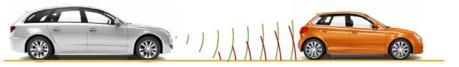


Developing Active Safety Systems Using MATLAB and Simulink

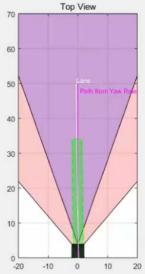


MathWorks **AUTOMOTIVE CONFERENCE 2015**

Marco Roggero Senior Application Engineer

marco.roggero@mathworks.de





Introduction & Motivation

- Active Safety Systems must operate consistent and robust also in unpredictable environments
- Testing these systems in a real world environment is dangerous and can cause serious damage
- Examples:
 - Lane Keeping Systems
 - Adaptive Cruise Control Systems
 - Automated Emergency Braking



Develop and verify your

active safety system functionality and

improve robustness using system level simulation.

MathWorks AUTOMOTIVE CONFERENCE 2015

3

What is Active Safety?

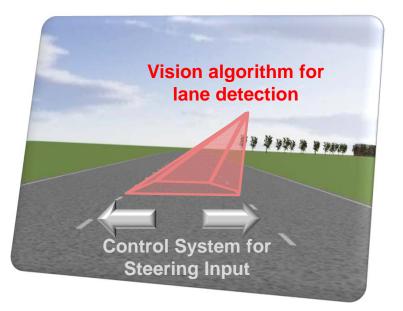
- Safety systems that are active *prior* to an accident
 - Use an understanding of the state of the vehicle and its environment to avoid and minimize the effects of a crash.
 - Interpret signals from various sensors and decide how to help the driver to control the vehicle.



A MathWorks

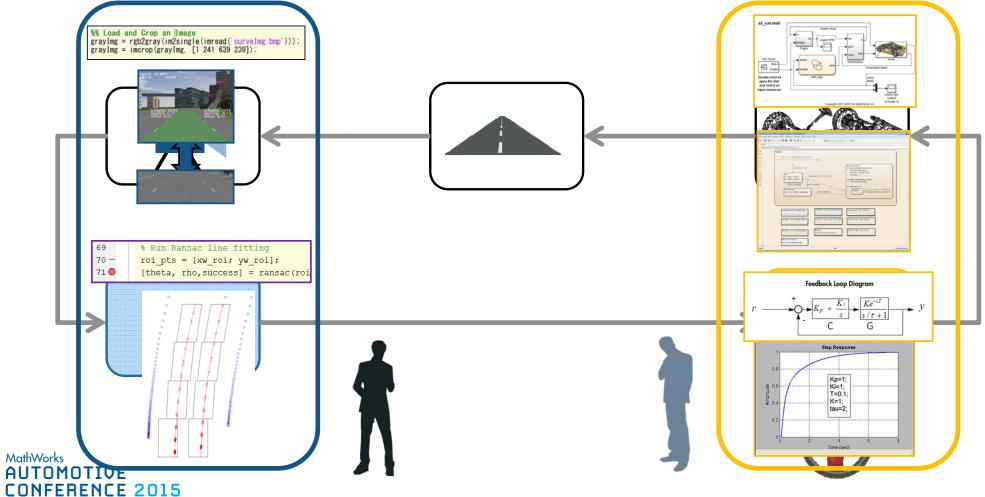
Case Study: Lane Keeping System

- Detect the vehicle's departure from its lane
- Warn the driver or actively steer the vehicle

