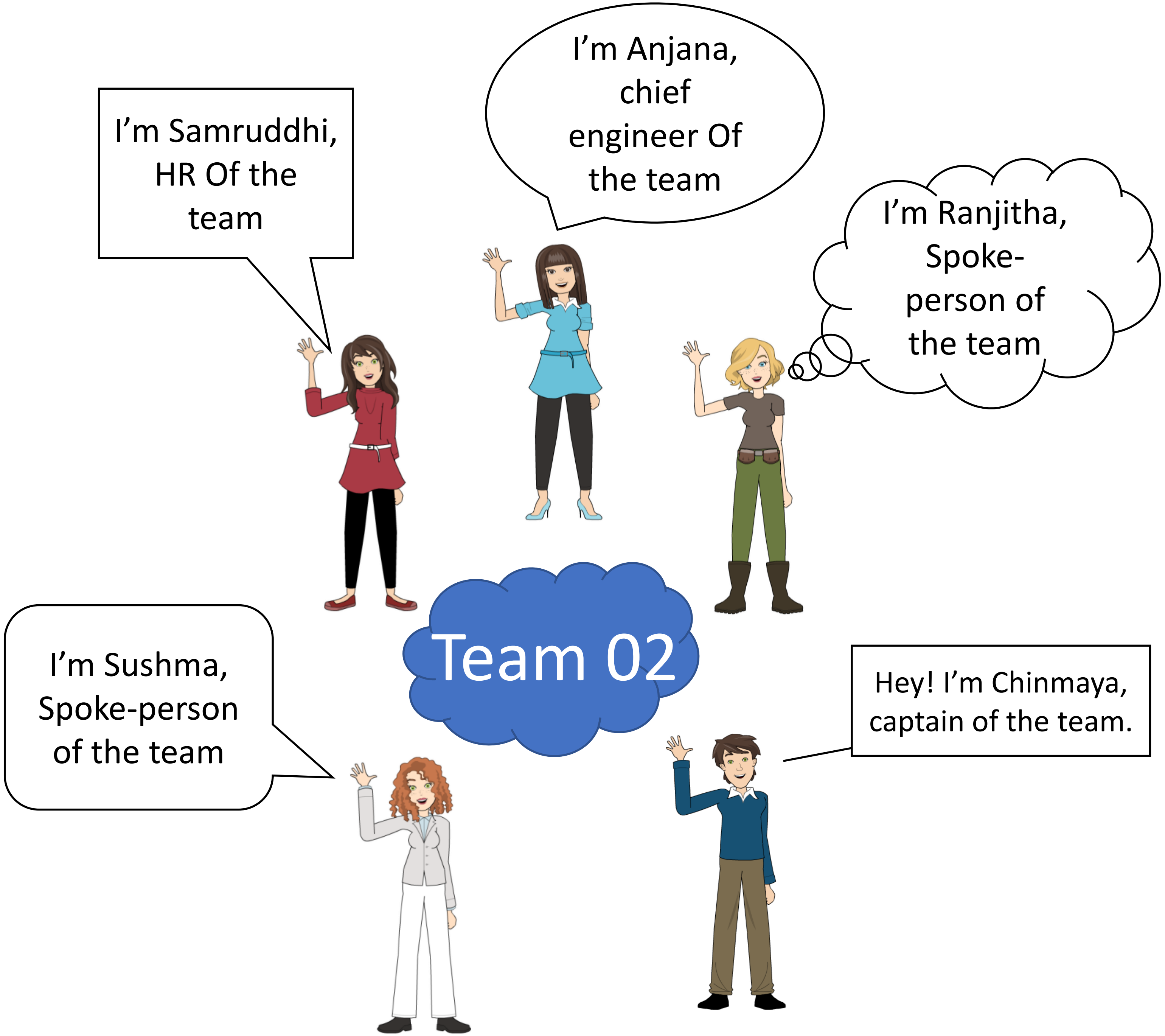
# Minor Project

# Simulation Of Sensor Fusion And Tracking Objects For KLE Tech AEV

By: Team 02

# Project Team

| USN | Name |
|---|---|
| 01FE17BAR013 | Chinmaya |
| 01FE17BAR029 | Anjana |
| 01FE17BAR048 | Ranjitha |
| 01FE17BAR051 | Samruddhi |
| 01FE17BAR057 | Sushma |



# Problem statement

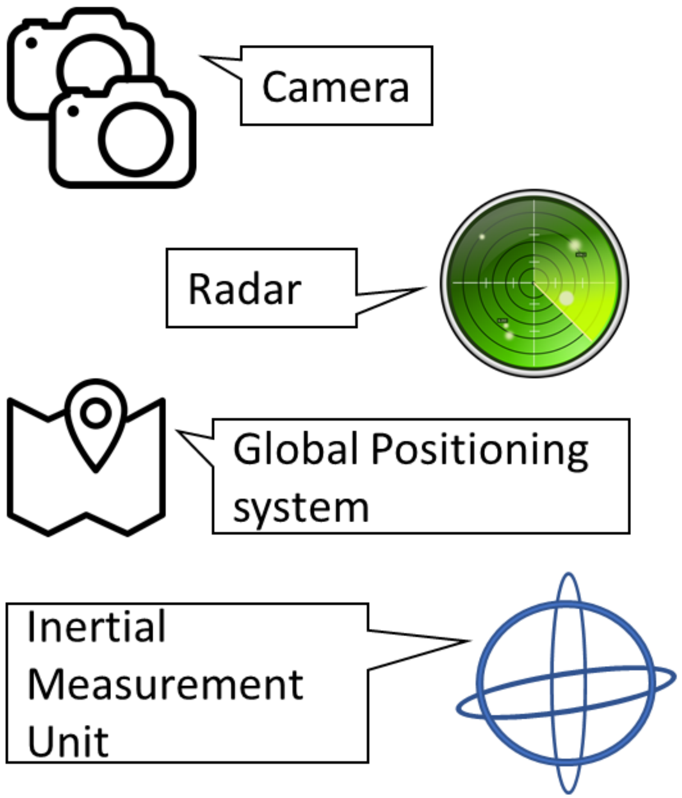
In the past decades, there has been a great interest and progress in the field of intelligent vehicles for both researches and industries. To reduce the traffic congestions and road accidents, we have to develop an efficient way for navigation in a specified area and test it for building an intelligent vehicle in a virtual scenario.

