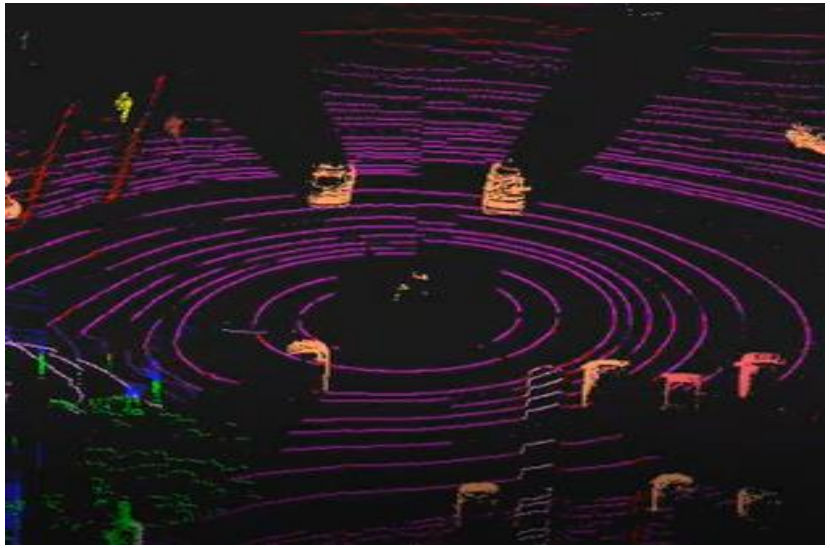
MATLAB EXPO 2021

레이더 및 라이다 데이터 처리를 위한 인공지능

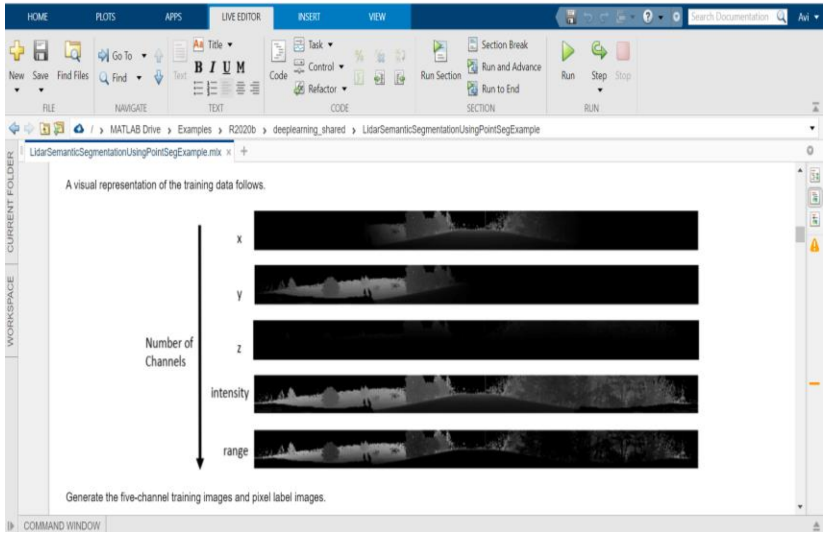
정승혁 차장

MathWorks

3 Things We’ll Cover Today



- Data Synthesis
- Labeling
- Pre-processing
- Model selection and training
- Full system deployment



Insight
AI Applications for Radar and Lidar

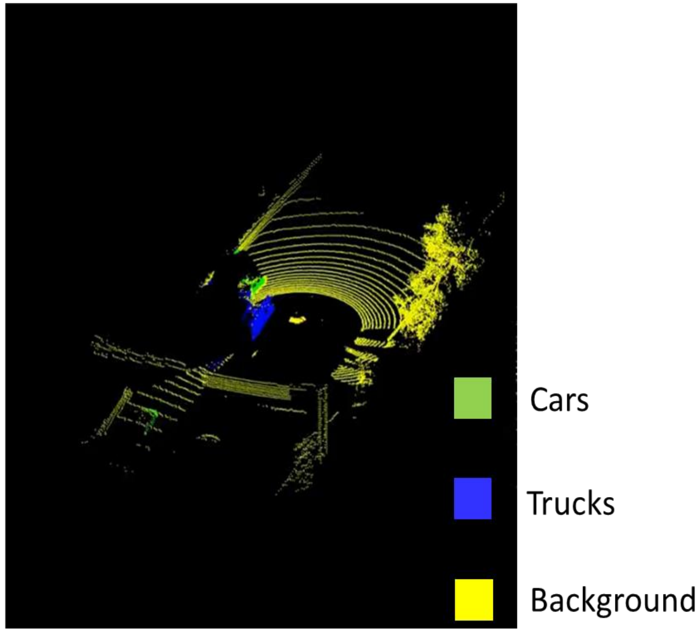
Challenges
Common issues engineers face in practice

Interaction
AI models for radar and lidar data

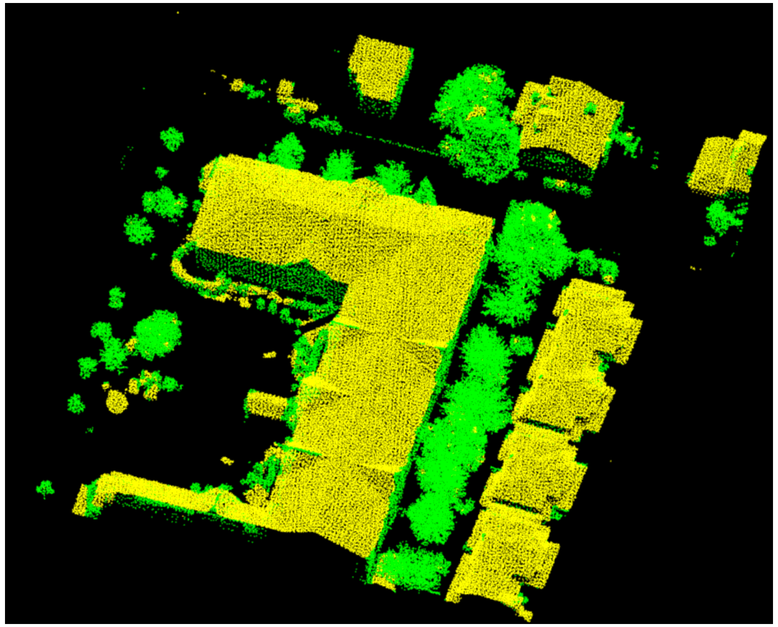
What is a lidar sensor and where is AI used ?

