



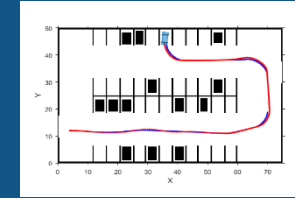
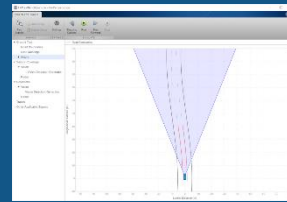
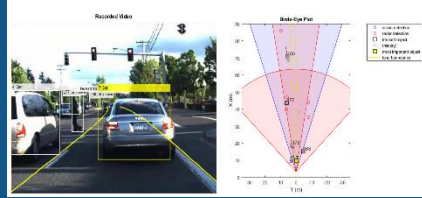
MathWorks
**AUTOMOTIVE
CONFERENCE 2019**

Virtual driving scenarios for verifying
and designing automated vehicles

Witek Jachimczyk
Development Manager
Computer Vision and Automated Driving

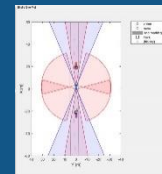
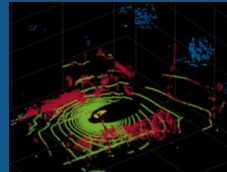
Automated Driving Toolbox

Examples

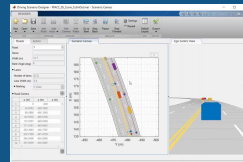


Algorithms

Visualizations

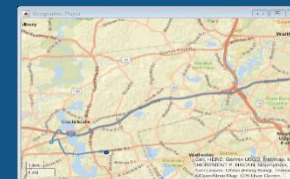


Virtual Scenario and Sensor Simulation



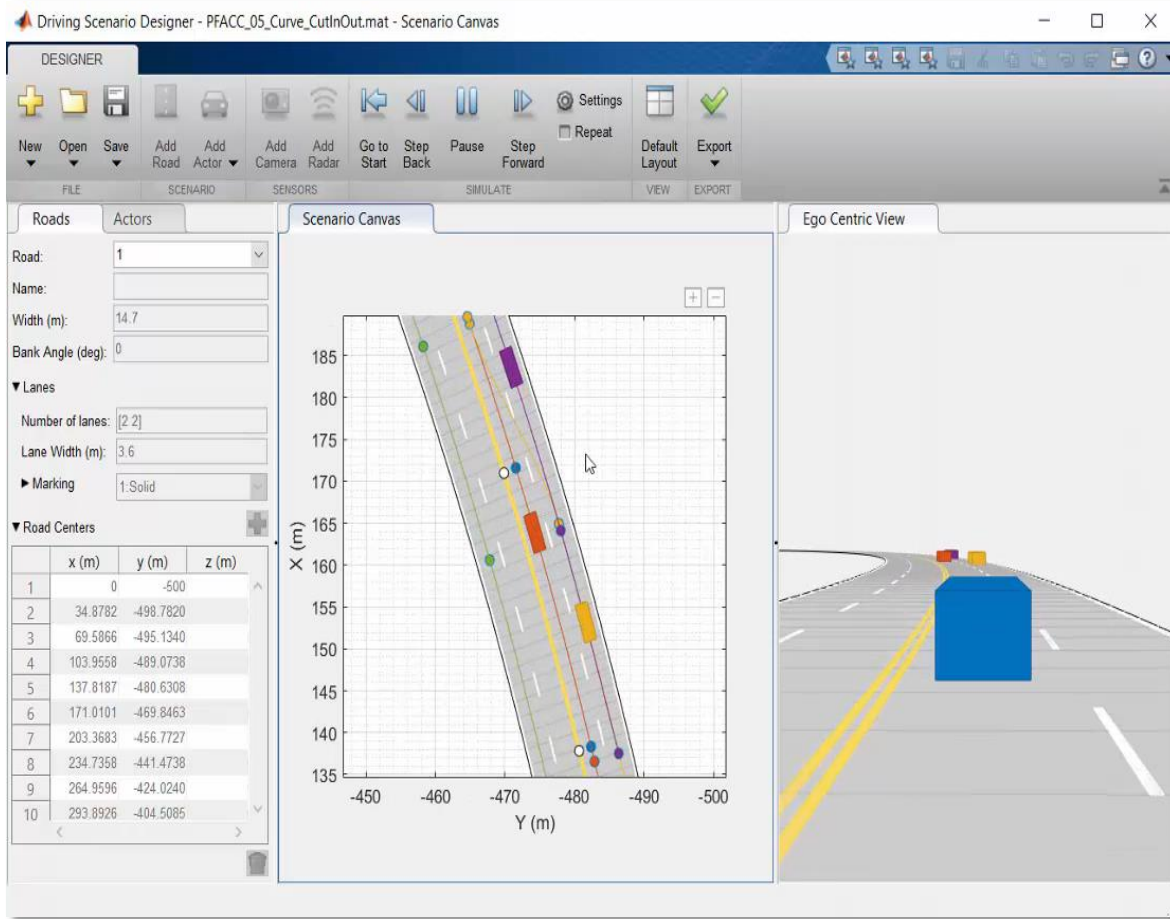
Ground Truth Labeling

Geographic Maps



The two simulation environments

Cuboid simulation environment



Gaming engine-based simulation environment



Cuboid simulation environment

Command line API

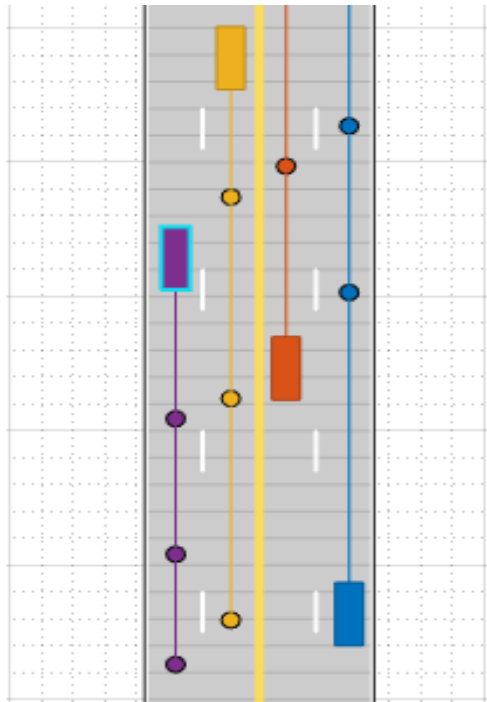
```

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

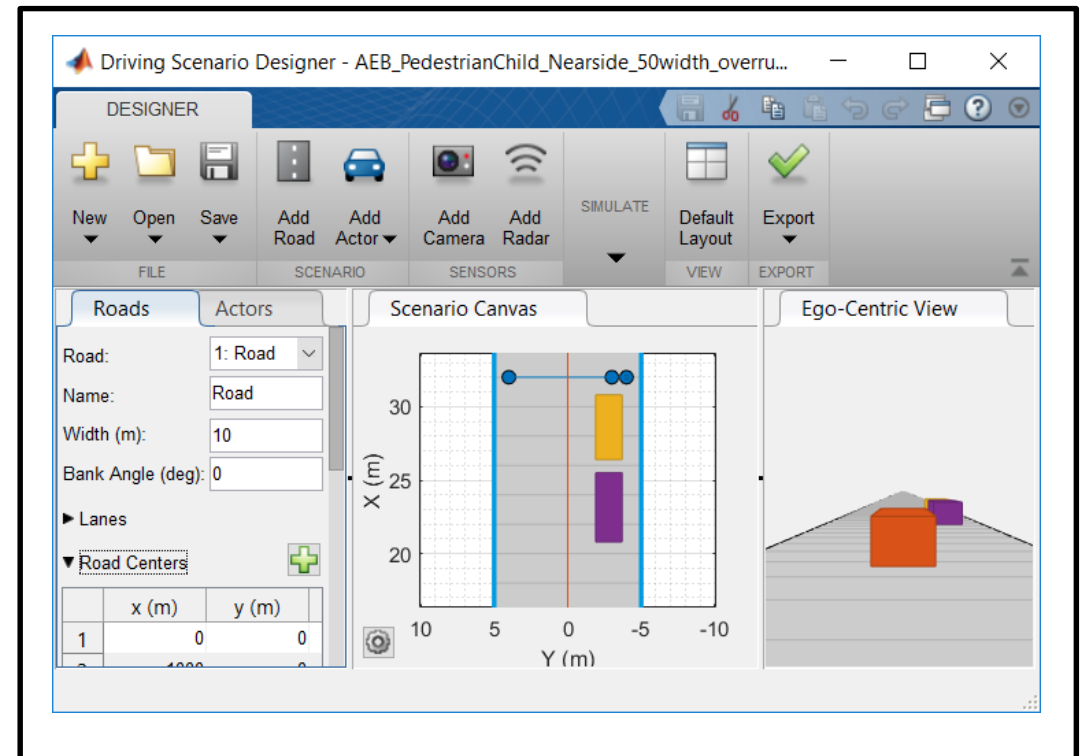
% Create a simple single lane road
roadCenters = [0 0; 10 0; 40 20; 50 20];
(m)
roadWidth = 5; % (m)
road(s,roadCenters,roadWidth)

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