# Objectives

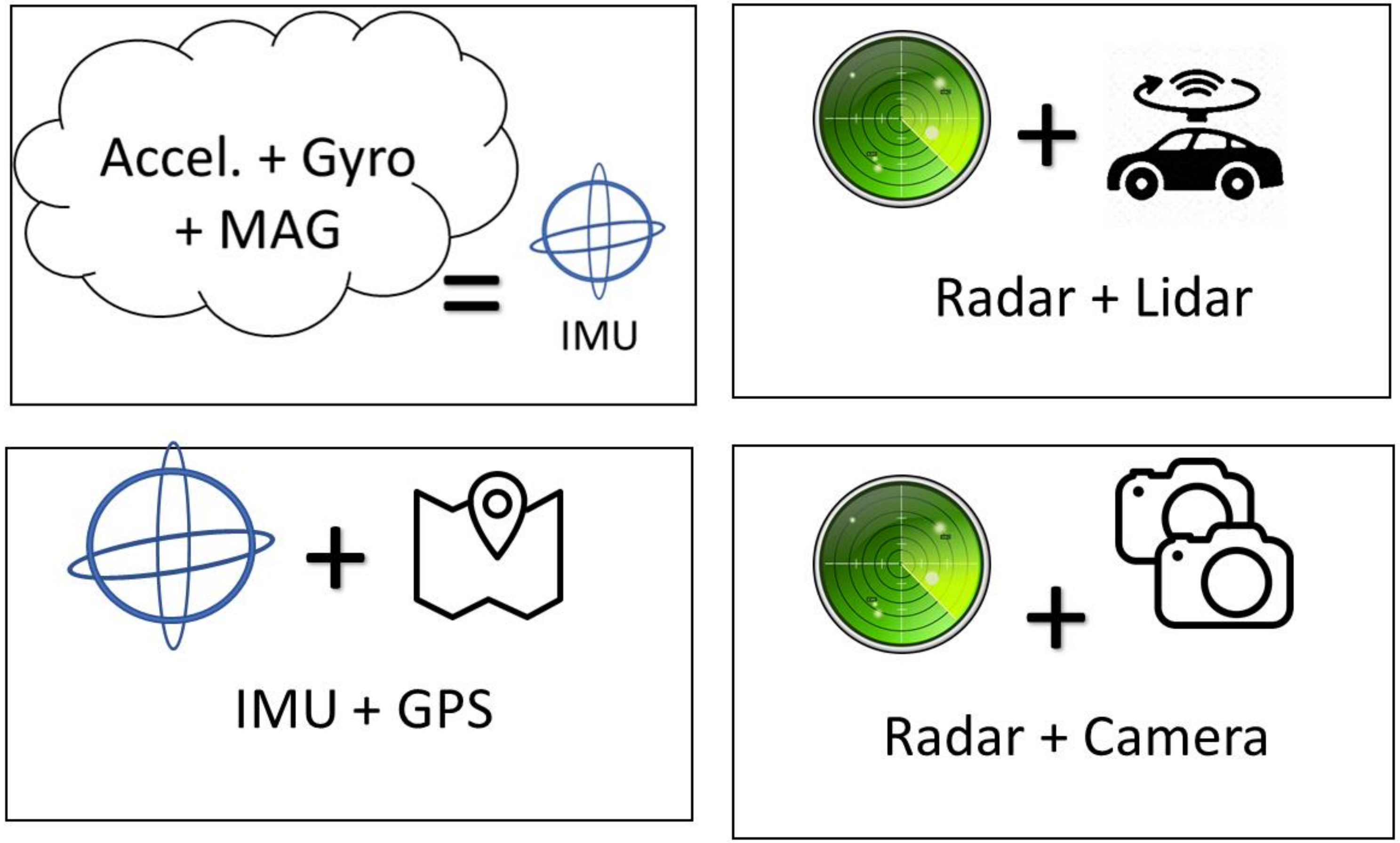
- Sensor modelling and simulation.
- Fuse multiple sensor data.
- Automate labelling of ground truth data.
- Build driving scenario.
- Generate vision detections.
- Simulate driving scenario in 3D environment.
- Develop drive by wire for KLE Tech AEV.

# Project Goals

## 1. Modelling of sensors

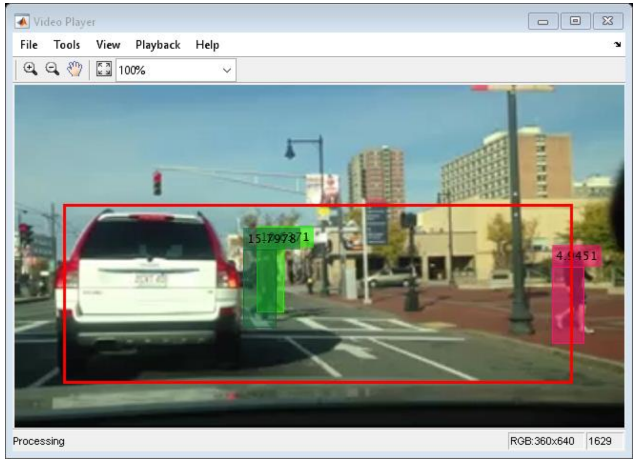


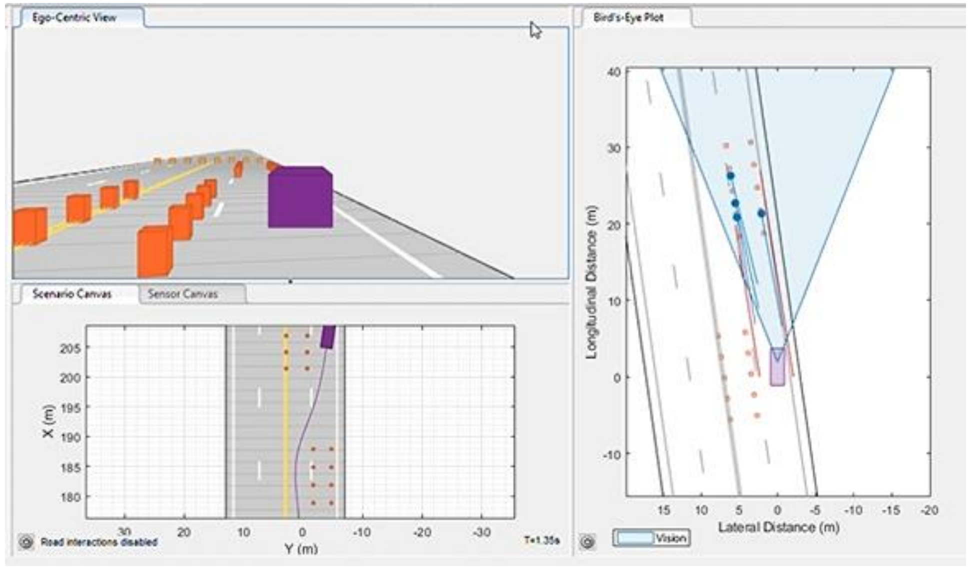

## 2. Simulate sensor fusion



# Project Goals

3. Tracking objects, lane and cars.




UNREAL ENGINE

4. Simulate in 3-D environment

# Introduction

- Autonomous systems processing loop with a focus on perception

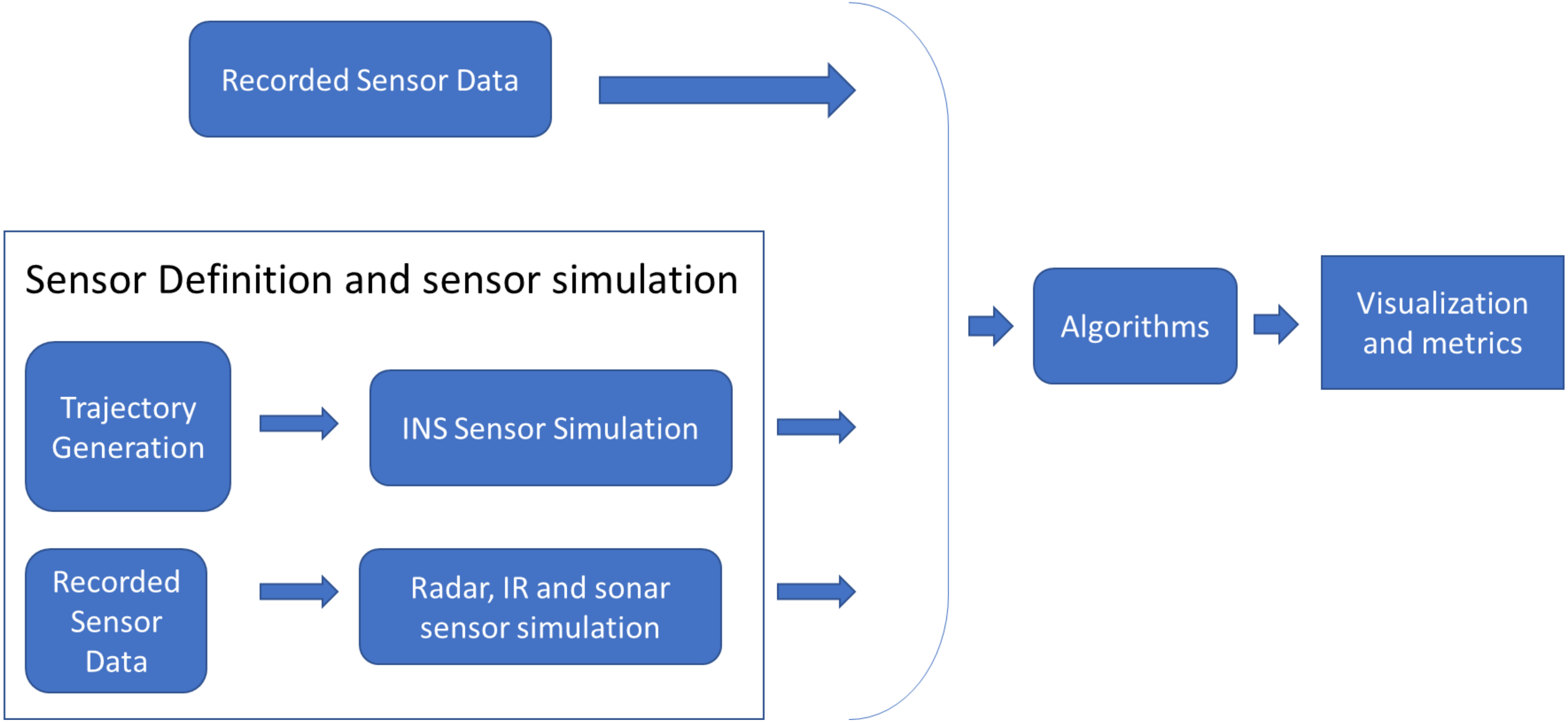
**Sensing**
- Detection
- Point Cloud
- Segmentation

