MATLAB EXPO 2018

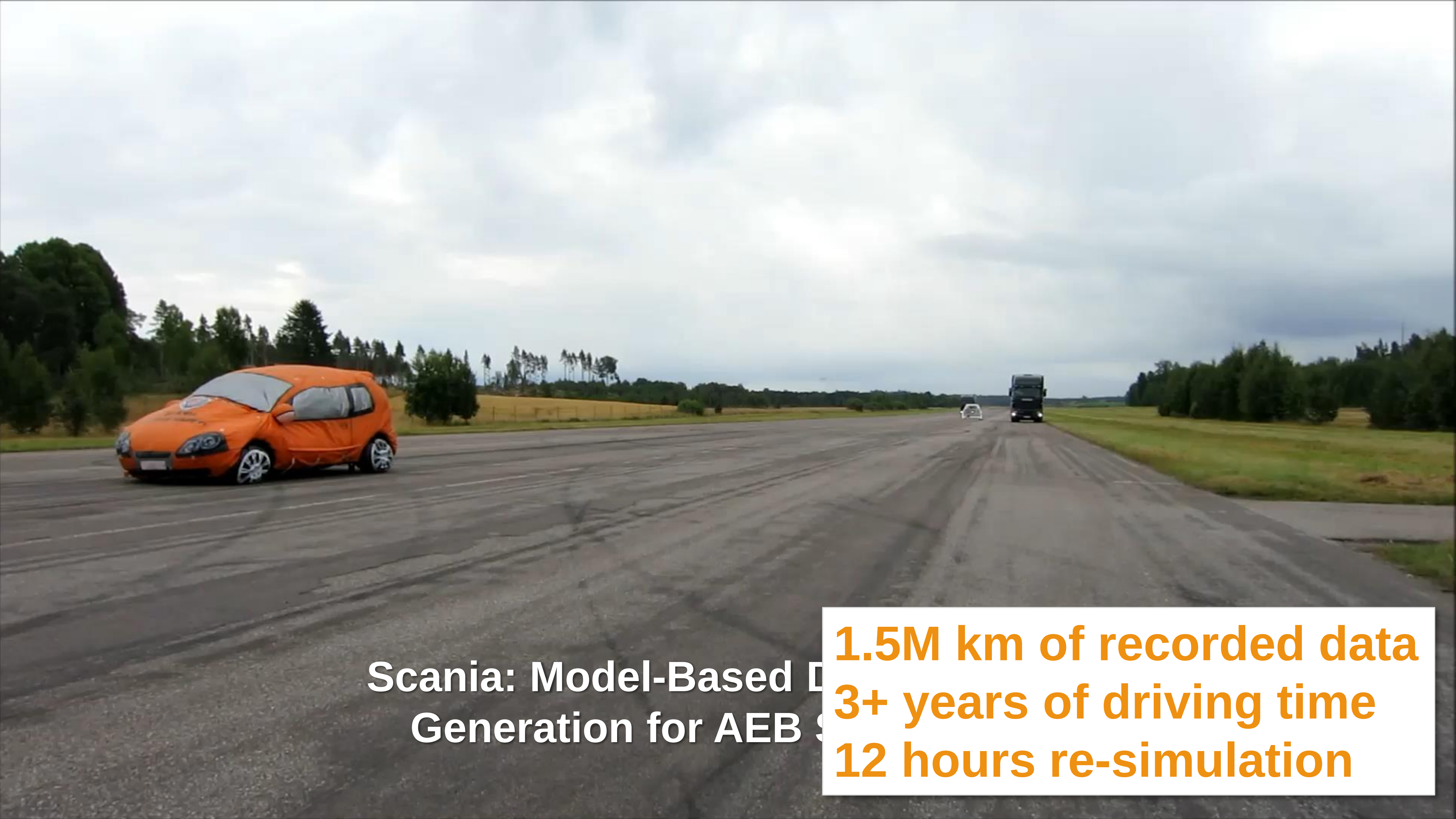
Automated Driving Development with MATLAB® and Simulink®

MANOHAR REDDY M

Using Model-Based Design to develop high quality and reliable Active Safety & Automated Driving Systems

Jonny Andersson
Senior Engineer
Scania

Thorsten Gerke
Automotive Industry Manager EMEA
MathWorks


Scania: Model-Based
Generation for AEB
1.5M km of recorded data
3+ years of driving time
12 hours re-simulation

Voyage Develops Longitudinal Controls for Self-Driving Taxis

Challenge

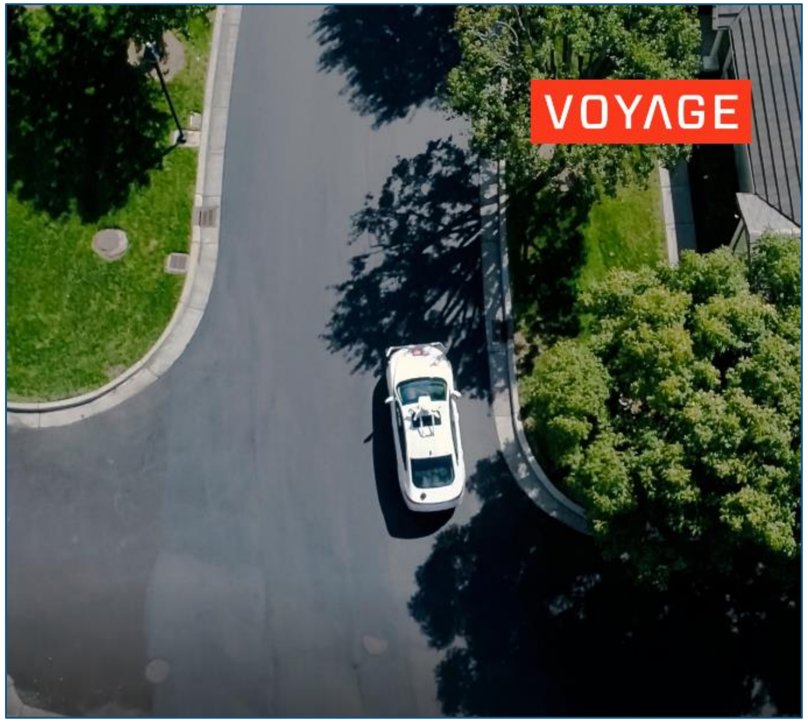
Develop a controller that enables a self-driving car to maintain a target velocity and keep a safe distance from obstacles.

Solution

Use Simulink® to design a longitudinal model predictive controller. Tune parameters based on experimental data imported into MATLAB®. Deploy the controller as an ROS node using Robotics System Toolbox™. Generate source code with Simulink Coder™, and package it as a Docker container.

Results

- Development speed tripled
- Easy integration with open-source software
- Simulink algorithms delivered as production software



Voyage's self-driving car in San Jose, California.

"We were searching for a prototyping solution that was fast for development and robust for production. We decided to go with Simulink for controller development and code generation, while using MATLAB to automate development tasks."

- Alan Mond, Voyage

How can you use MATLAB and Simulink to develop automated driving algorithms?

Perception

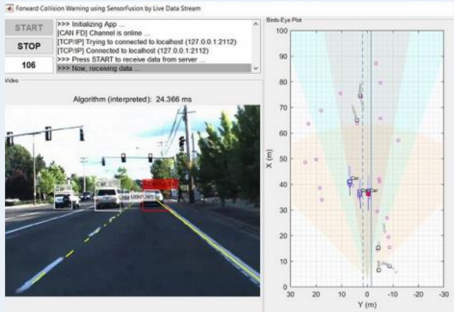
Control

Planning

Examples of how you can use MATLAB and Simulink to develop automated driving algorithms

Deep learning

Sensor fusion
with live data

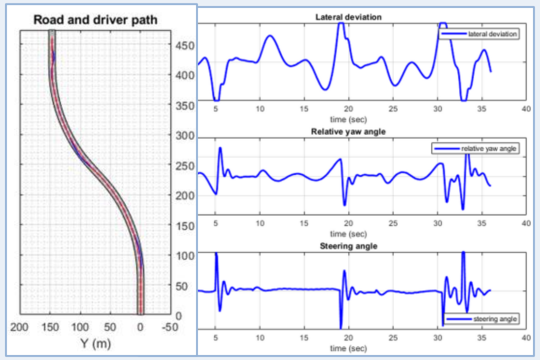


Perception

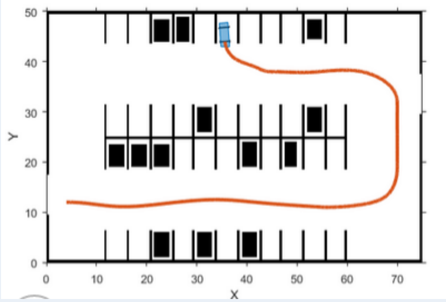
Control

Planning

Sensor models &
model predictive control



Path planning



How can you use MATLAB and Simulink to develop perception algorithms?

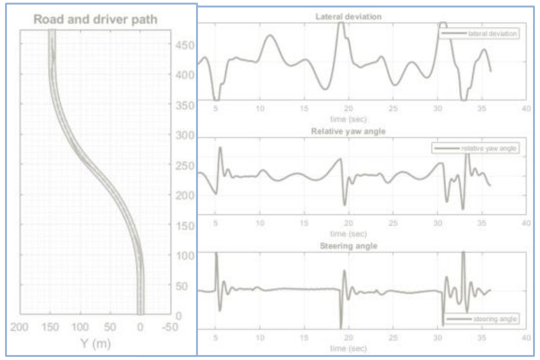
Deep learning

Perception

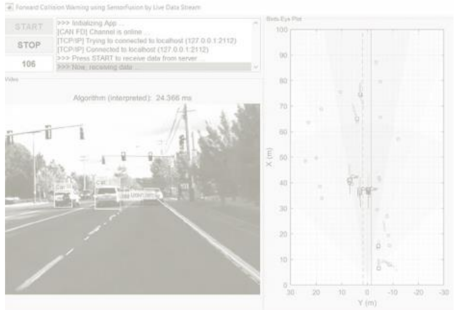
Control

Planning

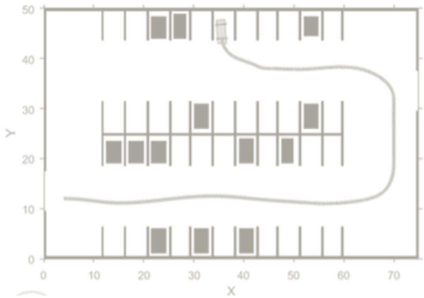
Sensor models & model predictive control



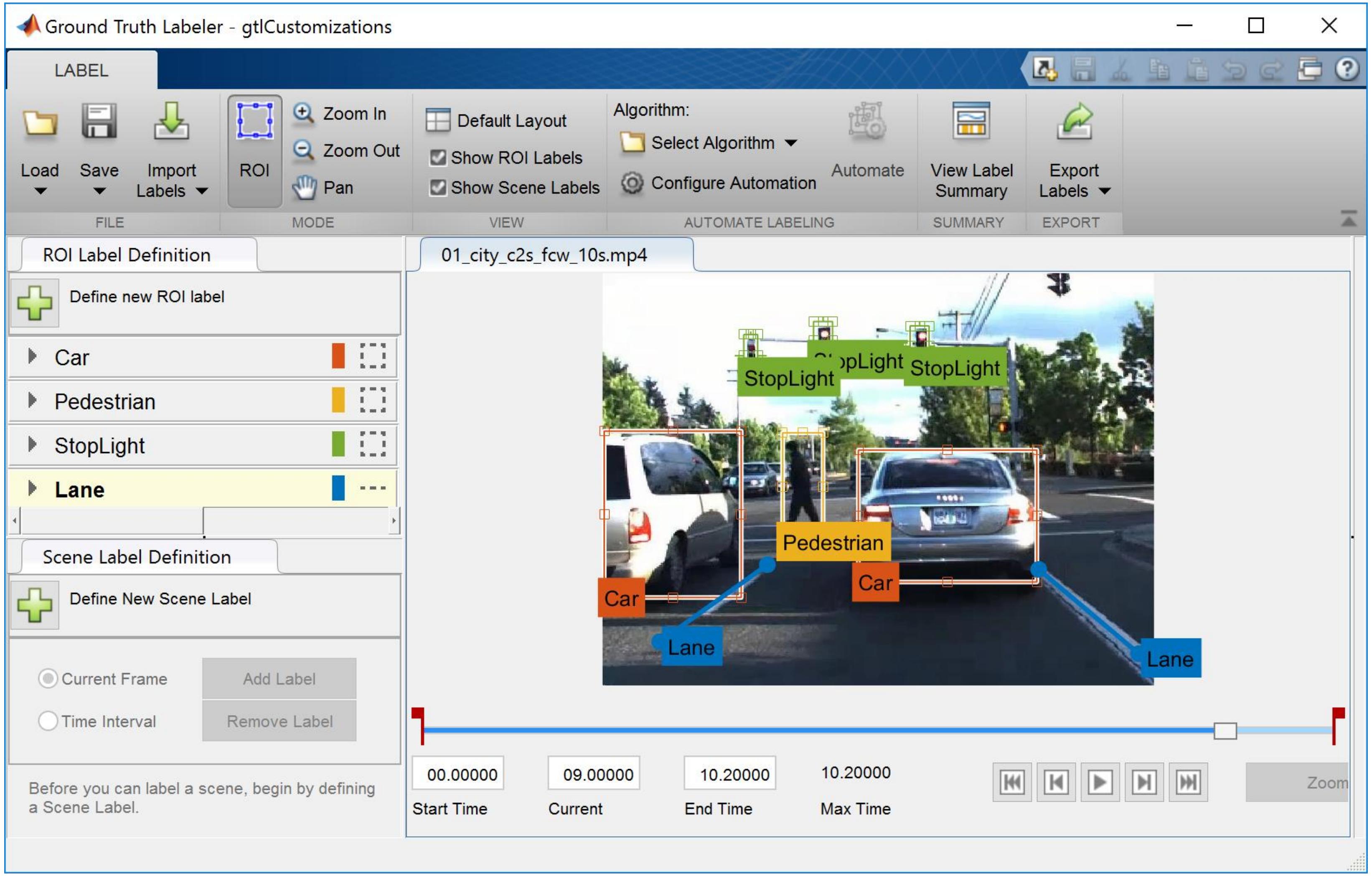
Sensor fusion with live data



Path planning

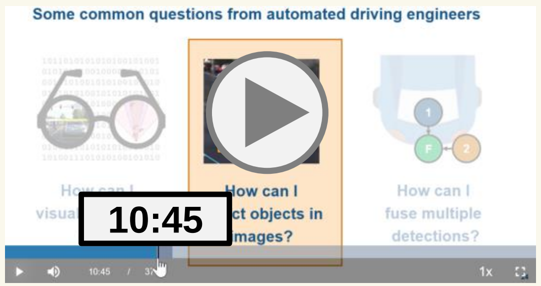


Automated Driving System Toolbox introduced:
Ground Truth Labeling App to label video data



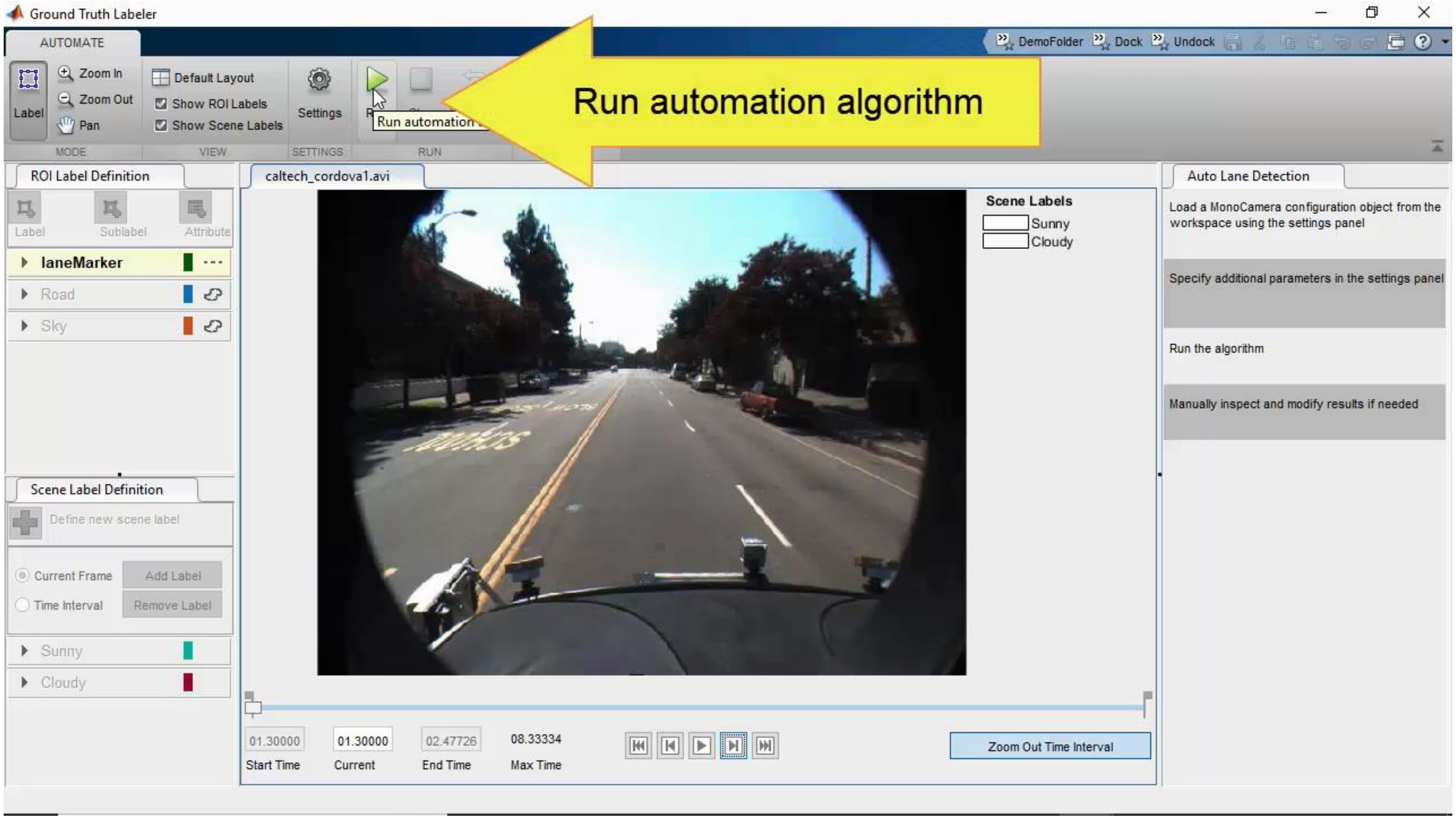
R2017a

Videos and Webinars



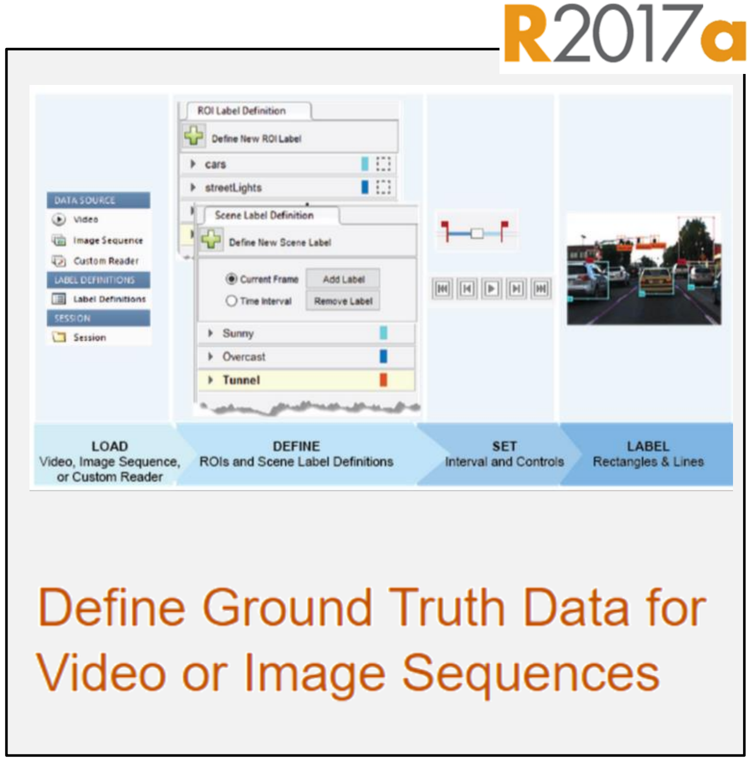
Introduction to Automated Driving System Toolbox