```



Driving Scenario Designer App



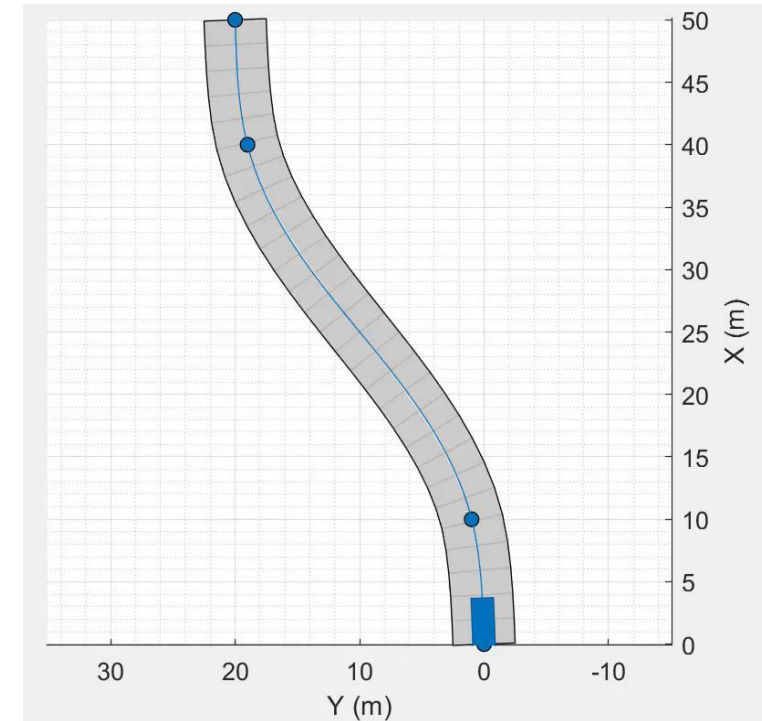
Programmatic API

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

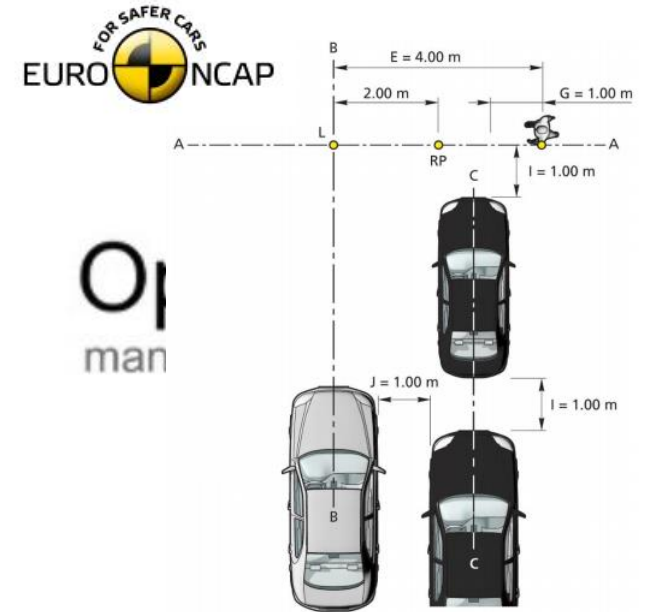
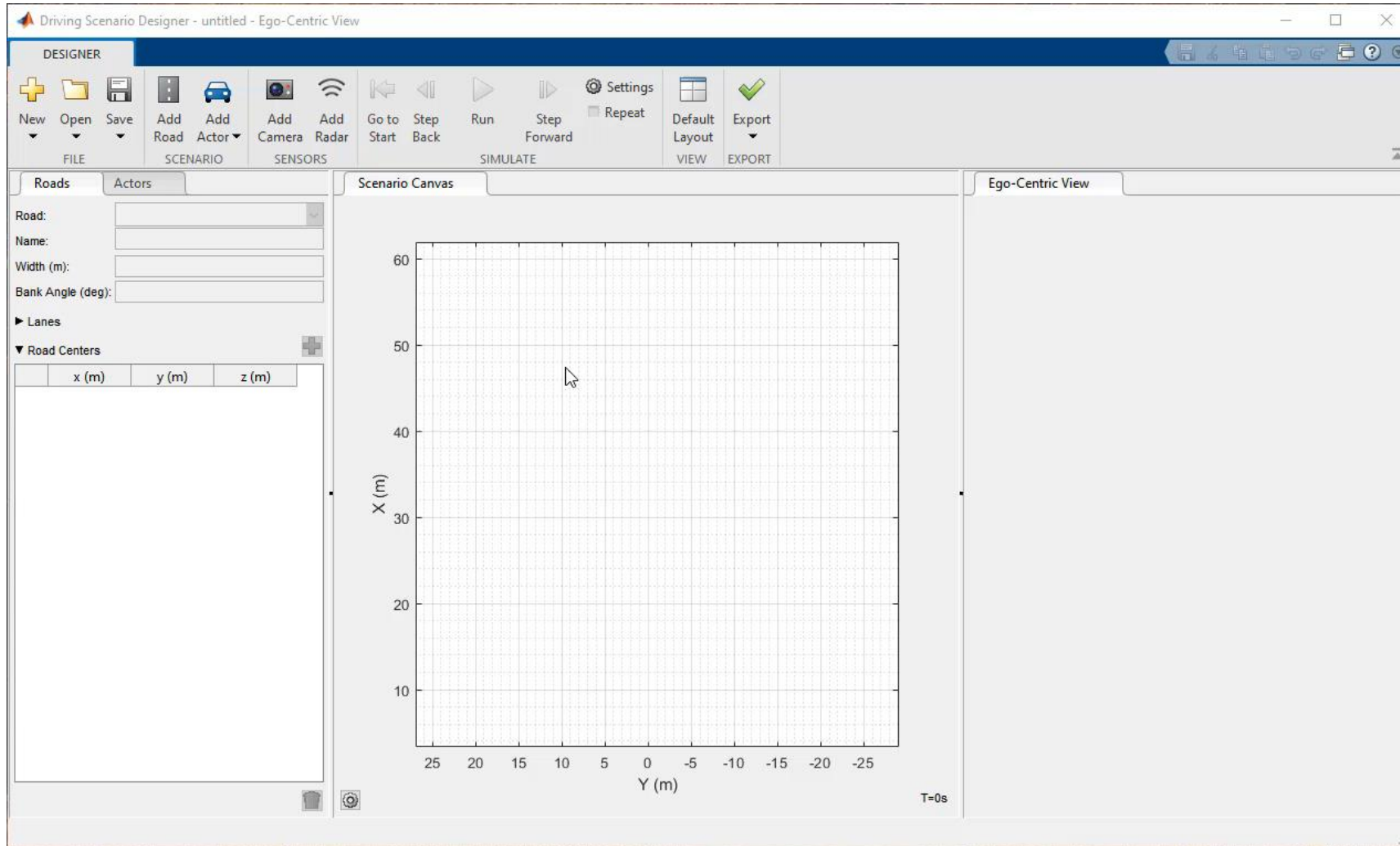
```
% Create a simple single lane road
roadCenters = [0 0; 10 0; 40 20; 50 20]; % (m)
roadWidth = 5; % (m)
road(s, roadCenters, roadWidth)
```