Lidar: Li*ght* d*etection* a*nd* r*anging*

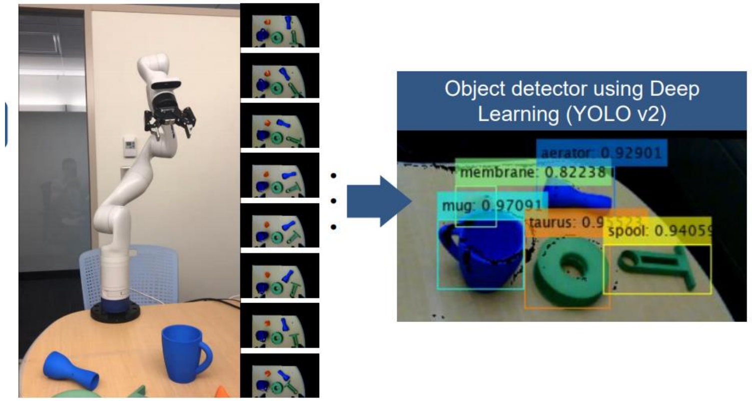
- Creates 2D or 3D point clouds representing depth using pulsed-light
- Also known as 3D laser scanner, laser scanner



Autonomous Perception

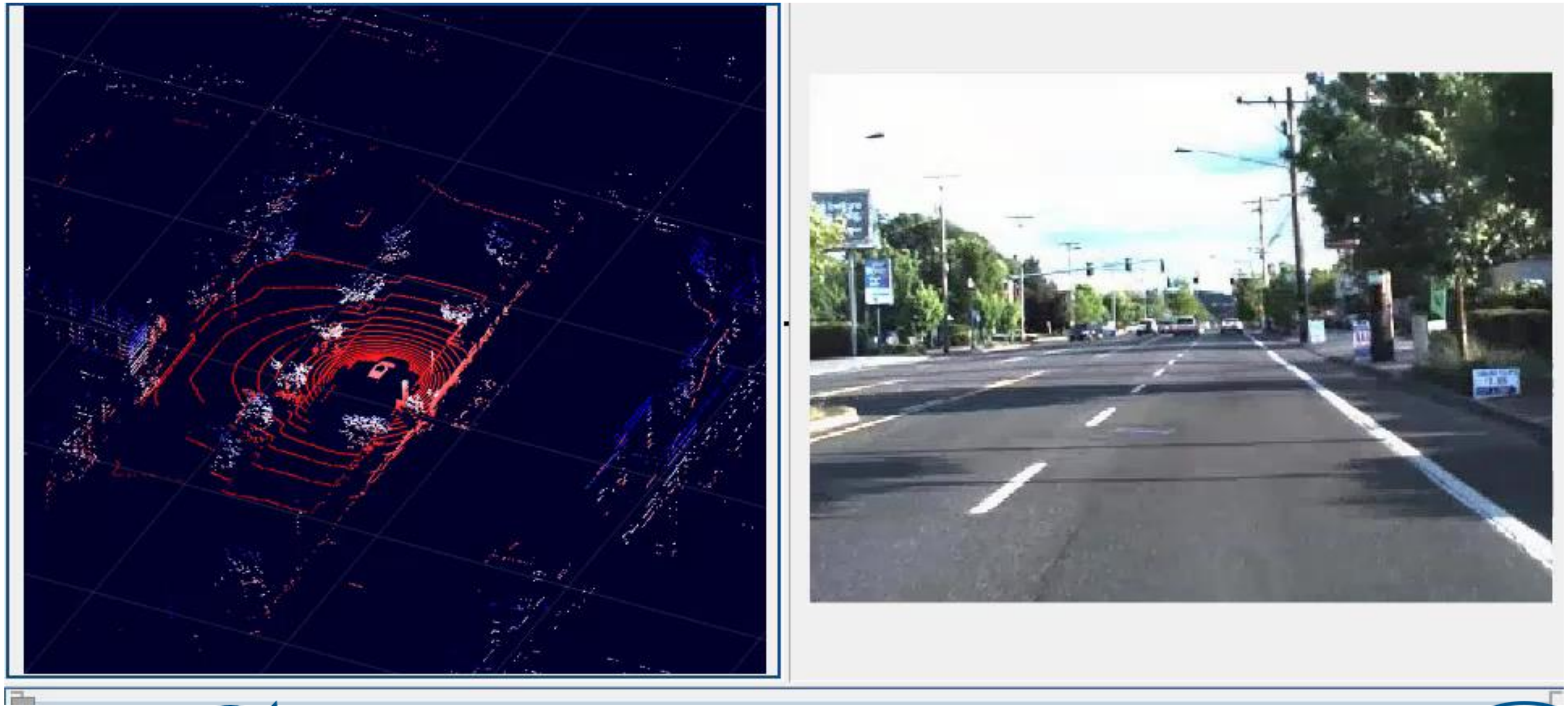


Aerial Imaging and Navigation



Robotics and Augmented Reality

What are the advantages and disadvantages of lidar sensors ?



Accurate Depth

Dense Data

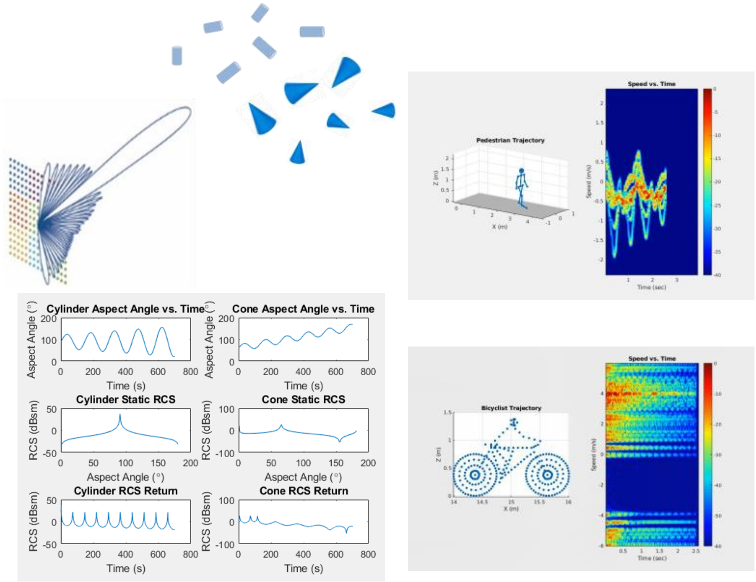
<u>Disadvantages of lidar sensors</u>

- Sensitive to rain, snow and weather effects
- Measurement effected by platform movement/vibration
- Accuracy drops as range increases

