Autonomous Driving with Lane Changing and Overtaking using Deep Reinforcement Learning

Project Overview

This project enhances a deep Q-learning-based autonomous driving model in the CARLA simulator. Starting from a baseline lane-keeping model, the agent is trained to perform lane-changing and overtaking maneuvers in dynamic traffic environments. The focus is on achieving safe, efficient, and accurate decision-making using the reinforcement learning technique DDQN.

Features

- **Lane Keeping:** Baseline functionality for maintaining lane discipline.
- **Lane Changing:** Dynamic decision-making for transitioning between lanes.
- **Overtaking:** Safe and efficient maneuvers to overtake slower vehicles.

The problem is modeled as a Markov Decision Process (MDP), where actions impact future states in a stochastic environment.

Prerequisites

- **CARLA Simulator:** Packaged CARLA Installation
- **For Windows, download** CARLA version 0.9.15
- **Download prerequisites as stated in the CARLA documentation**
- **Python 3.8**
- **Dependencies (remember to download python dependencies to specific python 3.8 version):**
    - `Torch v 2.3`
    - `numpy`
    - `gym`
    - `matplotlib`
    - `carla` (CARLA Python API)
    - `cuda v 11.8`

Repository Structure

```
root/
├── carla_lane_keeping_d3qn.py        # Environment and RL model source code
├── frontend.py     # Simpler method to run backend code
├── saves/          # Saved trained models that can be tuned and used for training
│   └── model.pth      # Trained models
├── executable/          # Executable scripts
│   └── run_model.sh      # Shell script to run the trained model
└── README.pdf            # Project description
```

Usage

Running the Simulation

1. Launch the CARLA server (make sure the server is running before training, the code will not run):
Navigate to the folder where the CARLA executable CarlaUE4.exe can be found. Either double-click on this file or run ./CarlaUE4.exe in the terminal.

2. Train the model (multiple options):
A. Use the terminal, py -3.8 carla_lane_keeping_d3qn.py --version OTv1 --operation new --reward-function 5 --map Town04 --epsilon-decrement 0.005 --num-episodes 600 --max-steps 300 --random-spawn False – you can edit the parameters

B. Run frontend.py, change parameters to your liking, and then click on run backend.

C. Run the executables located in the executable folder. Navigate to the executable folder, then to the dist folder. You can run the executable files by either double-clicking on them or through the terminal with ./filename.exe. Make sure you are in the dist folder if using terminal

3. Results:
Once the models finish training, figures will be displayed that can help judge their performance, including metrics such as average reward over time and driving behavior over time. If running by

using frontend.py, these figures can be displayed mid-training by pressing the show plots button to help gauge current model performance.

Contributions

- **Steve Wang (UIN: 402009097):**
- **Neel Vijay Pratap Singh (UIN: 735007592):**

Video Demonstration

A 5-minute project summary video is available here.