Special Topics - MSAI 495

Project Proposal

“On-road object detection for self-driving cars”

February 7th, 2022

Professor
Reda Al-Bahrani

Students
Ayushi Mishra
Preetham Pareddy
Ana Cheyre

Problem Statement

Self driving cars might be a luxury in the current world but the technology is rapidly changing to be available for everyone. They are useful because ideally, machines don't make the mistakes humans do, so there would be less accidents like crashes. It will also save a lot of time for humans, making their life much more efficient. Computer vision is a critical component for the functioning of such cars. There is an absolute need to create better models that minimize, if not completely eradicate, danger on the road. This is the reason we chose to work on object detection on the road. The goal of this project is to develop a computer vision model that solves some of the multiple tasks that the system of self-driving cars must execute to be able to drive autonomously. This refers to the detection of diverse elements found on the streets so that the car is able to identify if this type of obstacle is on the road or not.

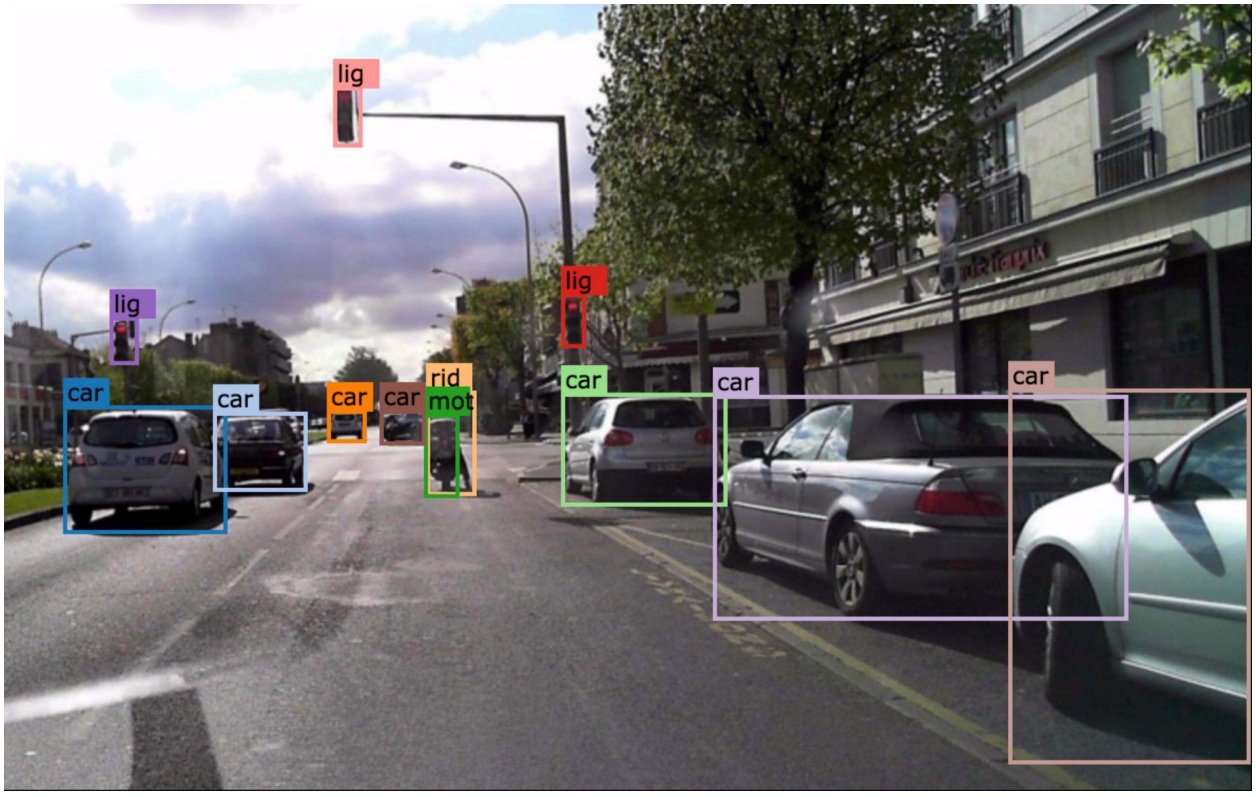
Dataset

The dataset we chose to move forward with is the “BDD100K Adas dataset” that contains around 120 million images of roads, with signs and different objects on them, obtained from 100,000 videos on roads. The data were collected from diverse locations in the United States, also covering different weather conditions, including sunny, overcast, and rainy, as well as different times of the day including daytime and nighttime.