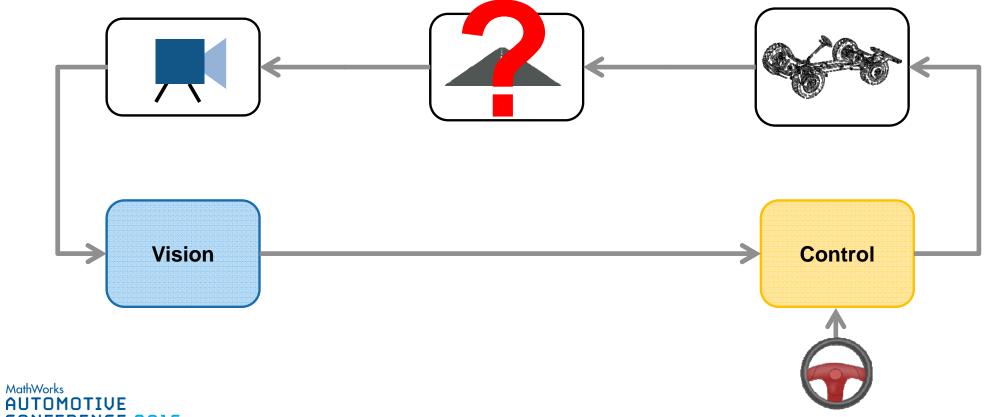


Developing Active Safety Systems

A Multi-Domain Problem



Developing Active Safety Systems The Challenge: Closed Loop with Environment



CONFERENCE 2015

Developing Active Safety Systems Developing the Vision Algorithm Part

O DUE Insert 🔛 fx 🙀 🔹 MATLAB based workflow provides ES 🕑 😐 Comment % 20 0 0 Co To - Breakpoints Run Stop Indent D - Free Q Find - Model Model Build Model 2MUD Print -MATLAB Function1 This file can be published to a formatted document. For more information, see the publishing video or help - Easy to debug scripting environment Eq = zeros(2,4); Valid = [1,1]; if isempty(LaneDetectBefore) LaneDetectBefore = 0; - Pixel level, 2D, 3D visualization ycenter_init = [2,-2]; slope_init = [0,0]; Count = 0: 14 end - Easy-to-use powerful image processing, and XX Detect Edges edgeImg = findLaneEdges(grayImg, filter_kernel); computer vision algorithms %% Find lane points if LaneDetectBefore == 0 ycenter_init = [2, -2]; slope_init = [0, 0]; - -Mage Tool 2 - I File Tools Window Help [Xcenter, Ycenter, slope, NSupportPts] = findLanePts2(edgeImg, invtform, 📄 💁 🕕 🌒 💡 🗖 🖉 🖉 🔍 🔍 🔍 👘 🐜 100% Curve Fitting Tool -% adjust ymin oral according File Fit View Tools Desktop Window Help LaneDetectBefore = 1; for lane = 1:2 if NSupportPts(lane) > 0 vcenter_init(lane) = slope_init(lane) = Xdata Degree 2 Eq(lane,2:4) = polyfi Y data Robust Off else 7 date Center and scale Fit Options... Pixel Region (Image Tool 2) imear model Poly2: f(x) = p1*x*2 + p2*x + p3 where x is normalized by me File Edit Window Help †m m 🕈 pefficients (with 95% carifidence b p1 = -4.693 (-11.55, 2.167) p2 = 17.65 (11.92, 23.37) p3 = 28.75 (21.85, 35.46) R:238 R: 32 G: 38 B: 17 R: 44 G: 50 B: 29 R: 66 G: 71 B: 58 R: 78 G: 83 B: 70 35 G:239 B:225 Goodhess of fit: SSE: 9.511 R-square: 0.9905 Adjusted R-square: 0.981 RMSE: 2.181 30 fo: (X, Y) [R G B] > 25 R:236 R:127 G:132 B:117 R: 55 G: 56 B: 44 R: 56 G: 59 B: 49 R:238 G:241 B:226 G:239 B:227 20 15 R: 59 G: 67 B: 53 R:232 R:213 R:160 R:184 G:159 B:146 G:221 B:207 G:231 B:218 G:187 B:175 -0.8 -0.6 -0.4 -0.2 0 0.2 0.4 0.5 -1.2 -1 R: 23 G: 30 B: 19 R: 59 G: 66 B: 55 R: 54 G: 57 B: 47 R: 35 G: 38 B: 28 Table of Fits MathW G: 4 B: 3 Fit na... = Data Fit type SSE R-square DFE Adj R-sq RMSE # Coeff 0.9905 AUT Guntet... Vvs.X poly2 9.3109 0.9810 2.1800 R: 45 G: 54 R: 36 G: 45 R:160 R:227 R:169 G:176 B:169 CON. G:163 B:153 G:230 B:220

