Final Project: Proposal

Rishabh Kaushal, Aidan Vickars, Anant Awasthy, Karthik Srinatha

1. List 3 questions that you intend to answer (1 point)

- Can you leverage the entire body of two dogs to determine if they are the same dog using computer vision techniques (Current work we have seen only uses the face)? We feel that by using only the face, useful information is being discarded and could result in a significant improvement compared to current work.
- Can we construct an entire deep learning pipeline that can accurately match “lost” dogs with “found” dogs in both an extremely large (i.e. an unlikely environment) and a smaller and more real world environment (i.e. a likely environment)? The core difference here, is that in the real world, “how many dogs are currently lost in the Vancouver area at any one time?”.
- Can we construct a proficient dog detector that can accurately create bounding boxes and extract images of dogs?

2. List all the datasets you intend to use (1 point)

- Stanford Dog Dataset
- DogFaceNet Dataset
- Pet Finder Data Scraping
- OpenImagesDataset

3. Give us a rough idea on how you plan to use the datasets to answer these questions. (2 points)

- Data Collection: Where/how to get data

Collecting the data is a fairly simple process. We have 2 datasets already:

- Stanford Dog Dataset
  - Contains 120 different dog breeds, for a total of 20 580 dog images. Each image has an annotated bounding box.
- DogFaceNet Dataset
  - Contains 4(ish) images of 1425 individual dogs. Each image is of a dogs face.
- Pet Finder Data Scraping
  - We are going to scrape Pet Finder to get multiple images of individual dogs that contains there face as well as their body. This is because the DogFaceNet Dataset contains images only of a dogs face where as Pet Finder Data Scraping images will contain a dogs entire body. We would like to train our dog

comparison model on the entire body of a dog rather than just its face since the body has significant information when comparing two dogs to determine if they are the same.

- OpenImagesDataset
    - Contains thousands of images contain dogs and a variety of others with annoted bounding box. We are looking to use this to supplement the Stanford Dog Dataset
- Data Exploration: Do you need to conduct EDA in order to understand the data?
    - None, its self-explanatory.
- Data Cleaning: Do you need to clean data? How to clean them?
    - We will need to normalize the images in some capacity. I.e. cropping them.
        - This is particularly important since we will be developing an app which can have images coming from a variety of phones (different resolutions).
        - However, it should be noted that we may have some flexibility here, as we are aiming to use the YoloV2 algorithm which theoretically is size invariant with respect to image inputs (to a point). Though, this will depend significantly on our choice of a feature extractor in the model (i.e. which pre-trained model we use as our base feature extractor).
    - There are a variety of tools that do this (ex: OpenCV).
- Data Integration: Do you need to integrate data from multiple sources?
    - Yes, we will need to integrate data from multiple sources. However, integration will be fairly streamlined as all pictures have color. The most significant integration task will be defining an image size standard (ex: 36 by 36) and cropping all images accordingly.
- Data Analysis: Do you need to analyze data? What analysis do you intend to do? (e.g., SQL, Statistics, Deep Learning) How to evaluate your analysis results? (e.g., evaluation metrics, confidence intervals, benchmark)

Significant analysis in 4 parts:

