Team VCR's Approach to NeurIPS Driving SMARTS Competition

Yuan Tian, Team VCR
University of British Columbia
yuant426@alumni.ubc.ca

Outline

1. Environment Setup
    a. Agent Action Space
    b. Agent Observation Space
2. Methodology
    a. Overall Workflow
    b. Trajectory Planning
    c. Safety Score
    d. Speed Selection
3. Future Work

1. Environment Setup: Agent Action Space

$$a = [x, y, \theta, \delta t]$$

Table 1: Action Space

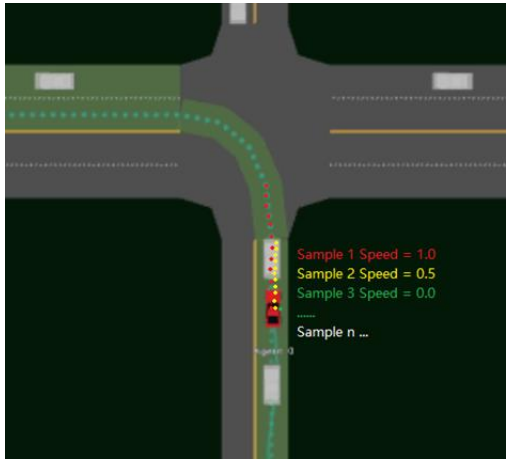
Name	Min	Max
x	-10^{10}	10^{10}
y	-10^{10}	10^{10}
θ	$-\pi$	π
δt	0.1	0.1

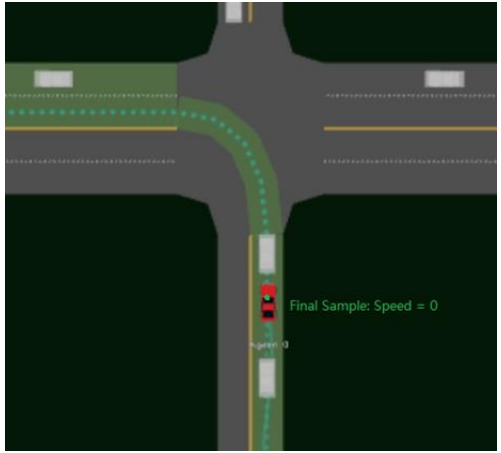
1. Environment Setup: Agent Observation Space

Name	Space	Description
ego[pos]	$[x_e, y_e, 0]$	The current location of the ego car with respect to the map coordinate system.
ego[heading]	θ_e	The current heading of the ego car with respect to the map coordinate system.
ego[lane index]	i	The current lane index of the ego car.
waypoints	$[[w_{11}, w_{12}, ..., w_{120}]], ..., [w_{m1}, w_{m2}, ..., w_{m20}]]]$	The next 20 way points ahead of the ego car for m paths.
waypoint[pos]	$[x_w, y_w, \theta_w]$	A way point contains its x, y coordinates and the heading.
waypoints[lane index]	$[[l_{11}, l_{12}, ..., l_{120}], ..., [l_{m1}, l_{m2}...]]$	Each way point's lane index
mission	$[x_m, y_m]$	The final goal position for the ego car.
ego's bird view	images [3, 256, 256]	The overhead bird view centered at the ego car for the box view within 50 meters.