Editor - Block LaneDetectionTest SL/MATLAR Fund

* * * *

8080

Z Auto fit

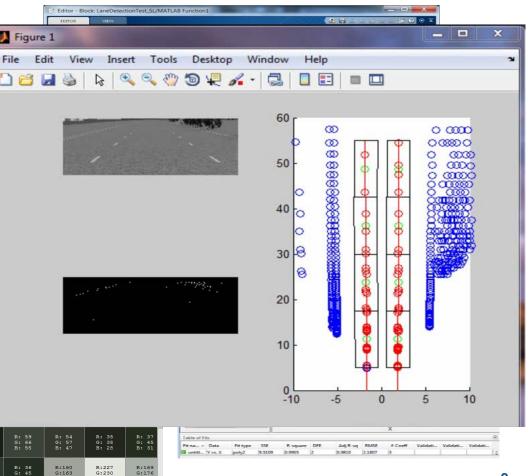
Step

📣 MathWorks

9

Developing Active Safety Systems Developing the Vision Algorithm Part

 MATLAB based workflow provides Figure 1 - Easy to debug scripting environment - Pixel level, 2D, 3D visualization - Easy-to-use powerful image processing, computer vision algorithms 🚺 Image Tool 2 - I File Tools Window Help 📄 💁 🕕 🌒 💡 🙇 🔗 🔍 🔍 🔍 👘 100% File Edit m m 🦻 fo: (X, Y) [R G B] R:21 G:22 B:20 R: 23 G: 30 B: 19 MathW AUT

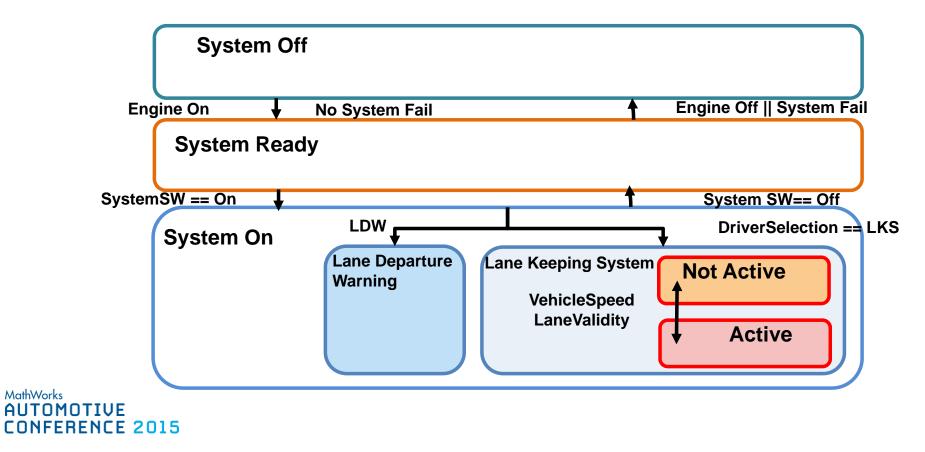


Developing Active Safety Systems Developing the Controller Part

- Goal
 - Keep the vehicle within a lane by controlling steering input
- Controller configuration
 - Mode Selector (Stateflow)
 - Risk Assessment (Stateflow)
 - Steering angle compensator (Simulink)
 - Feed-back steer angle using heading & lateral offset
 - Feed-forward steer angle using curvature & vehicle speed
- Control Parameter Tuning

