Virtual driving scenarios for verifying and designing automated vehicles

Witek Jachimczyk
Development Manager
Computer Vision and Automated Driving
Automated Driving Toolbox

Examples

Algorithms

Visualizations

Virtual Scenario and Sensor Simulation

Ground Truth Labeling

Geographic Maps
The two simulation environments

Cuboid simulation environment

Gaming engine-based simulation environment
Cuboid simulation environment

Command line API

% Create driving scenario
s = drivingScenario('SampleTime', 0.05);

% Create a simple single lane road
roadCenters = [0 0; 10 0; 40 20; 50 20];
roadWidth = 5; % (m)
road(s,roadCenters,roadWidth)

% Add vehicle
trajectory(egoCar, waypoints, speed);
Programmatic API

% Create driving scenario
s = drivingScenario('SampleTime', 0.05);

% Create a simple single lane road
roadCenters = [0 0; 10 0; 40 20; 50 20]; % (m)
roadWidth = 5; % (m)
road(s,roadCenters,roadWidth)

% Add vehicle
egoCar = vehicle(s);
waypoints = roadCenters; % (m)
speed = 13.89; % (m/s) = 50 km/hour
trajectory(egoCar, waypoints, speed);

% Play scenario
while advance(s)
    pause(s.SampleTime);
end
Driving Scenario Designer App
Integrate driving scenarios into Simulink
Closed-loop: AEB scenario
Learn more about creating scenarios by exploring examples in the Automated Driving Toolbox.
HERE HD Live Map Reader

```matlab
>> reader = hereHDLMReader(latitude, longitude)
```

**hereHDLMReader** with properties:

- TileIds: 309106790
- Layers: [10x1 string]
- WriteLocation: "C:\Users\akurian\AppData\...
- Configuration: 1=n hereHDLMConfigurations

**HD Lane Model**

```matlab
>> read(reader, 'AdasAttributes')
```

**AdasAttributes** with properties:

- HereTileId: 309106790
- LinkAttribution: [603x1 struct]
- NodeAttribution: [443x1 struct]

Use HERE HD Live Map Data to Verify Lane Configurations

Read and visualize lane configurations for a recorded driving route from the HERE HD Live Map (HDLM) service.
Create roads from geographic maps

HERE HD Live Map

Import

OpenStreetMap
Here is how it might look…
Gaming engine-based simulation environment

Simulink library

Rendered scene
Example: automated parking valet
Core components comprising the simulator

- Simulation 3D Scene Configuration
- Simulation 3D Camera
- Simulation 3D Probabilistic Radar
- Simulation 3D Fisheye Camera
- Simulation 3D Probabilistic Radar Configuration
- Simulation 3D Lidar
Scene configuration

Configures the 3D simulation environment. You must have this block in models that have sensor blocks to test perception, control, and planning algorithms with data from the 3D environment. The sensor blocks and visualization environment inherit the sample time parameter value from this block.
Vehicle control

Simulation 3D Vehicle with Ground Following

Implements a vehicle with four wheels that follows the ground in the 3D visualization environment. Uses the vehicle position to adjust the vehicle elevation, roll, and pitch so that the vehicle follows the ground terrain. Determines the vehicle velocity and heading and adjusts the steering angle and rotation for each wheel. You can select the type of vehicle, color, and initial position and rotation.

Vehicle Parameters

- **Type:** Muscle car
- **Color:** Sedan
- **Initial:** Sport utility vehicle
- **Initial rotation [Roll, Pitch, Yaw] (deg):** [0, 0, 0]
- **Name:** SimulinkVehicle1
- **Sample time:** -1

[X Y Yaw]
Sensor example: video camera
Putting it all together in a simple model
More in Automated Driving Toolbox
Cuboid vs. gaming engine simulation environment

Key takeaways

• Both environments have their uses. One does not replace the other.

• Both environments offer virtual sensors. Sensors in the gaming environment provide richer output.

• Cuboid simulation lets you rapidly define and simulate your scenarios and it does not require high powered GPU.
Thank you!