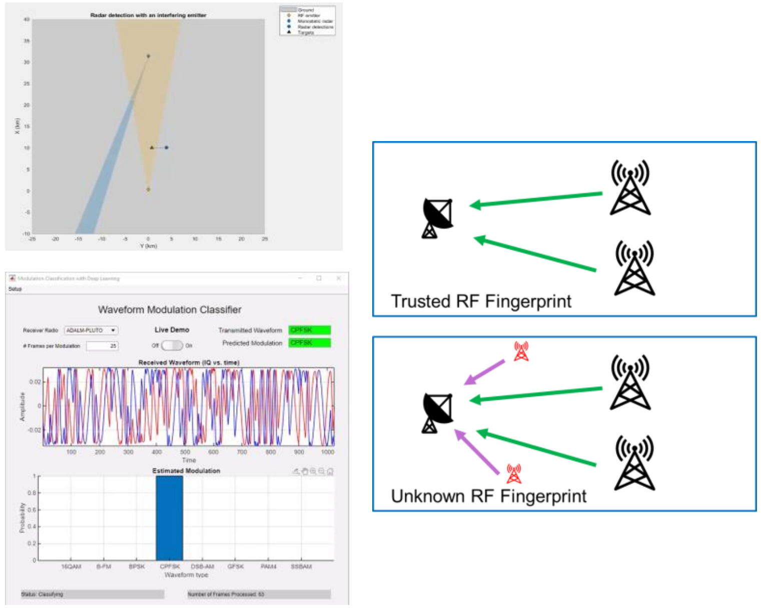
What is a radar sensor and where is AI used ?

Radar: Radio *detection* and *ranging*

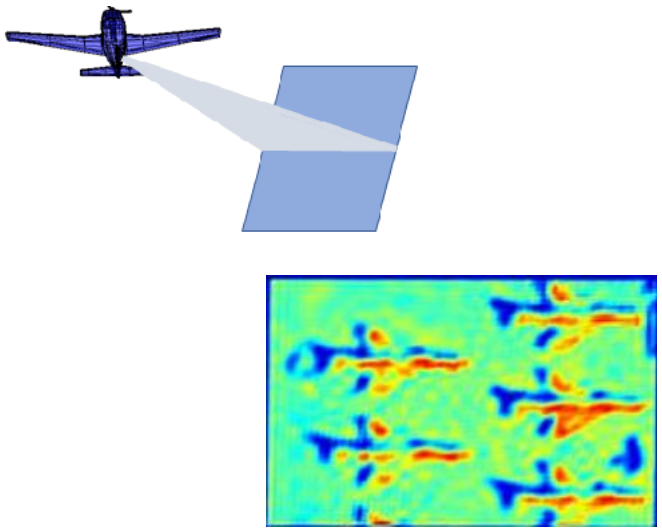
- Use radio frequency echos to detect objects at a distance
- Estimate position, Doppler, and micro-Doppler.
- Generate images with 4D radar



Target classification

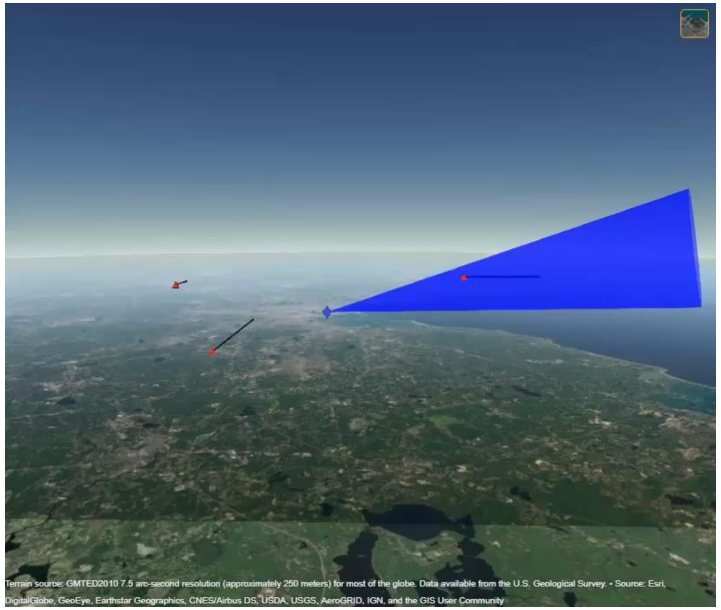


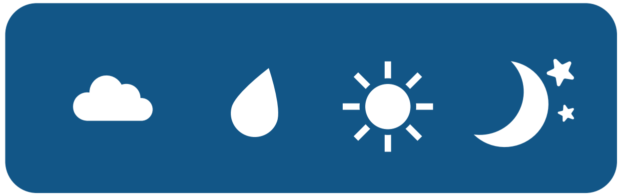
Signal identification

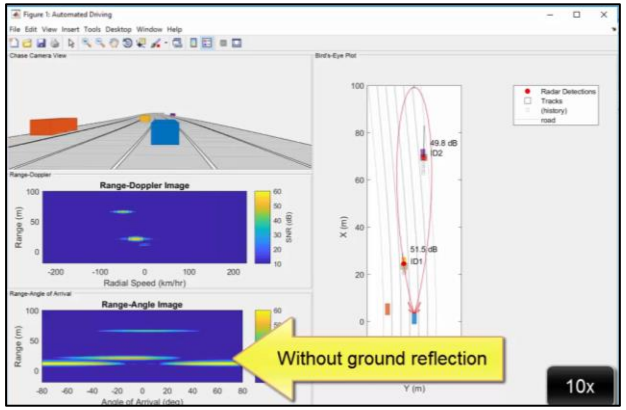


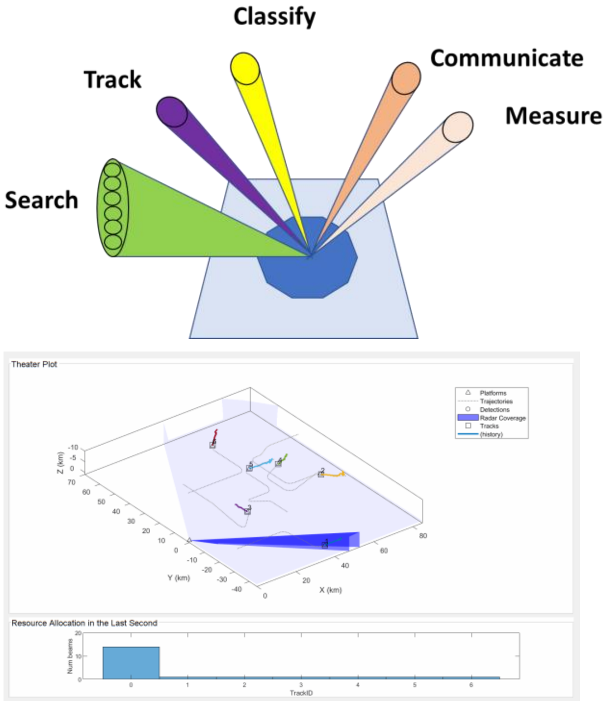
SAR imaging

What are the advantages and disadvantages of radar sensors?