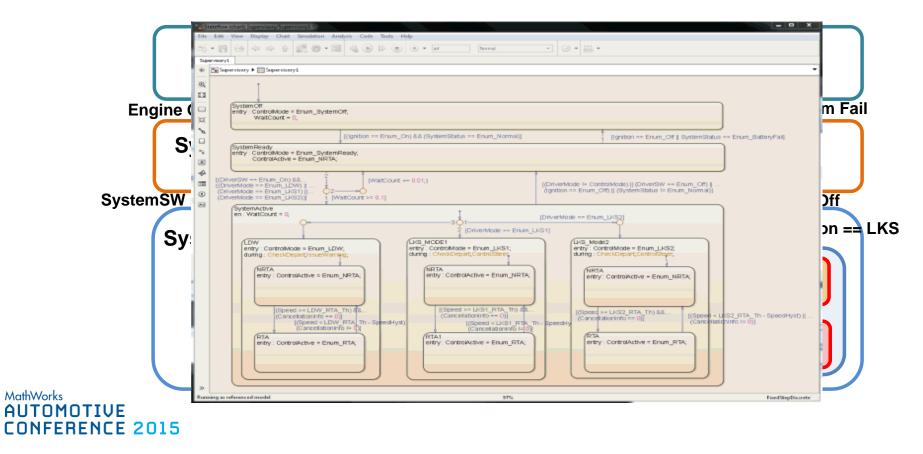
Developing Active Safety Systems Developing the Controller Part

Determining Control Mode – creating the model



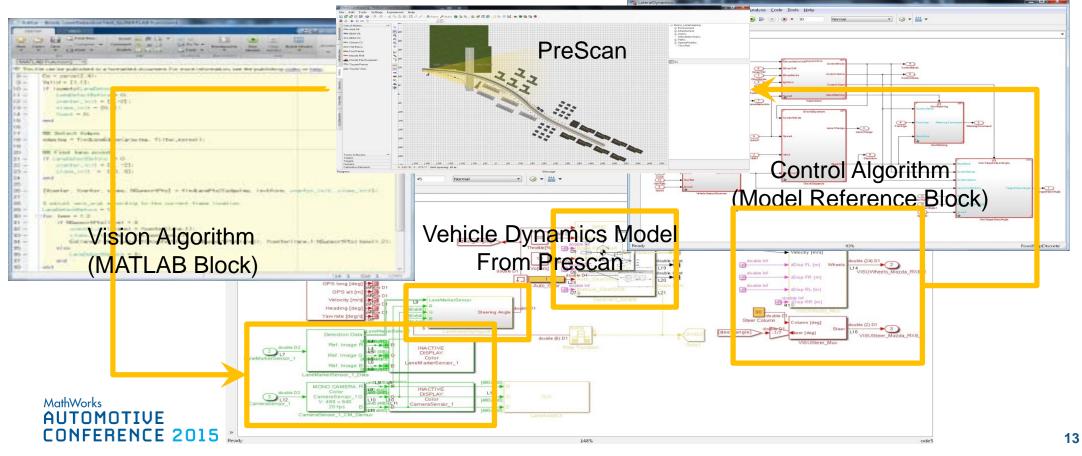
Developing Active Safety Systems Developing the Controller Part