**Perception**
- Tracks
- Occupancy map

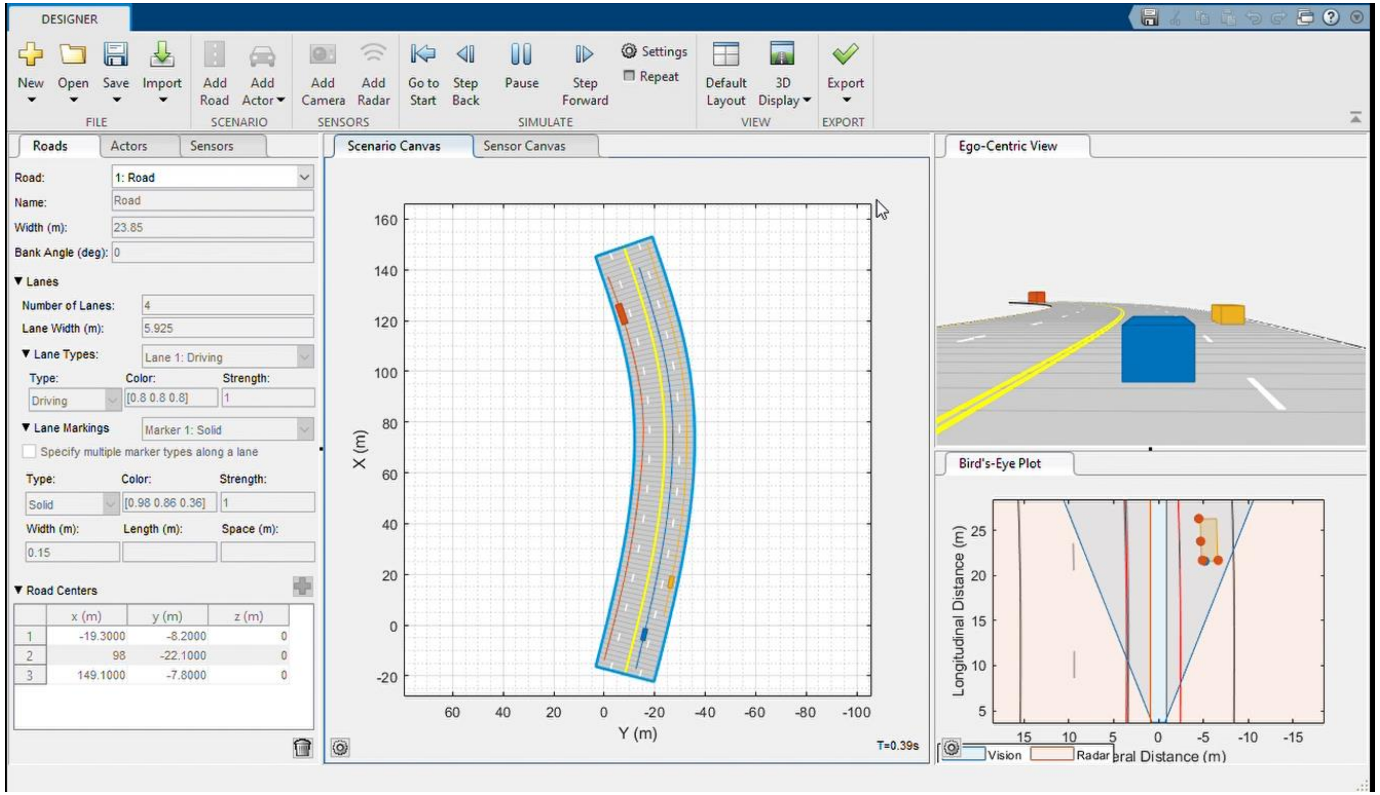
**Decision**
- Mission planning
- Path Planning

**Action**
- Execution
- Control

# Sensor fusion and tracking

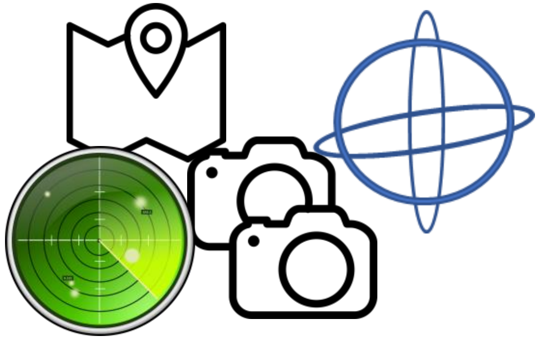


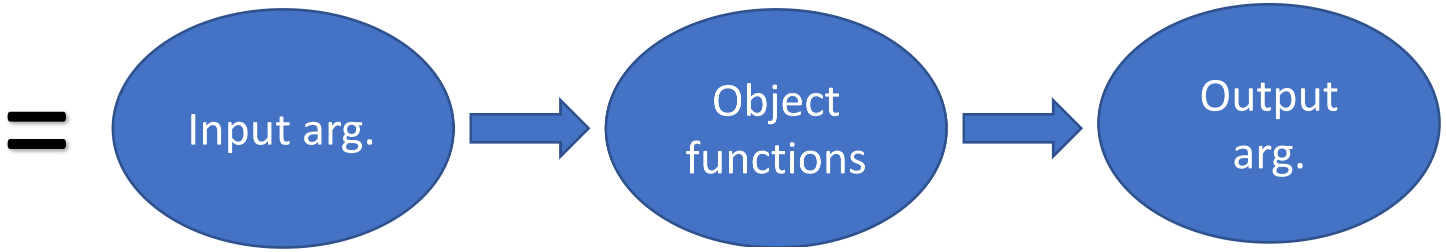
# Generating scenarios




Screenshots taken from the simulation.