Long range operations

All weather, night and day

Flexibility

Disadvantages of radar sensors

- Lower resolution than lidar
- Lower azimuthal resolution at longer ranges
- Multipath and clutter cause ghost detections and false detections

What are the common challenges engineers face using AI with radar and lidar ?

1. **Labeling recorded data** for AI training is manual and time consuming

2. Little-**no recorded data** to train models for safety-critical applications

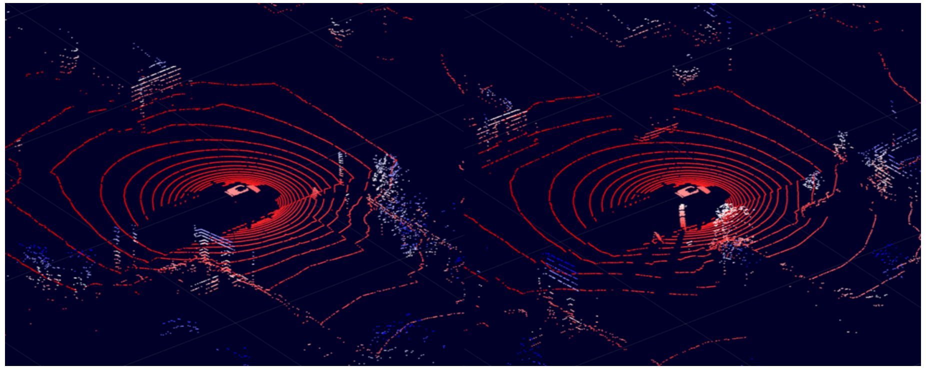
3. **Unfamiliarity with AI models** for radar and lidar

4. Unclear how to **pre-process sensor signals** for best results

5. Real-world systems require **deployment of more than AI model**

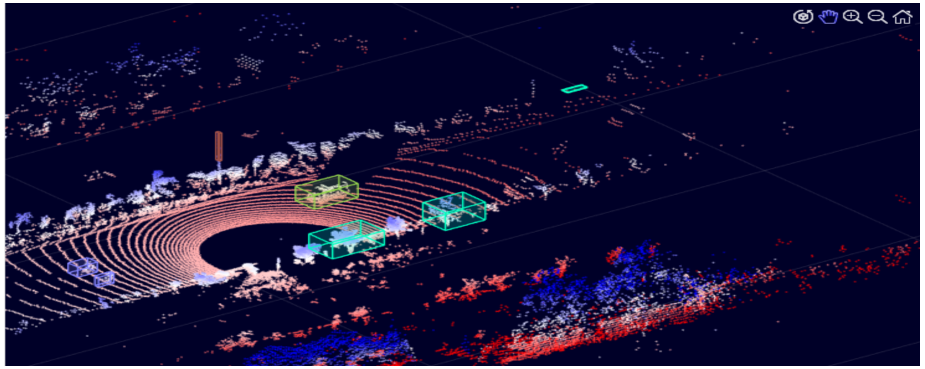
Challenge

Labeling data is repetitive, manual and time consuming



Repetitive and manual

Very little variation frame-frame

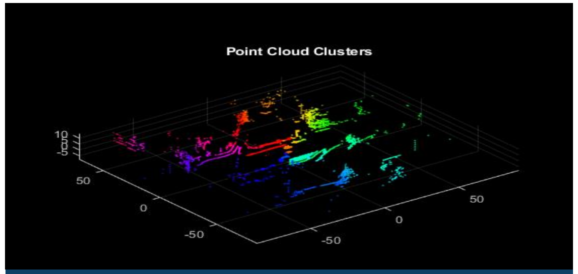


Noise

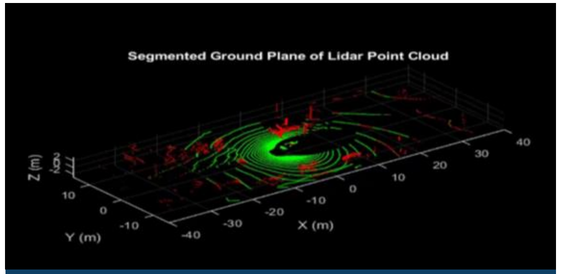
Majority of points not required to train AI model