Automate labeling lanes with Ground Truth Labeler



Automated Driving System Toolbox introduced examples to:

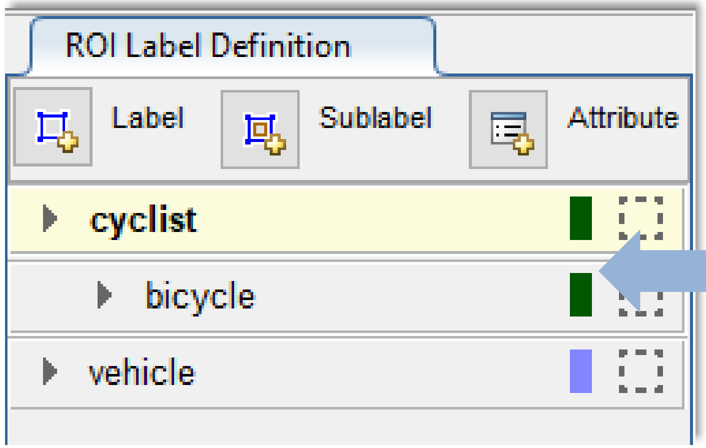
Accelerate the process of Ground Truth Labeling

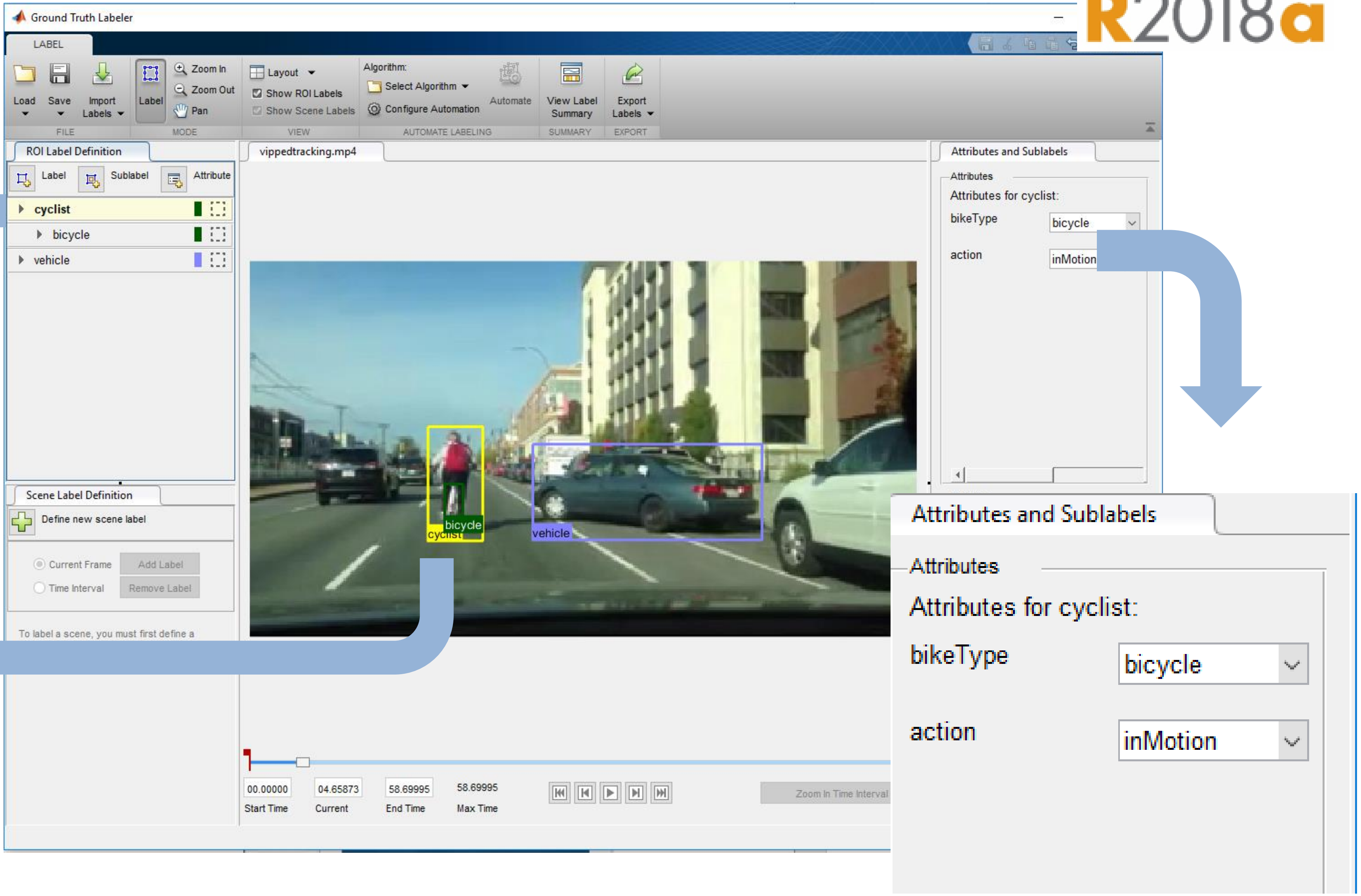


- **Label detections** with Ground Truth Labeler App
- **Add your own automation algorithm** to Ground Truth Labeler App
- **Extend connectivity** of Ground Truth Labeler App to visualize lidar data

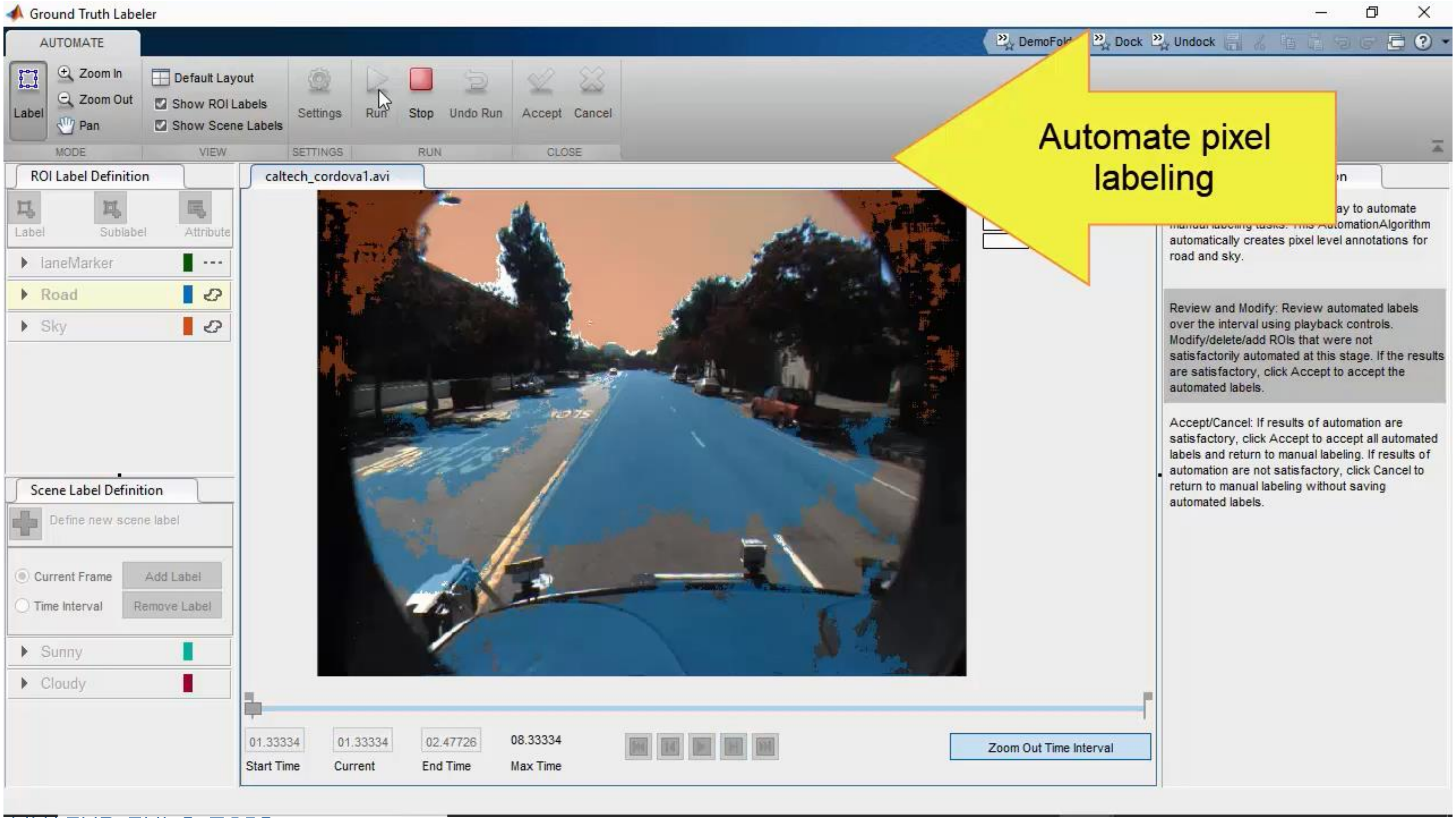
Specify attributes and sublabels in Ground Truth Labeler App

R2018a

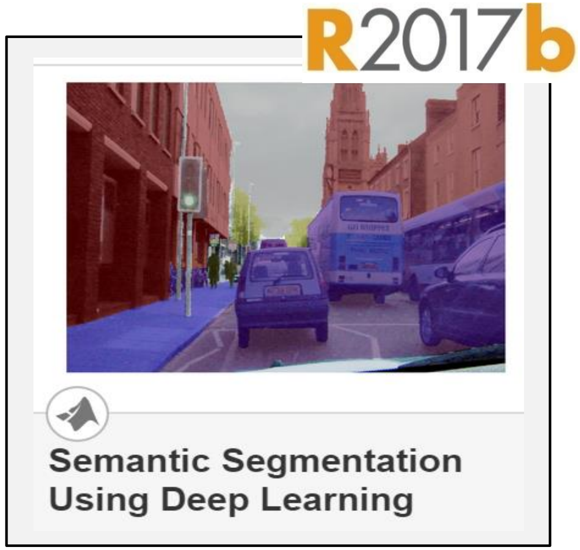




Automate labeling pixels with Ground Truth Labeler

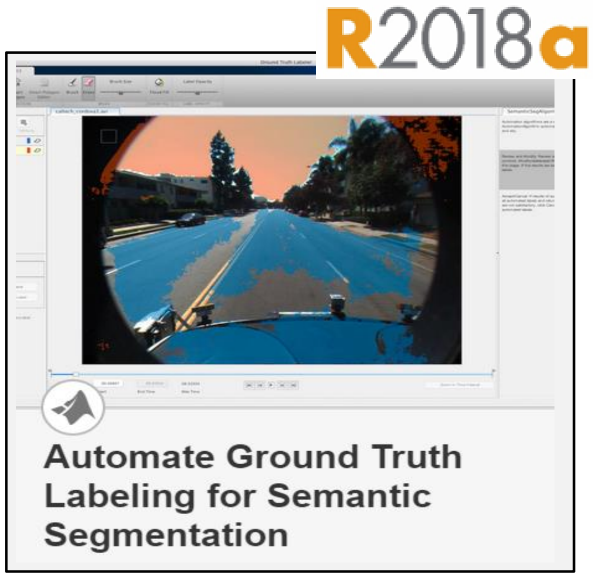


Learn more about developing deep learning perception algorithms with these examples



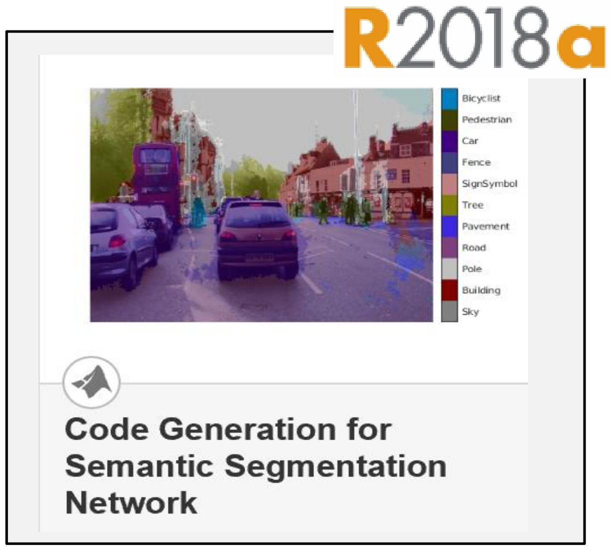
- **Train free space detection network** using deep learning

Computer Vision System Toolbox™



- **Add semantic segmentation** automation algorithm to Ground Truth Labeler App

Automated Driving System Toolbox™



- **Generate CUDA® code** to execute directed acyclic graph network on an NVIDIA GPU

GPU Coder™

Robotics and Autonomous Systems	
14:30	Demystifying Deep Learning *Dr. Amod Anandkumar, MathWorks*

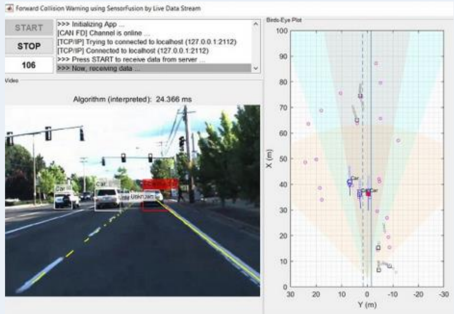
Robotics and Autonomous Systems	
15:30	Deploying Deep Neural Networks to Embedded GPUs and CPUs *Rishu Gupta, Ph.D, MathWorks*

Free Space Detection Using Semantic Segmentation

How can you use MATLAB and Simulink to develop perception algorithms?

Deep learning

Sensor fusion with live data

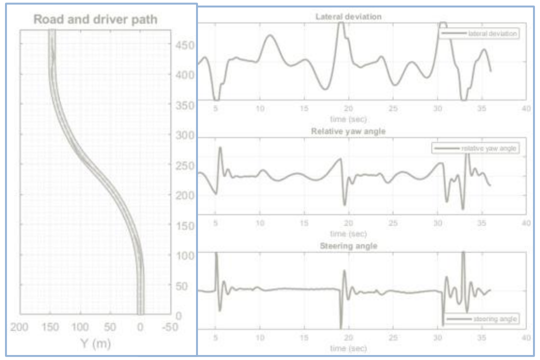


Perception

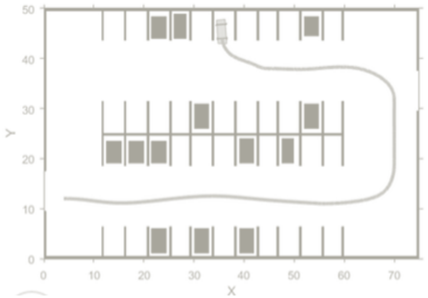
Control

Planning

Sensor models & model predictive control

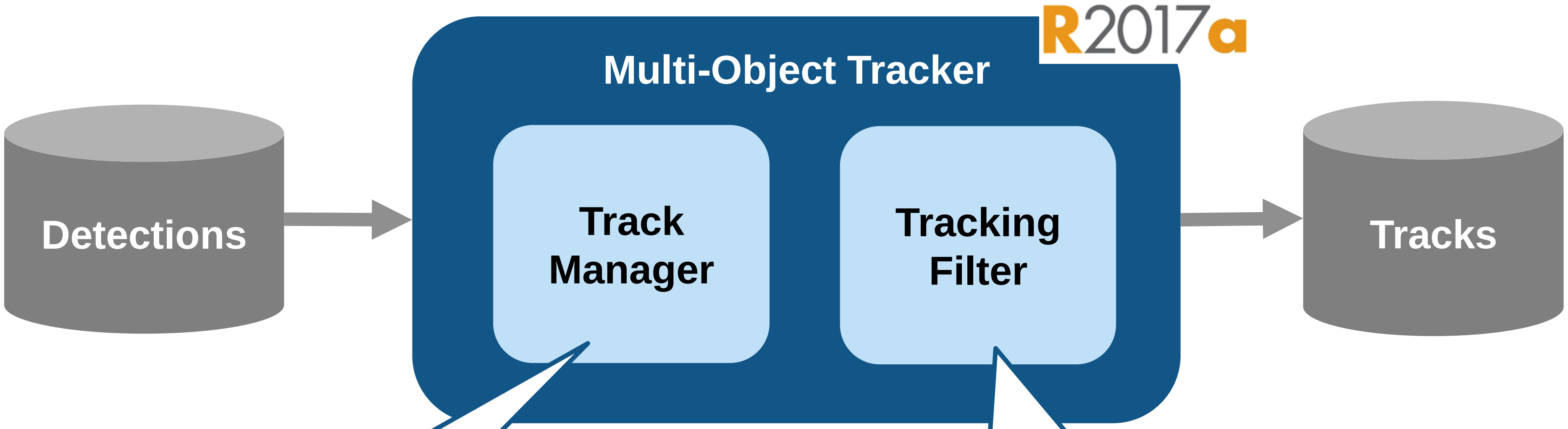


Path planning



Automated Driving System Toolbox introduced:

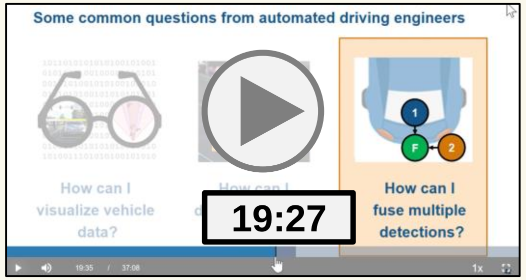
Multi-object tracker to develop sensor fusion algorithms



- Assigns detections to tracks
- Creates new tracks
- Updates existing tracks
- Removes old tracks

- Predicts and updates state of track
- Supports linear, extended, and unscented Kalman filters