Determining Control Mode – creating the model

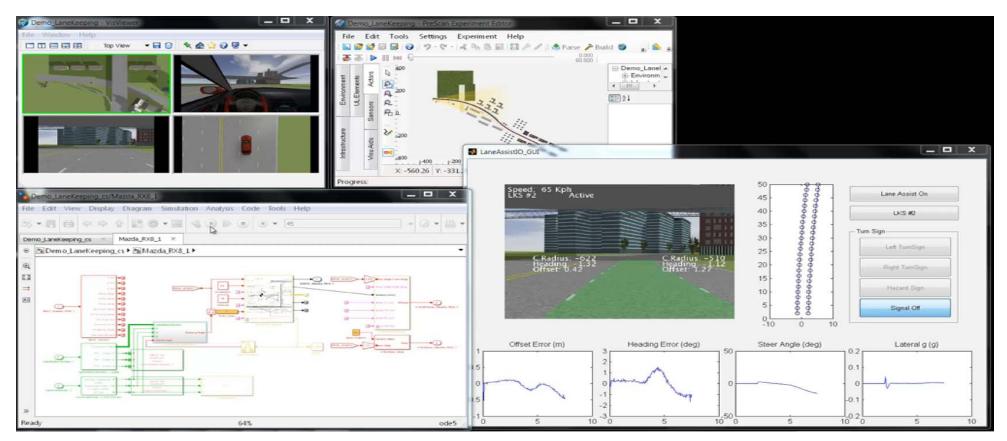


Developing Active Safety Systems System Level Simulation

Closed-loop test harness model for system level validation

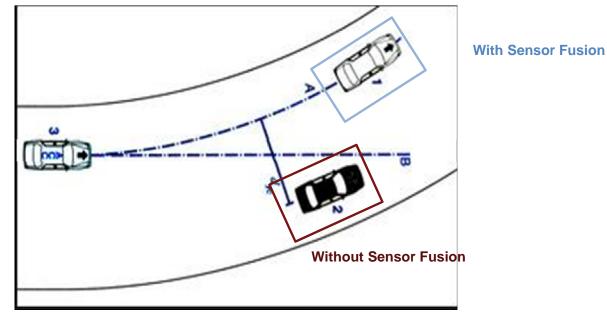


Developing Active Safety Systems System Level Simulation

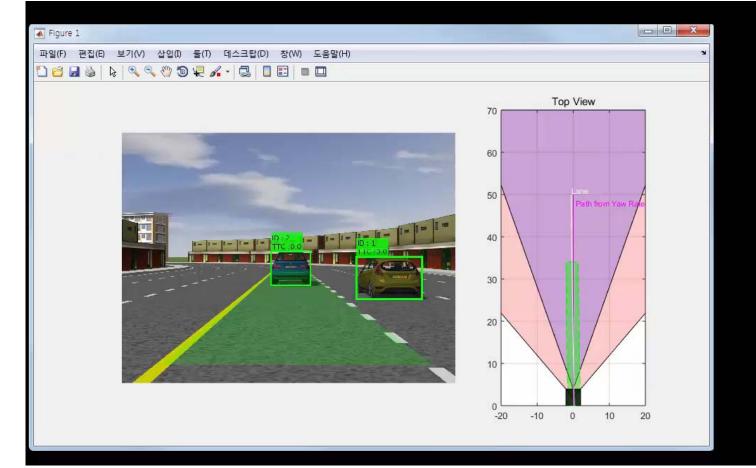


Need for Sensor Fusion Case study: Automated Emergency Braking

Accurate CIPV (Critical In-Path Vehicle) selection



Example: Radar and Camera Data Fusion

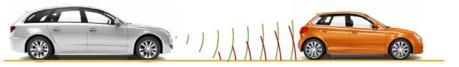




16



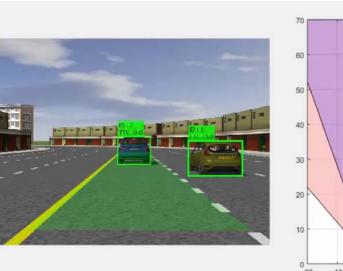
Developing Active Safety Systems Using MATLAB and Simulink

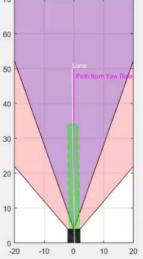


MathWorks **AUTOMOTIVE CONFERENCE 2015**

Marco Roggero Senior Application Engineer

marco.roggero@mathworks.de





Top View