Some common questions from automated driving engineers

- How can I visualize vehicle data?
- How can I detect objects in images?
- How can I fuse multiple detections?
Some common questions from automated driving engineers

How can I visualize vehicle data?

How can I detect objects in images?

How can I fuse multiple detections?
Examples of automated driving sensors

- Camera
- Vision-based object detector
- Lane detector
- Radar-based object detector
- Lidar
- Inertial measurement unit
Examples of automated driving sensor data

**Camera** (640 x 480 x 3)

- SensorID = 1;
- Timestamp = 1461634696379742;
- NumDetections = 6;

**Vision Detector**

- TrackID: Left
  - Position: [22.8 3.12 2.24]
  - Classification: 5
  - TrackStatus: 1
  - Amplitude: 3
  - Offset: 1.68
- TrackID: Right
  - Position: [22.61 0.43 2.24]
  - Classification: 5
  - TrackStatus: 0
  - Amplitude: 6

**Lane Detector**

- Left
  - IsValid: 1
  - Confidence: 3
  - BoundaryType: 3
  - HeadingAngle: 0.002
- Right
  - IsValid: 1
  - Confidence: 3
  - Curvature: 0.000

**Radar Detector**

- SensorID = 2;
- Timestamp = 1461634696407521;
- NumDetections = 23;

**Lidar** (47197 x 3)

- TrackID: Left
  - Position: [25.7134 3.0970 0.34]
  - Velocity: 9.2795
  - Amplitude: 23
- TrackID: Right
  - Position: [18.8008 2.2849 0.34]
  - Velocity: 9.86
  - Amplitude: 8
- TrackID: Center
  - Position: [14.8815 1.8245 0.34]
  - Velocity: 12.2911
  - Amplitude: 0.8369

**Inertial Measurement Unit**

- Timestamp: 1461634696379742
- Velocity: 9.2795
- YawRate: 0.0040
Visualize sensor data
Visualize differences in sensor detections
Explore logged vehicle data

- **Load video data** and corresponding **mono-camera parameters**

  ```matlab
  >> video = VideoReader('01_city_c2s_fcw_10s.mp4')
  >> load('FCWDemoMonoCameraSensor.mat', 'sensor')
  ```

- **Load detection sensor data** and corresponding **parameters**

  ```matlab
  >> load('01_city_c2s_fcw_10s_sensor.mat', 'vision', 'lane', 'radar')
  >> load('SensorConfigurationData.mat', 'sensorParams')
  ```

- **Load lidar point cloud data**

  ```matlab
  >> load('01_city_c2s_fcw_10s_Lidar.mat', 'LidarPointCloud')
  ```
Learn more about visualizing vehicle data by exploring examples in the Automated Driving System Toolbox

- Plot object detectors in vehicle coordinates
  - Vision & radar detector
  - Lane detectors
  - Detector coverage areas

- Transform between vehicle and image coordinates

- Plot lidar point cloud

MATLAB EXPO 2017
Some common questions from automated driving engineers

How can I visualize vehicle data?

How can I detect objects in images?

How can I fuse multiple detections?
How can I detect objects in images?

Object detector

Classification
Left
Bottom
Width
Height

Classification
Left
Bottom
Width
Height
Train object detectors based on ground truth
Train object detectors based on ground truth

Design object detectors with the Computer Vision System Toolbox

<table>
<thead>
<tr>
<th>Machine Learning</th>
<th>Aggregate Channel Feature</th>
<th>trainACFOBJECTDETECTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cascade</td>
<td></td>
<td>trainCASCADEOBJECTDETECTOR</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Deep Learning</th>
<th>R-CNN (Regions with Convolutional Neural Networks)</th>
<th>trainRCNNOBJECTDETECTOR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast R-CNN</td>
<td></td>
<td>trainFastRCNNOBJECTDETECTOR</td>
</tr>
<tr>
<td>Faster R-CNN</td>
<td></td>
<td>trainFasterRCNNOBJECTDETECTOR</td>
</tr>
</tbody>
</table>
Specify ground truth to train detector

How can I create ground truth?
Specify ground truth to train detector

Video → Ground Truth Labeler App → Ground Truth → Train detector → Object detector
Automate labeling based on a manually labeled frame with point tracker
Ground truth labeling to **train** detectors

Video → Ground Truth Labeler App → Ground Truth → Train detector → Object detector

Ground truth labeling to **evaluate** detectors

Video → Object detector → Detections → Ground Truth Labeler App → Ground truth → Evaluate detections
Customize Ground Truth Labeler App

Add custom image reader with `groundTruthDataSource`
Customize Ground Truth Labeler App

Add custom automation algorithm

driving.automation.AutomationAlgorithm
Customize Ground Truth Labeler App

Add connection to other tools with `driving.connector.Connector`
Learn more about detecting objects in images by exploring examples in the Automated Driving System Toolbox.