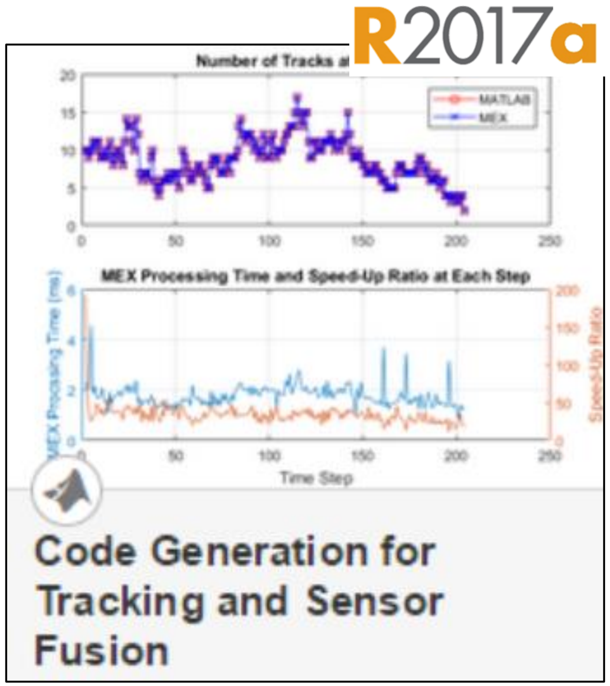
Videos and Webinars



Introduction to Automated Driving System Toolbox

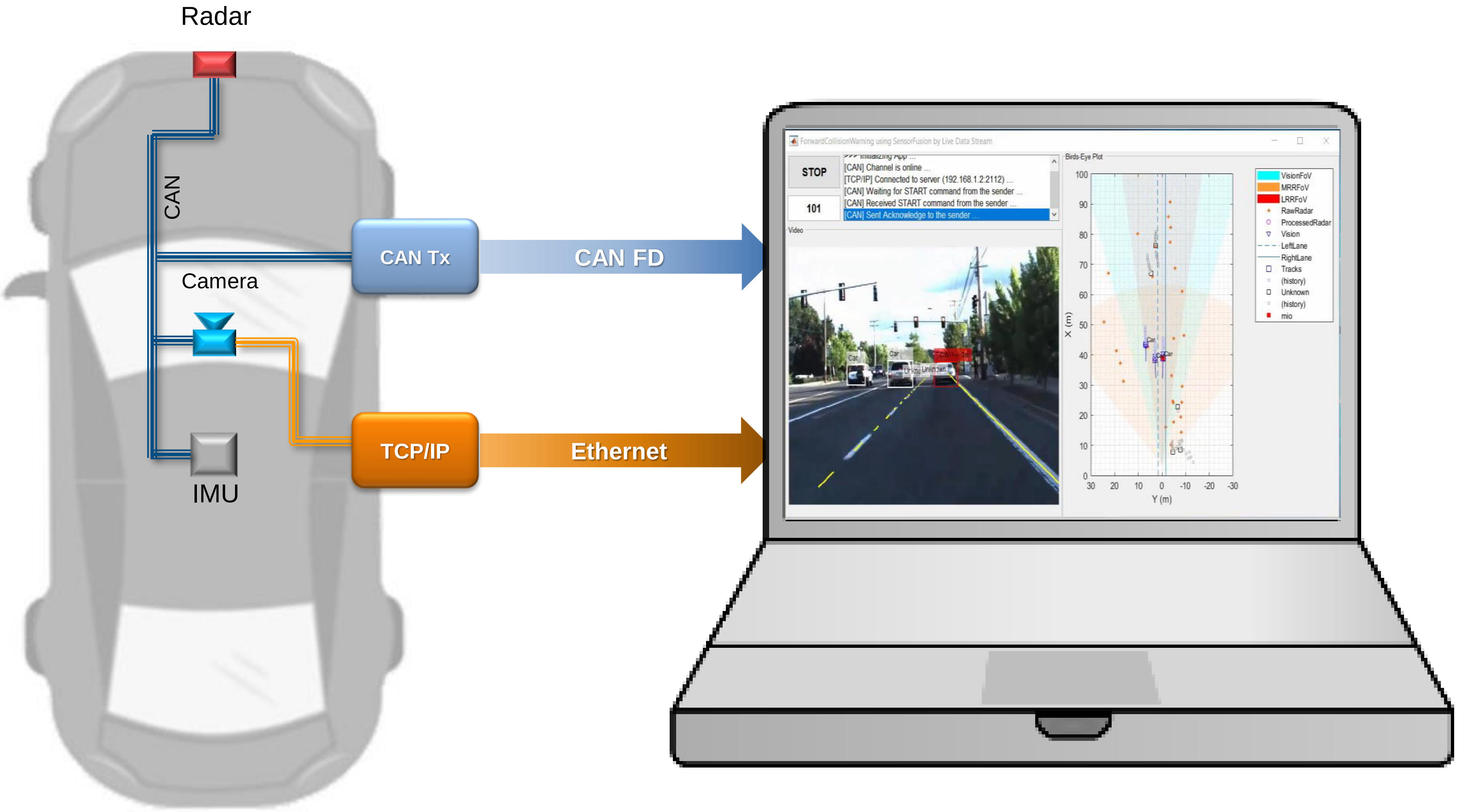
Automated Driving System Toolbox introduced examples to:

Develop sensor fusion algorithms with recorded data

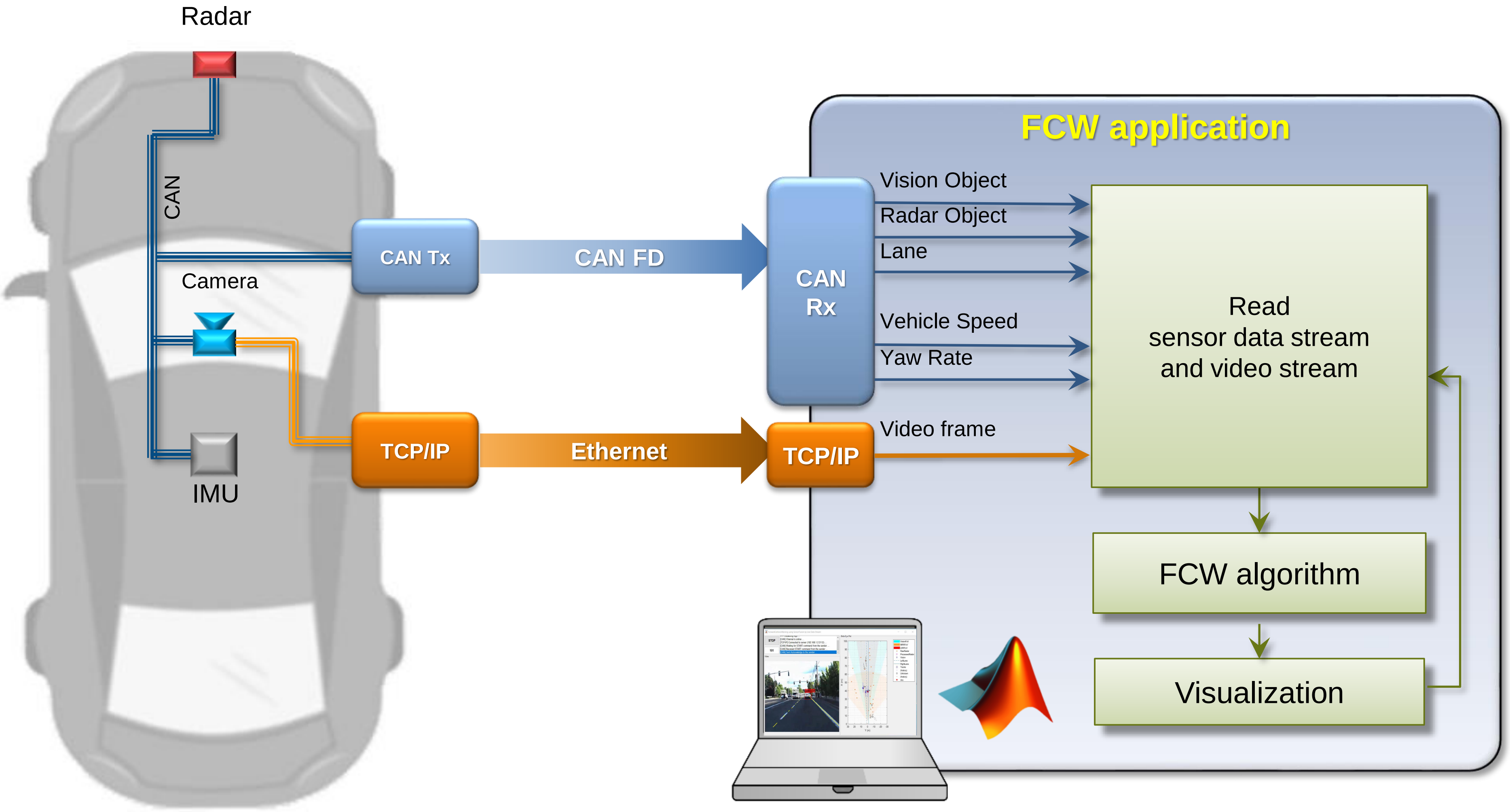


- **Design** multi-object tracker based on logged vehicle data
- **Generate C/C++** code from algorithm which includes a multi-object tracker

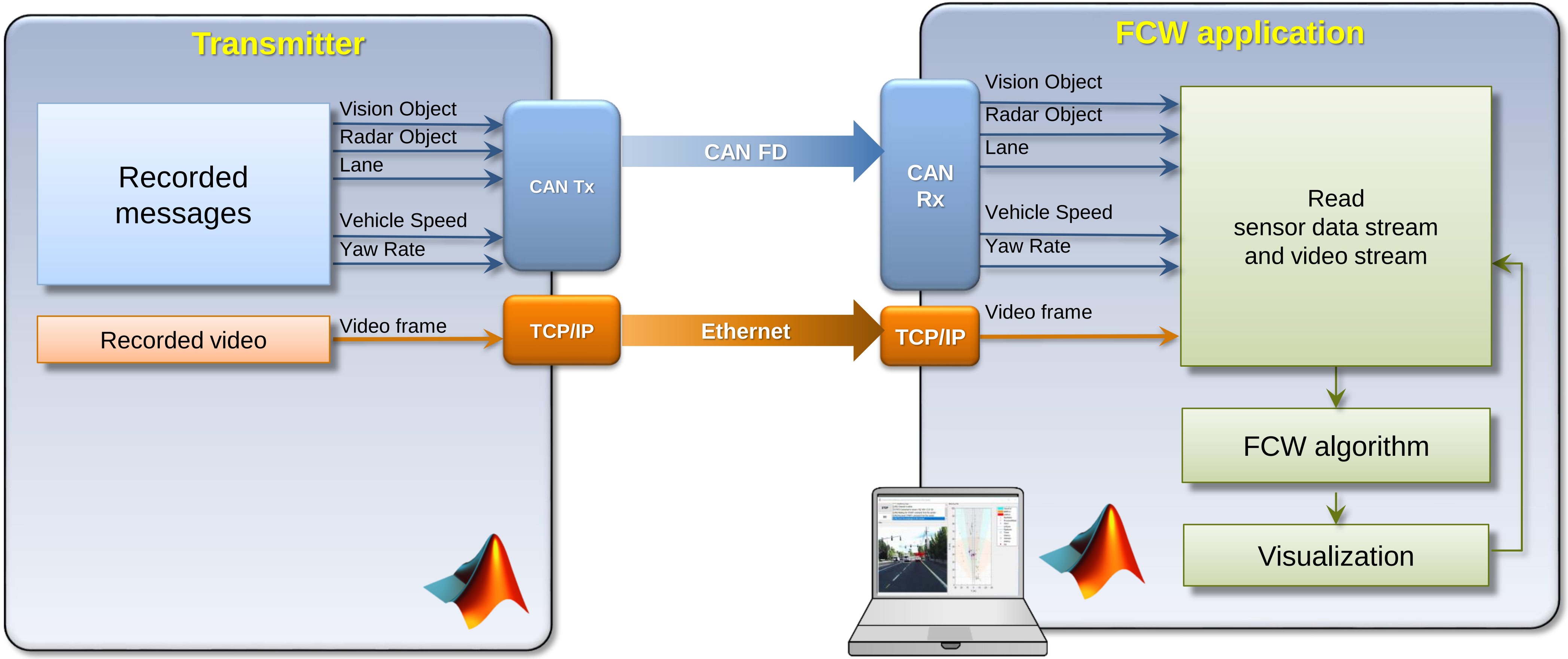
How can I test my sensor fusion algorithm with live data?



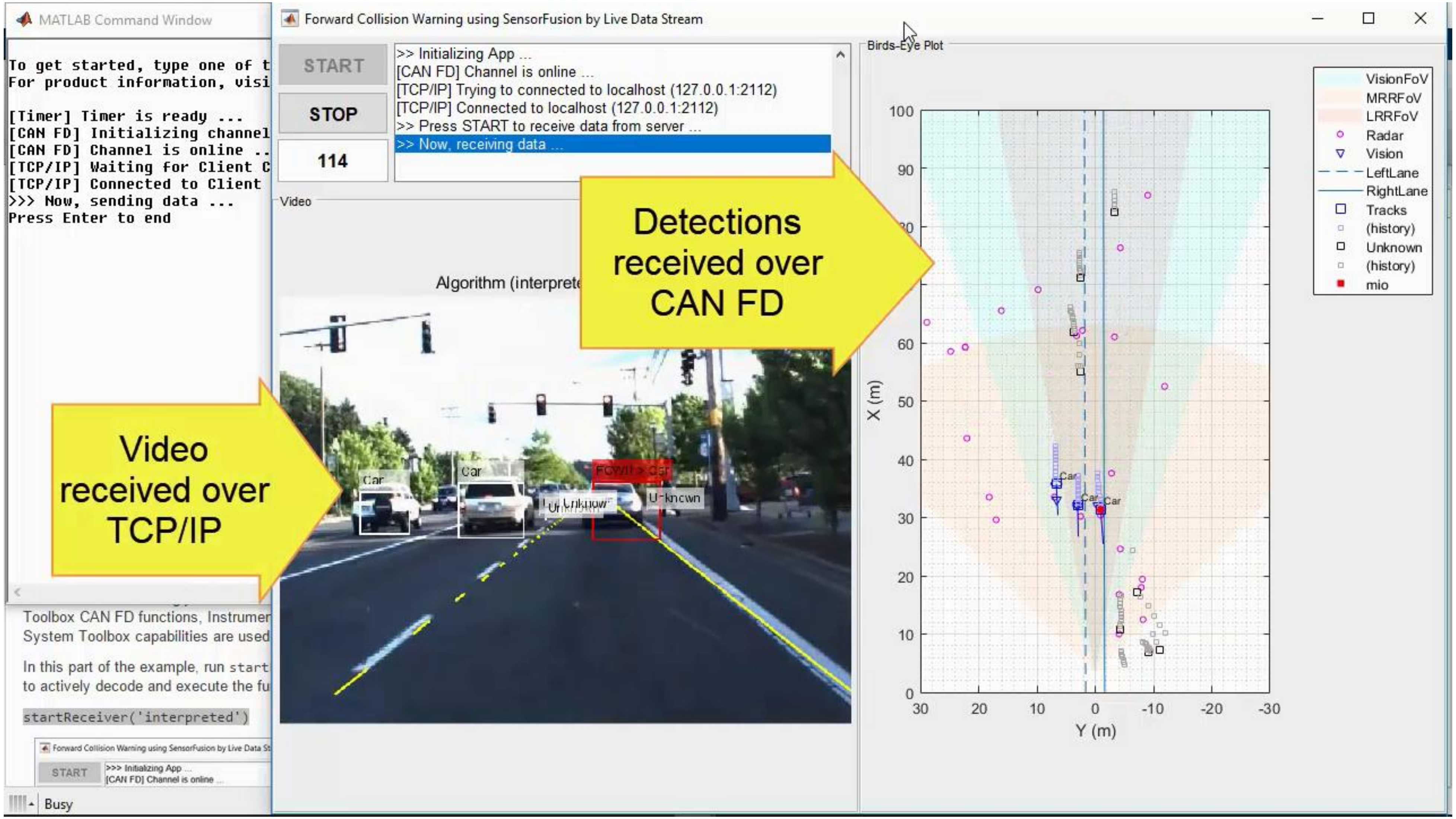
Test forward collision warning algorithm with live data from vehicle



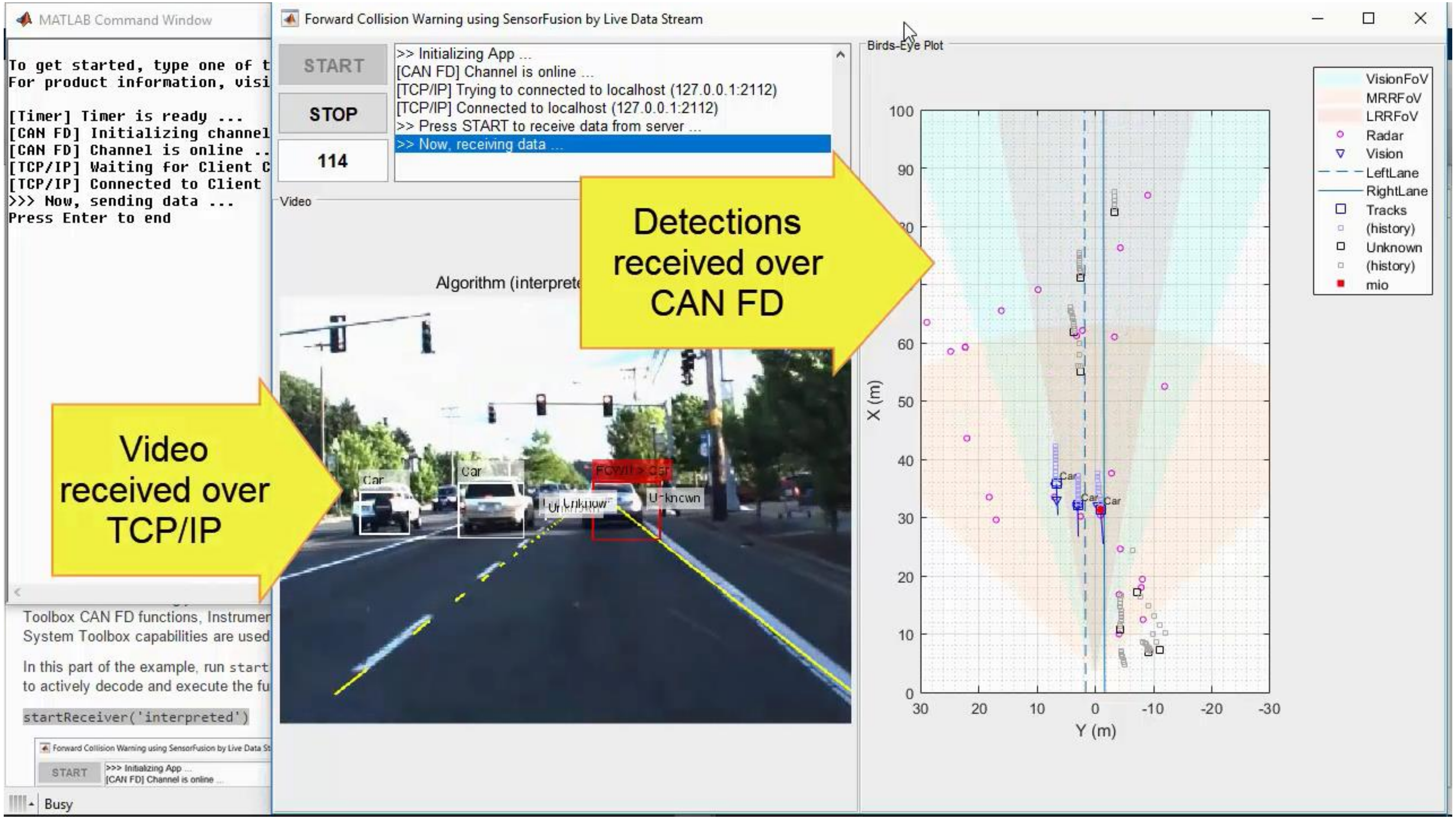
Test forward collision warning algorithm with live data from "surrogate" vehicle