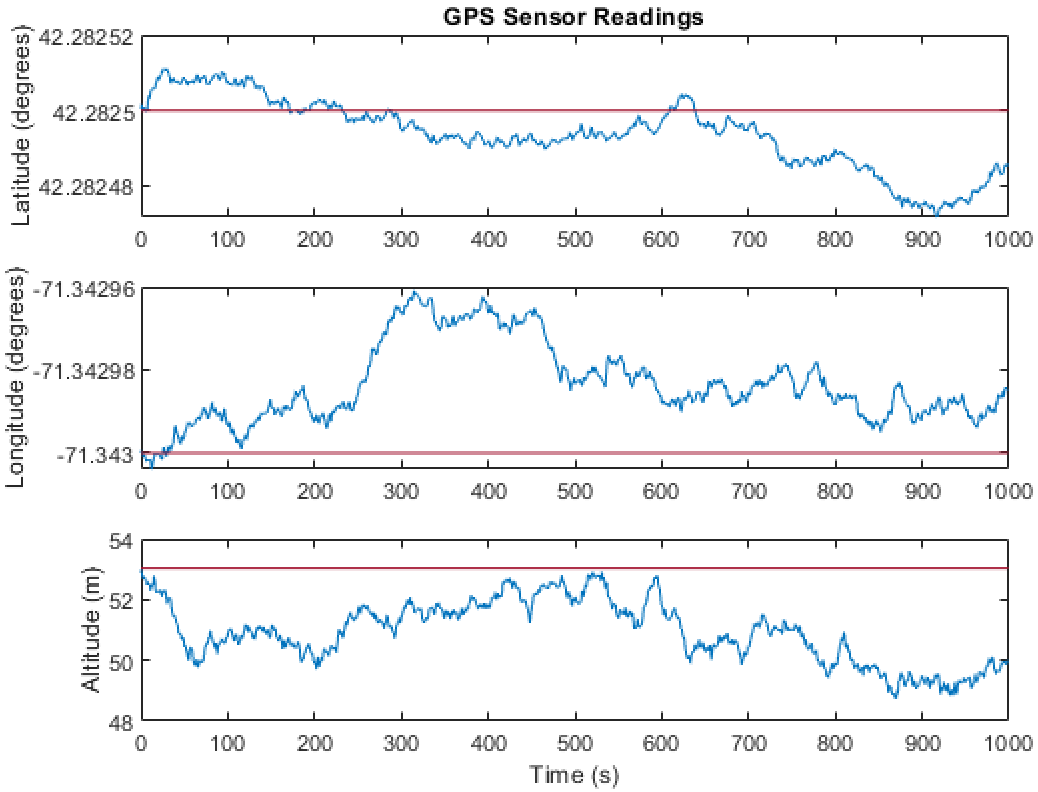
# Modeling of sensors



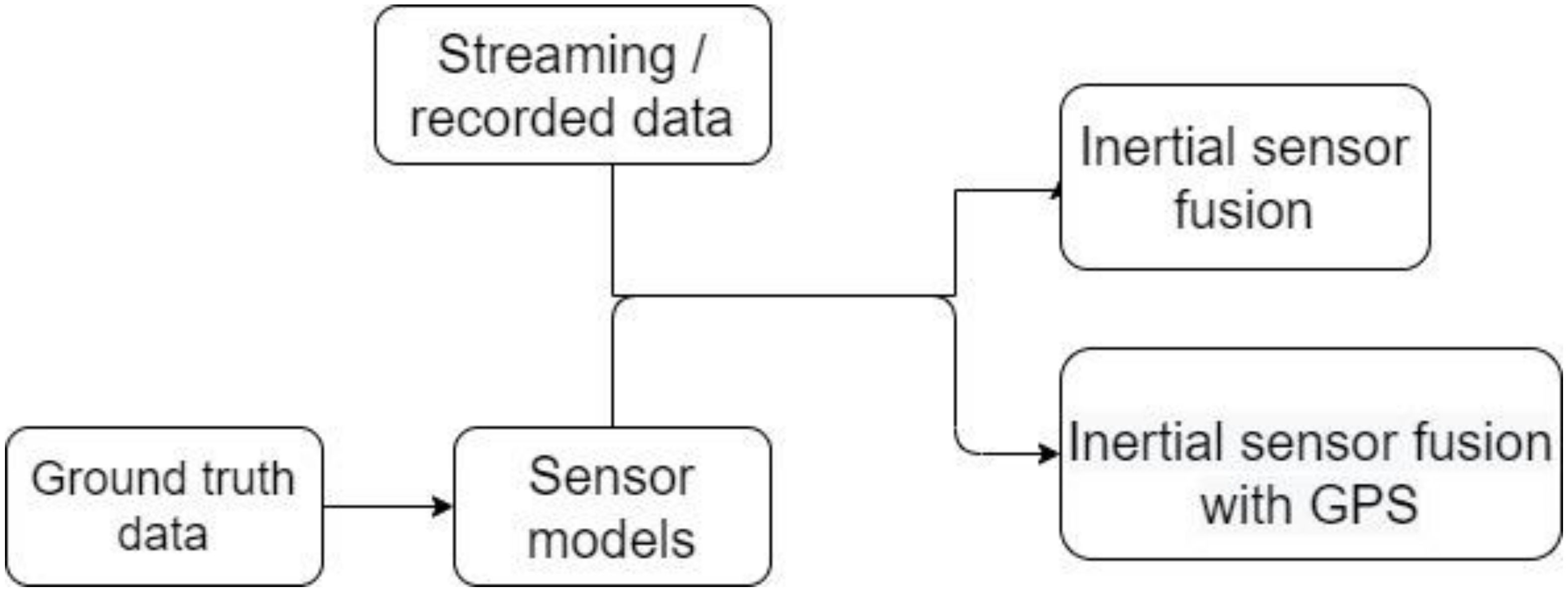



```
GPS =
  gpsSensor with properties:

                 UpdateRate: 1                   Hz
          ReferenceLocation: [0 0 0]             [deg deg m]
 HorizontalPositionAccuracy: 1.6                 m
   VerticalPositionAccuracy: 3                   m
           VelocityAccuracy: 0.1                 m/s
               RandomStream: 'Global stream'
                DecayFactor: 0.999
```

```
IMU =
  imuSensor with properties:

          IMUType: 'accel-gyro-mag'
       SampleRate: 100
      Temperature: 25
    MagneticField: [27.5550 -2.4169 -16.0849]
    Accelerometer: [1x1 accelparams]
        Gyroscope: [1x1 gyroparams]
     Magnetometer: [1x1 magparams]
     RandomStream: 'Global stream'
```

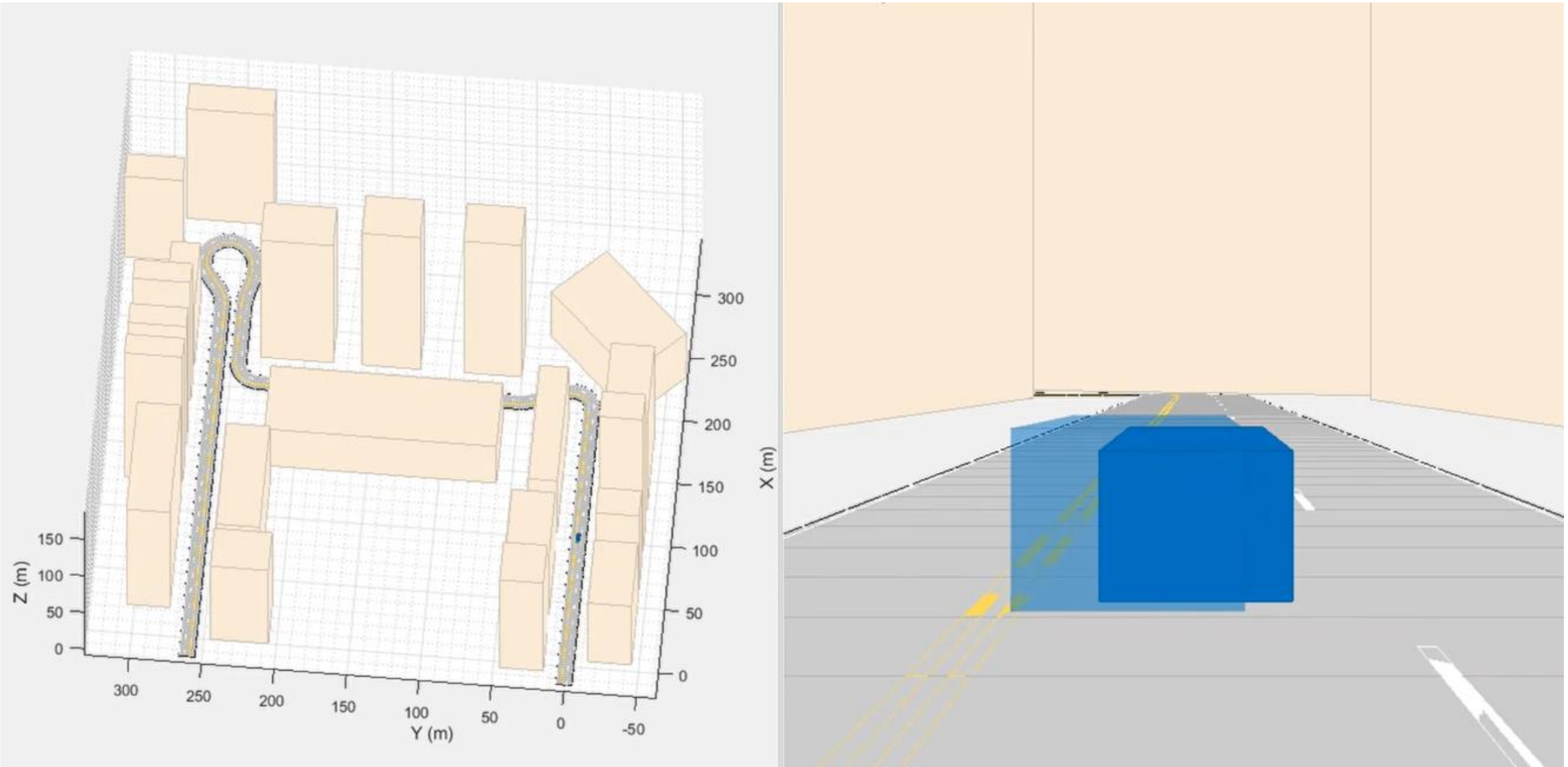


# localization of an autonomous system




Workflow for localization of an autonomous system

# localization of an autonomous system




Urban canyon where GPS signal is lost (left) and ground truth vs.
estimated position after visual odometry-inertial fusion (right).

