시뮬레이션 과제

CARLA Simulator의 leaderboard와 연동된 route scenario에서 동작하는 agent를 구현합니다.

Requirements

- agent는 1개의 camera sensor로부터 들어오는 영상을 매 프레임 디스플레이에 보여줍니다.
- agent는 PythonAPI를 이용하여 World에 있는 모든 Vehicle과 Walker의 정보를 읽어옵니다.
- agent는 ego vehicle로부터 50m 반경내에 위치하는 모든 Vehicle과 Walker의 bounding box를 camera sensor로부터 들어오는 영상 위에 overlay하여 그려줍니다.
- agent는 keyboard 입력을 받아, ego vehicle의 steer, throttle, break를 제어합니다.
- agent는 ego vehicle로부터 전방 0m ~ 10m, 좌우 -3m ~ 3m 내에 다른 actor (Vehicle, Walker)가 존재할 경우, break 값을 1.0으로 유지합니다.
- agent는 ego vehicle과 다른 actor (Vehicle, Walker)와의 CollisionEvent를 count하여 디스플레이상에 충돌 횟수를 점수로 보여줍니다.

Installation

1. System setup
    a. Get CARLA 0.9.10.1
        i. Download the binary `CARLA 0.9.10.1 release`.
            - https://carla-releases.s3.eu-west-3.amazonaws.com/Linux/CARLA_0.9.10.1.tar.gz
        ii. Unzip the package into a folder, e.g. CARLA.
            - `${CARLA_ROOT}` corresponds to your CARLA root folder. Change this for your CARLA root folder when copying the commands below.
        iii. In order to use the CARLA Python API you will need to install some dependencies in your favorite environment. As a reference, for *conda*, start by creating a new environment:

```
conda create -n py37 python=3.7
conda activate py37
cd ${CARLA_ROOT}  # Change ${CARLA_ROOT} for your CARLA
root folder

pip3 install -r PythonAPI/carla/requirements.txt
easy_install PythonAPI/carla/dist/carla-0.9.10-py3.7-
linux-x86_64.egg
```

    b. Get the Leaderboard and Scenario_Runner
        i. Download the Leaderboard repository.

```
git clone -b stable --single-branch https://github.com
/carla-simulator/leaderboard.git
```

        `${LEADERBOARD_ROOT}` corresponds to your Leaderboard root folder. Change this for your Leaderboard root folder when copying the commands below.
        ii. Install the required Python dependencies.

```
cd ${LEADERBOARD_ROOT} # Change ${LEADERBOARD_ROOT} for
your Leaderboard root folder
pip3 install -r requirements.txt
```

iii. Download the Scenario_Runner repository.

```
git clone -b leaderboard --single-branch https://github.
com/carla-simulator/scenario_runner.git
```

`${SCENARIO_RUNNER_ROOT}` corresponds to your Scenario_Runner root folder. Change this for your Scenario_Runner root folder when copying the commands below.

iv. Install the required Python dependencies using the same Python environments.

```
cd ${SCENARIO_RUNNER_ROOT} # Change
${SCENARIO_RUNNER_ROOT} for your Scenario_Runner root
folder
pip3 install -r requirements.txt
```

c. **Define the environment variables**

We need to make sure that the different modules can find each other.

i. Edit your `~/.bashrc` profile, adding the following definitions:

```
export CARLA_ROOT=PATH_TO_CARLA_ROOT
export SCENARIO_RUNNER_ROOT=PATH_TO_SCENARIO_RUNNER
export LEADERBOARD_ROOT=PATH_TO_LEADERBOARD
export PYTHONPATH="${CARLA_ROOT}/PythonAPI/carla
/":"${SCENARIO_RUNNER_ROOT}":"${LEADERBOARD_ROOT}":${PYTHO
NPATH}
```

ii. Open the `~/.bashrc` profile with the following command. Remember to save your changes before closing.

```
gedit ~/.bashrc
```

iii. Remember to source `.bashrc` for these changes to take effect.

```
source ~/.bashrc
```

2. **Create Your Own Agent with the Leaderboard**

a. **First steps with the Leaderboard**

The Leaderboard will take care of running your autonomous agent and evaluate its behavior in different traffic situations across multiple routes. To better understand this process, let's run a basic agent.

i. Run the CARLA server in one terminal.

```
cd ${CARLA_ROOT}
./CarlaUE4.sh -quality-level=Epic -world-port=2000 -
resx=800 -resy=600
```

ii. Create a Python script that sets some environment variables for parameterization, and runs the `run_evaluation.sh`. The script should be similar to the following.

```
#Parameterization settings. These will be explained in
2.2. Now simply copy them to run the test.
export SCENARIOS=${LEADERBOARD_ROOT}/data
/all_towns_traffic_scenarios_public.json
export ROUTES=${LEADERBOARD_ROOT}/data/routes_devtest.xml
export REPETITIONS=1
export DEBUG_CHALLENGE=1
export TEAM_AGENT=${LEADERBOARD_ROOT}/leaderboard
/autoagents/human_agent.py
export PYTHONPATH="${CARLA_ROOT}/PythonAPI/carla
/":"${SCENARIO_RUNNER_ROOT}":"${LEADERBOARD_ROOT}":${PYTHO
NPATH}
export CHECKPOINT_ENDPOINT=${LEADERBOARD_ROOT}/results.
json
export CHALLENGE_TRACK_CODENAME=SENSORS

./scripts/run_evaluation.sh
```

This will launch a pygame window giving you the option to manually control an agent. Follow the route indicated by colorful waypoints in order to get to your destination.

References

- https://leaderboard.carla.org/get_started/
- https://carla.readthedocs.io/en/0.9.10/python_api/