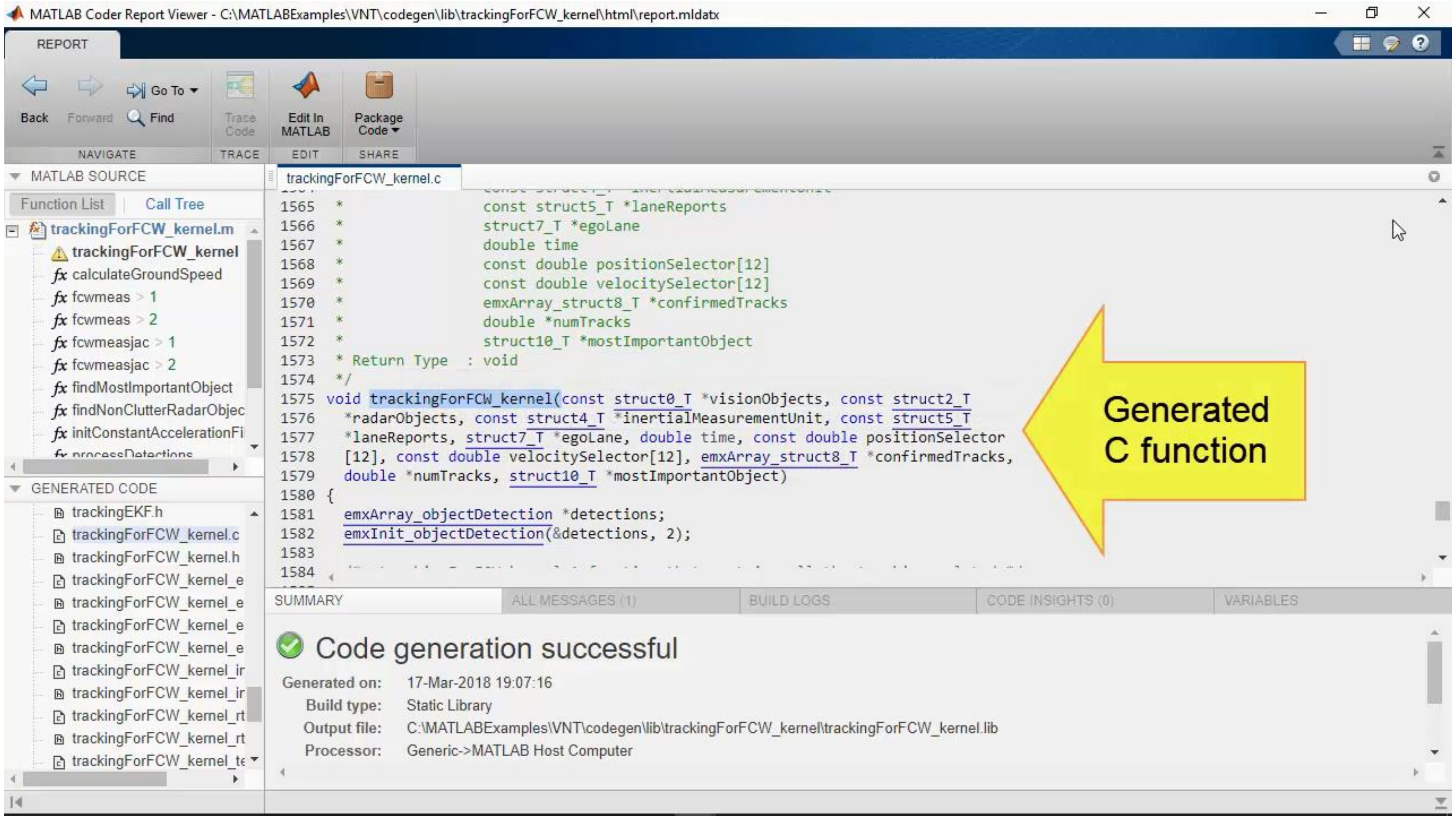
Send live CAN FD and TCP/IP data



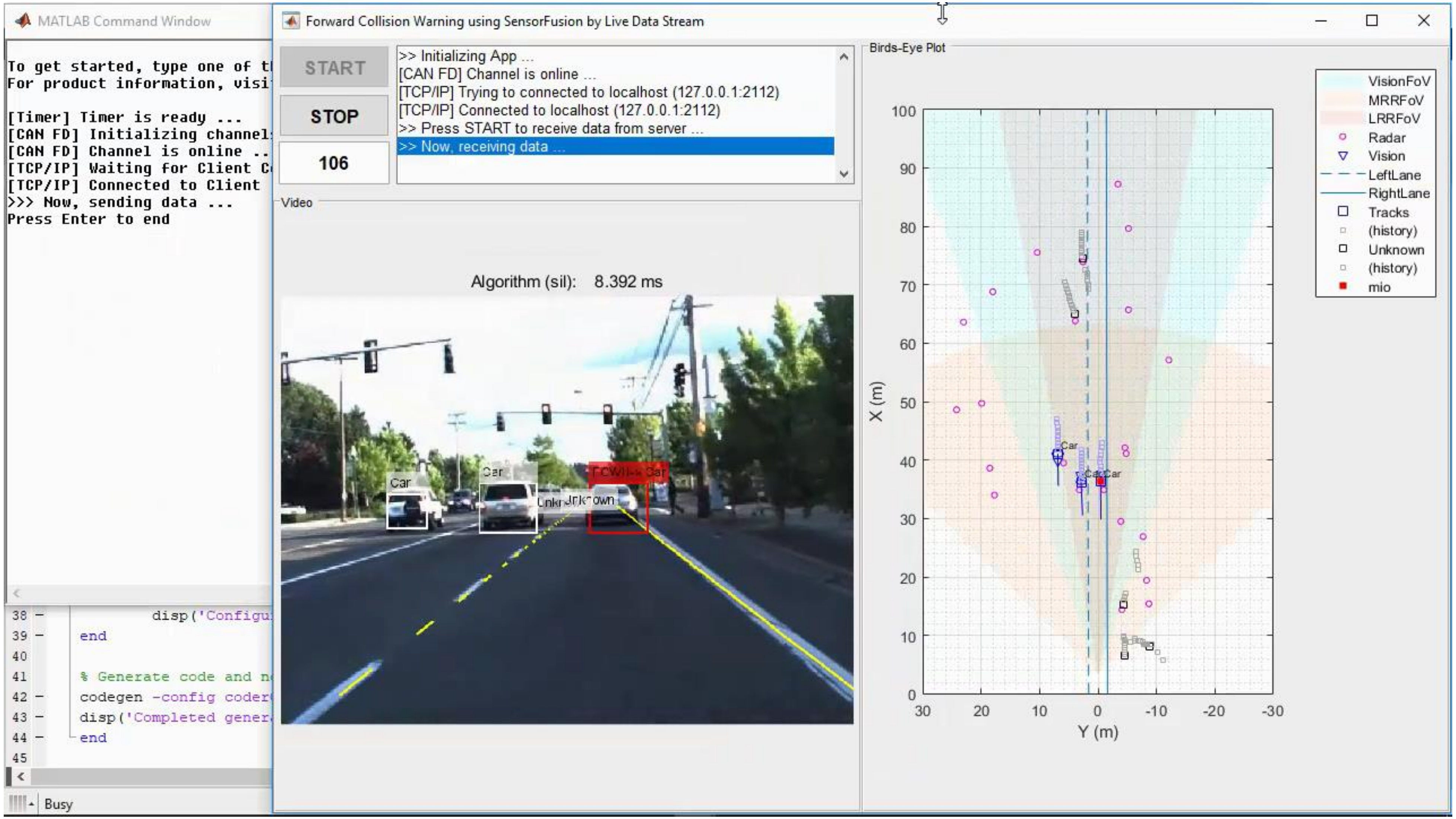
Receive live CAN FD and TCP/IP data



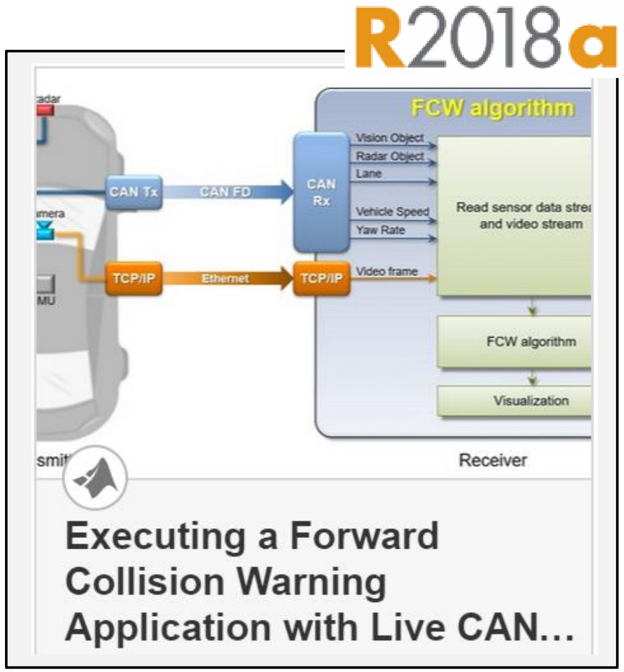
Generate C/C++ code for algorithm

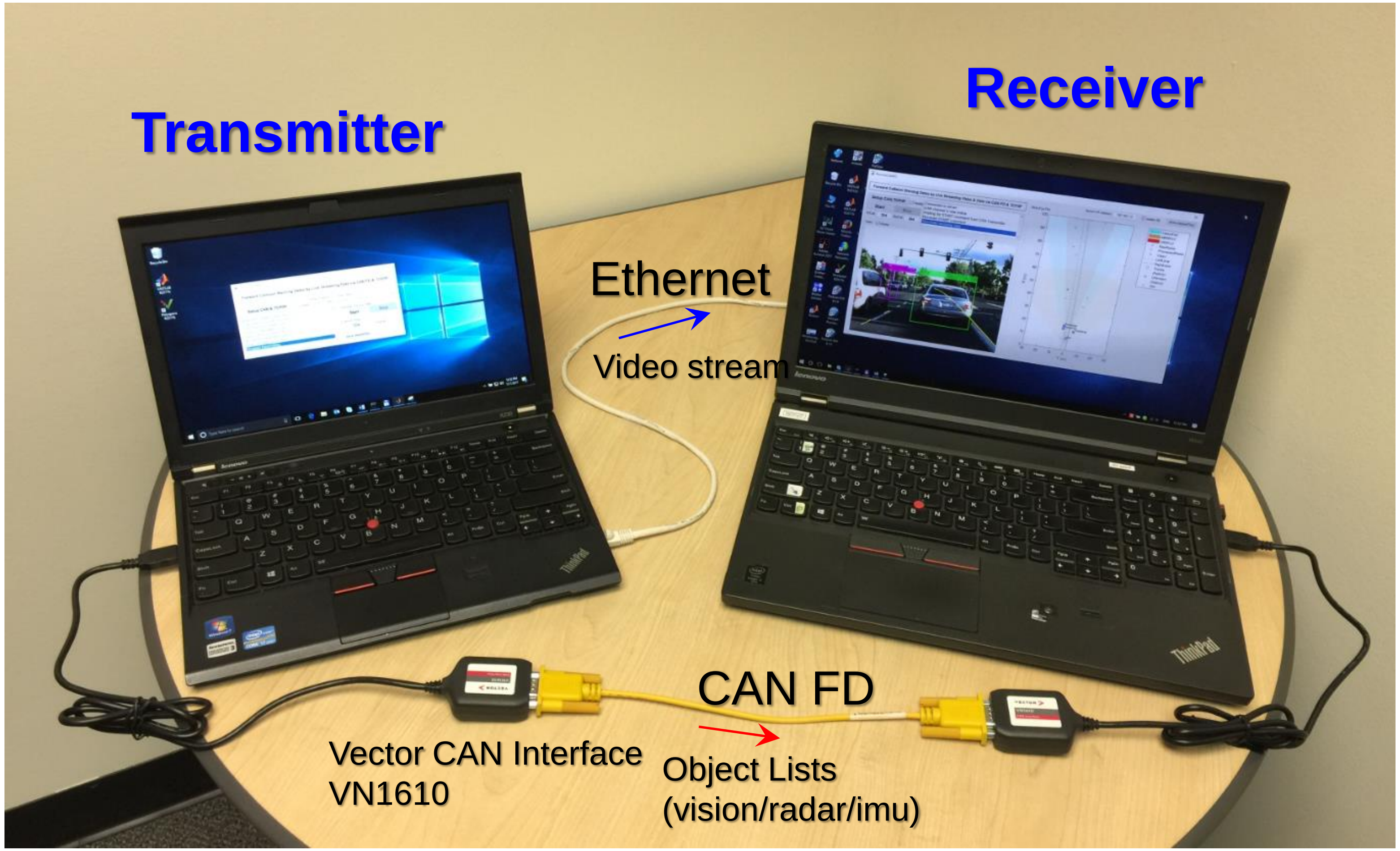


Stream live CAN FD and TCP/IP data into compiled algorithm code



Learn about developing sensor fusion algorithms with live data using this example





- **Stream CAN FD** data to prototype algorithms on your laptop

Vehicle Network Toolbox™

How can you use MATLAB and Simulink to develop control algorithms?

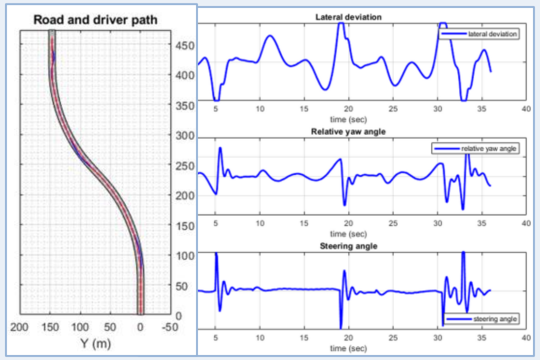
Deep learning

Sensor fusion
with live data

Perception

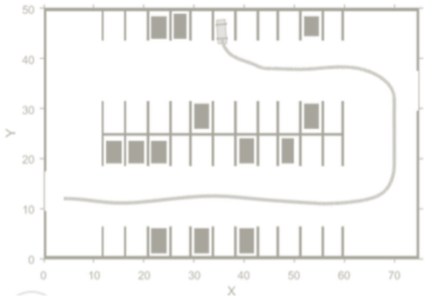
Control

Sensor models &
model predictive control



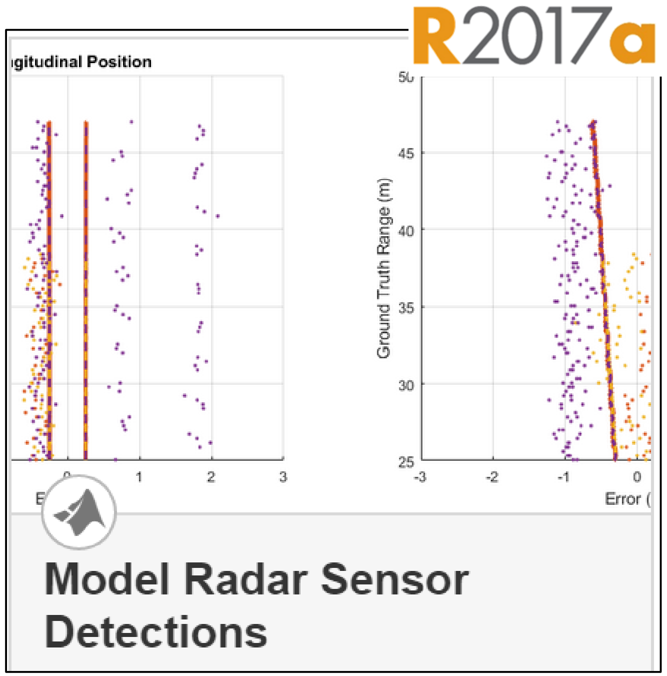
Planning

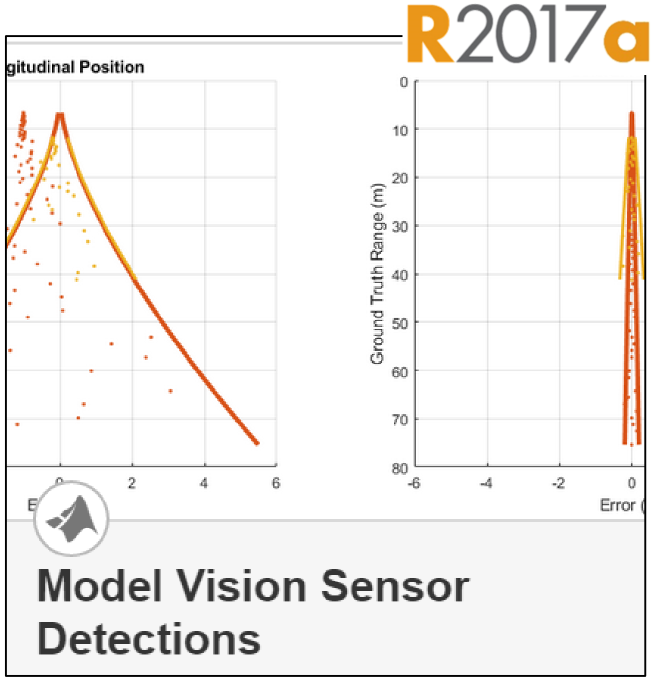
Path planning

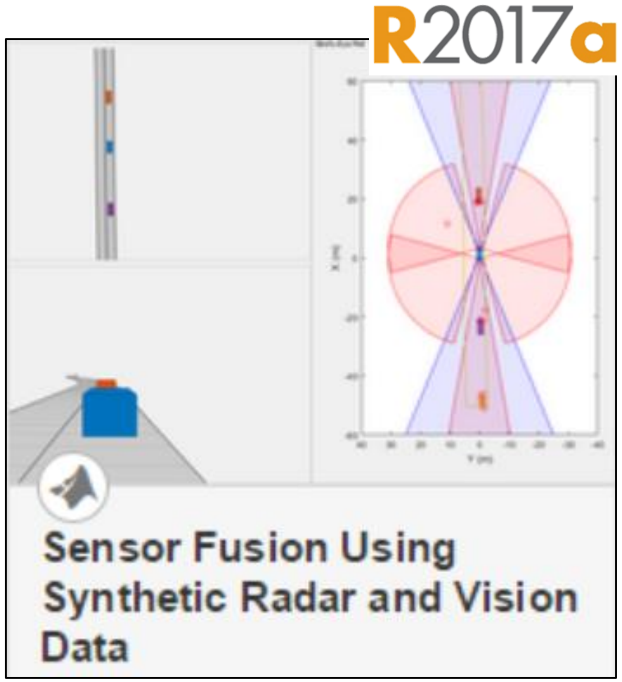


Automated Driving System Toolbox introduced examples to:

Synthesize detections to test sensor fusion algorithms



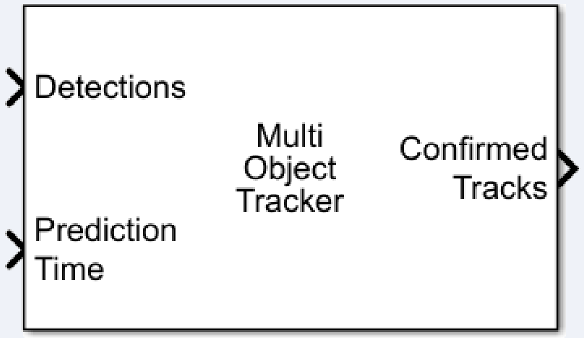


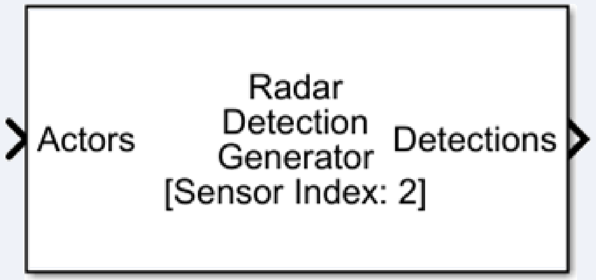


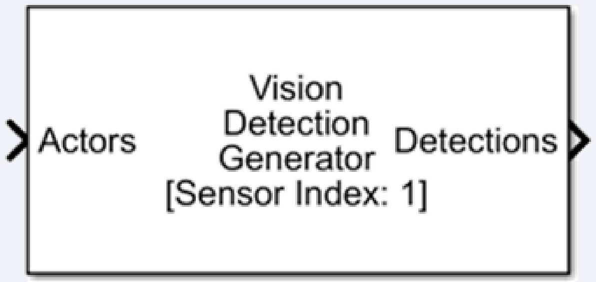
- **Synthesize radar** detections with probabilistic impairments
- **Synthesize vision** detections with probabilistic impairments
- **Synthesize scenario** to test multi-object tracker

Automated Driving System Toolbox introduced:

Radar and vision detections for closed loop simulation

<table>
<tr><th>Release</th><th>Fuse</th><th>Synthesize</th><th>Visualize</th></tr>
<tr><td>R2017a</td><td><code>multiObjectTracker</code></td><td><code>radarDetectionGenerator</code>
<code>visionDetectionGenerator</code></td><td><code>birdsEyePlot</code></td></tr>
<tr><td>R2017b</td><td>

Multi-Object Tracker</td><td>

Radar Detection Generator


Vision Detection Generator</td><td>

Bird's-Eye Plot</td></tr>
</table>

Voyage Develops Longitudinal Controls for Self-Driving Taxis

Challenge

Develop a controller that enables a self-driving car to maintain a target velocity and keep a safe distance from obstacles.