The images contain labels for 10 different types of objects: traffic light, traffic sign, car, person, bus, truck, rider, bike, motor and train. Each image has object bounding boxes, and these boxes contain the respective label.

Label	Frequency
Traffic light	265,906
Traffic sign	343,777
Car	1,021,857
Person	129,262
Bus	16,505
Truck	42,963
Rider	6,461
Bike	10,229
Motor	4,296
Train	179

Frequency of labels in database



Example of labeled image from database

This extensive database with images of the streets is perfect for training the model that we will develop, the images would be very similar to what a camera from a self driving car would see, and it will be able to identify 10 different objects, which correspond to the most frequent objects that are found on the streets when driving.

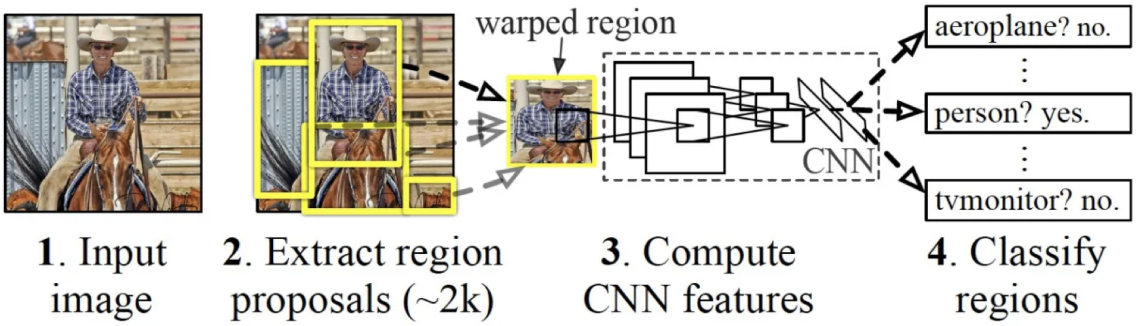
Proposed Solution

The networks that will be used to develop the model will be convolutional neural networks, specifically R-CNN (Region based Convolutional Neural Networks), which is an algorithm that proposes multiple boxes in the image and checks if any of these boxes contain any object, so then it will work with a smaller section of the image.

The R-CNN model is composed of three components:

1. A **region selector** uses a “selective search,” algorithm that finds regions of pixels in the image that might represent objects, also called “regions of interest” (RoI). The region selector generates around 2,000 regions of interest for each image.
2. The Regions of interest are warped into a predefined size and passed on to a **convolutional neural network (CNN)**. The CNN processes every region separately and extracts the features through a series of convolution operations. The CNN uses fully connected layers to encode the feature maps into a single-dimensional vector of numerical values.

3. A **classifier machine learning model** maps the encoded features obtained from the CNN to the output classes. The classifier has a separate output class for “background,” which corresponds to anything that isn’t an object.



R-CNN architecture

We plan to dive into AlexNet convolutional neural network for feature extraction since it reduces the parameters proportion of the full connection layer, and simplifies the parameters number of the model along with increasing network training speed and model classification accuracy over other models. We will be using a support vector machine (SVM) for classification.

Note: Mask Region-based Convolutional Neural Networks (Mask R-CNN) and You Only Look Once v4 (YOLOv4) are some image processing algorithms we might consider in the future based on output metrics and progress of the current proposed algorithm. We might venture down this path to improve speed and accuracy. The main improvement in YOLO particularly from a theoretical standpoint is the integration of the entire object detection and classification process in a single network. Instead of extracting features and regions separately, YOLO performs everything in a single pass through a single network, hence the name “You Only Look Once.”

We also plan to familiarize ourselves with the other existing models like AttentionNet and OverFeat. The main idea of OverFeat is to (i) do image classification at different locations on regions of multiple scales of the image in a sliding window fashion, and (ii) predict the bounding box locations with a regressor trained on top of the same convolution layers. It is pretty similar to Alexnet. In the case of AttentionNet, the model casts an object detection problem as an iterative classification problem, which is the most suitable form of a CNN. AttentionNet provides quantized weak directions pointing to a target object and iterative predictions from AttentionNet converge to an accurate object boundary box. We want to explore these architectures to determine what works best on our data and possibly find out why.

The metrics used for the object detection task would be: Precision (P) and recall (R) for each object category, in this case we will be particularly interested in the amount of false negatives (looking for high recall), because in a self driving car there is no space for errors because it can risk people's lives. Also the mean average precision (MAP) which is a very common metric in object recognition as it measure how good the model is at performing the classification of objects. Finally, we will keep track of accuracy and the training loss after each iteration.