2. Methodology: Overall Workflow

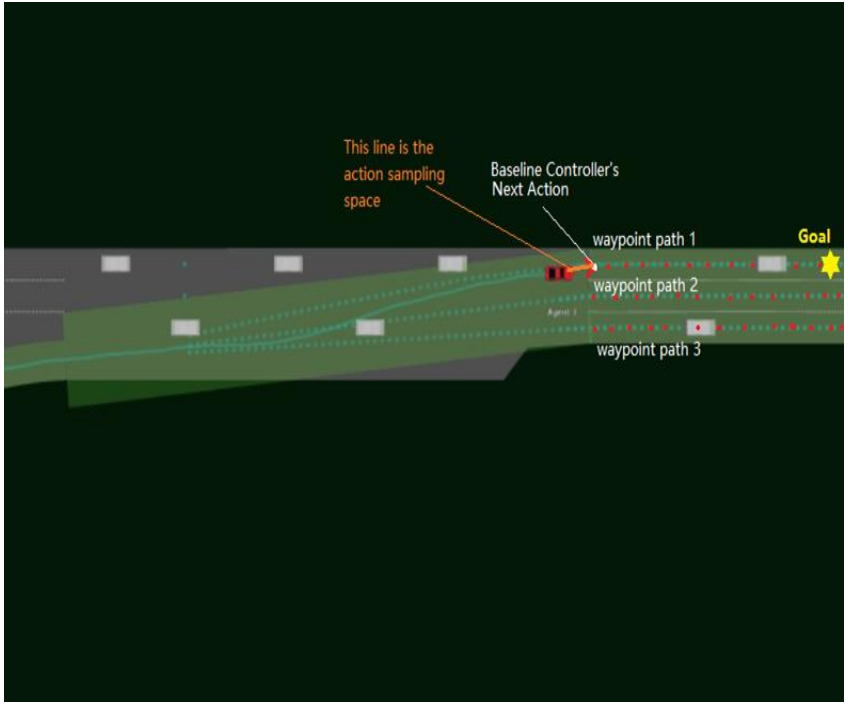
Original Design





Generate Random Trajectories (20ish) -> Score Each Trajectories Based on Safety -> Resample from the trajectories

2. Methodology: Trajectory Planning



1. Select the waypoints path of which's end point is closest to the goal position.

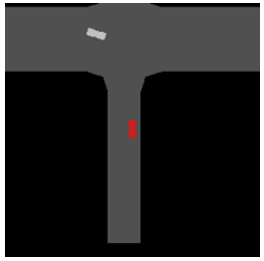
1. Generate smoothed bezier curve given ego pos and the future waypoints. Then resample the points given different speed.

1. If no waypoints are provided, then generate the path from the ego position and goal position with one turn.

2. Methodology: Safety Score

Input & Output

Current Birdview. [3, 256, 256]



Current Ego Position: [x, y]

Current Ego Heading: Θ

Next Step Action: [x, y, Θ]

Next 5 Waypoints: 5 ✕ [x, y, Θ]

Classify

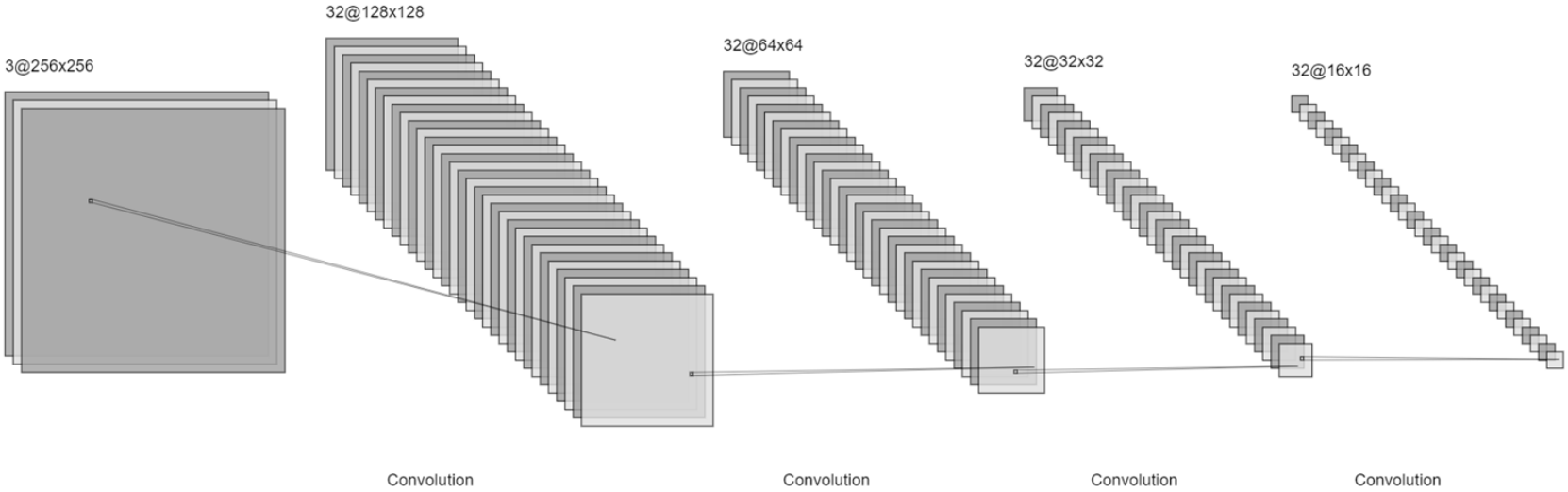
Whether the car will

1. Collide
2. Wrong way
3. On Shoulder
4. Off Road
5. Off route
6. Safe

Within the future 8 steps.