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

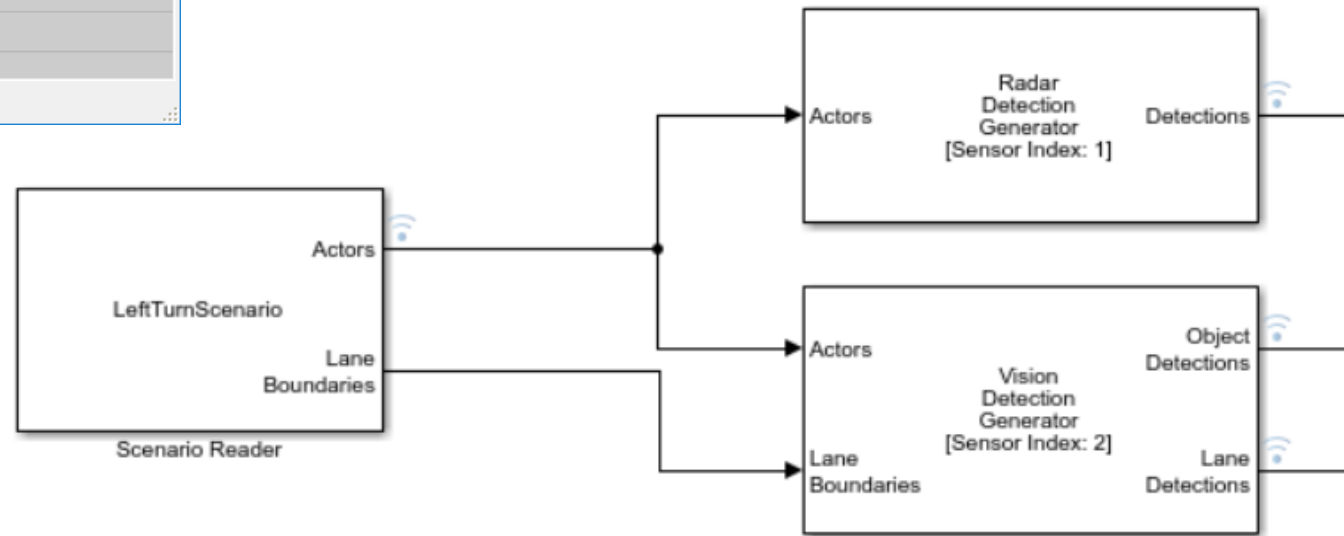
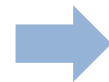
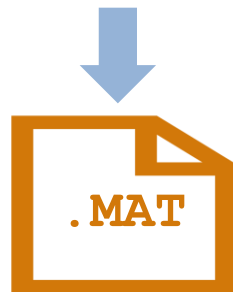
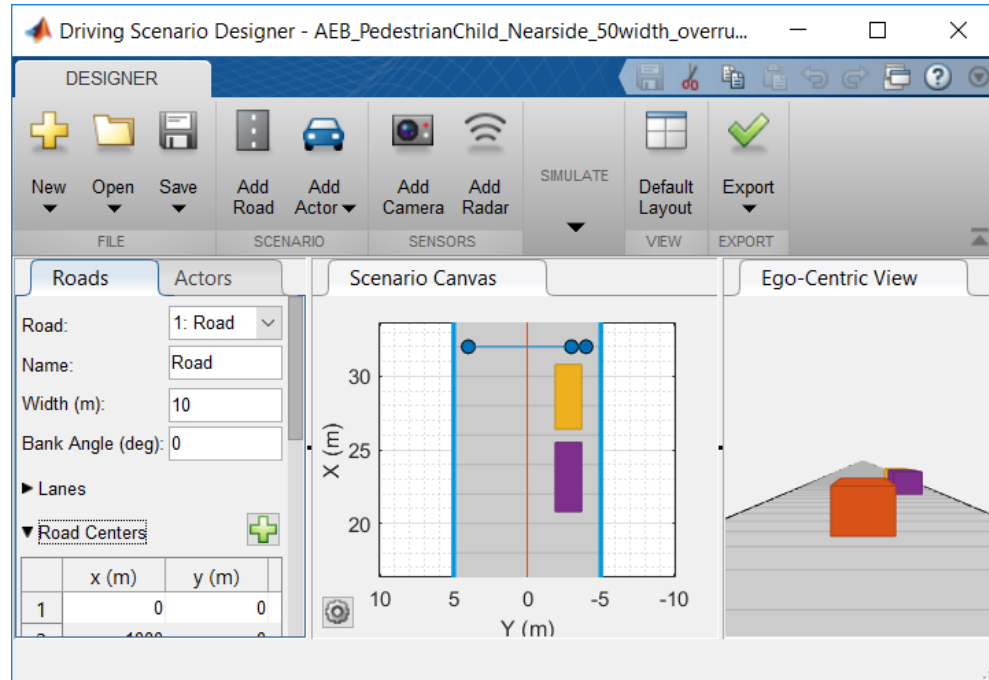
```
% Play scenario
while advance(s)
    pause(s.SampleTime);
end
```