Two steps to improving accuracy and efficiency of labeling process

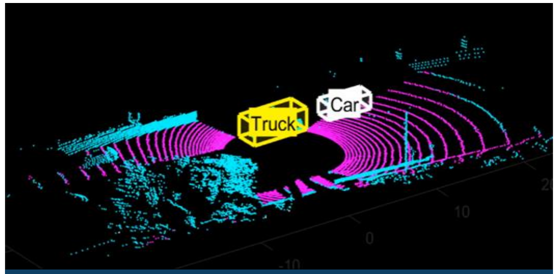
1. Automation using non-AI techniques



Clustering

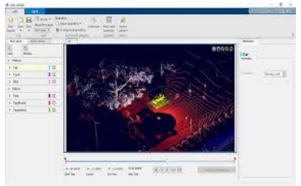


Ground Plane Removal



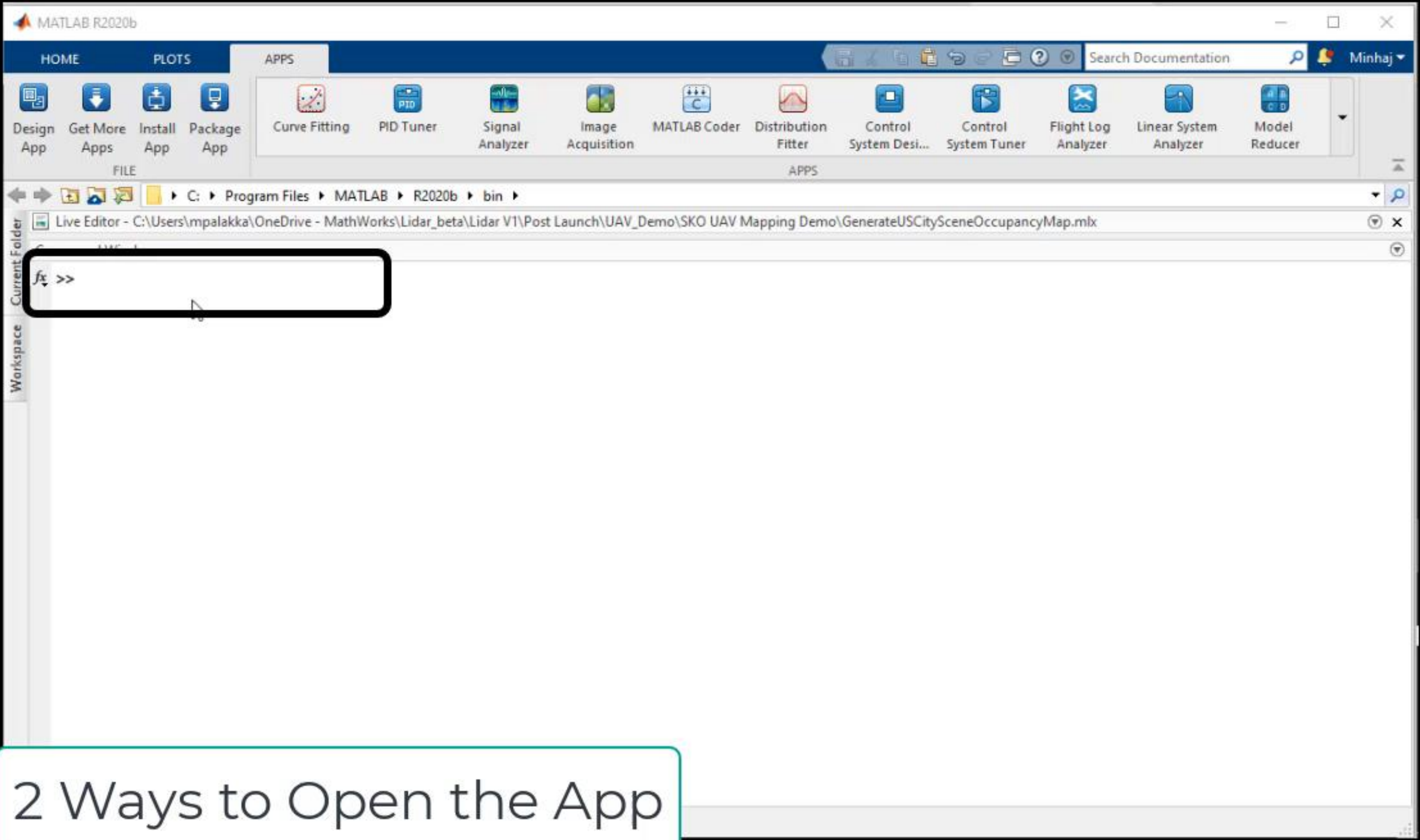
Tracking

2. Iterative training and labeling



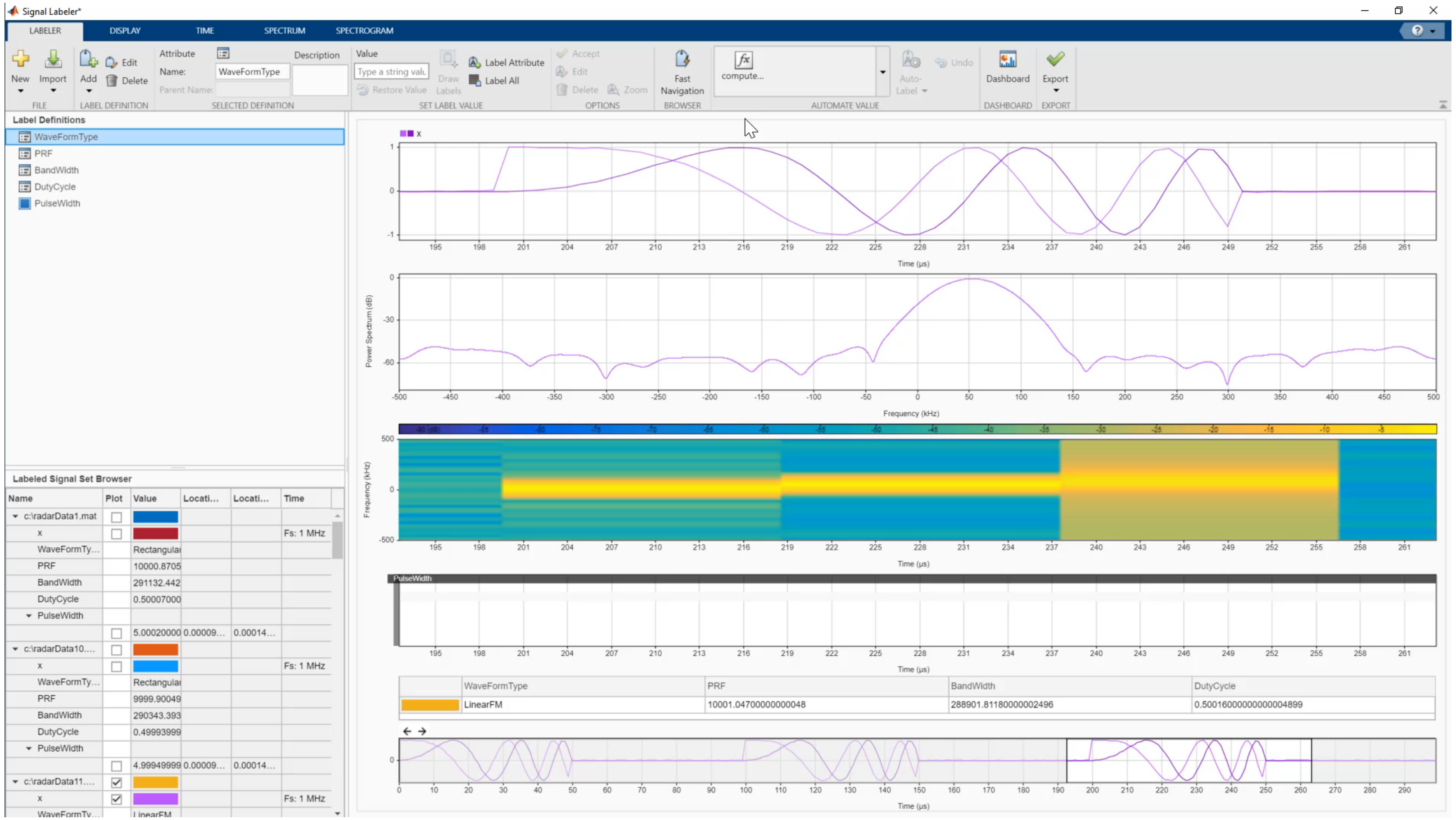
Train Model

Iteration and Refinement



2 Ways to Open the App

Labelling radar signals can also be done automatically