# Interfacing to the Trackers

| Property | Description |
| --- | --- |
| Time | Time at which measurement was taken |
| Sensor Index | Unique identifier of sensor in the system |
| Measurement | What sensor measures |
| Measurement Noise | Uncertainty covariance of measurement |
| Measurement Parameters | List of parameters required for nonlinear measurement function |
| Object Class-Id | Integer representing the classification of the object |
| Object Attributes | Additional information sensor can provide |

Defining properties using the objectDetection class.

```
objectDetection with properties:

                 Time: 1
          Measurement: [3×1 double]
     MeasurementNoise: [3×3 double]
          SensorIndex: 1
        ObjectClassID: 0
MeasurementParameters: [1×1 struct]
     ObjectAttributes: {[1×1 struct]}
```

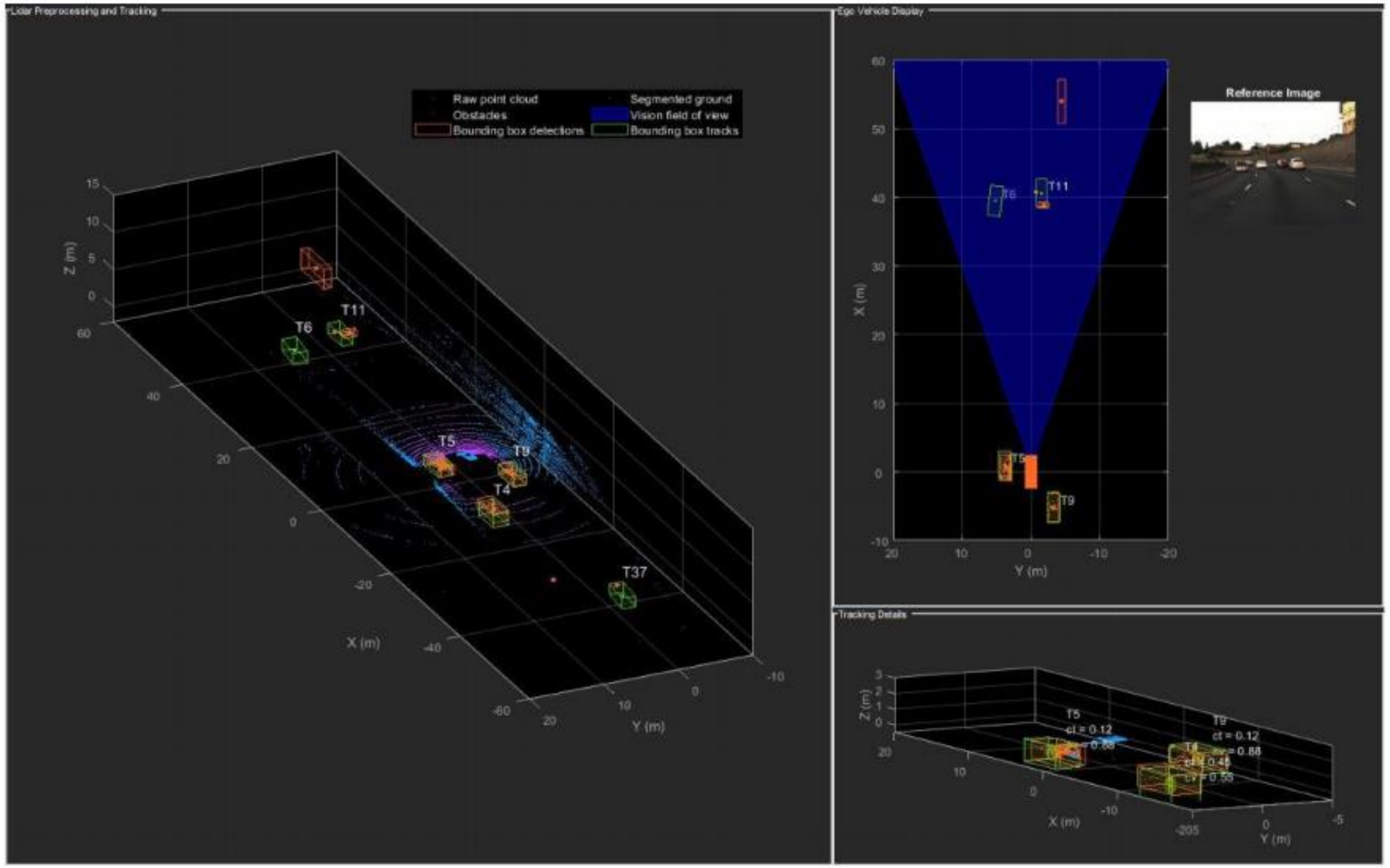
Screenshot taken from MATLAB command window defining objectDetection class.

# Tracking Algorithms

- Joint probabilistic data association tracker (JPDA).

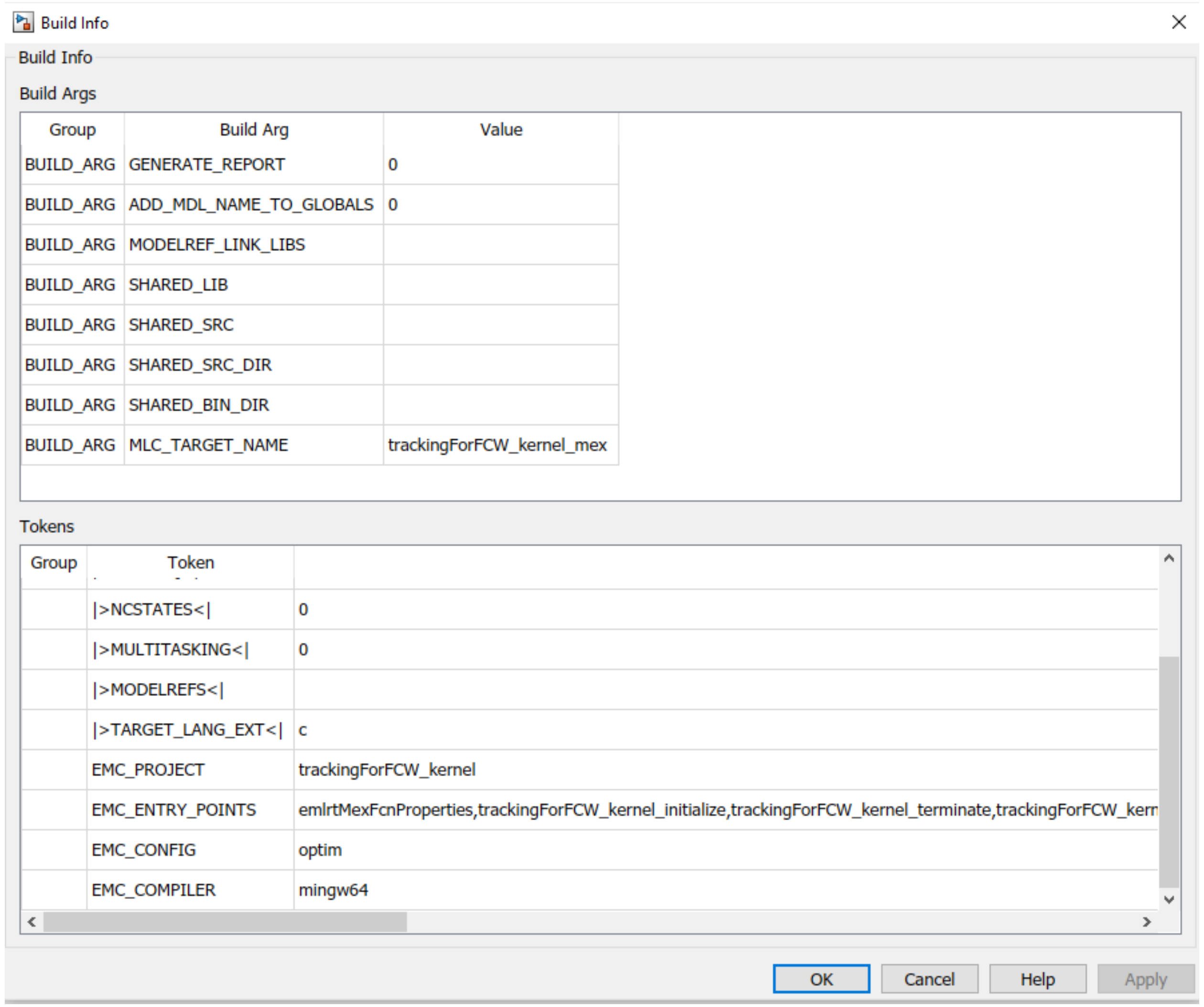
| Input Arguments | Output Arguments |
|---|---|
| detections — Detection list | confirmedTracks — Confirmed tracks |
| time — Time of update | tentativeTracks — Tentative tracks |
| costMatrix — Cost matrix | allTracks — All tracks |
| detectableTrackIDs — Detectable track IDs | analysisInformation |

