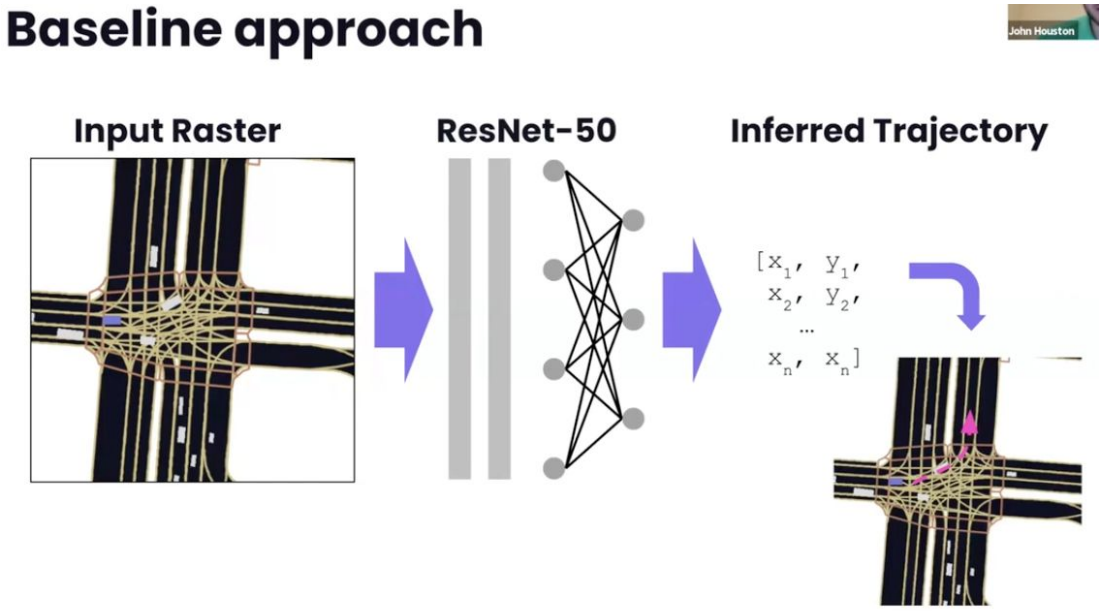
Midterm Update:

Abstract: Currently, many self-driving cars rely on rule based models to make decisions about about when, where, and how to maneuver; however, the real world often doesn't follow these rules and comes with many uncertainties. In order to make a future where self-driving cars make transportation safer, environment-friendly and more accessible for everyone, they must be able to reliably predict the movement of traffic agents around the autonomous vehicle (AV), such as cars, cyclists, and pedestrians. We are using a multimodal model to generate multiple hypotheses (up to 3) - further described by a confidence vector - to predict the trajectories of both an AV and other traffic participants in a given scene.

Teaser figure:



Introduction:

Our group aims to help make autonomous vehicles (AV) safer for our communities. As AVs have become more prevalent in our society. Companies such as Tesla, Waymo, Lyft and several others are working diligently to create Level 5 (full self driving/no human input) autonomous vehicles for the world to enjoy; however, no task like this has ever been done before and safety is of the utmost importance. Thus, fast and accurate results are a must for any autonomous vehicle. None of the members have previous experience in the field of autonomous vehicles; however, we have several members who have experience with the methods and procedures used to create and train the model.

Approach:
We chose to use two existing models for our baseline and development. Our baseline model is based on the Resnet-18 architecture while the model we wish to use for development is based on Sandler et al's (2018) MobileNet-v2. We chose a pretrained Resnet-18 model because training a model from scratch on all the data would take an enormous amount of time and it was the fastest way to get started in the limited amount of time we have for the project. The Resnet-18 model takes bird's-eye view (BEV) rasters, described in section 3.1.1 of Djuric et al's paper (2020), of an autonomous vehicle's surroundings in a given scene along with 10 frames of historical data, each frame occurring 0.1s before the other, as input. It outputs a confidence vector of predicted trajectories for the AV and the other traffic actors in the given scene. We chose MobileNet-v2 as our development model because it has been shown to be effective by Djuric et al (2020). Our development model takes in the BEV rasters for every actor in each frame, along with the encoded state information (velocity, acceleration, and heading change rate) for the actor captured in the dataset. Djuric et al (2020) has shown that while the BEV rasters include a wealth of semantic information for the CNN models to learn from, including state information along with the rasters significantly improves model performance. The MobileNet-v2 model will output the same data as the Resnet-18 for comparison. Most of the obstacles faced have been grasping the complex ideas conveyed in the relevant literature and understanding what's captured in the dataset and how to then process and feed that data into our model. Luckily, there are several forums, many of which published by employees at Lyft themselves, which have described the dataset in depth and have provided several examples on working with and visualizing the data.

Experiments and results:
The Resnet-18 model is trained on 100 input scenes, 24,838 frames, and 1,893,736 agents, which is a significant subset of our extremely large data set, allowing us to run our model in a short period of time: roughly 3 minutes and 40 seconds. We chose the "adam" optimizer for our model because this model was specifically designed to deal with training deep neural networks to leverage the power of adaptive learning rates to find the individual learning rates for each parameter. Our loss function was the negative log likelihood (NLL) of the probability of the prediction assigned to the actual trajectory. NLL works well since it works by giving a "happiness" value. So, when our confidence is high for our correct prediction then our unhappiness is low, but when the network assigns low confidence to the correct class, the unhappiness is high.

Qualitative results:

Conclusion and future work:

In the future we hope to develop a custom baseline model such as one that uses an Unscented Kalman Filter in order to mirror the practices in the literature. In addition we would like to try various settings for our hyperparameters. As currently we are simply using 2 epochs with 80 iterations. We can change these settings as well as our optimizers, loss function schedule learning rate and much more in hopes to increase the accuracy. Lastly, we want to train our model on the full set of data as we have only been using a small subset for training purposes.

References:

Djuric, Nemanja, et al. "Uncertainty-aware short-term motion prediction of traffic actors for autonomous driving." *The IEEE Winter Conference on Applications of Computer Vision*. 2020

Houston, John, et al. "One Thousand and One Hours: Self-driving Motion Prediction Dataset." arXiv preprint arXiv:2006.14480 (2020).

Sandler, Mark, et al. "Mobilenetv2: Inverted residuals and linear bottlenecks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.