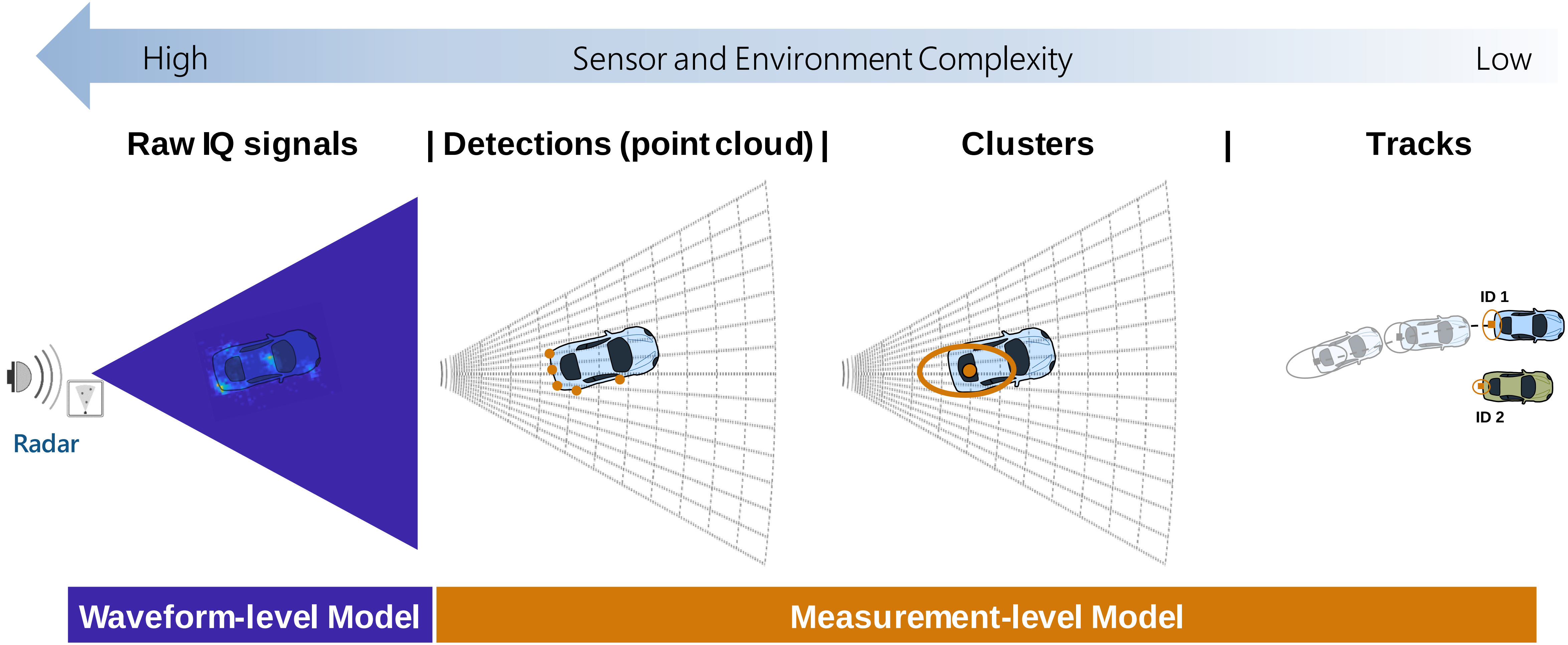


Automatically label signals with custom functions

Explore and label signals with time, frequency, and time-frequency views

Track labelling statistics with integrated dashboards

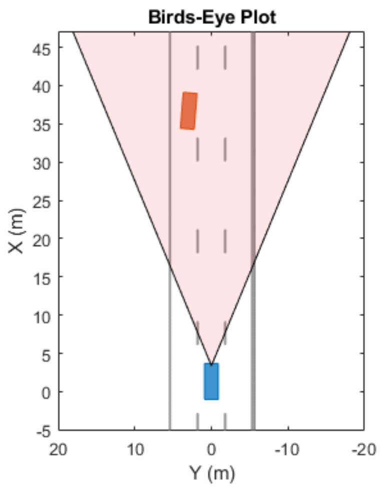
Simulating radar data in MATLAB and Simulink

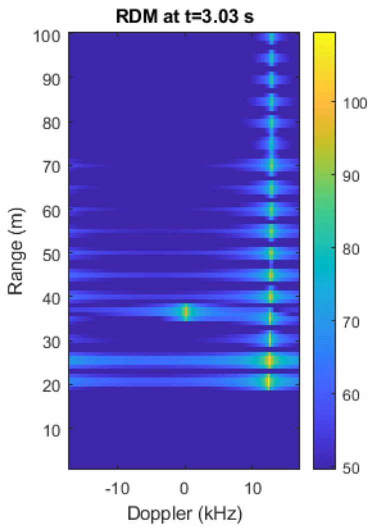


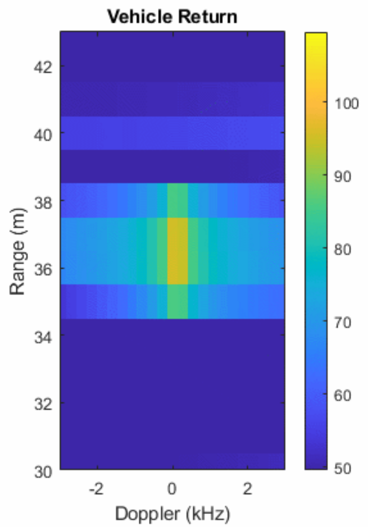
Simulating radar data in MATLAB and Simulink

