

CEE 678 CARLA Lab 3: Customize Autopilot Behavior and Set the Spectator

Prerequisites:

1. Success installation and running of CARLA. (version 0.9.12 is used in this tutorial)
2. A Python IDE to edit the Python scripts.

Load the CARLA Server and Client:

Server side:

1. Open the Command Prompt window, type or copy and paste the two commands and press Enter:

```
>> cd C:/Your_Directory/CarlaSimulator
>> CarlaUE4.exe
```

Since running CARLA requires strong graphical computing power, we recommend you run CARLA with low graphics quality to speed up the simulation:

```
>> CarlaUE4.exe -quality-level=Low
```

Client side:

1. Open a **new** CMD window, type the following commands to run the example Python client:

```
>> cd C:/Your_Directory/PythonAPI/examples
>> python manual_control.py
```

2. To run the customized code, you can type the following commands in the same way:

```
>> cd C:/Your_Directory/PythonAPI/examples
>> python set_autopilot_behavior.py
```

Procedures to Set the Autopilot Behavior:

Launch the server and follow steps in the example code [set_autopilot_behavior.py](#). Feel free to change the parameters.

1. Spawn two example vehicles

- a. Initialize [carla.BlueprintLibrary](#) class

```
blueprint_library = world.get_blueprint_library()
```

- b. `vehicle_1` is generated using the Tesla Model 3 blueprint and at the first default spawn point. The autopilot mode is turned on.

```
vehicle_1_bp = blueprint_library.filter('model3')[0]
spawn_point_1 = spawn_points[0]
vehicle_1 = world.spawn_actor(vehicle_1_bp, spawn_point_1)
vehicle_1.set_autopilot(True)
actor_list.append(vehicle_1)
```

- c. `vehicle_2` is generated using the Tesla Cybertruck blueprint. The spawn point is 6 meters away from `spawn_point_1` in x-axis. The autopilot mode is turned on.

```
vehicle_2_bp = blueprint_library.filter('cybertruck')[0]
spawn_point_2 = carla.Transform(spawn_point_1.location-carla.Location(x=6),
spawn_point_1.rotation)
vehicle_2 = world.spawn_actor(vehicle_2_bp, spawn_point_2)
vehicle_2.set_autopilot(True)
actor_list.append(vehicle_2)
```

The two vehicles will be generated at this spot.



2. Set autopilot behavior

- a. Get the traffic manager. Class: [carla.TrafficManager](#)

```
tm = client.get_trafficmanager(8000)
```

- b. Set the autopilot behavior. Check this link for more examples:

https://carla.readthedocs.io/en/latest/adv_traffic_manager/#configuring-autopilot-behavior.

- i. Set `vehicle_1` as a general automated vehicle, which keeps at least 2 meters from other vehicles, and drives 20% faster than the current speed limit.

```
current_veh = vehicle_1
tm.distance_to_leading_vehicle(current_veh,2)
tm.vehicle_percentage_speed_difference(current_veh,-20)
```

- ii. Set `vehicle_2` as a dangerous vehicle, which ignores all traffic lights, keeps no safety distance from other vehicles, and drive 40% faster than the current speed limit.

```
danger_car = vehicle_2
tm.ignore_lights_percentage(danger_car,100)
tm.distance_to_leading_vehicle(danger_car,0)
tm.vehicle_percentage_speed_difference(danger_car,-40)
```

3. Set the spectator.

- a. Get the spectator. The spectator allows you to freely navigate in the simulation world.

We are going to fix the view on a specific vehicle. Check this link:

https://carla.readthedocs.io/en/latest/tuto_G_getting_started/#the-spectator.

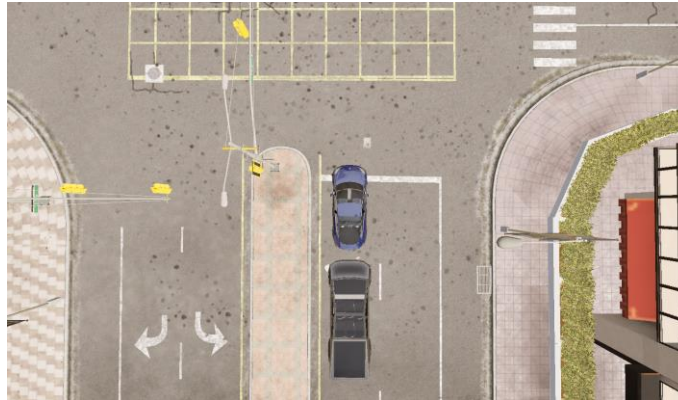
```
spectator = world.get_spectator()
```

- b. Use the while loop to update the location and rotation of the spectator.

```
while True:  
    transform = vehicle_1.get_transform()
```

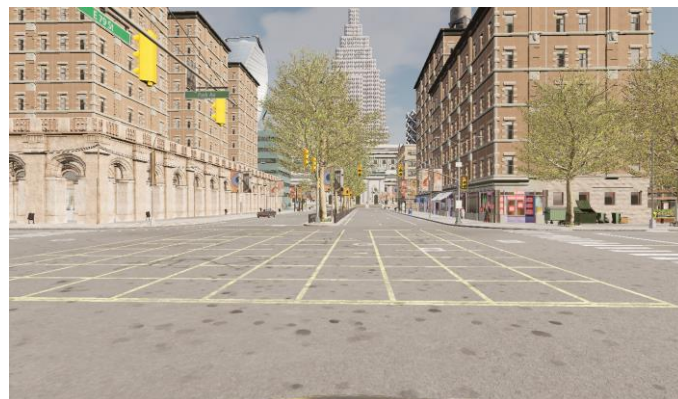
Spectator transform 1: bird's-eye view

```
spectator_location = transform.location + carla.Location(z=20)  
spectator_rotation = carla.Rotation(pitch=-90)
```



Spectator transform 2: first-person view

```
spectator_location = transform.location + carla.Location(z=2)  
spectator_rotation = carla.Rotation(pitch=0, yaw=transform.rotation.yaw)
```



Update the transform with an interval of 0.02 second.

```
spectator.set_transform(carla.Transform(spectator_location,  
spectator_rotation))  
time.sleep(0.02)
```

Tasks:

1. Read and run the sample code: `set_autopilot_behavior.py`.
2. Set a group of three vehicles with different behaviors (aggressive, moderate, and conservative, L1-L5, mixed).

3. Set appropriate spectators for the vehicles.
4. Submit the ZIP file including following:
 - a. Code to set the autopilot behavior.
 - b. Video recordings of the three vehicles.