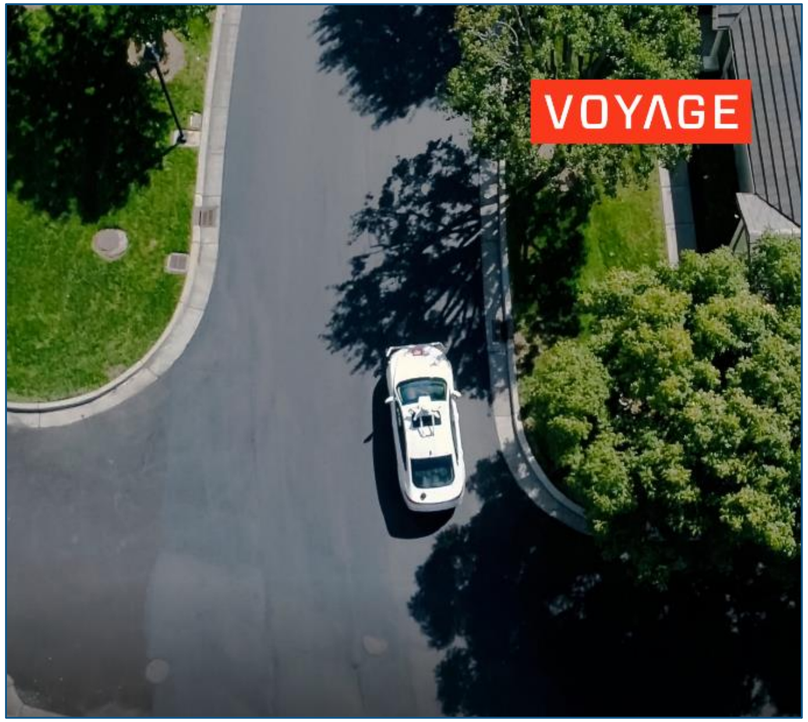
Solution

Use Simulink® to design a longitudinal model predictive controller. Tune parameters based on experimental data imported into MATLAB®. Deploy the controller as an ROS node using Robotics System Toolbox™. Generate source code with Simulink Coder™, and package it as a Docker container.

Results

- Development speed tripled
- Easy integration with open-source software
- Simulink algorithms delivered as production software

[Technical Article]



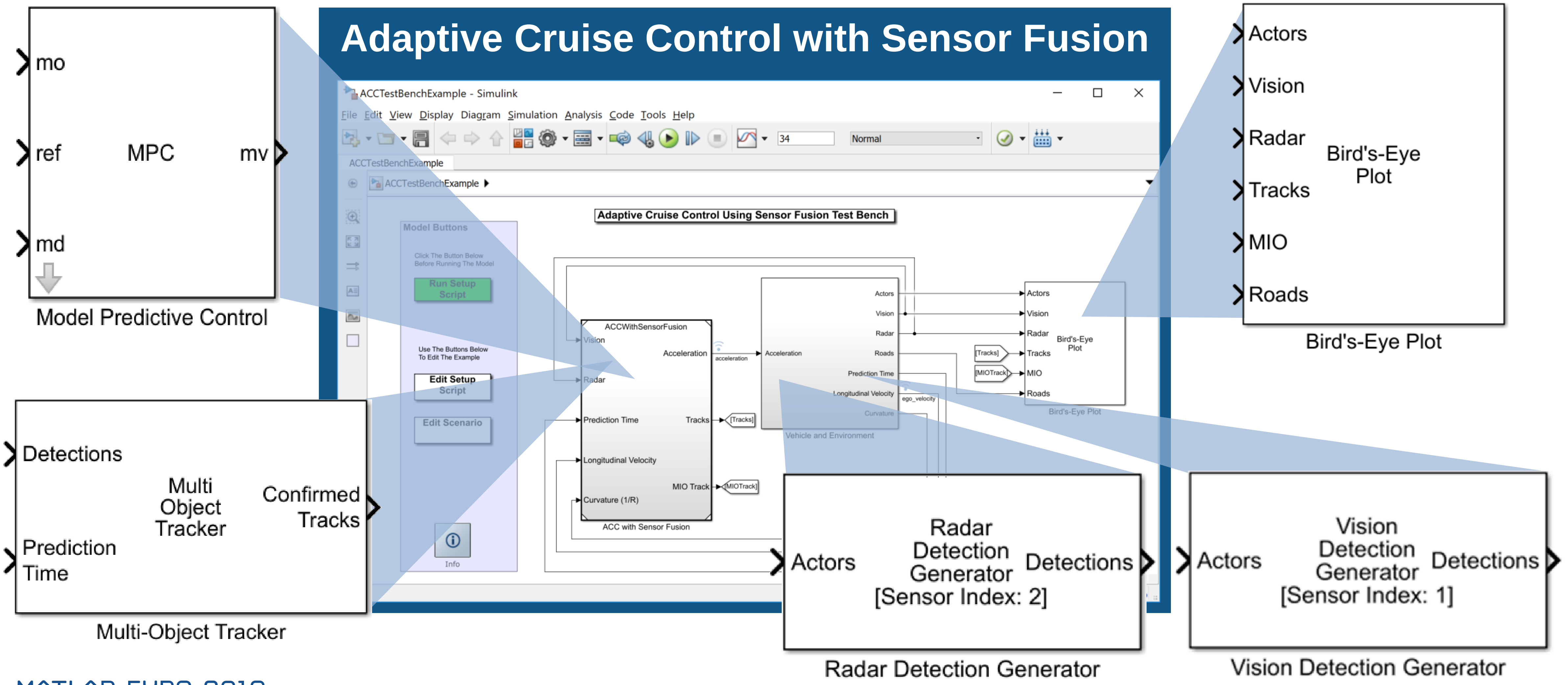
Voyage's self-driving car in San Jose, California.

"We were searching for a prototyping solution that was fast for development and robust for production. We decided to go with Simulink for controller development and code generation, while using MATLAB to automate development tasks."

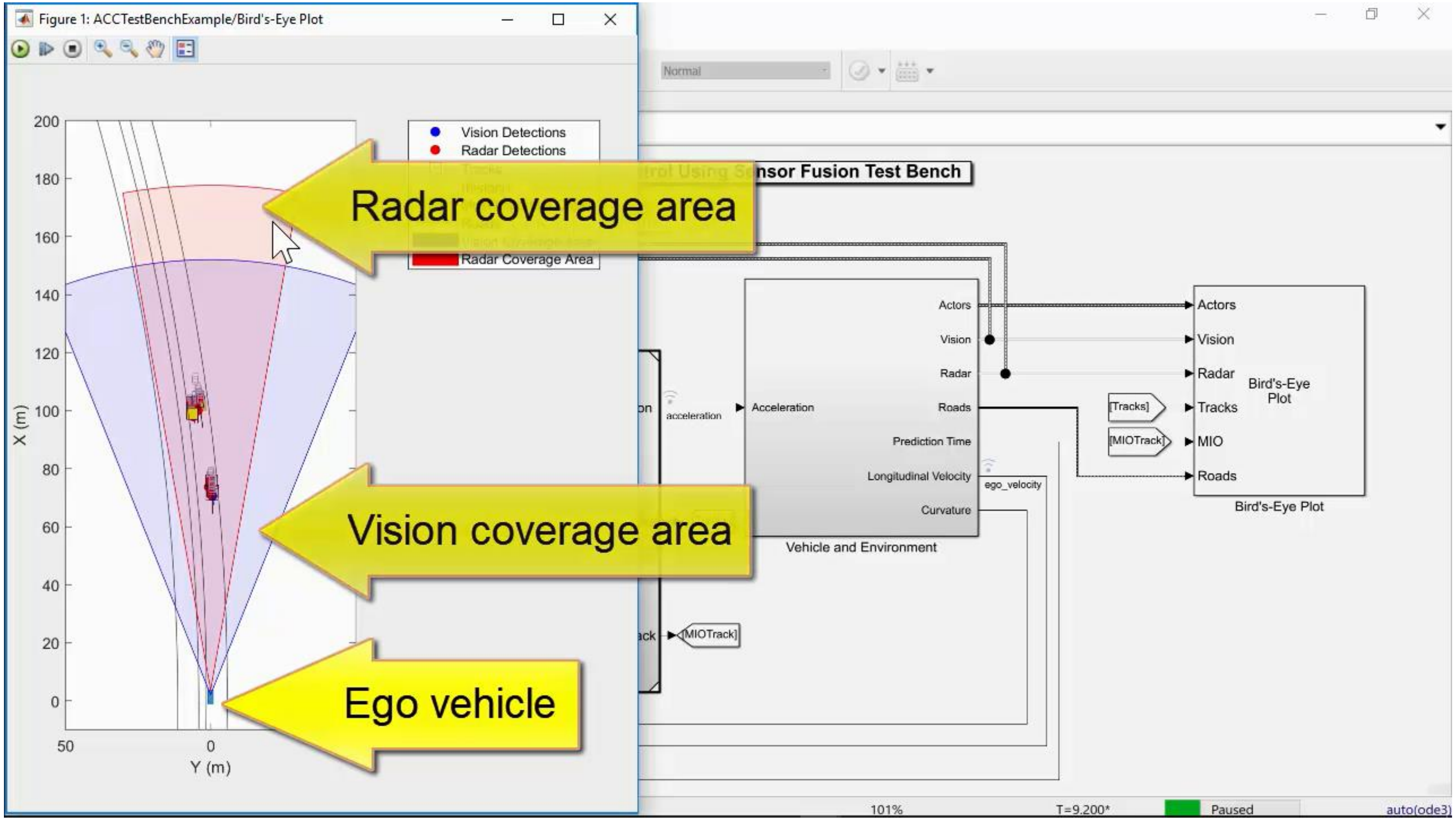
- Alan Mond, Voyage

Simulate closed loop system with radar/vision detections, sensor fusion, and model-predictive control

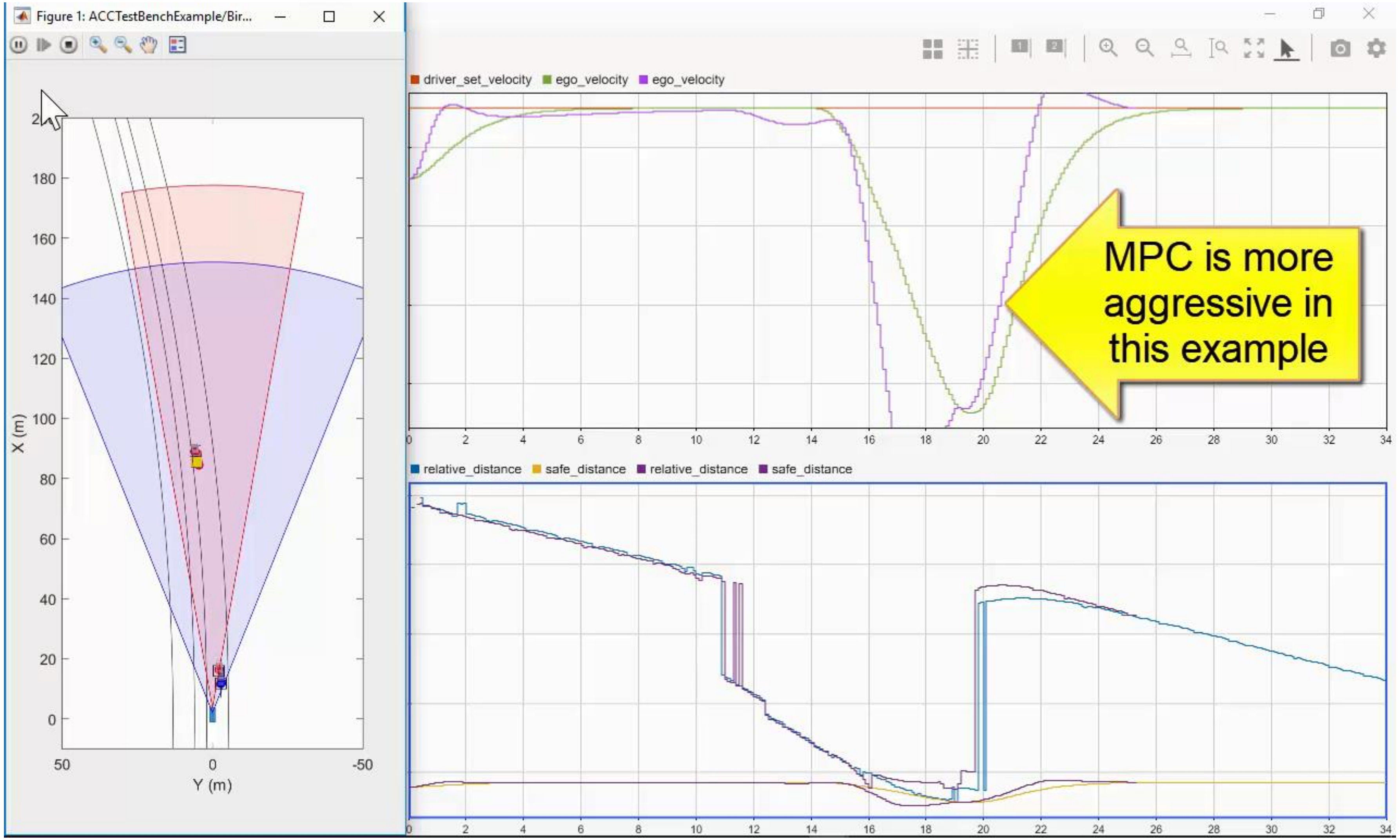
R2017b



Synthesize detections to test sensor fusion and model-predictive controller

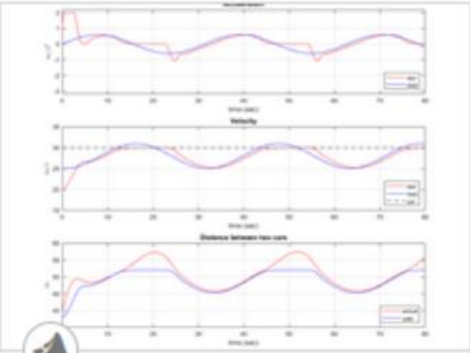


Compare classical and model predictive control algorithms



Automated Driving Applications with Model Predictive Controls

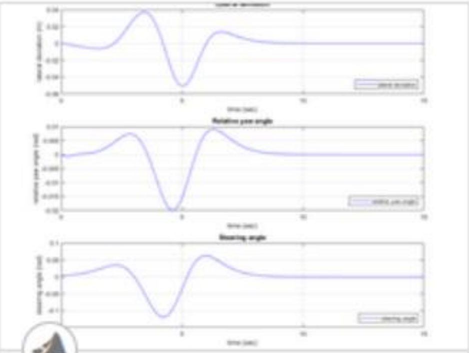
Automated Driving Applications



Adaptive Cruise Control System Using Model Predictive Control

Use the block in Simulink® and demonstrates the control objectives and constraints of this block.

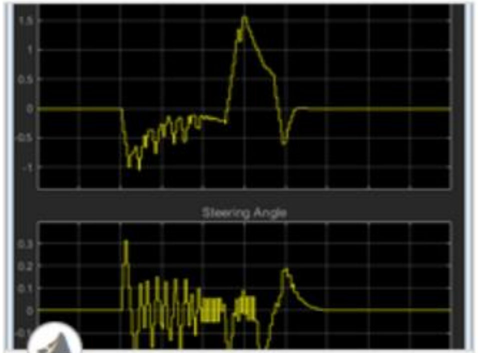
Open Script



Lane Keeping Assist System Using Model Predictive Control

Use the block in Simulink® and demonstrates the control objectives and constraints of this block.

Open Script



Obstacle Avoidance Using Adaptive Model Predictive Control

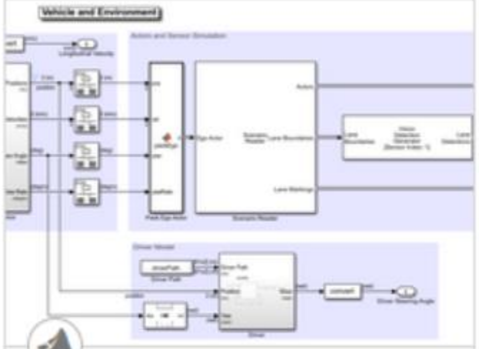
Make a vehicle (ego car) follow a reference velocity and avoid obstacles in the lane using adaptive MPC. To do so, you update the

Open Script

Adaptive Cruise Control with Sensor Fusion

Implement a sensor fusion based automotive adaptive cruise controller for a vehicle traveling on a curved road using sensor fusion.

Open Model



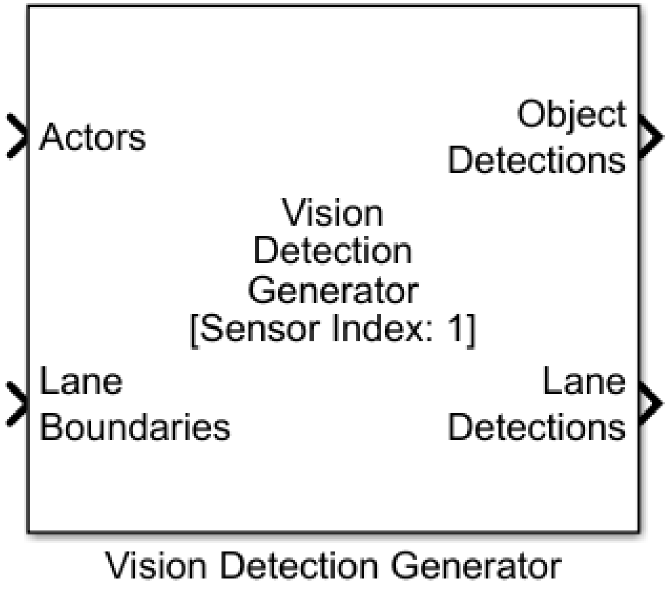
Lane Keeping Assist with Lane Detection

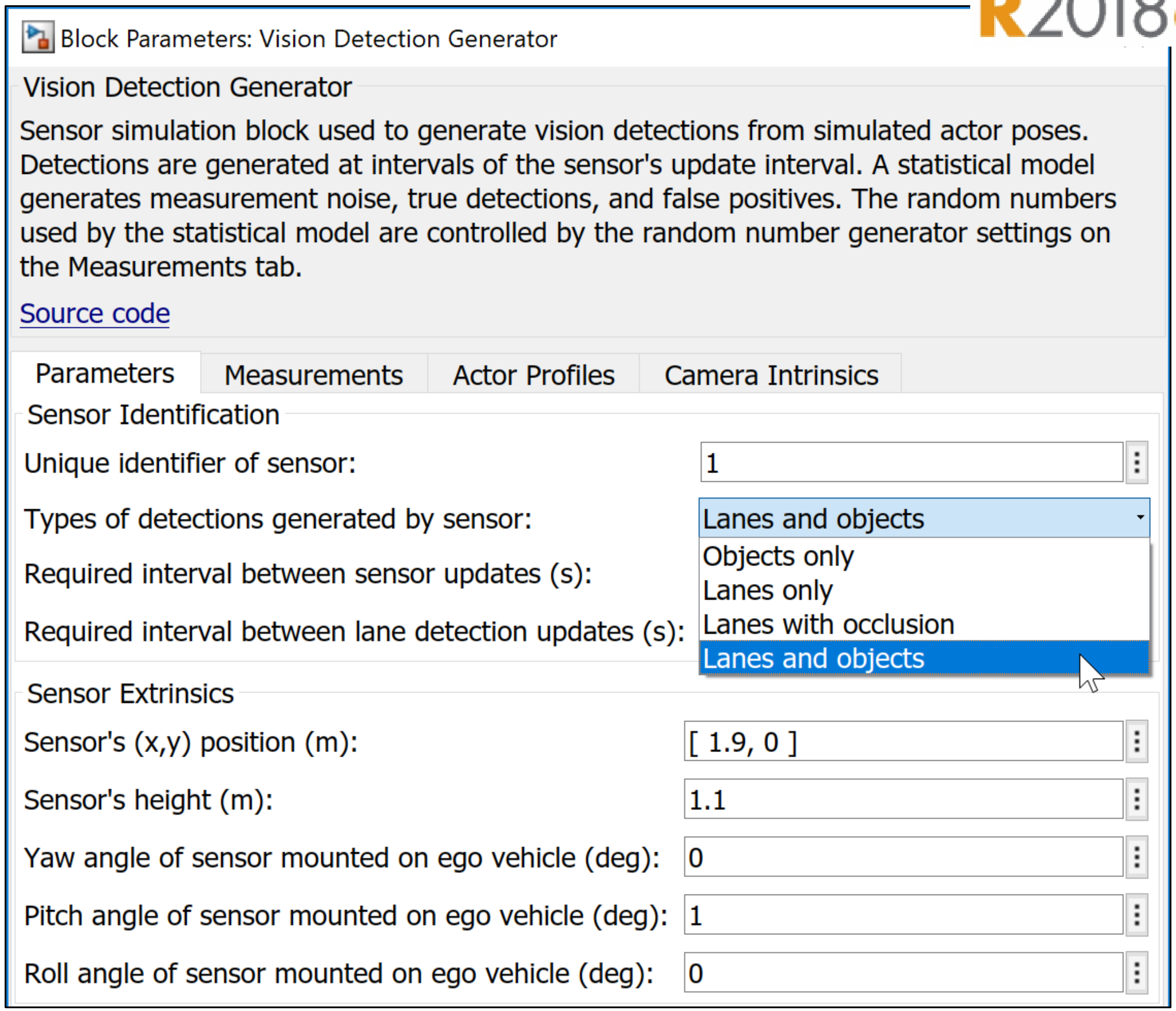
Simulate and generate code for an automotive lane keeping assist (LKA) controller.