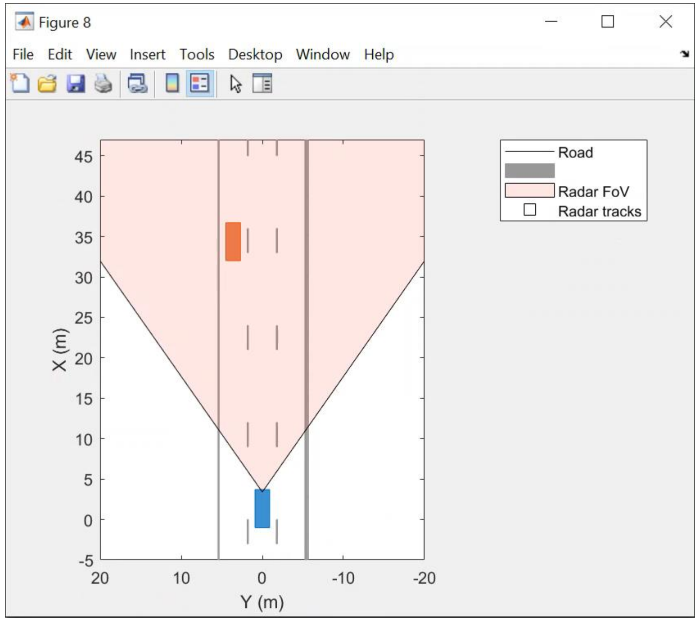
High — Sensor and Environment Complexity — Low

Raw IQ signals | **Detections (point cloud)** | **Tracks**



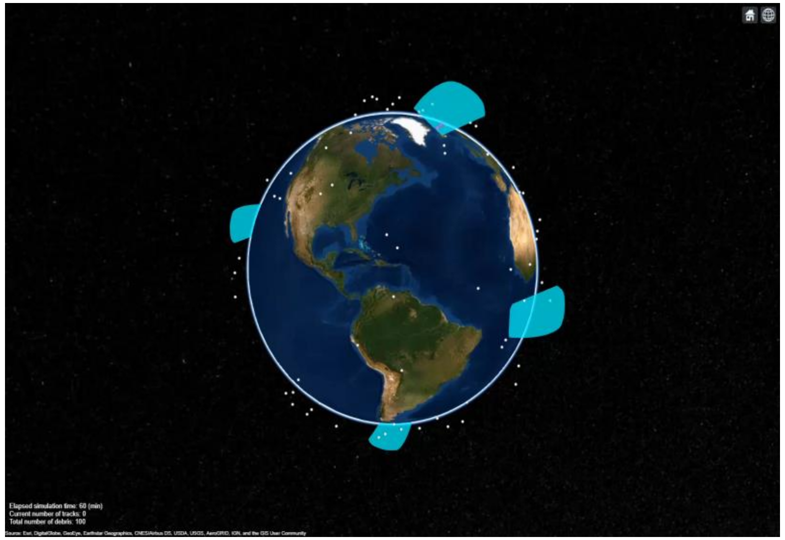




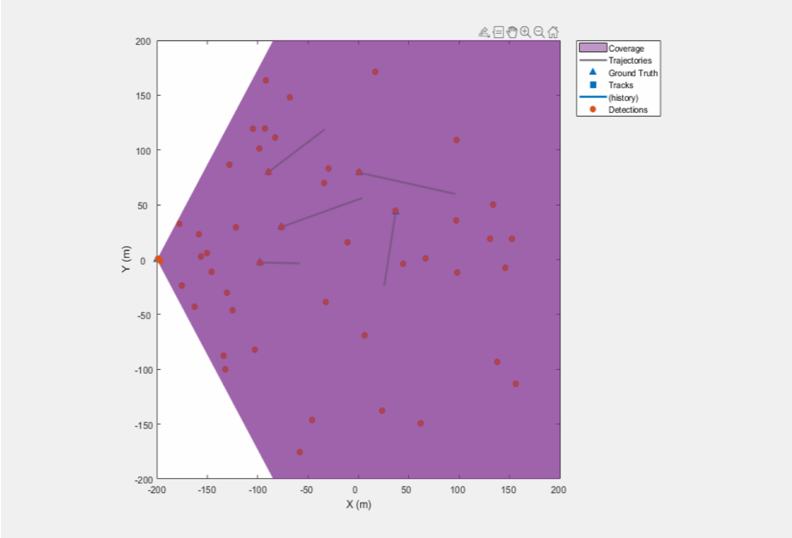


Waveform-level Model | **Measurement-level Model**

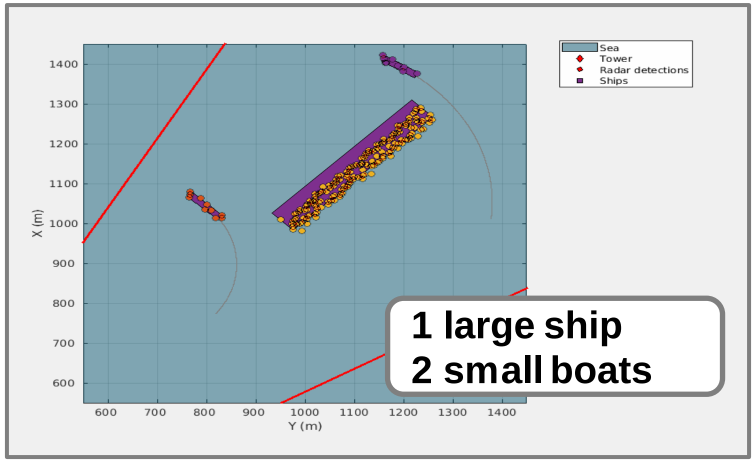
Wide range of data synthesis options for radar systems



