with defining a three layer CNN with a 2x2 max pooling layer. A dropout layer will be required to prevent overfitting. Finally, two fully connected linear layers at the output. The last linear layer will have 133 class scores as outputs (corresponding to the breeds being classified). Depending on the results of my first model, I will adjust the number of layers to optimize the network.