Open Model

Vision Detection Generator models lane detection sensor

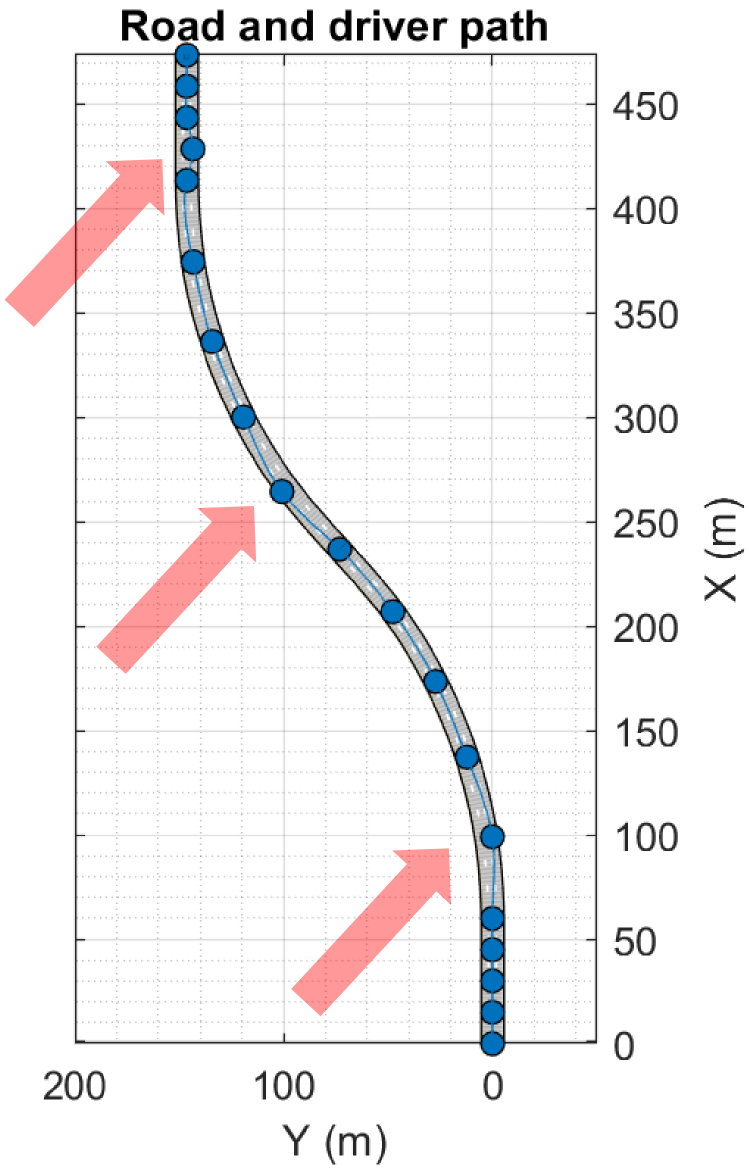
R2018a

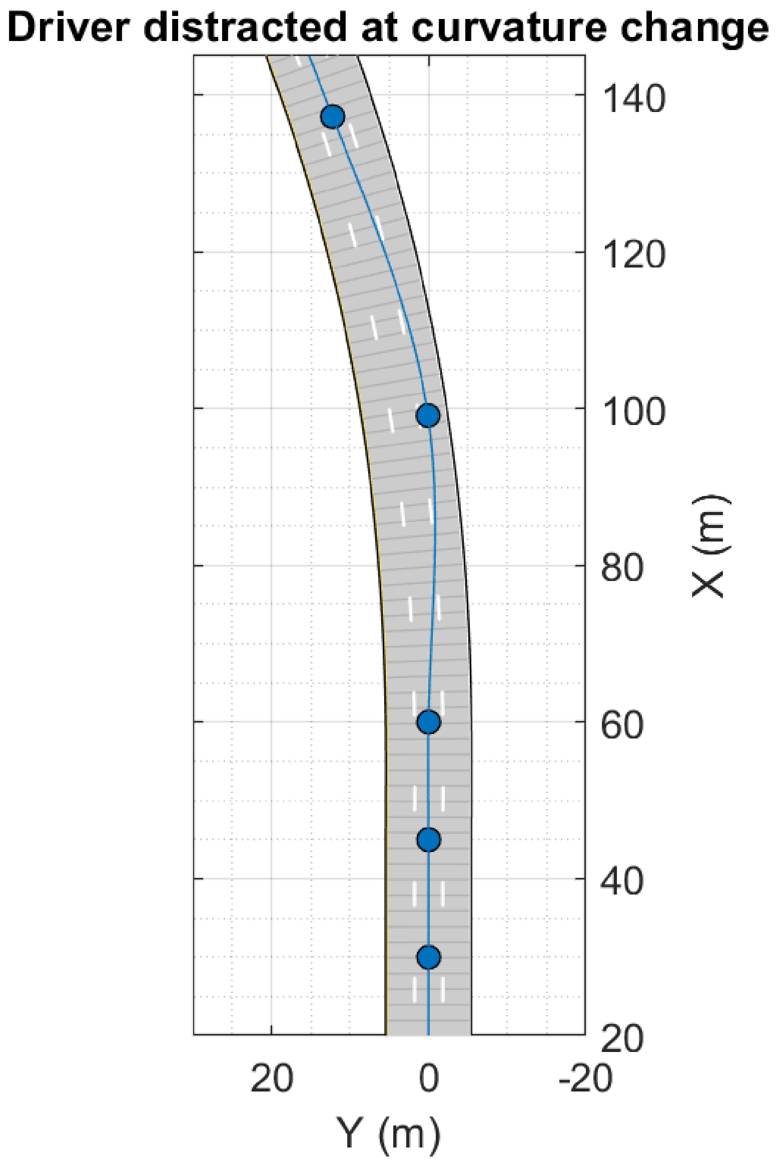




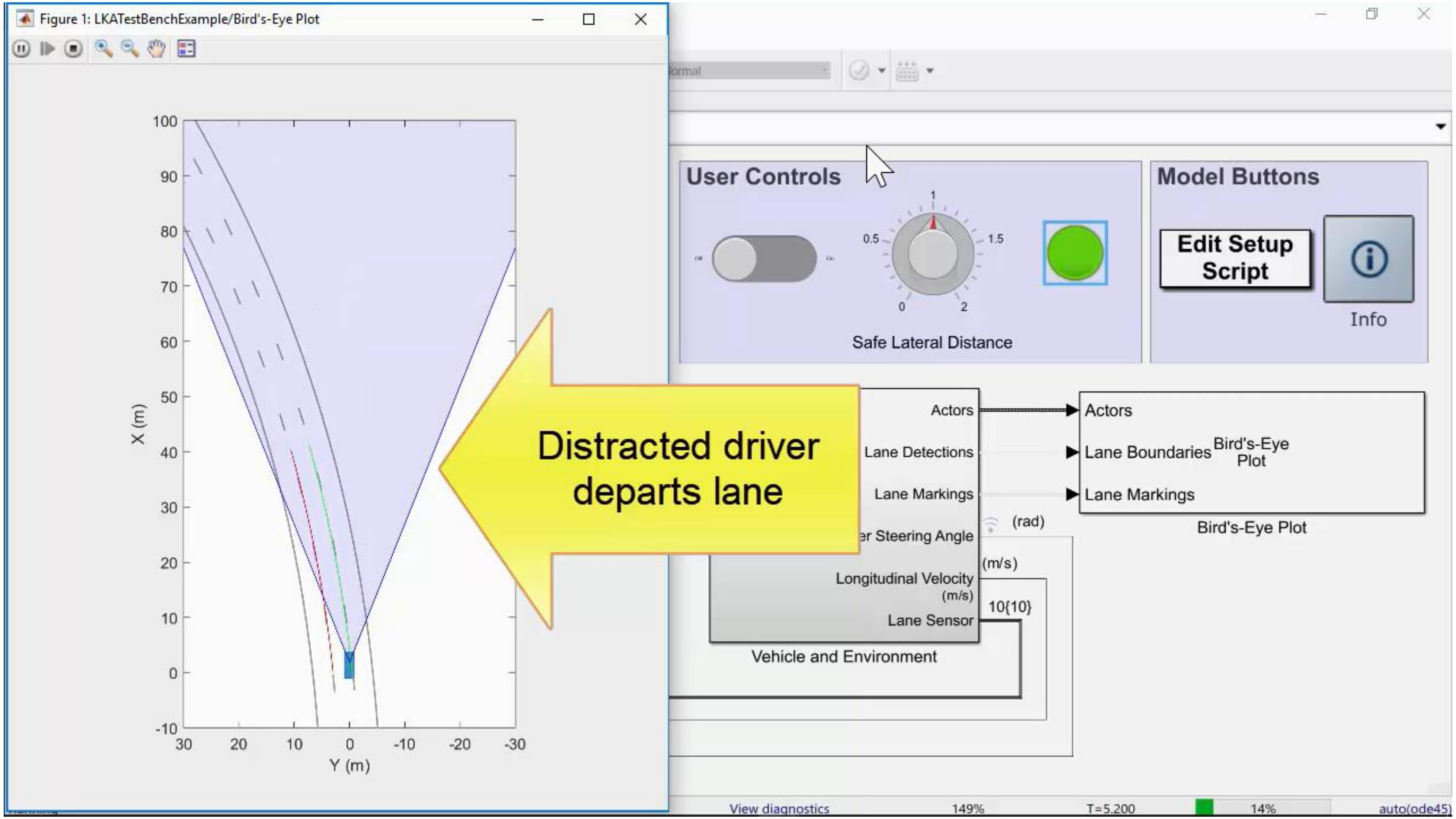
Create highway double curve with drivingScenario