# Lidar Tracking




- Lidar tracker using bounding box detections.
- Joint probabilistic data association tracker (JPDA).

# Generating C code


Build Info

Build Info

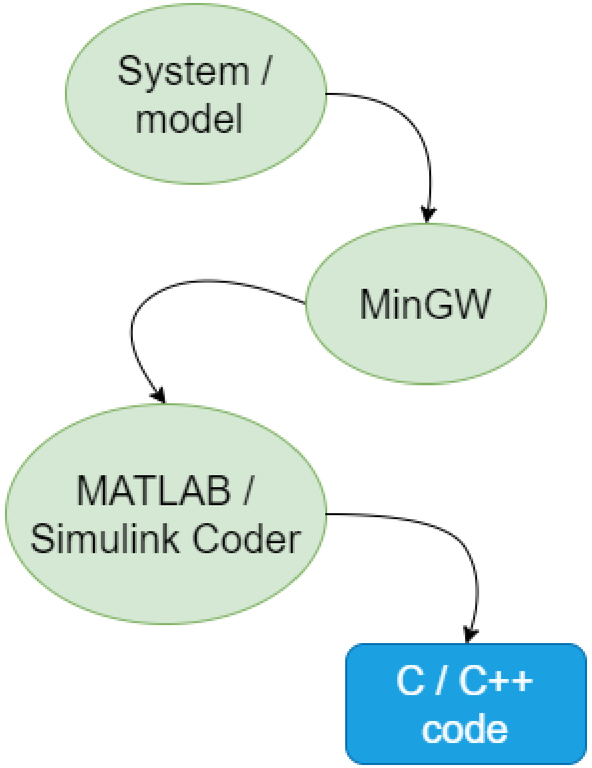
Build Args

| Group | Build Arg | Value |
|---|---|---|
| BUILD_ARG | GENERATE_REPORT | 0 |
| BUILD_ARG | ADD_MDL_NAME_TO_GLOBALS | 0 |
| BUILD_ARG | MODELREF_LINK_LIBS | |
| BUILD_ARG | SHARED_LIB | |
| BUILD_ARG | SHARED_SRC | |
| BUILD_ARG | SHARED_SRC_DIR | |
| BUILD_ARG | SHARED_BIN_DIR | |
| BUILD_ARG | MLC_TARGET_NAME | trackingForFCW_kernel_mex |

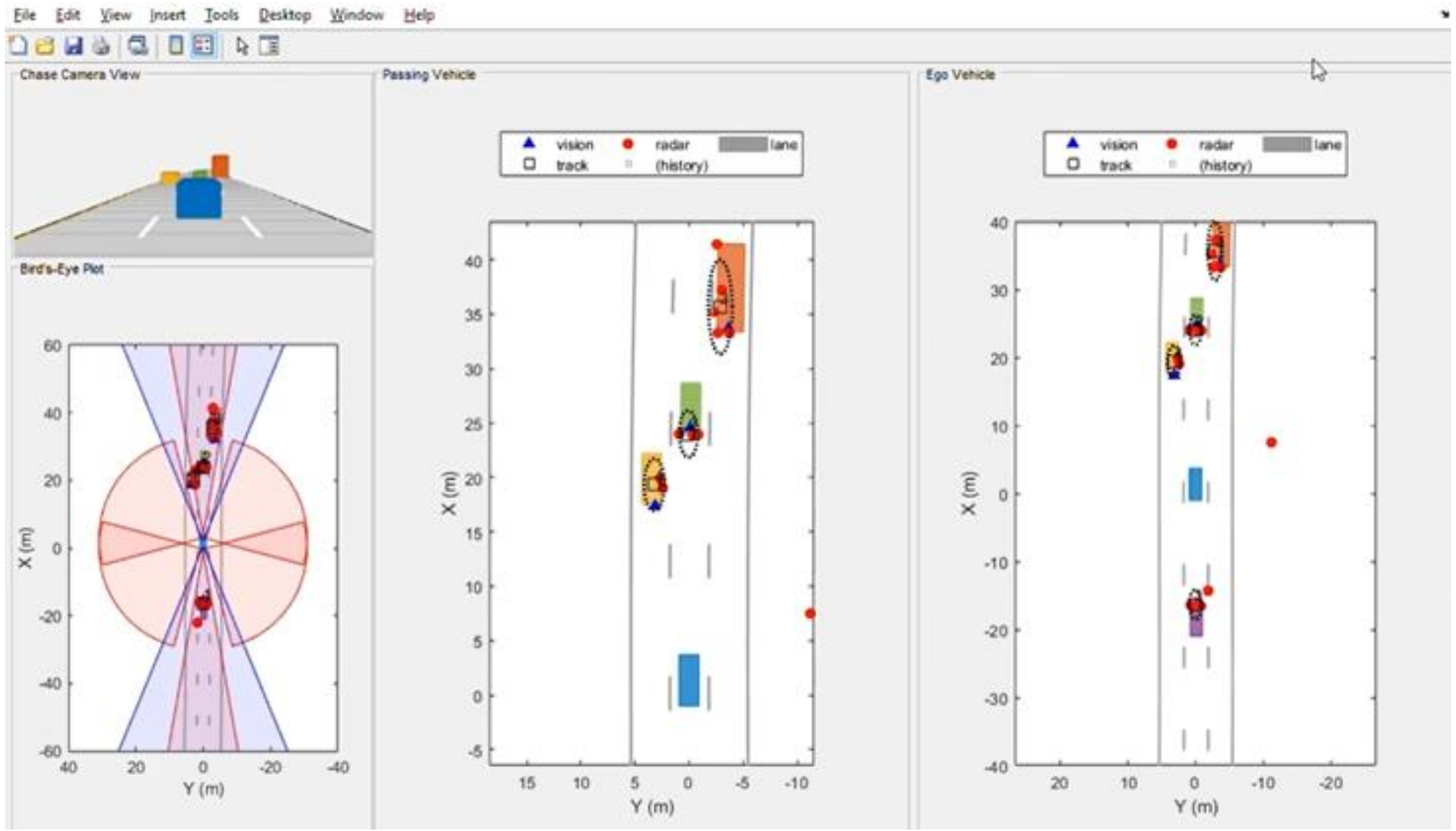
Tokens

| Group | Token | |
|---|---|---|
| | \|>NCSTATES<\| | 0 |
| | \|>MULTITASKING<\| | 0 |
| | \|>MODELREFS<\| | |
| | \|>TARGET_LANG_EXT<\| | c |
| | EMC_PROJECT | trackingForFCW_kernel |
| | EMC_ENTRY_POINTS | emlrtMexFcnProperties,trackingForFCW_kernel_initialize,trackingForFCW_kernel_terminate,trackingForFCW_kern |
| | EMC_CONFIG | optim |
| | EMC_COMPILER | mingw64 |

