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

Bird’s-Eye Scope
Learn more about creating scenarios
by exploring examples in the Automated Driving Toolbox
HERE HD Live Map Reader

>> reader = hereHDLMReader(latitude, longitude)

hereHDLMReader with properties:

  TileIds: 309106790
  Layers: [10x1 string]
  WriteLocation: "C:\Users\akurian\AppData\"C:\Users\akurian\AppData\Read and visualize lane configurations for a recorded driving route from the HERE HD Live Map (HDLM) service. This visualization

>> read(reader, 'Attributes')

ans =

  Attributes with properties:

    HereTileId: 309106790
    LinkAttribution: [603x1 struct]
    NodeAttribution: [443x1 struct]
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 Vehicle with Ground Following
- Simulation 3D Camera
- Simulation 3D Fisheye Camera
- Simulation 3D Probabilistic Radar
- Simulation 3D Probabilistic Radar Configuration
- Simulation 3D Lidar

Note: Simulation 3D (Windows only)
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.

Simulation Configuration

Scene description:
- Straight road
- Curved road
- Parking lot
- Double lane change
- Open surface
- US city block
- US highway
- Virtual Moty
- Large parking lot
- Custom

Sample time: 1/60

OK  Cancel  Help  Apply
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.

Vehicle Parameters

- **Type:** Muscle car
- **Color:** Sedan
- **Initial:** Sport utility vehicle
- **Initial rotation [Roll, Pitch, Yaw] (deg):** [0, 0, 0]
Sensor example: video camera
Putting it all together in a simple model

Lane Marker Detection in Mcity

Simulation 3D Scene Configuration

Simulation 3D Camera

Image

poses.refPosesX

poses.refPosesY

poses.refPosesZ

X

Y

Yaw

Simulation 3D Vehicle With Ground Following

Monocular Camera Sensor with Visualizations

Active Area

(0,0,0)
More in Automated Driving Toolbox

- **Select Waypoints for 3D Simulation**
  Select waypoints from a scene and visualize the path of a vehicle following these waypoints in a 3D simulation environment.

- **Design of Lane Marker Detector in 3D Simulation Environment**
  Use a 3D simulation environment to record synthetic sensor data and develop and test a lane marker detection system.

- **Visualize Automated Parking Valet Using 3D Simulation**
  Visualize vehicle motion in a 3D simulation environment using an automated parking valet system constructed in Simulink.

- **Simulate Radar Sensors in 3D Environment**
  Implement a synthetic data simulation for tracking and sensor fusion using Simulink and a 3D simulation environment.

- **Simulate Lidar Sensor Perception Algorithm**
  Develop a lidar perception algorithm using data recorded from a 3D simulation environment, and simulate within that environment.
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!