- Driver waypoints simulate distraction at curvature changes

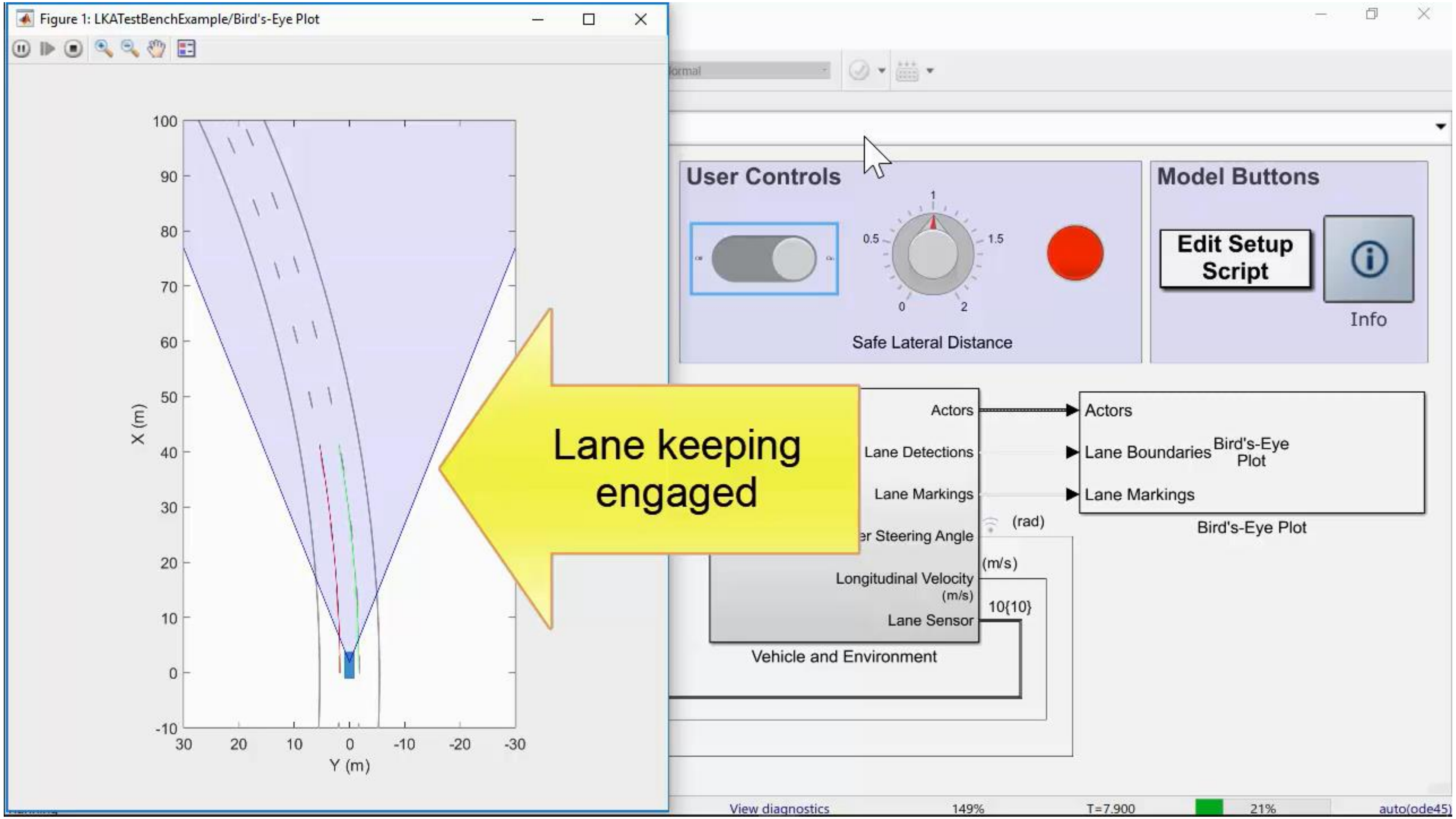




Simulate distracted driver

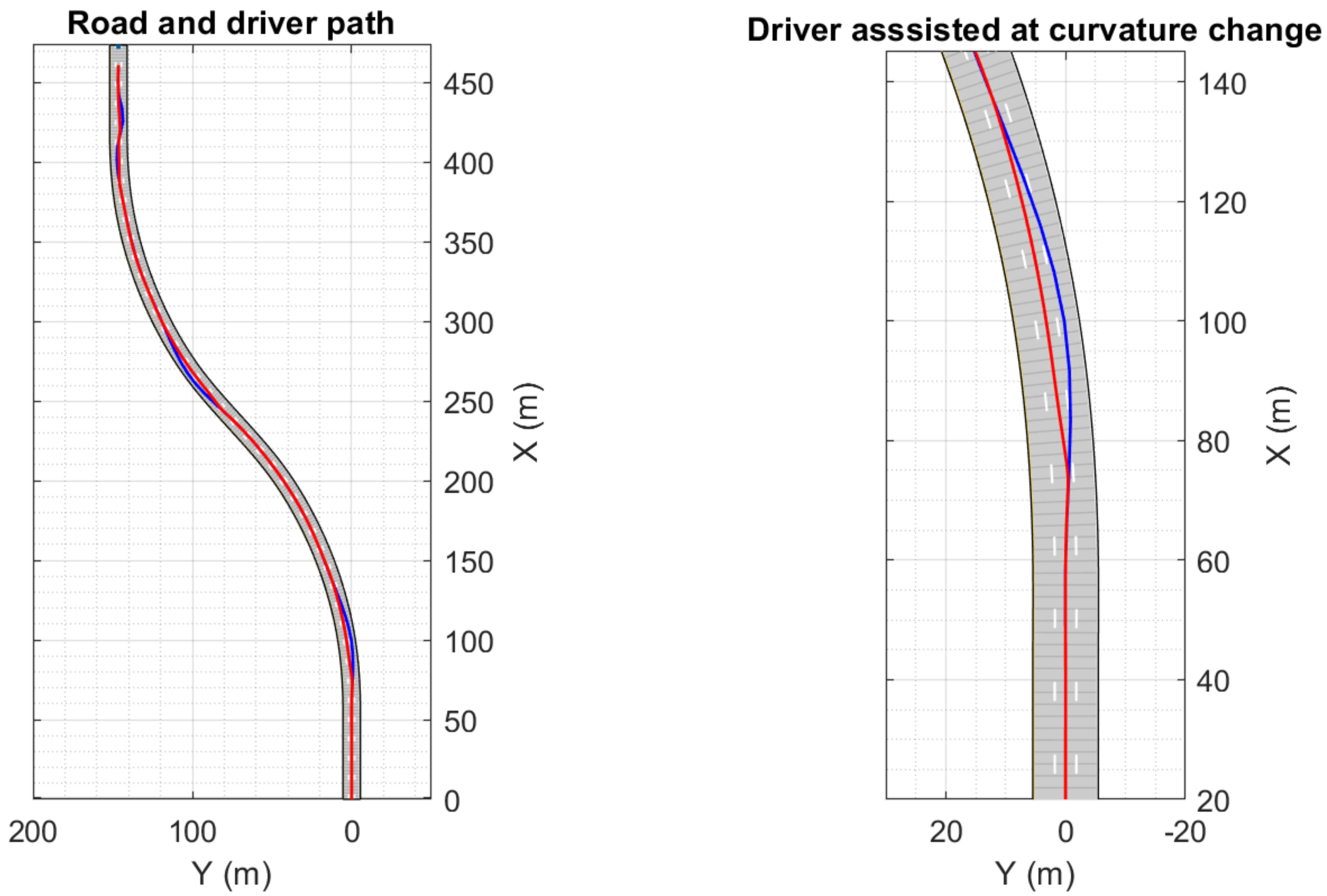


Simulate lane keep assist at distraction events

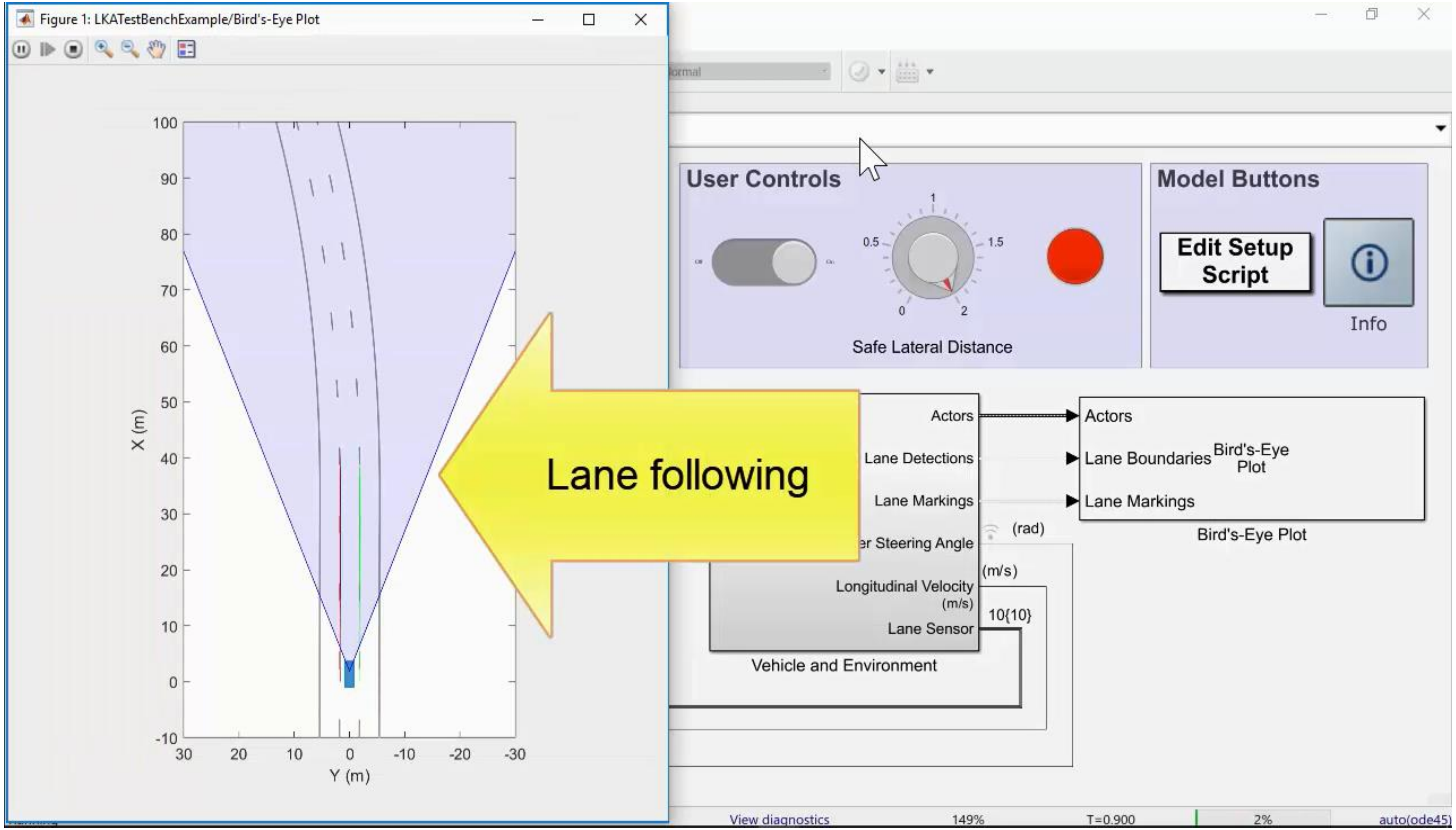


Compare distracted and assisted results

- Detect lane departure and maintain lane during distraction

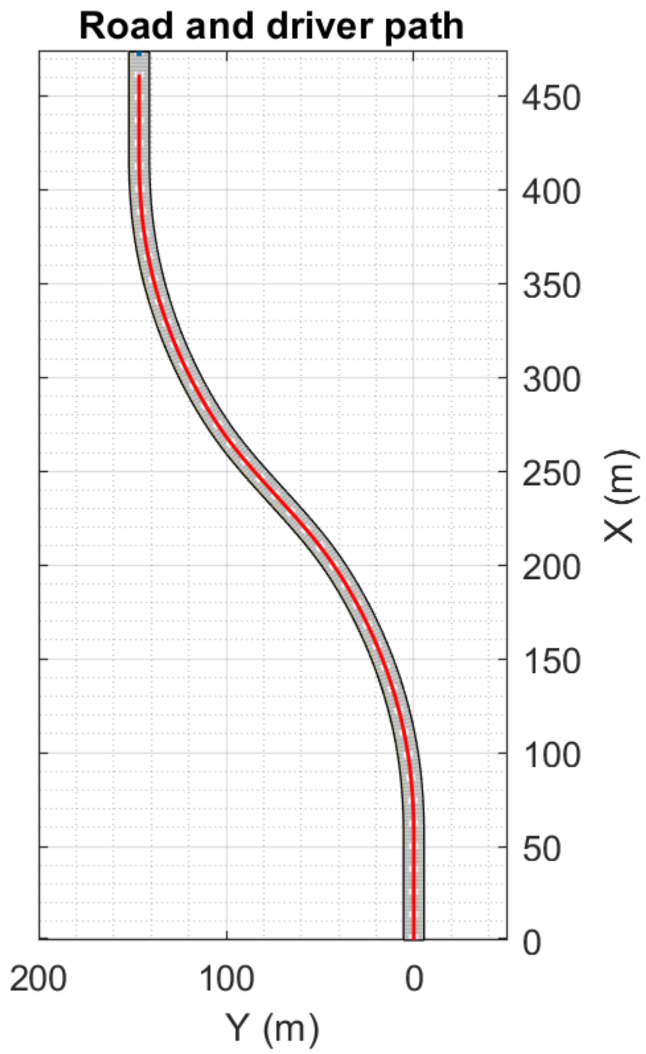


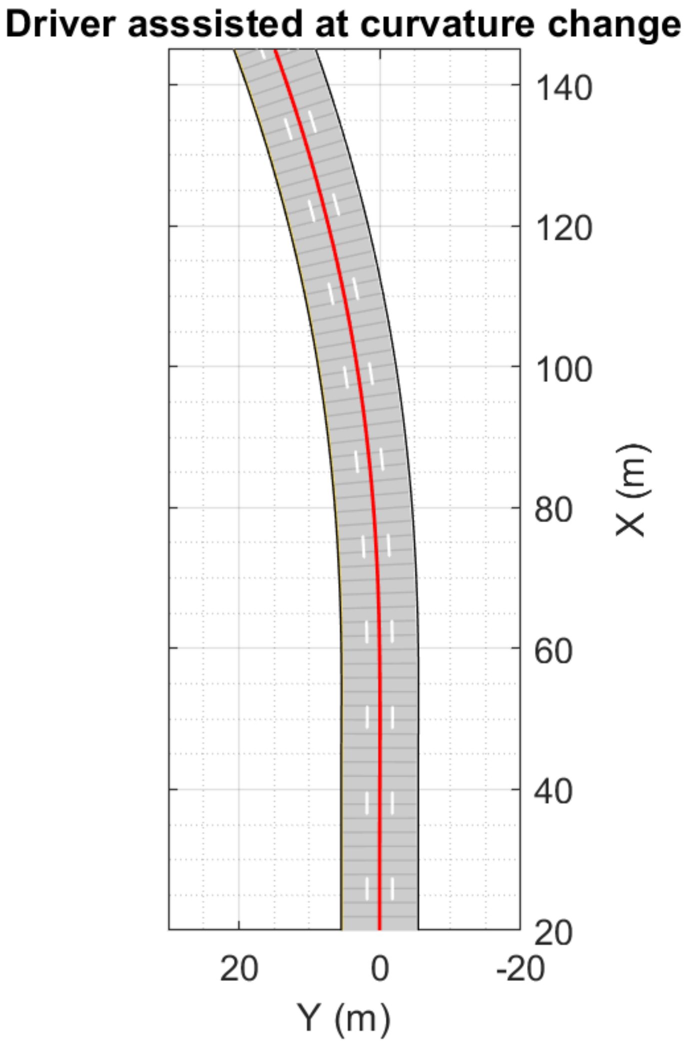
Simulate lane following by increasing minimum safe distance



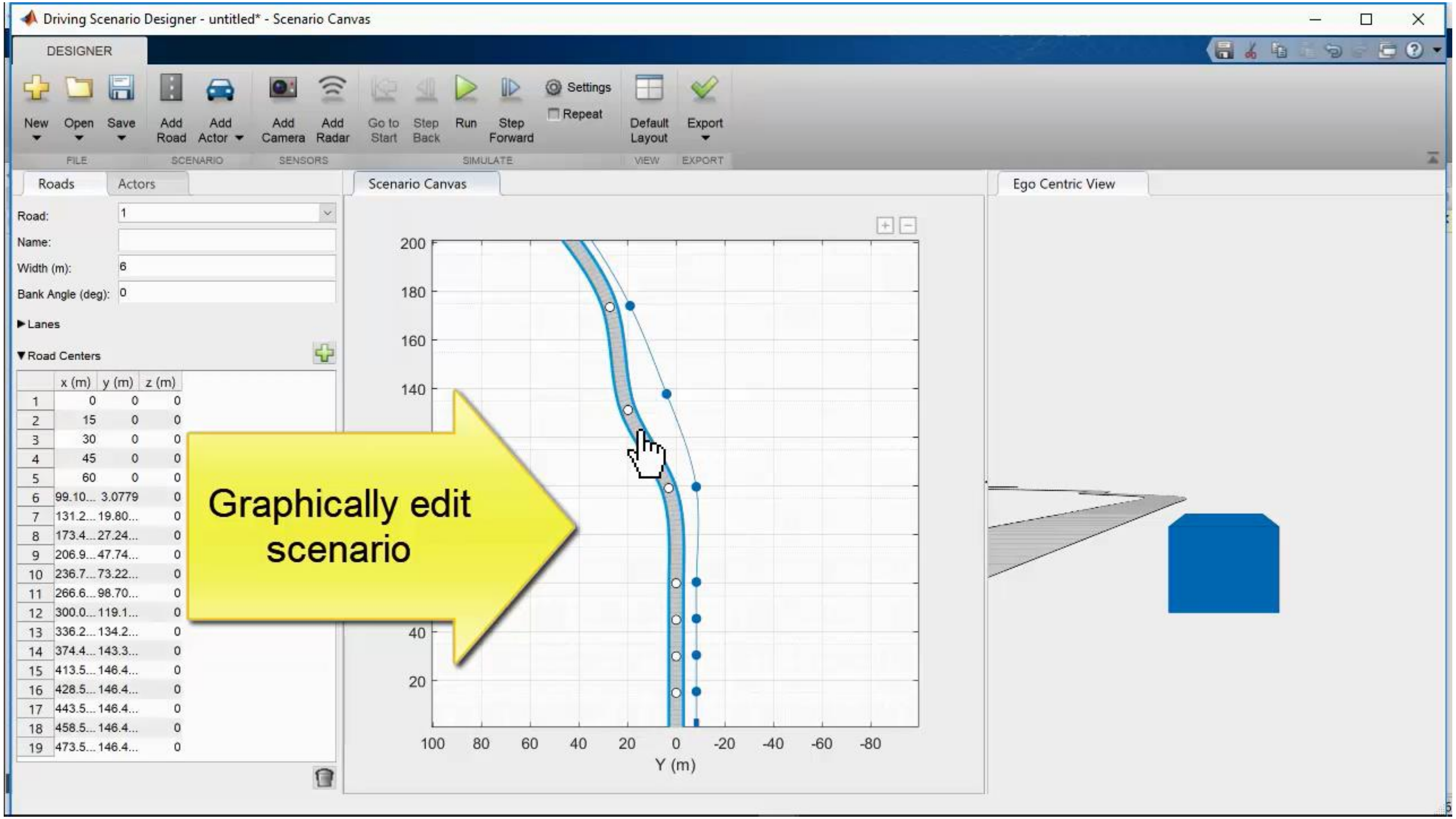
Explore lane following results

- Vehicle stays within lane boundaries

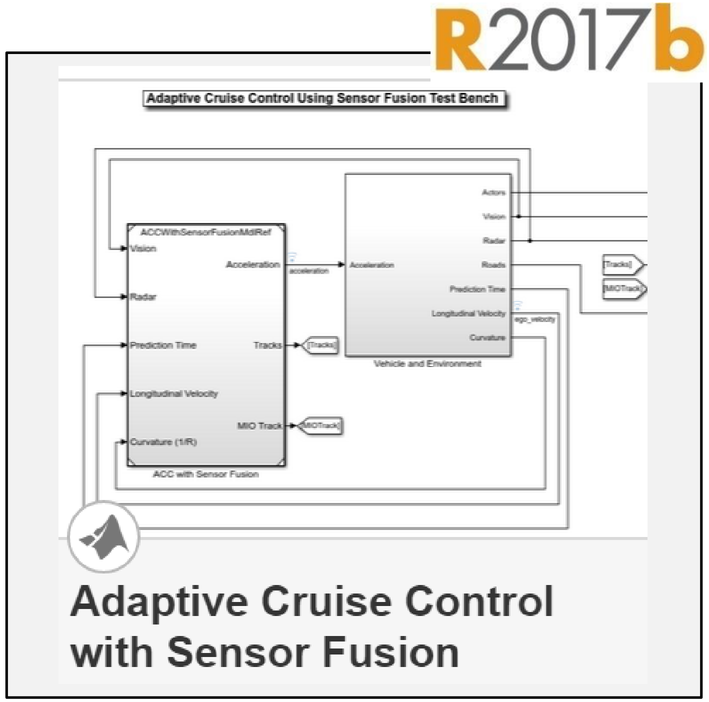




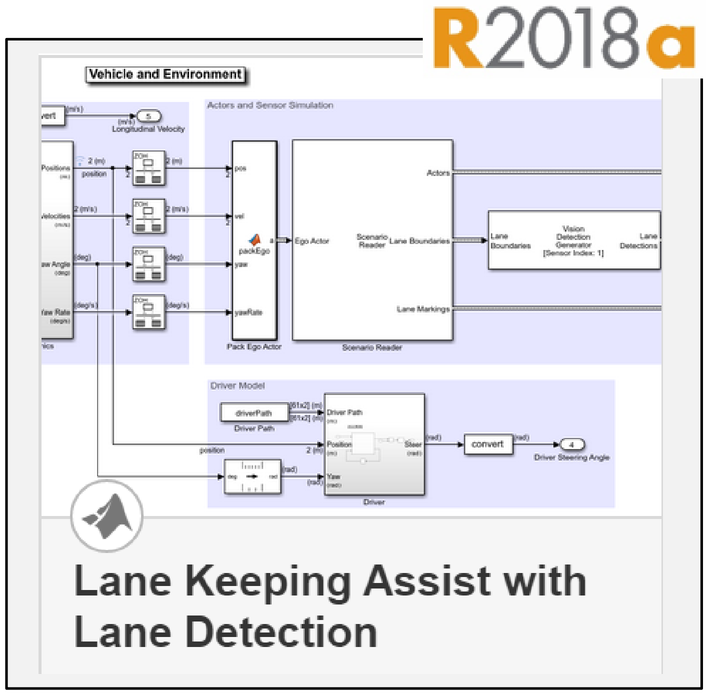
Graphically edit scenarios with Driving Scenario Designer



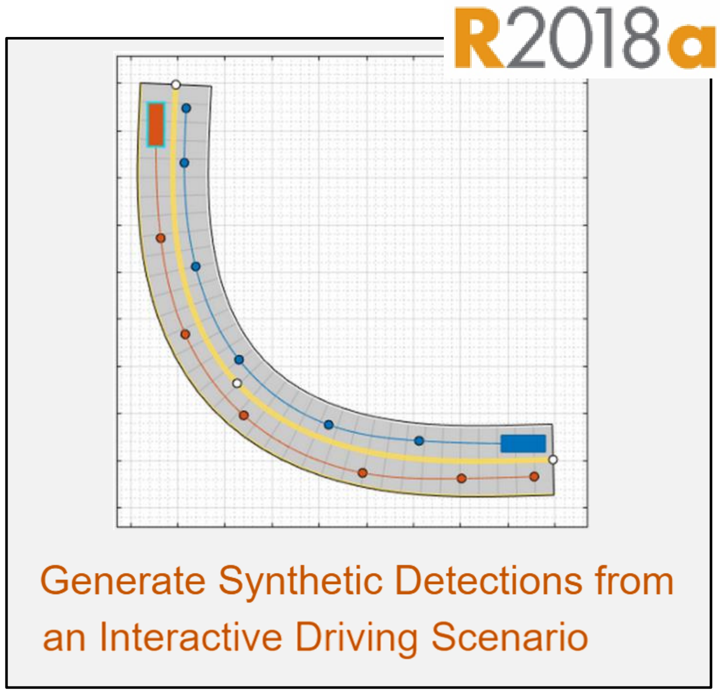
Learn about synthesizing sensor detections to develop control algorithms with these examples



- **Simulate and generate C++** for model-predictive control and sensor fusion algorithms

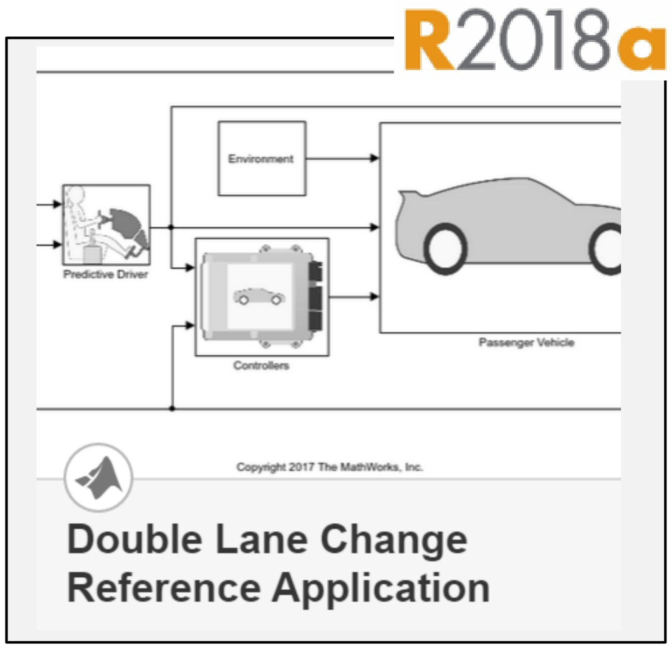


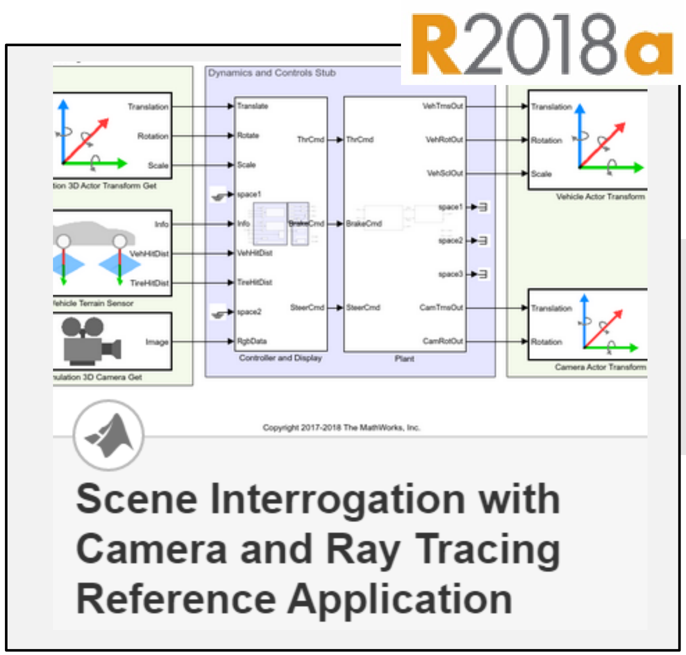
- **Simulate and generate C++** for model-predictive control with lane detections



- **Edit roads, cuboid actors, and sensors** with Driving Scenario Designer App `drivingScenarioDesigner`

Learn about modeling vehicle dynamics to develop control algorithms with these examples





- **Simulate vehicle dynamics** for closed loop design

  Vehicle Dynamics Blockset™

- **Co-simulate with Unreal Engine** and to set actor positions get camera image

  Vehicle Dynamics Blockset™

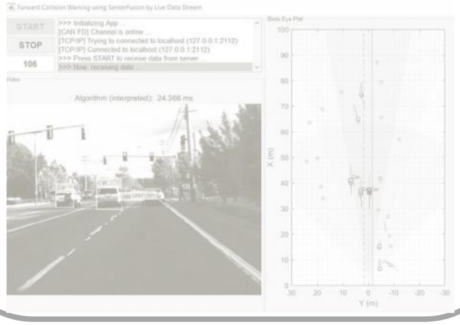
Controls and Embedded Systems

14:30	Full Vehicle Simulation for Electrification and Automated Driving Applications *Prasanna Deshpande, MathWorks* *R Vijayalayan, MathWorks*

How can you use MATLAB and Simulink to develop planning algorithms?