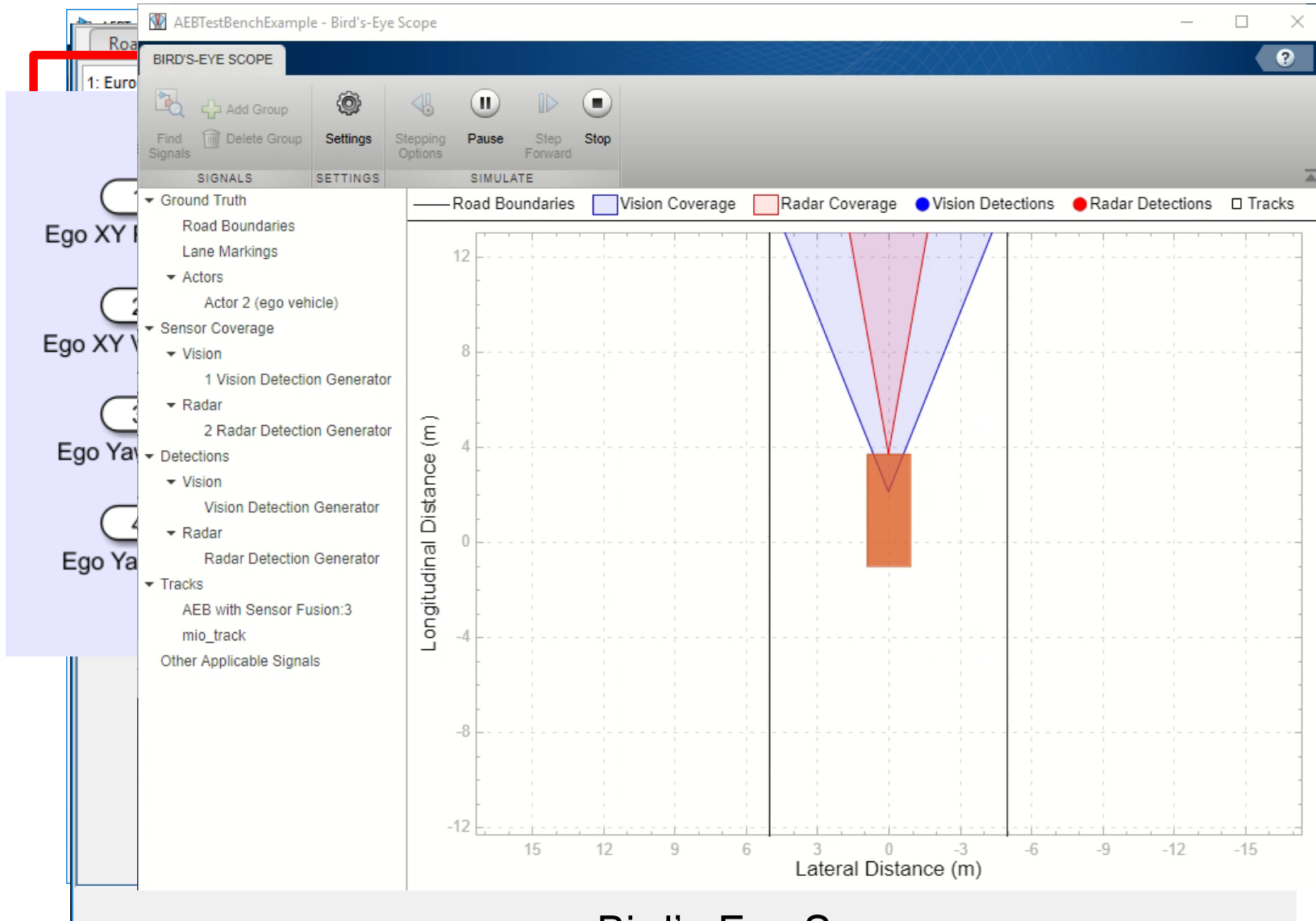
Driving Scenario Designer App



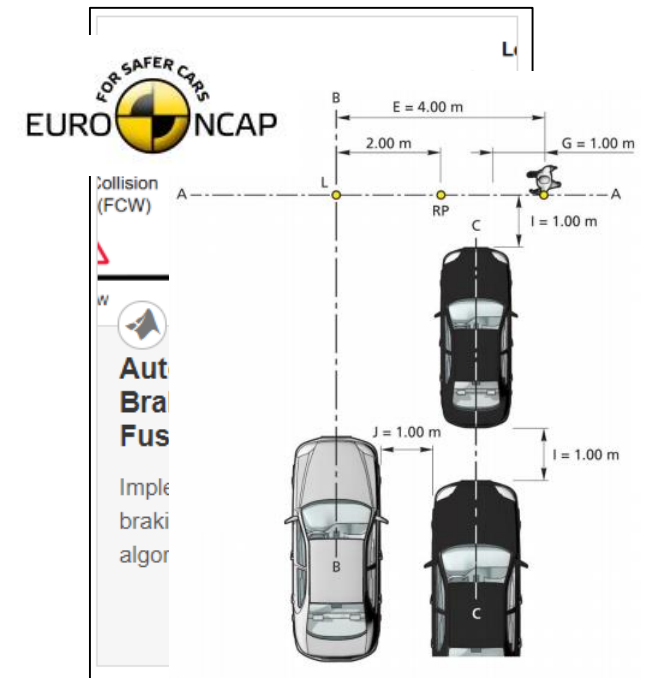
Integrate driving scenarios into Simulink



Closed-loop: AEB scenario

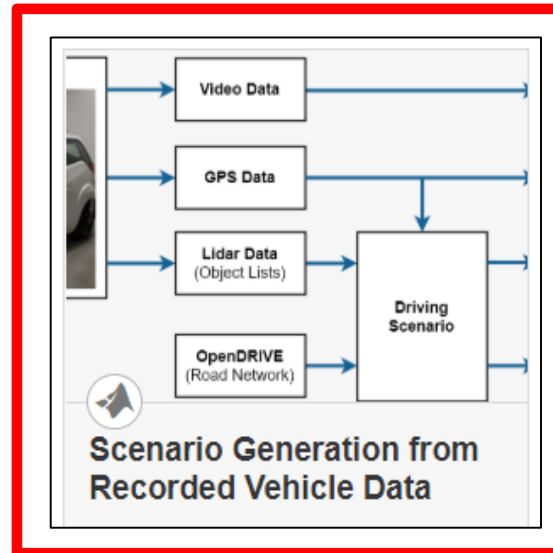
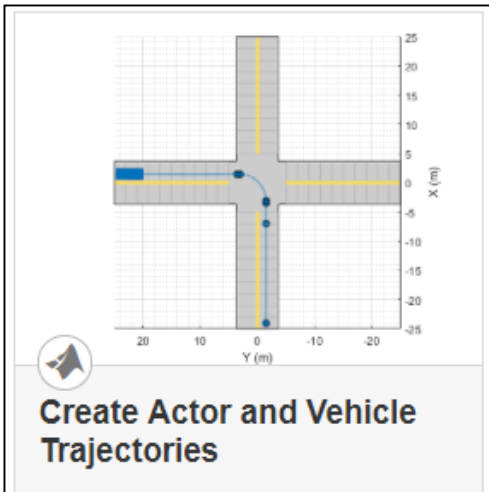
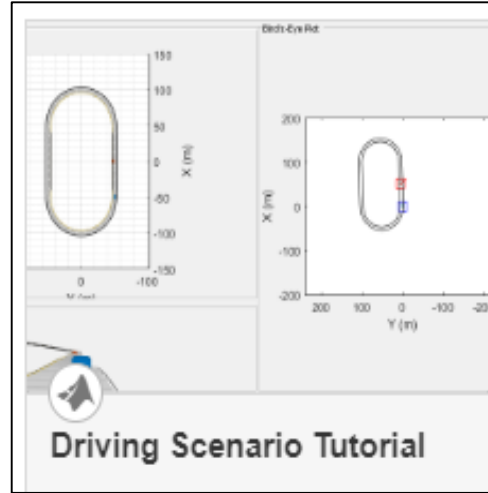
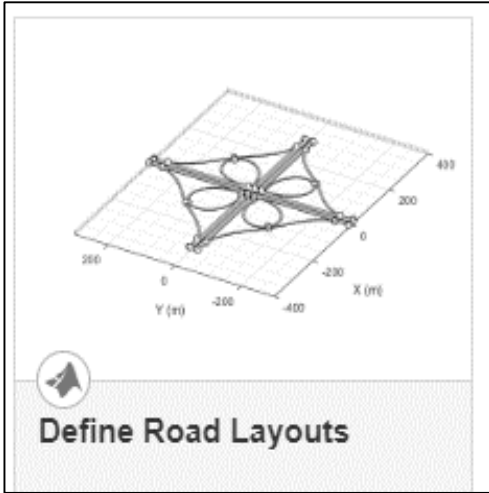