OK Cancel Help Apply

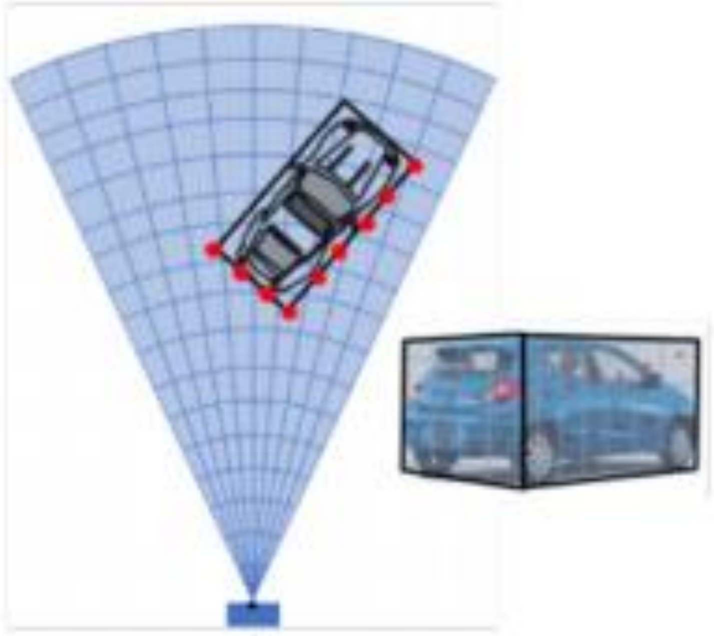


# Tracking Extended Objects



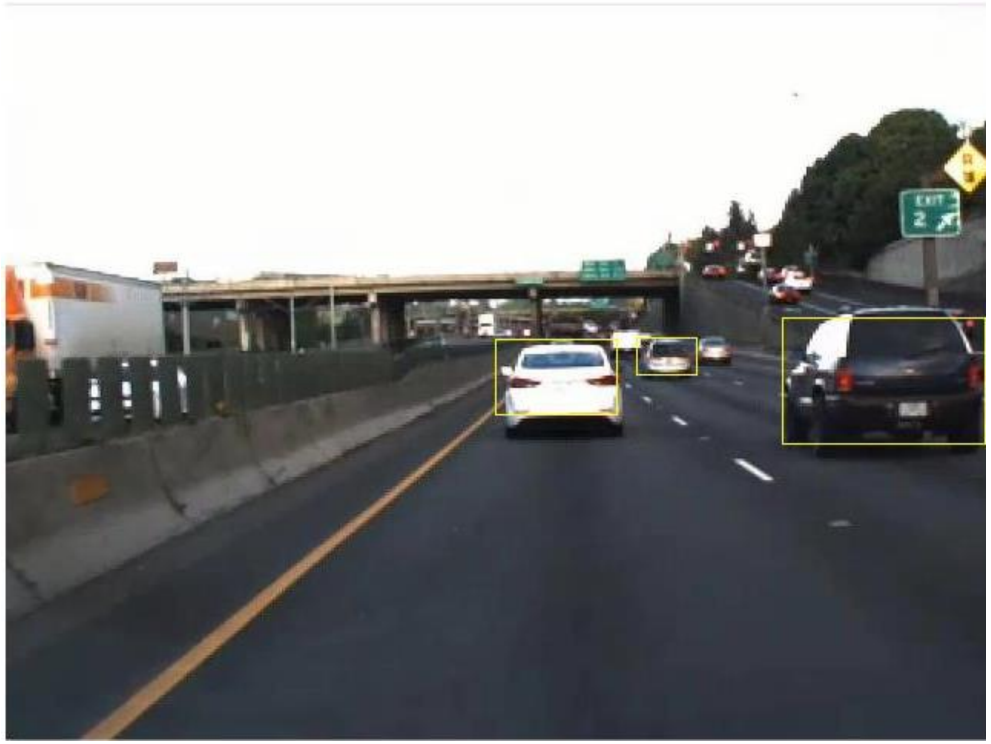

Results of extended object tracker using an ellipse

Why use Extended objects instead point objects?



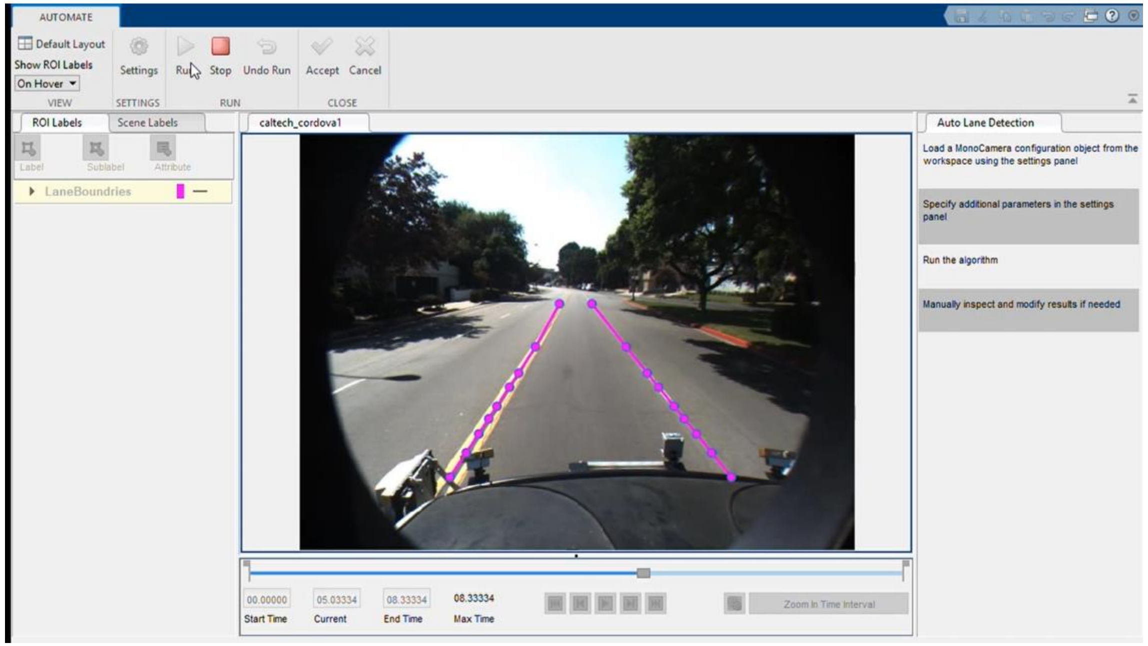
# Ground Truth Labeller: Dist. estimation