- **Label detections** with Ground Truth Labeler App
- **Add automation algorithm** for lane tracking
- **Extend connectivity** of Ground Truth Labeler App
Learn more about detecting objects in images by exploring examples in the Automated Driving System Toolbox

- **Train object detector** using deep learning and machine learning techniques
  
  **MATLAB EXPO 2017**

- **Explore pre-trained pedestrian detector**

- **Explore lane detector** using coordinate transforms for mono-camera sensor model
Some common questions from automated driving engineers

How can I detect objects in images?

How can I fuse multiple detections?

How can I visualize vehicle data?
Example of radar and vision detections of a vehicle

Can we fuse detections to better track the vehicle?
Fuse detections with multi-object tracker

Vision and radar detections to be fused
Synthesize scenario to test tracker
Test tracker against synthesized data

All detections fused into a single track
Track multiple object detections

- 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
Examples of Kalman Filter (KF) initialization functions

Object Detections → Multi-Object Tracker:
- Track Manager
- Tracking Filter → Tracks

<table>
<thead>
<tr>
<th></th>
<th>Linear KF (trackingKF)</th>
<th>Extended KF (trackingEKF)</th>
<th>Unscented KF (trackingUKF)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Constant velocity</strong></td>
<td>initcvmkf</td>
<td>initcmvekf</td>
<td>initcmvukf</td>
</tr>
<tr>
<td><strong>Constant acceleration</strong></td>
<td>initcmaekf</td>
<td>initcaemkf</td>
<td>initcaemukf</td>
</tr>
<tr>
<td><strong>Constant turn</strong></td>
<td>Not applicable</td>
<td>initctekf</td>
<td>initctekf</td>
</tr>
</tbody>
</table>

MATLAB EXPO 2017
Fuse and track multiple detections from different sensors

- Radar Detections
  - Time
  - Position
  - Velocity

- Vision Detections
  - Time
  - Position
  - Velocity

Object Packer

- Time
- Position
- Velocity

Multi-Object Tracker

- Track Manager
- Kalman Filter

- Object Detections
  - Time
  - Measurement
  - Measurement Noise

- Tracks
  - Time
  - State
  - State Covariance
  - Track ID
  - Age
  - Is Confirmed
  - Is Coasted

- Typically unique to application and sensors
- Map sensor readings into measurement matrix
- Specify measurement noise for each sensor
Explore demo to learn more about fusing detections

Forward Collision Warning Using Sensor Fusion
product demo illustrates
• Packing sensor data into object detections
• Initializing Kalman filter
• Configuring multi-object tracker
Virtual scenario generation

- Specify driving scenario and roads
- Add ego vehicle
- Add target vehicle and pedestrian actor
- Play scenario with chase plot
- Create birds eye plot to view sensor detections
- Play scenario with sensor models
Simulate effects of vision detection sensor

**Range Effects**
- Range measurement accuracy degrades with distance to object

**Occlusion Effects**
- Partially or completely occluded objects are not detected

**Road Elevation Effects**
- Objects in coverage area may not be detected because they appear above the horizon line
- Large range measurement errors may be introduced for detected objects
Learn more about sensor fusion by exploring examples in the Automated Driving System Toolbox

- Design multi-object tracker based on logged vehicle data
- Generate C/C++ code from algorithm which includes a multi-object tracker
- Synthesize driving scenario to test multi-object tracker

MATLAB EXPO 2017
The Automated Driving System Toolbox helps you…

- Visualize vehicle data
  - Plot sensor detections
  - Plot coverage areas
  - Transform between image and vehicle coordinates

- Detect objects in images
  - Train deep learning networks
  - Label ground truth
  - Connect to other tools

- Fuse multiple detections
  - Design multi-object tracker
  - Generate C/C++
  - Synthesize driving scenarios

MATLAB EXPO 2017