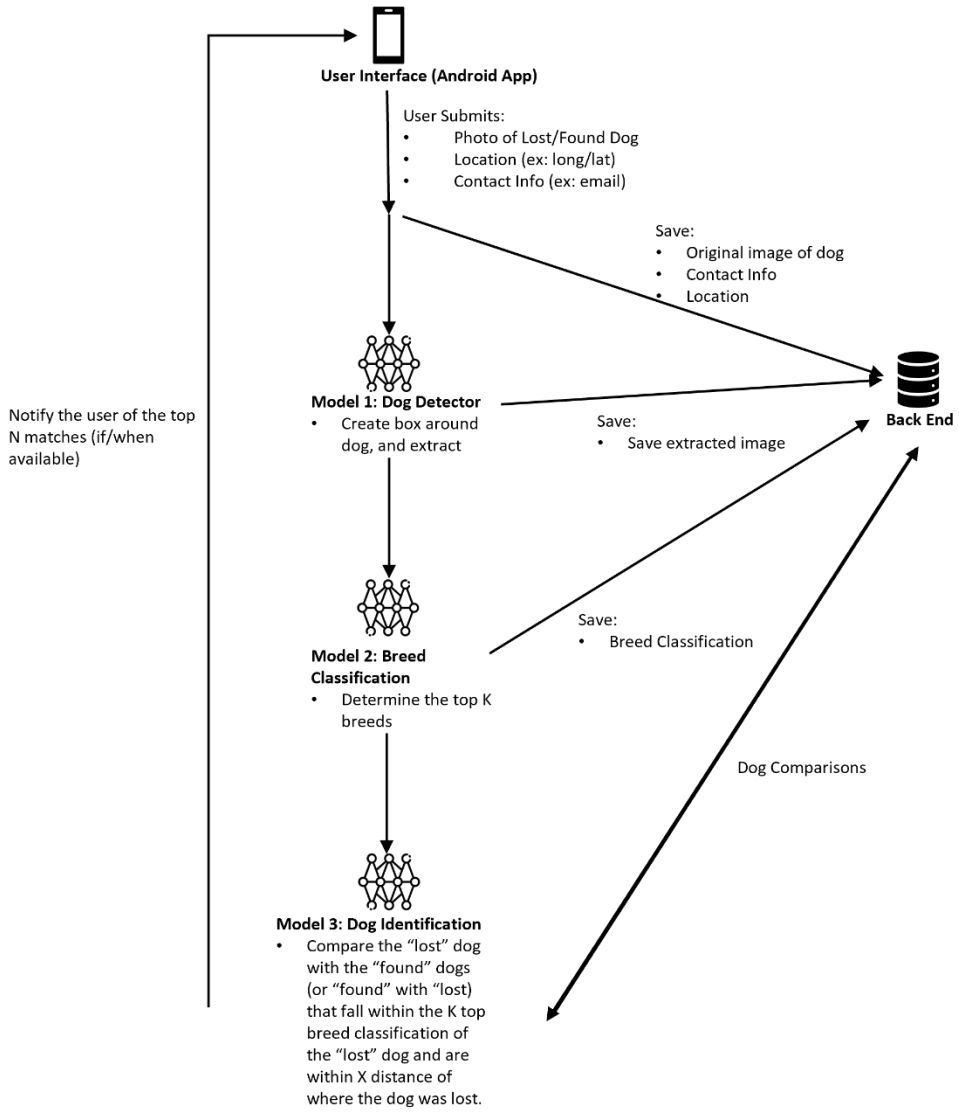
- Model 1: Dog Detector
    - The first model we are looking to build will given an image of a dog(s) will extract the bounding box of the dog(s). This model will serve as a normalization model for the subsequent models so that they are only working with an image of the actual dog in question and no other features (i.e. a person, a second dog etc).
    - Will use a convolutional neural network in combination via transfer learning with yoloV2. Since we are aiming to use yoloV2, we will be computing our own anchor boxes to use as well, likely via K-means clustering as was done in the original paper.
- Model 2: Breed Classification
    - The second model will be a straightforward breed classification model.
    - Will use a simple convolution network likely using transfer learning.
- Model 3: Dog Identification
    - The third model will be a dog identification model that given two images of two dogs, it will output the probability that the dogs are the same.

        - Will use a convolution network likely using transfer learning with triplet loss.
    - Model Analyses
        - One of the most significant aspects of this will be to assess the accuracy of each model both individually and together (see diagram of application below).
        - Individual Model Analysis
            - We will assess the accuracy of the models individually. The assessment of models 2 and 3 will of course be the accuracy of the model (I.e. how accurately does it assess the breed and how accurately does it assess if two dogs are the same dog). Additional analysis will be done to see where the model performs well and where it does not (ex: may perform worse on specific breeds).
            - For model 1, the assessment here will be how well it performs object detection on dogs. An example evaluation metric is the MAP (mean average precision).
        - All model analysis
            - We will need to assess on a large scale how well the entire pipeline is able to match “lost” dogs with “found” dogs (and vice versa) as a function of tunable parameters in the pipeline. For instance:
                - Suppose person A loses their dog, they will submit an image of their dog which will then be piped through model 1 (extract image of just the dog in question to use in subsequent models)
                - Then the extracted image will be piped through model 2 (determine the top K breeds that the model determines the dog belongs to)
                - Then the extracted image from model 1 of person A’s dog will be compared to all dogs that have been submitted as “found” that fall within both the top K breeds as determined by model 2 and were found within X distance from where person A’s dog was lost.
                - Then we will return (if any) the top N matches of “found” dogs likely to be person A’s dog.

              By tuning the K number of breeds and the X distance variables that dictate the number of comparisons that are made as well as how many potential matches are returned, we can determine the minimum number of total comparisons and matches returned that need to be made while still finding the “lost” dog among the “found” dogs. To do this we will need to leverage the validation and test data to construct a large-scale simulation (i.e., a multitude of both “lost” and “found” dogs).

- Data Product: What data product do you want to build? (e.g., visualization, an interactive web app, a report, a model)

Our idea is to create an app to match “lost” dogs with “found” dogs and vice versa. We are looking to incorporate a signification degree of deep learning (computer vision to be specific). We feel this is a significant project that is difficult for us but achievable. This is because from the app development perspective, within the group we have significant experience in both app building and back-end construction as well and do not anticipate significant difficulties or roadblocks. With respect to model construction, this provides a good balance for us with respect to difficulty. This is because as a group, we have minimal experience constructing computer vision models and deploying them in production. However, the models that we will be constructing are not original (individually, does not appear to have been done together in a production environment) and as a result there are resources that will be of great use for us. It is in this sense that this project allows us to experiment with constructing and deploying deep learning in a production environment, while also having a strong underlay of previous research to support us and ensure success. To get a clearer idea of the entire app, please see the diagram below:



4. Think about that once your project is complete, what impacts it can make. Pick up the greatest one and write it down. (1 point)

The most significant impact our project will make is the ability to reunite people with their lost dogs. Our aim once the app is completed, is to contact the BC SPCA and ask if they have any interest in adopting the app.