Deep learning

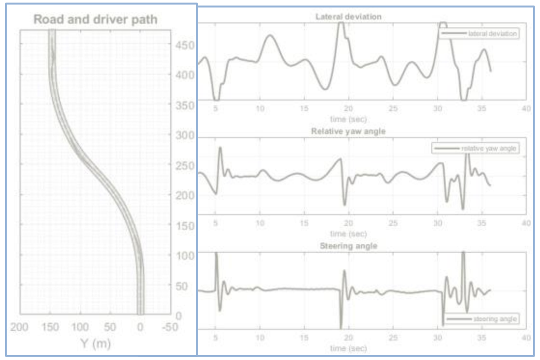
Sensor fusion
with live data



Perception

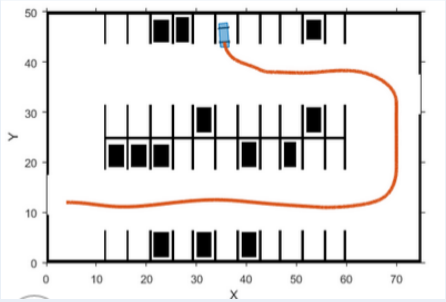
Control

Sensor models &
model predictive control



Planning

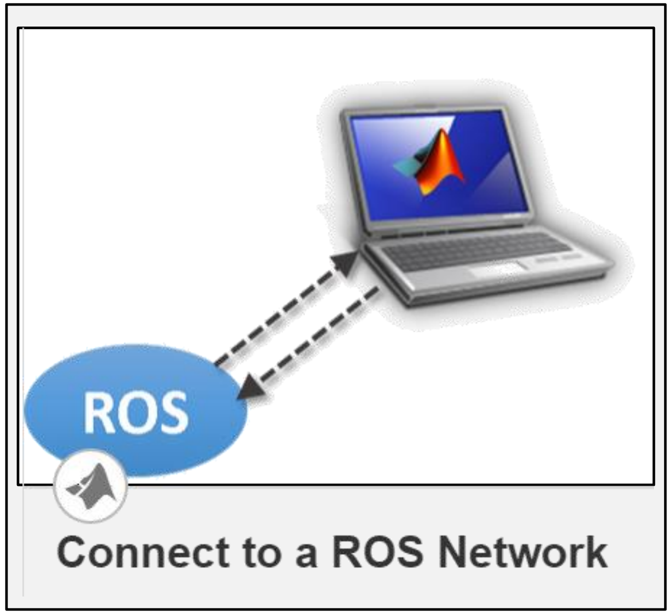
Path planning

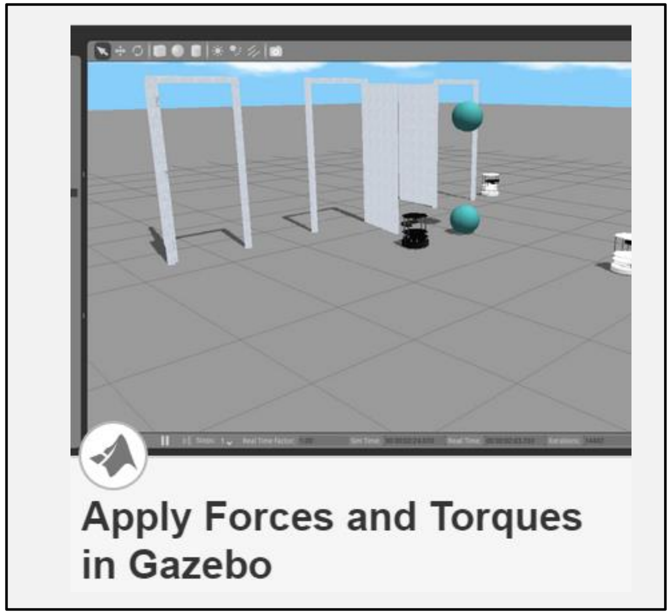


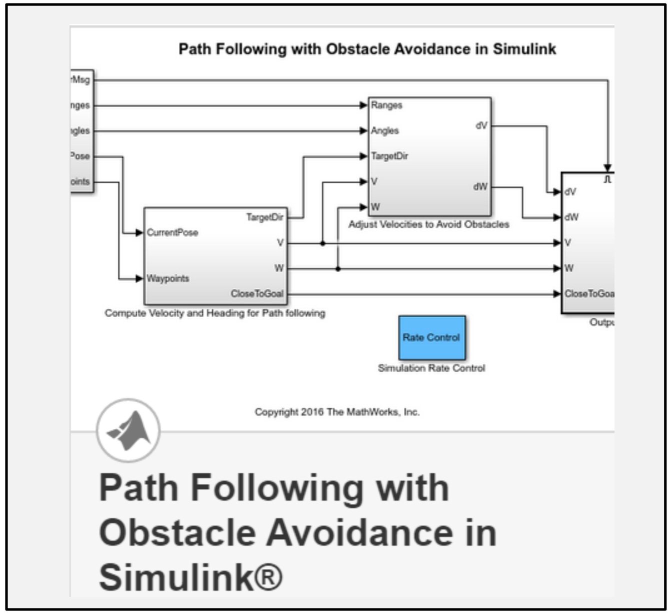
MathWorks®

Robotics System Toolbox introduced:

Connectivity with the ROS ecosystem

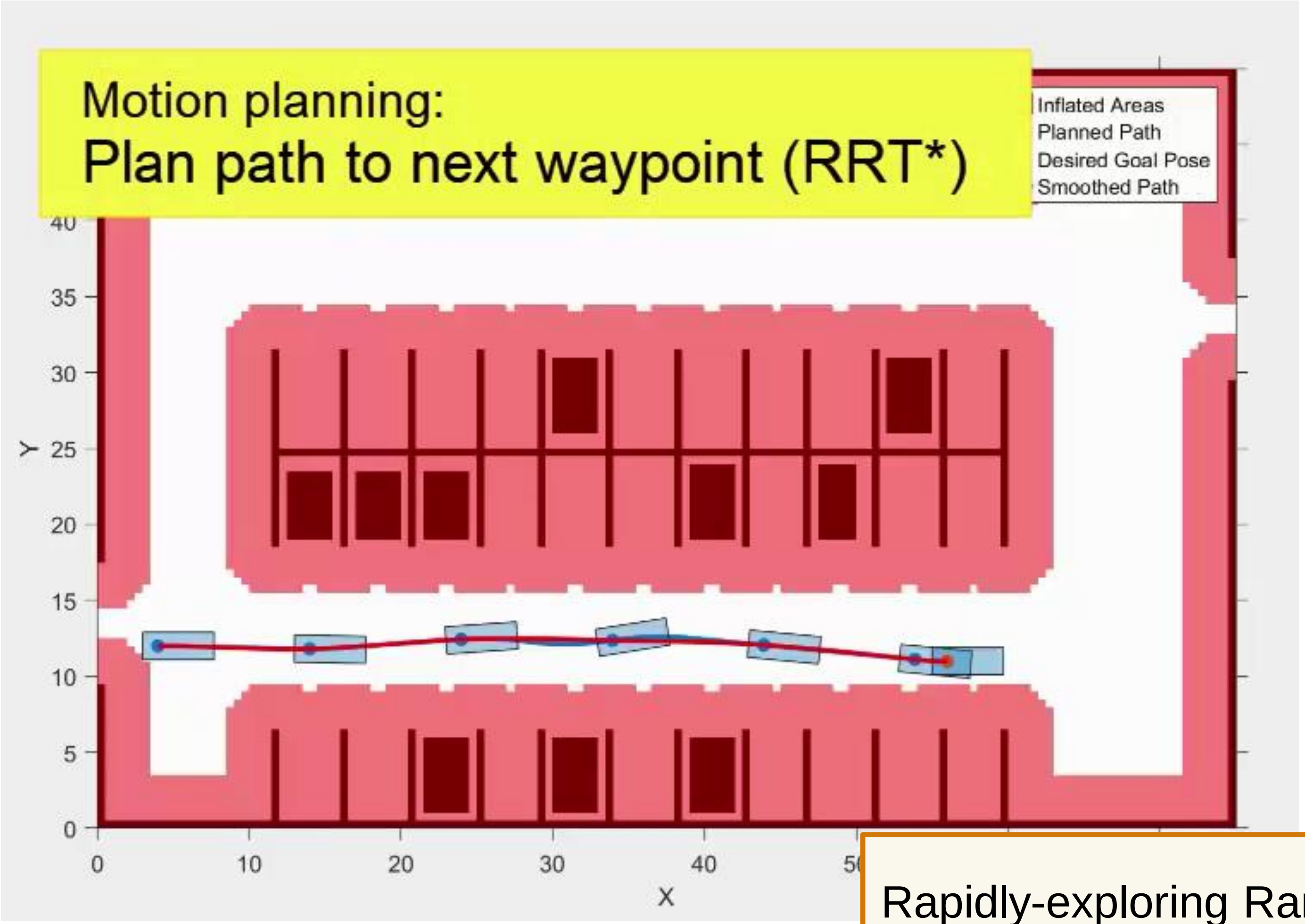






- **Communicate via ROS** to integrate with externally authored ROS components
- **Communication with Gazebo** to visualize and simulated system
- **Follow path** for differential drive robot with ROS based simulator

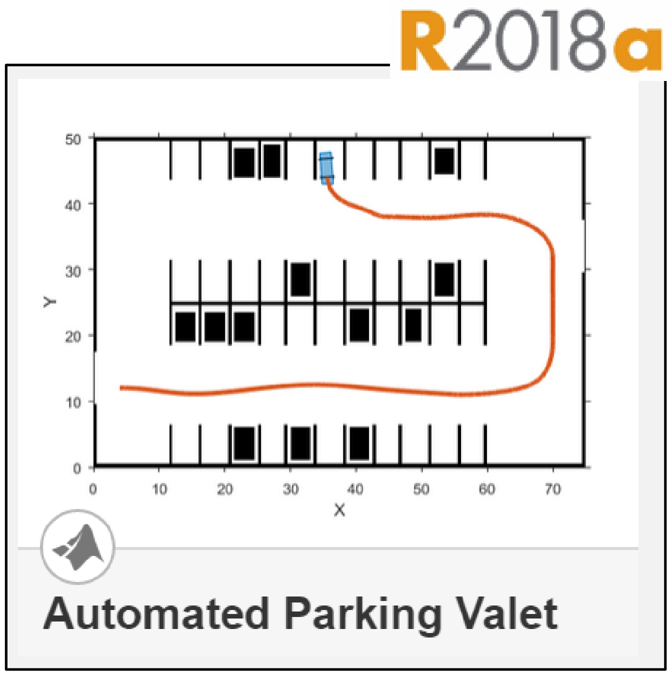
We are investing in design and simulation of path planning for automobiles



R2018a

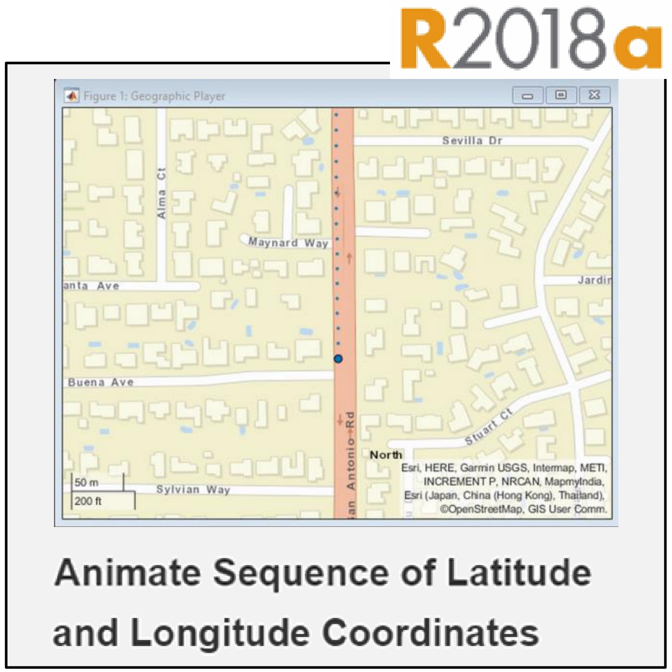
Rapidly-exploring Random Tree (RRT*)

Learn about developing path planning algorithms with these examples



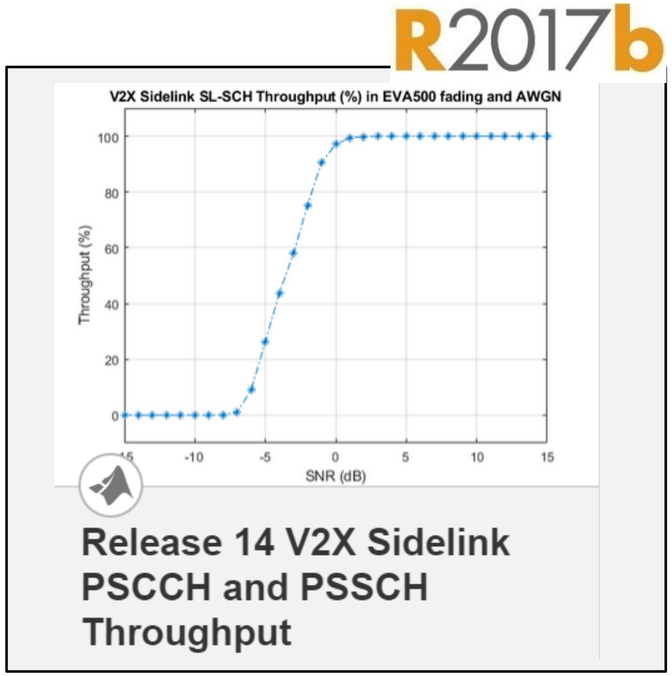
- **Plan path** for automobile given pre-defined map

  Automated Driving System Toolbox™



- **Plot map tiles** using World Street Map (Esri)

  Automated Driving System Toolbox™



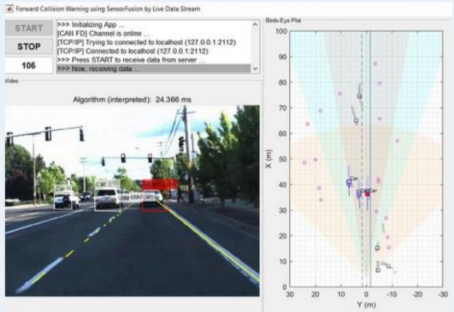
- **Simulate V2X communication** to assess channel throughput

  LTE System Toolbox™

Examples of how you can use MATLAB and Simulink to develop automated driving algorithms

Deep learning

Sensor fusion with live data

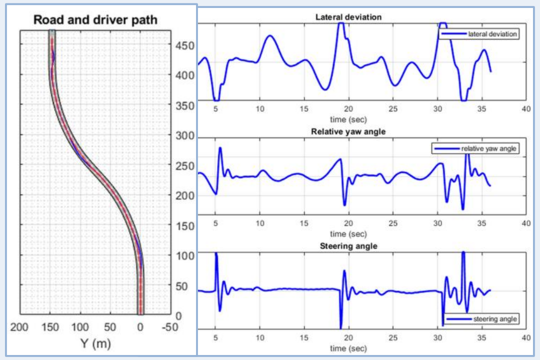


Perception

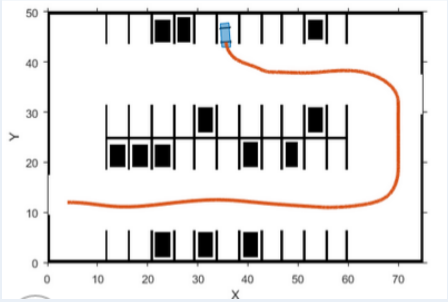
Control

Planning

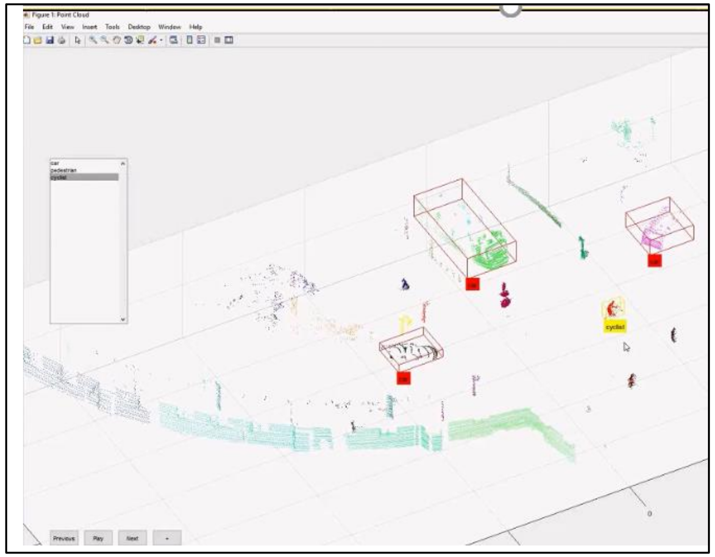
Sensor models & model predictive control

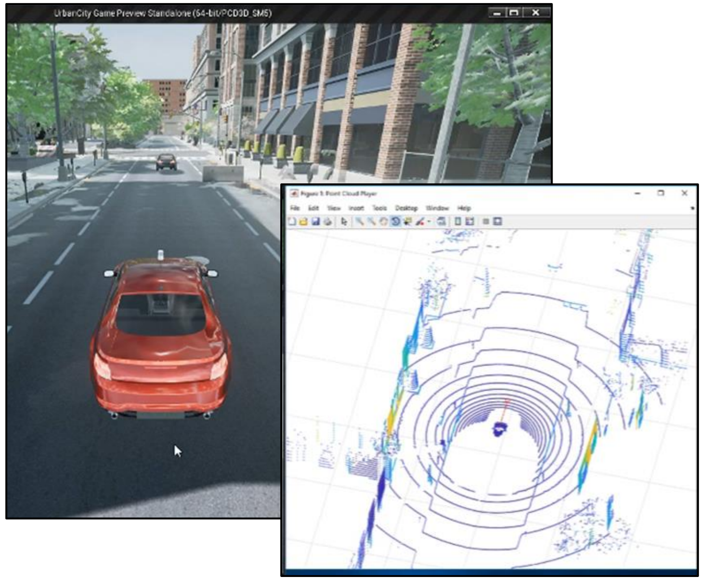


Path planning



MathWorks can help you customize MATLAB and Simulink for your automated driving application





- **Web based ground truth labeling**
  - Consulting project with Caterpillar
  - 2017 MathWorks Automotive Conference

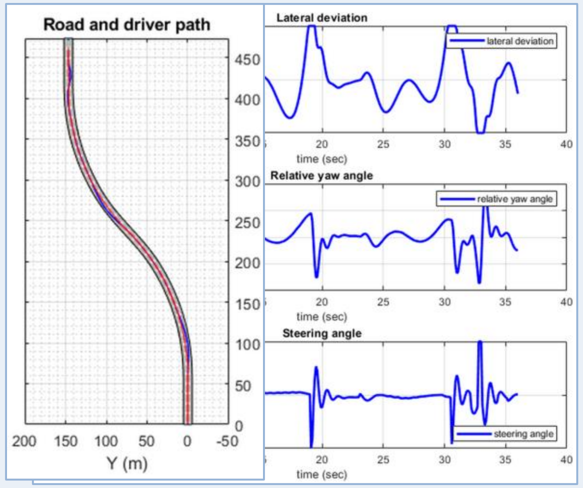
- **Lidar ground truth labeling**
  - Joint presentation with Autoliv
  - 2018 MathWorks Automotive Conference (May 2nd, Plymouth MI)

- **Lidar sensor model for Unreal Engine**
  - Joint paper with Ford
  - SAE Paper 2017-01-0107

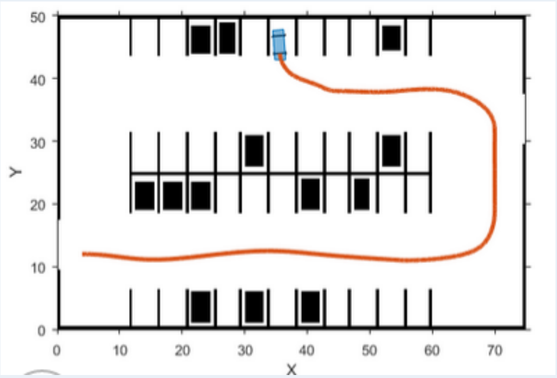
How can we help you can use MATLAB and Simulink to develop automated driving algorithms?

Perception

Control



Planning



MathWorks®

Accelerating the pace of engineering and science

Speaker Details

MANOHAR REDDY M

Email: Manohar.Reddy@mathworks.in

LinkedIn: https://www.linkedin.com/in/manoharreddym

Contact MathWorks India

Products/Training Enquiry Booth

Call: 080-6632-6000

Email: info@mathworks.in

Your feedback is valued.

Please complete the feedback form provided to you.