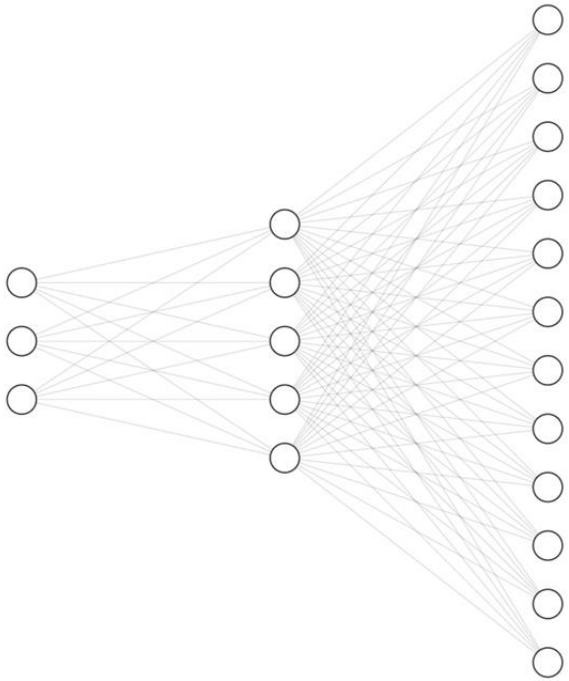
2. Methodology: Safety Score

Convolutional Neural Network for Birdview Images



2. Methodology: Safety Score

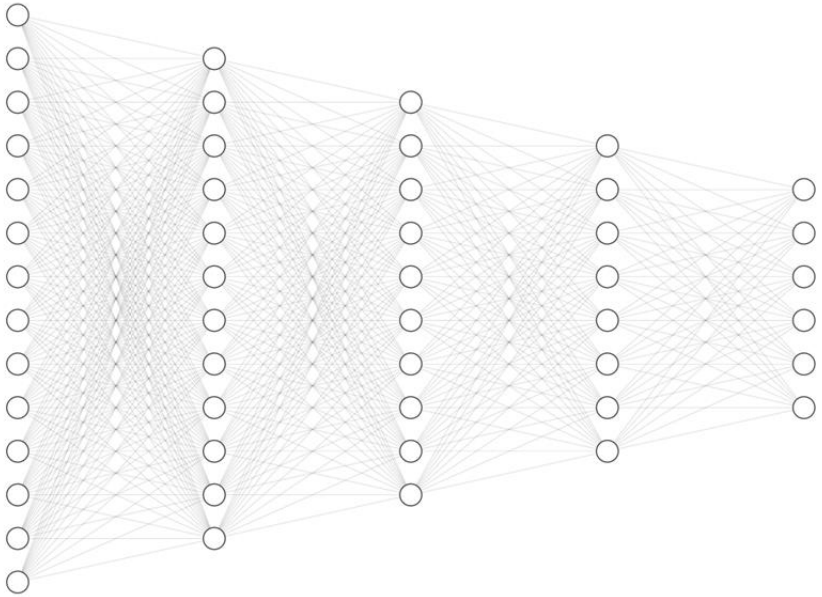
Multi Layer Perceptron for Numeric Features



Numeric Feature MLP	
	Size
Input Layer	3 + 3 + 5 * 3 = 21
Hidden Layer 1	64
Output Layer	64 x 64

2. Methodology: Safety Score

Final Combined Network



Combined Feature MLP	
	Size
Input Layer	64 x 64 + 32 x 16 x 16
Hidden Layer 1	64 x 64 x 2 x 2
Hidden Layer 2	64 x 64 x 2
Hidden Layer 3	64 x 64
Output Layer	6

2. Methodology: Safety Score

Training Data Collection

1. Data was collected for 7 scenarios provided in the track1 training set.
    - 1_to_2lane_left_turn_c
    - 1_to_2lane_left_turn_t
    - 3lane_cruise_single_agent
    - 3lane_cut_in
    - 3lane_merge_multi_agent
    - 3lane_merge_single_agent
    - 3lane_overtake