Bird's-Eye Scope

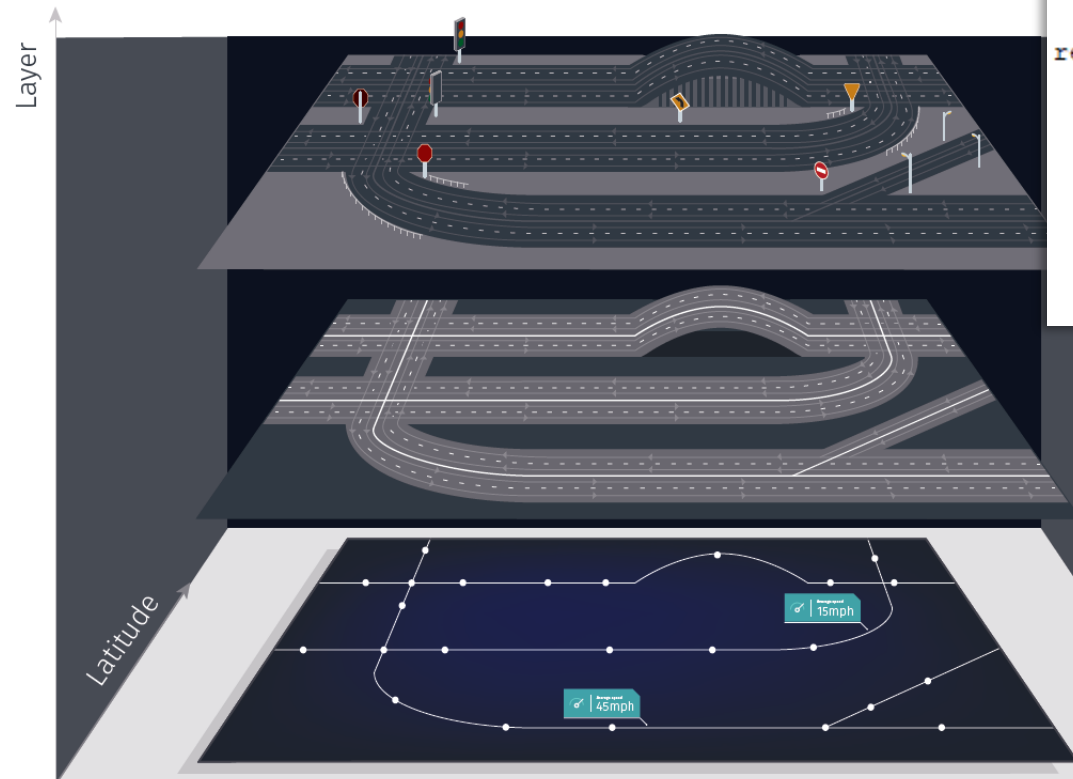


Learn more about creating scenarios

by exploring examples in the Automated Driving Toolbox



HERE HD Live Map Reader

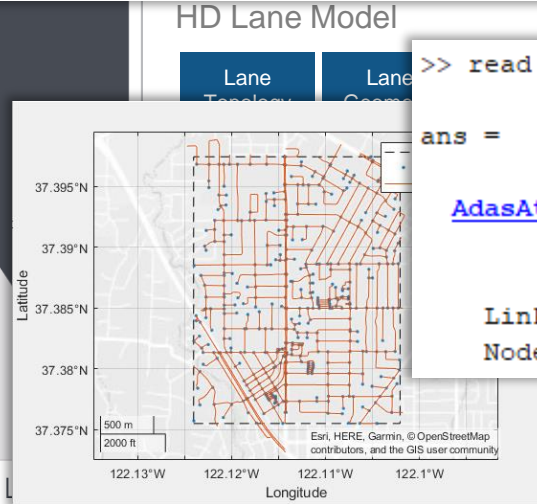


```
>> reader = hereHDLReader(latitude, longitude)

reader =

hereHDLReader with properties:

    TileIds: 309106790
    Layers: [10x1 string]
    WriteLocation: "C:\Users\akurian\AppData\Local\Temp\
    Configuration: [1x1 hereHDLConfiguration]
```



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

ans =

AdasAttributes with properties:

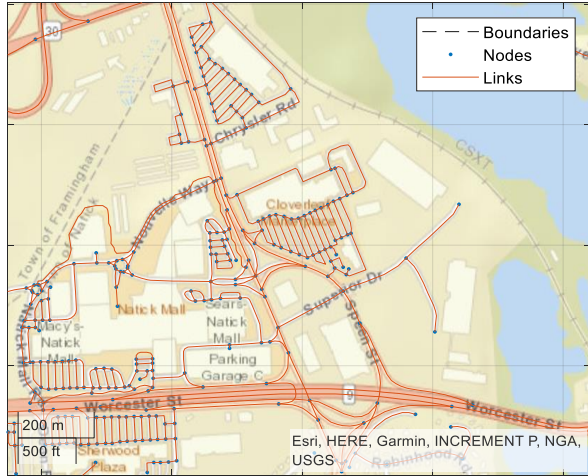
    HereTileId: 309106790
    LinkAttribution: [603x1 struct]
    NodeAttribution: [443x1 struct]
```

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 (HDL) service. This visualization

 >> hereHDLReader 

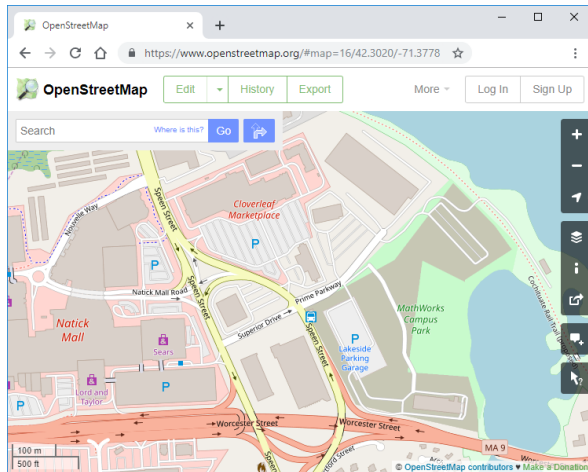
Create roads from geographic maps



HERE HD Live Map



Import



OpenStreetMap

Driving Scenario Designer - untitled* - Scenario Canvas

DESIGNER

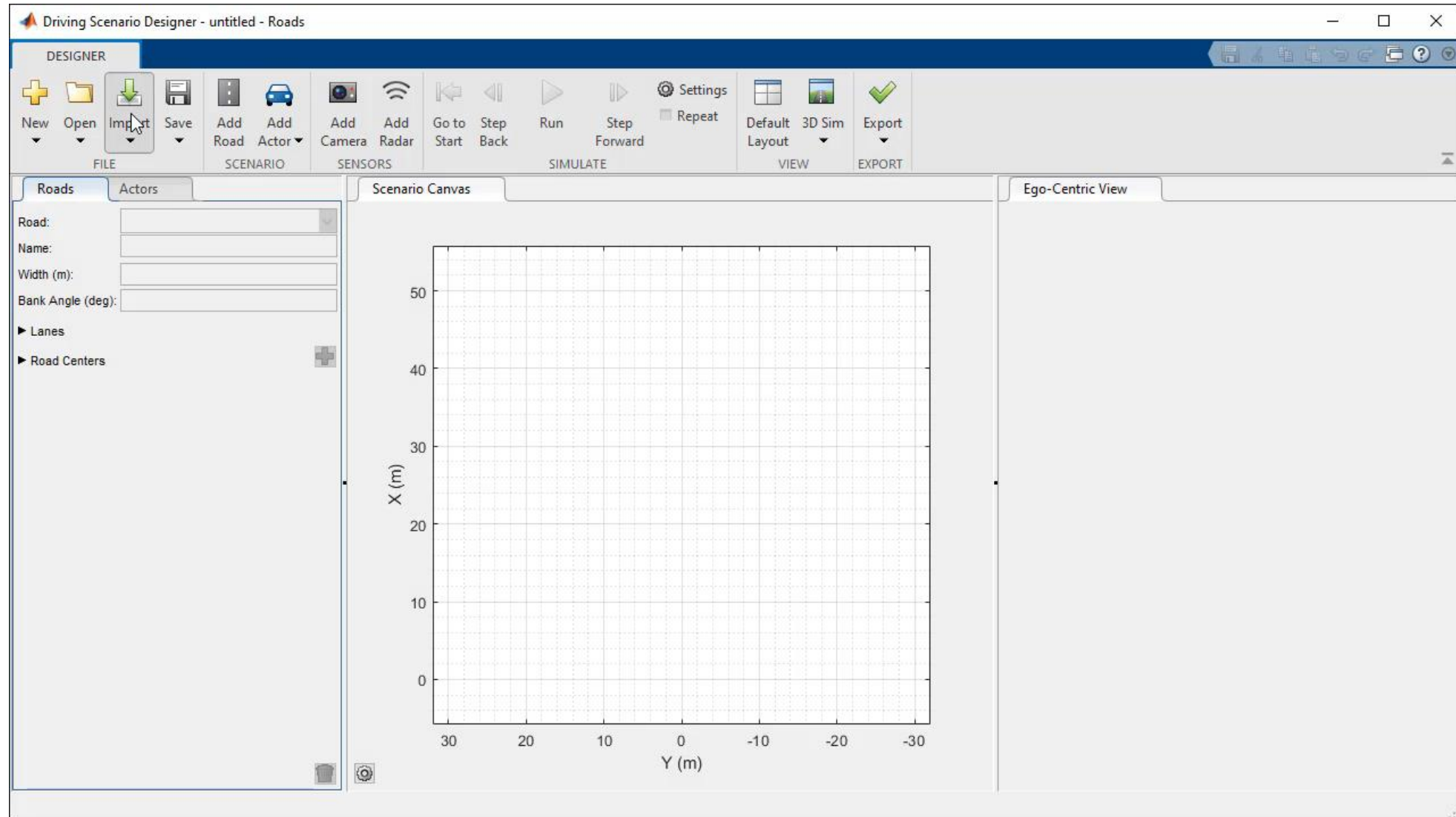
FILE SCENARIO SENSORS SIMULATE VIEW EXPORT

Roads Actors Scenario Canvas Ego-Centric View

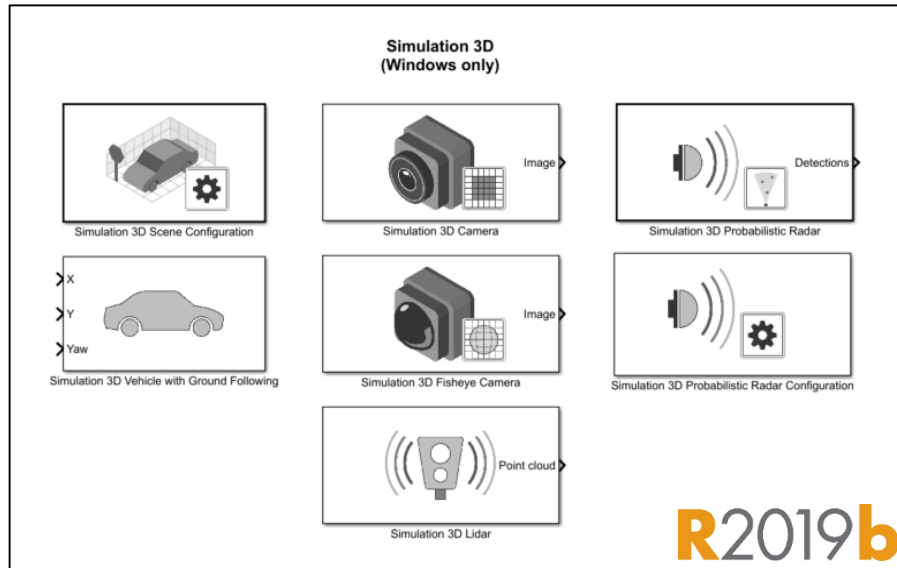
1: Car (ego vehicle)
 Name: Car
 Class: Car
 Constant Speed (m/s): 30

Waypoints	x (m)	y (m)	z (m)
1	25.7000	65.8000	0
2	32.8000	93.2000	0
3	32.1000	110.2000	0

Here is how it might look...



Gaming engine-based simulation environment



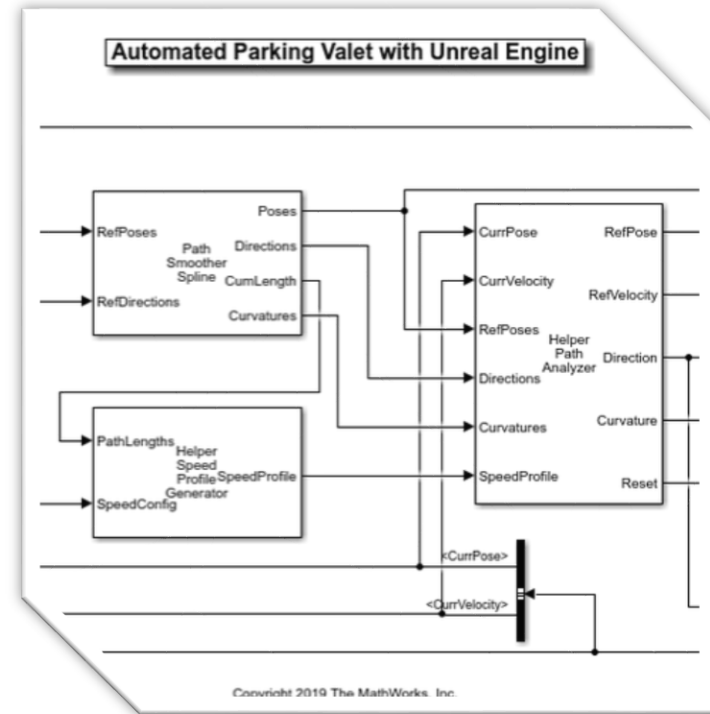
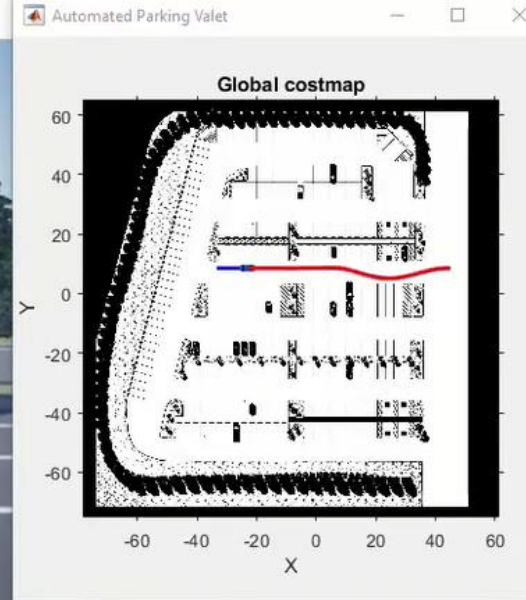
Simulink library



Rendered scene

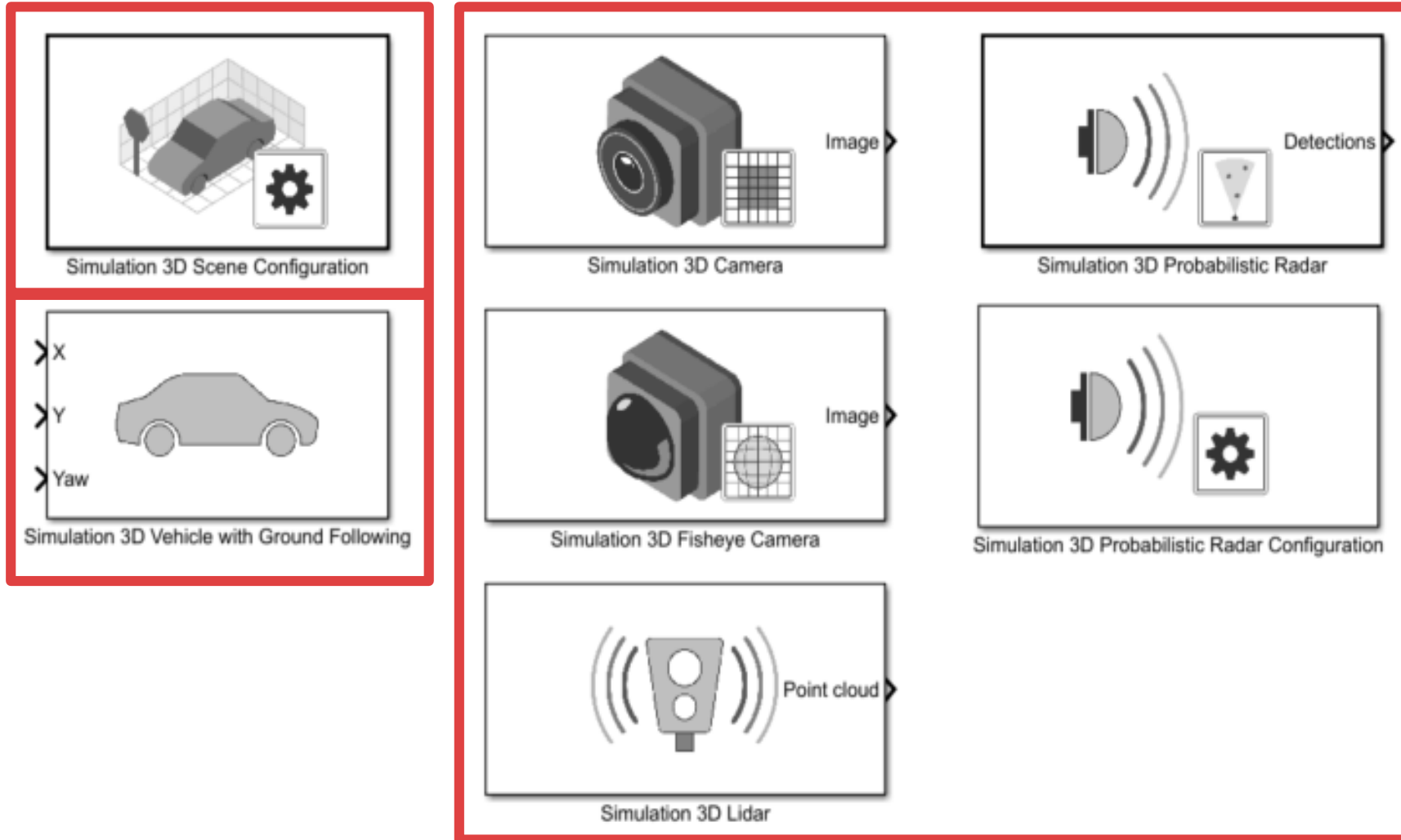
Example: automated parking valet

AutoVrtlEnv (64-bit, PCD3D_SM5)



Core components comprising the simulator

Simulation 3D (Windows only)





Simulation 3D Scene Configuration

Scene configuration

Block Parameters: Simulation 3D Scene Configuration