Detected Vehicles



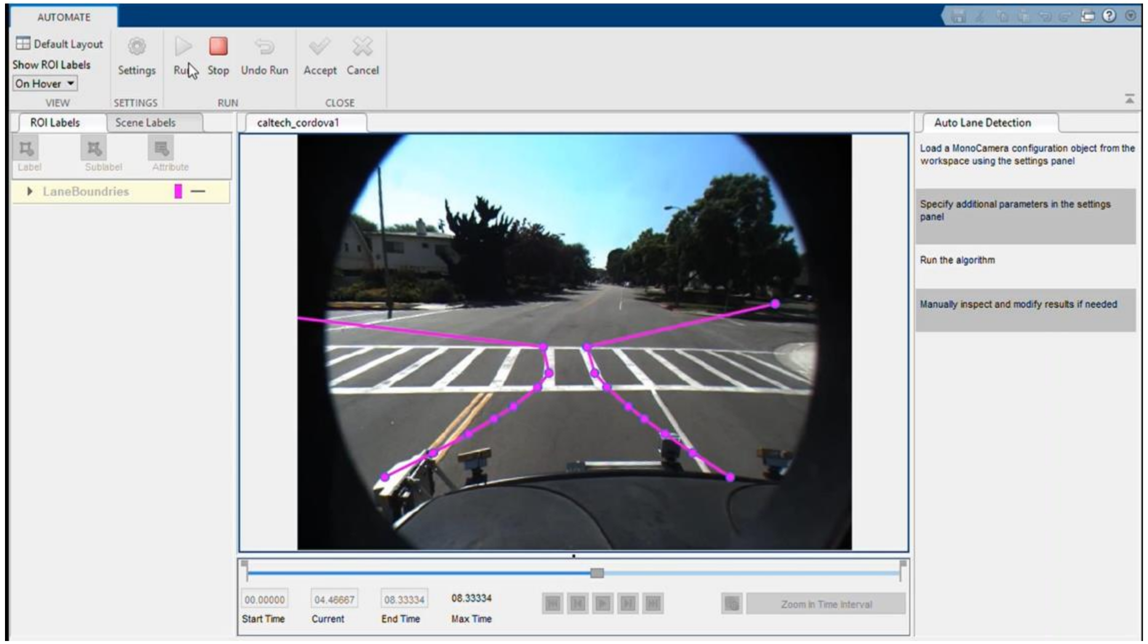
Distances of Vehicles from Camera

# Ground Truth Labeller: Lane Detection

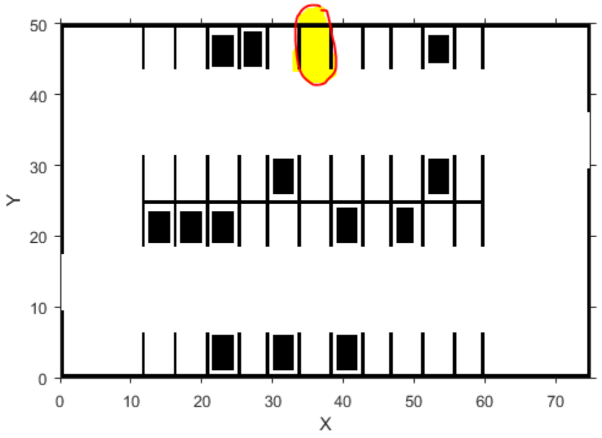



Lane detections with complete lane markings

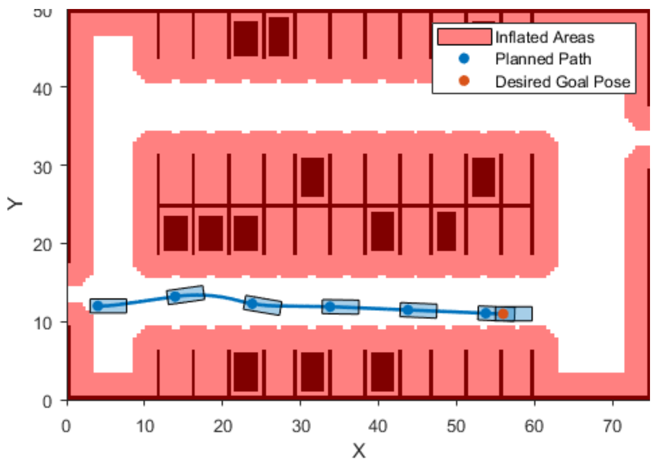
Lane detections with incomplete lane markings



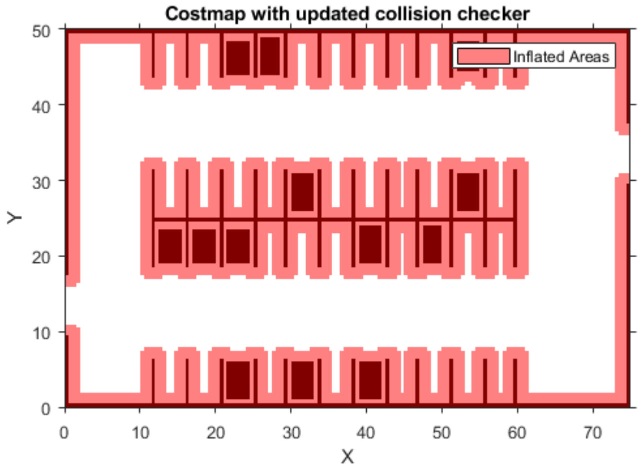
# Motion Planning



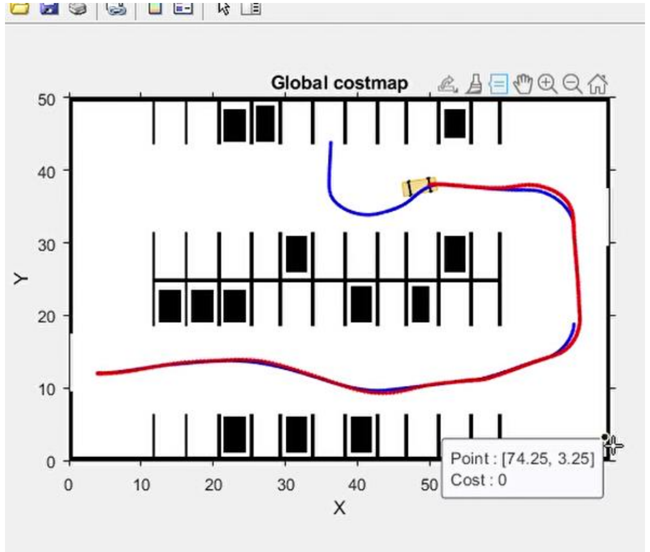

1



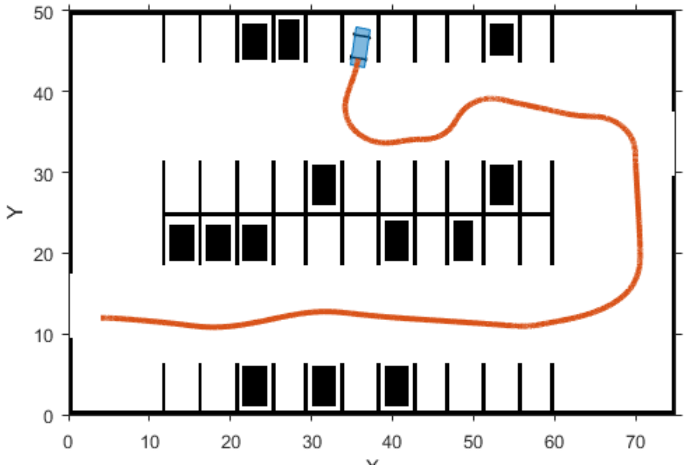

2



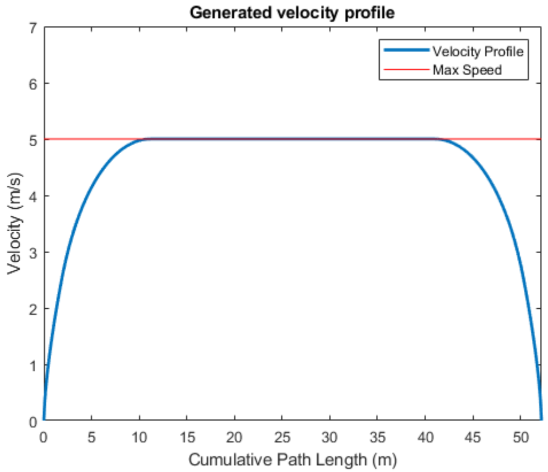

3




4




5




6

# Conclusion

- Understandings or Learnings:

1. Detailed architecture and components of a self-driving car software stack.
2. Realistic vehicle physics, complete sensor suite.
3. Demonstrate skills in CARLA and build programs with Python.

- Implement methods:

1. static and dynamic object detection.
2. localization and mapping.
3. behaviour and maneuver planning.
4. vehicle control.