1. Total 17035 data was collected and labeled.

1. Important lesson learned:
    a. Manually made sure the portion of each 6 events in the dataset in roughly equal.

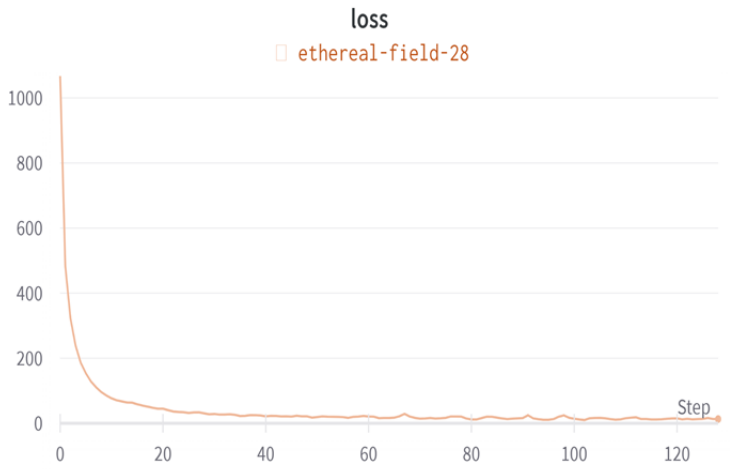
2. Methodology: Safety Score

Training Algorithm and Configuration

Configuration	Value
Learning Rate	0.003
Batch Size	100
Epoch	150
Loss Function	Cross Entropy Loss
Optimizer	Adam

2. Methodology: Safety Score

Results



Training Loss

Good News:

Validated with 1000 test data with accuracy of ~90%

Bad News:

Trained network is not sensitive to the numeric features. The prediction is heavily weighted for the birdview image data.

2. Methodology: Speed Selection Strategy

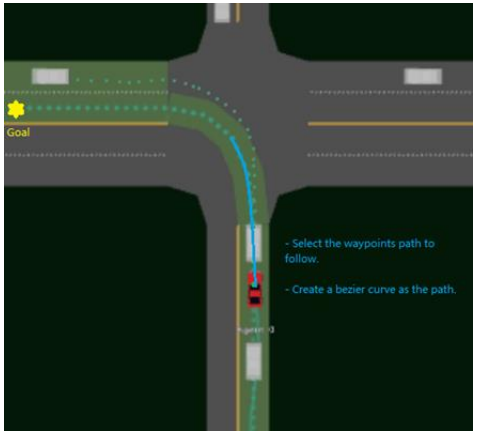
Here is the HACKY strategy I used to resolve the **bad news** due to time constraint.

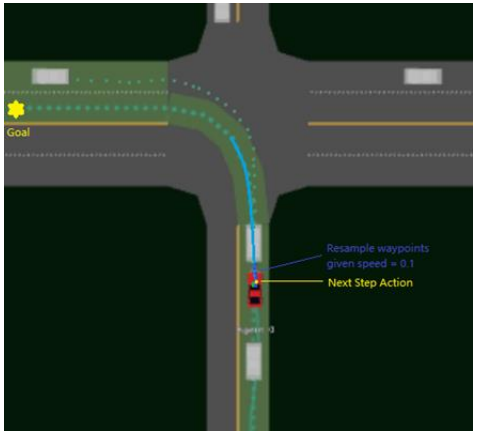
Final speed selection algorithm:

1. Tell whether the threats are from the front or back
    a. Calculated based on the surrounding neighbors location
2. If the front, map the **average** safety score from the **past 5 steps** with a sigmoid function to select speed from [0,1]
3. If the back, map the **average** safety score from the **past 5 steps** with a sigmoid function to select speed from [1, 1.2]

2. Methodology: Overall Workflow

Submitted Solution





Select waypoints path and get bezier curve -> Score the state, and select the speed -> Resample waypoints and get action

3. Future Work

1. Set up proper ***Reinforcement Learning*** agent training
    a. Didn't achieve it due to some IT issue for computing power and setting up parallel computation environment.

1. Integrate **Inverted.ai DRIVE** (https://www.inverted.ai/drive)with SMARTS simulation
    a. Inverted.ai provides Non-playable characters for simulation that are reactive, realistic, and behaviorally diverse - just like humans.
    b. Founded from UBC PLAI lab
    c. I'm halfway done the integration, still need to integrate this to work with the bubble feature in SMARTS.

1. Compare the results between using and not using Inverted.ai Drive

Thank You!

Yuan Tian, Team VCR
University of British Columbia
yuant426@alumni.ubc.ca