Simulation 3D Scene Configuration (mask) (link)

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 Co-Simulation

Scene description: **Straight road**

Scene view: Simul

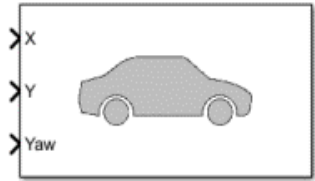
Show state and s

Sample time: 1/60

- Straight road
- Curved road
- Parking lot
- Double lane change
- Open surface
- US city block
- US highway
- Virtual Mcity**
- Large parking lot
- Custom

OK Cancel Help Apply





Simulation 3D Vehicle with Ground Following

Vehicle control

Block Parameters: Simulation 3D Vehicle with Ground Following

Simulation 3D Vehicle with Ground Following (mask) (link)

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: Small pickup truck

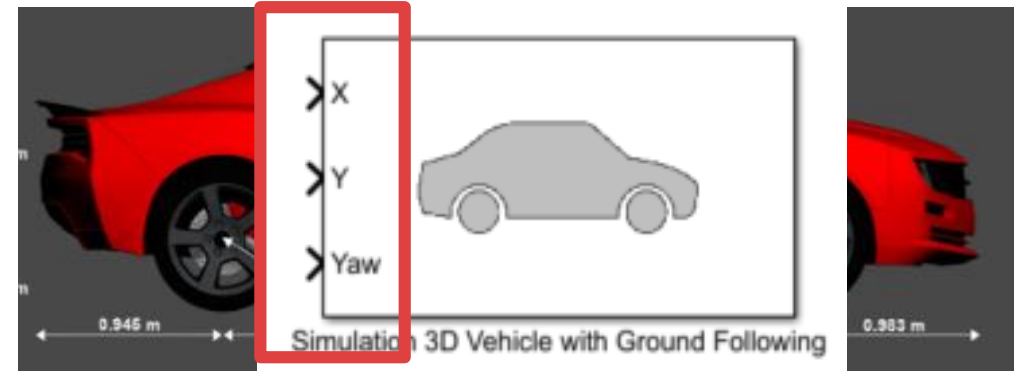
Hatchback

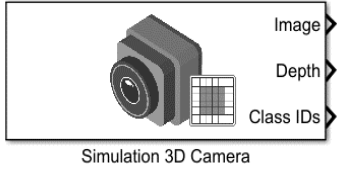
Initial rotation [Roll, Pitch, Yaw] (deg): [0, 0, 0]

Name: SimulinkVehicle1

Sample time: -1

OK Cancel Help Apply





Sensor example: video camera

Block Parameters: Simulation 3D Camera

Simulation 3D Camera (mask) (link)

Provides an interface to a camera with a lens in the 3D visualization environment. The block uses the focal length, radial distortion, and tangential distortion to model the lens. If you set the sample time to -1, the block uses the sample time specified in the Simulation 3D Scene Configuration block. To use this sensor, ensure that the Simulation 3D Scene Configuration block is in your model.

