Capstone Proposal
DATS 6501 – Summer 2020
Alex Cohen

For this capstone project, I want to examine the effect that ambiguity and distortion has on common computer vision tasks – specifically object detection. Most existing computer vision models are trained on clear, often high-resolution images taken from stock photo libraries, with a number of common image classification models being trained on the ILSVRC-2012 dataset (frequently referred to as the Imagenet dataset). Similarly, object detection models are also frequently trained on high resolution, unambiguous images such as the PASCAL VOC or Microsoft COCO dataset, with any manipulation often restricted to image rotation, cropping, or color correction. However, little work has been done to understand the effect that additional, extreme image distortion (such as blurring or noise addition) has on object detection models. I want to better understand how these additional forms of image manipulation and distortion impact the performance of these models and attempt to find strategies to mitigate the (likely) decrease in performance.

There are two standard datasets for object detection: Microsoft's COCO (Common Objects in Context), and the PASCAL VOC (Visual Object Classification) dataset. As most object detection models are trained on the COCO dataset, I will be using this dataset for evaluation, image distortion, and any model training throughout the project. It contains over 330,000 images from 80 categories (however the 2017 train/val set includes ~140,000 training and 5,000 validation images), compared to the 20 object classes and 11,540 images of the VOC dataset. The larger size of the COCO dataset and the higher number of image categories, combined with the fact that many pretrained object detection models (such as those in the torchvision model library) were developed using the COCO dataset mean it is likely the better dataset to use for this task. Additionally, the COCO API (pycocotools pip package) will allow for easier JSON parsing and label/bounding box extraction.

To solve the issue of understanding how models respond to warped or distorted image input, some data engineering paired with deep learning techniques will be used to understand and act upon these Convolutional Neural Networks (CNNs). The set of tasks to accomplish this project will require image manipulation (likely using the openCV or Pillow packages to perform these transformations), the understanding of predicted bounding boxes and CNN model architecture (using PyTorch), the potential training of new neural networks (again using PyTorch and deep learning techniques), as well as data engineering to understand how to most effectively store this collection of images and results. All of these techniques will be combined to better generate and understand the project results.

For the deep learning framework, PyTorch will be used to test the existing object detection models as well as develop any new object detection models. PyTorch is the framework of choice as I have experience using it for image classification using CNNs during the Machine Learning 2 course, as well as the torchvision package which provides access to some of the pretrained object detection models. Additionally, some of the pretrained models that have not been implemented in the torchvision package were written using PyTorch, so it will facilitate an easier implementation if the scope of the project expands outside of the torchvision models.

To better understand the specific methodology that will need to be used for this project will require a combination of background material on training and evaluating object detection models, as

well as any research previously done on the response to the distortion of image inputs to CNNs. There are numerous papers explaining SotA object detection models to review, as well as internet articles and PyTorch documentation to better understand the actual training and evaluation of the object detection models. The core of the work is based of the paper "ON CLASSIFICATION OF DISTORTED IMAGES WITH DEEP CONVOLUTIONAL NEURAL NETWORKS", which undertook the same task with image classification models. This will serve as a starting point for much of the methodology and image manipulation techniques to be used, whereas the Microsoft COCO GitHub repository will help with the dataset loading and evaluation. Other GitHub users that have implemented object detection models will serve as reference points in the training and prediction stages and credited accordingly.

Evaluation of my work will involve the calculation of overlapping bounding boxes and the Intersection Over Union score (often referred to as the Jaccard index). This will be used to compute the area of the predicted bounding box(es) and the corresponding ground truth bounding box(es) to estimate the fraction of the area that was correctly predicted by the model. The higher the score, the better our estimated bounding box. Success for this project will be to understand any potential decline in the Jaccard Score of the pretrained object detection models when applied to distorted images, as well as (hopefully) any increase to the Jaccard Scores of any fine-tuned or new object detection model when applied to a set of distorted input images. Additionally, the Microsoft COCO API provides precision and recall metrics that will need further exploration before incorporation into the project.

As the semester is approximately 14 weeks, the schedule for completing the project is the following:

Weeks 1-3:

- Finalizing the project proposal and scope of the project
- Understanding the dataset and existing literature on object detection models
- Building prototype code to explore existing bounding box predictions

Weeks 4-6:

- Develop evaluation pipeline for image transformations, prediction, and output intersection calculations
- Evaluate various image distortion techniques on predictions and output intersection calculations
- (Potentially) implement models from sources outside torchvision (some SSDs)
- Begin journal writing around distorted predictions for existing models

Weeks 7-11:

- Fine tune existing models using transfer learning techniques to measure improvement in distorted image object detection
- Attempt development of a novel model trained exclusively on distorted images
- Compile results and begin journal writing around novel/tuned model results

Weeks 12-14:

- Create draft presentation and refine to final presentation
- Continue and finalize the final project report
- Finalize the journal article and submit for review