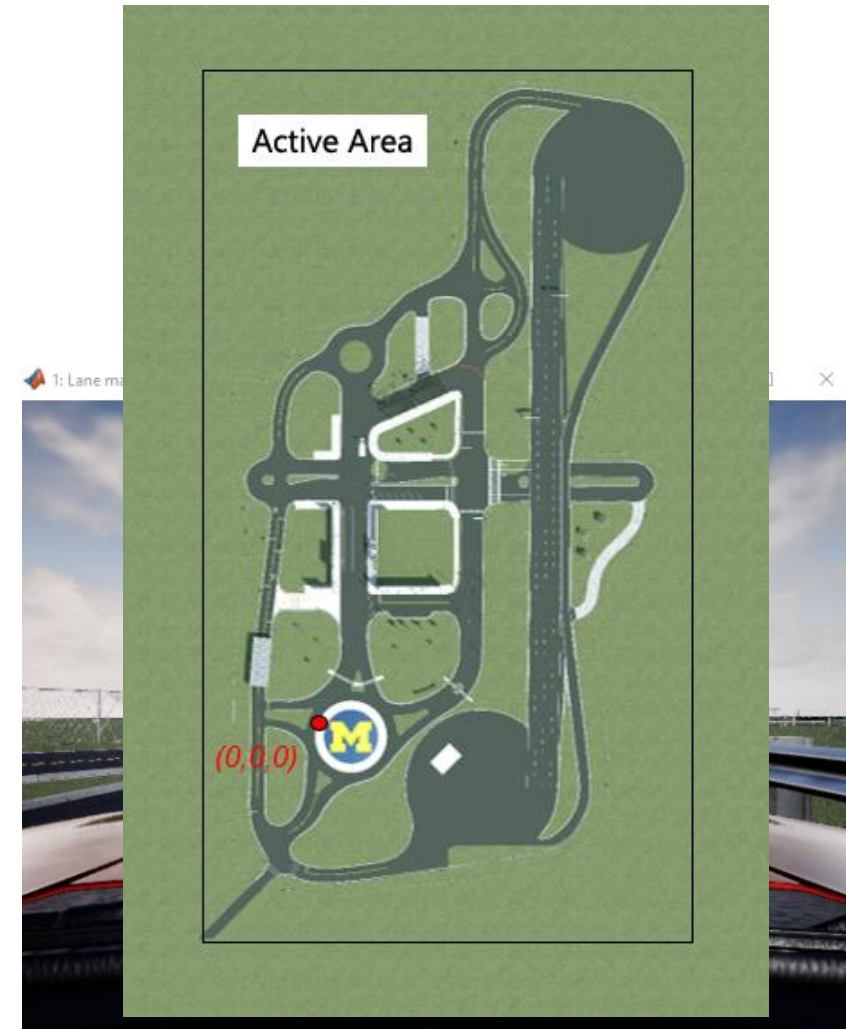
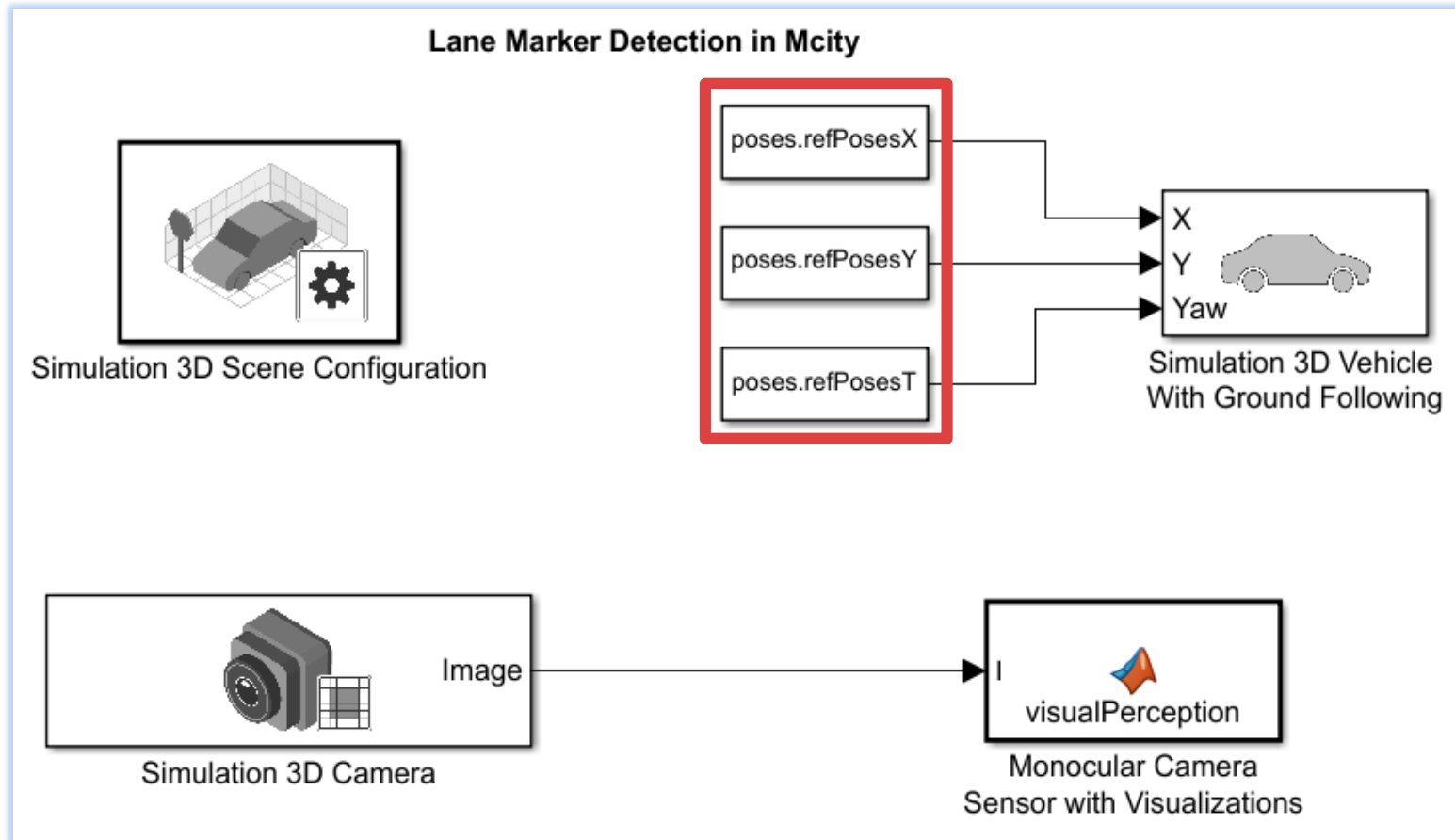
Mounting Parameters Ground Truth

- Output depth:
- Output semantic segmentation:
- Output location (m) and orientation (rad):

Sample time:

OK Cancel Help Apply

Putting it all together in a simple model



More in Automated Driving Toolbox

Featured Examples



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.

[Open Model](#)



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.

[Open Script](#)



Visualize Automated Parking Valet Using 3D Simulation

Visualize vehicle motion in a 3D simulation environment using an automated parking valet system constructed in Simulink.

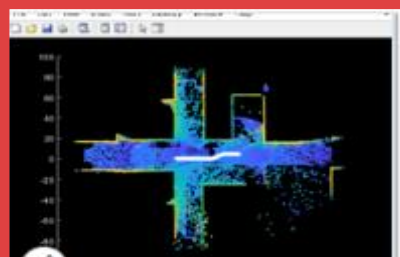
[Open Model](#)



Simulate Radar Sensors in 3D Environment

Implement a synthetic data simulation for tracking and sensor fusion using Simulink and a 3D simulation environment.

[Open Model](#)



Simulate Lidar Sensor Perception Algorithm

Develop a lidar perception algorithm using data recorded from a 3D simulation environment, and simulate within that environment.

[Open Model](#)

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!