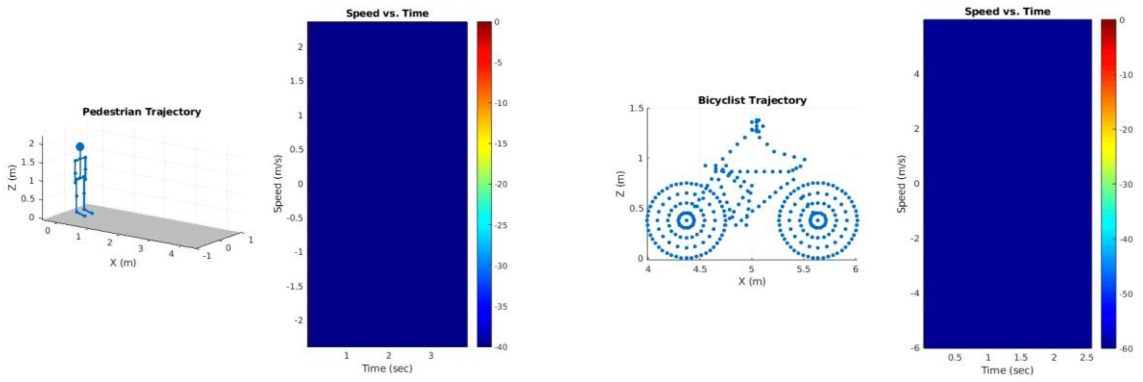
Long distance, multi-object operations



High clutter environments



Extended objects



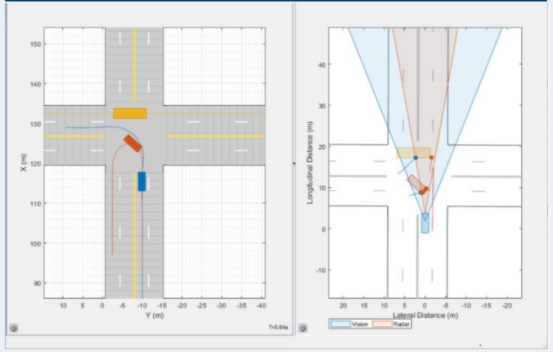
Micro-Doppler signatures

Simulating lidar sensor data in MATLAB and Simulink

Automated Driving Toolbox

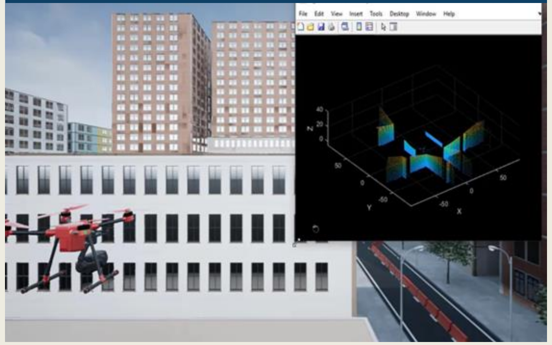
Unreal Engine®

Cuboid Environment

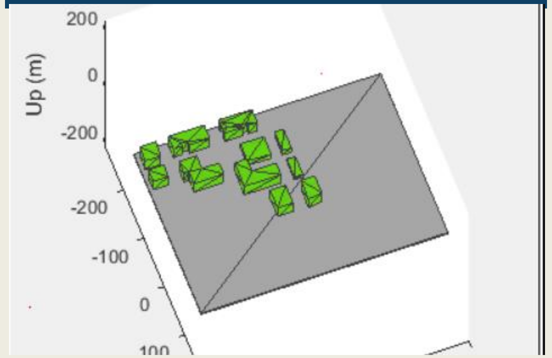


UAV Toolbox

Unreal Engine®

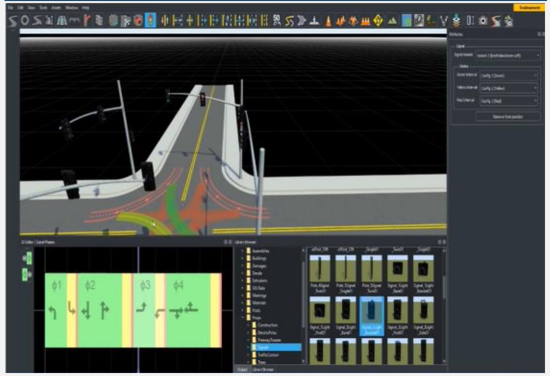


Cuboid Environment



3D Scene Creation

RoadRunner



What are the common challenges engineers face using AI with radar and lidar ?

1. **Labeling recorded data** for AI training is manual and time consuming

2. Little-**no recorded data** to train models for safety-critical applications

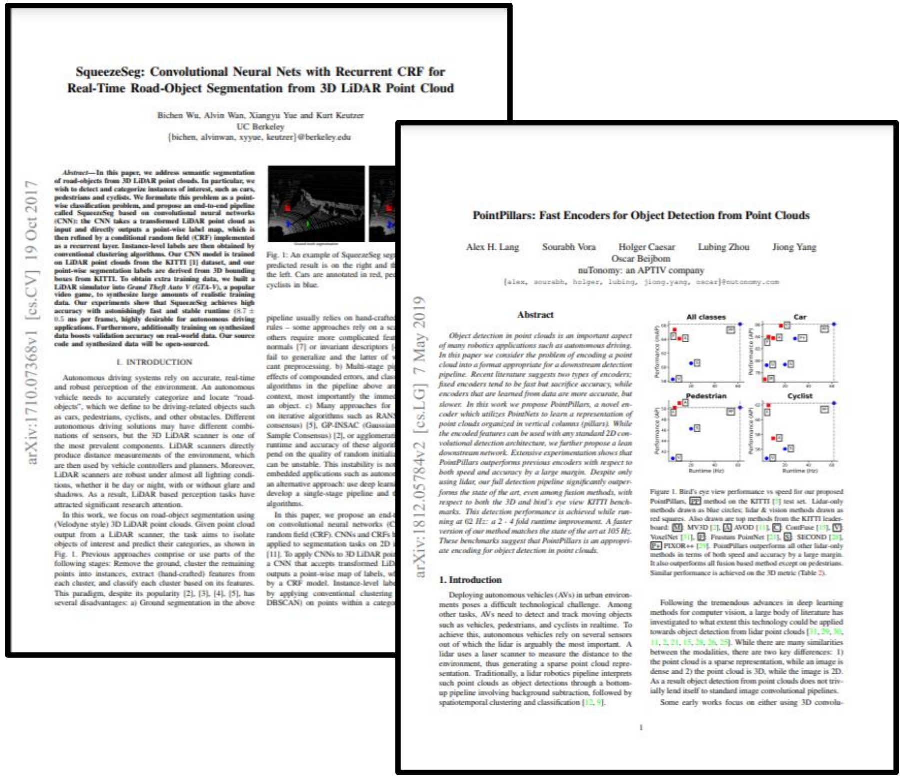
3. **Unfamiliarity with AI models** for radar and lidar

4. Unclear how to **pre-process sensor signals** for best results

5. Real-world systems require **deployment of more than AI model**

Challenge

Lack of knowledge on combination of model-type and data format best results


SqueezeSeg: Convolutional Neural Nets with Recurrent CRF for Real-Time Road-Object Segmentation from 3D LiDAR Point Cloud

Bichen Wu, Alvin Wan, Xiangyu Yue and Kurt Keutzer
UC Berkeley

arXiv:1710.07368v1 [cs.CV] 19 Oct 2017

I. INTRODUCTION

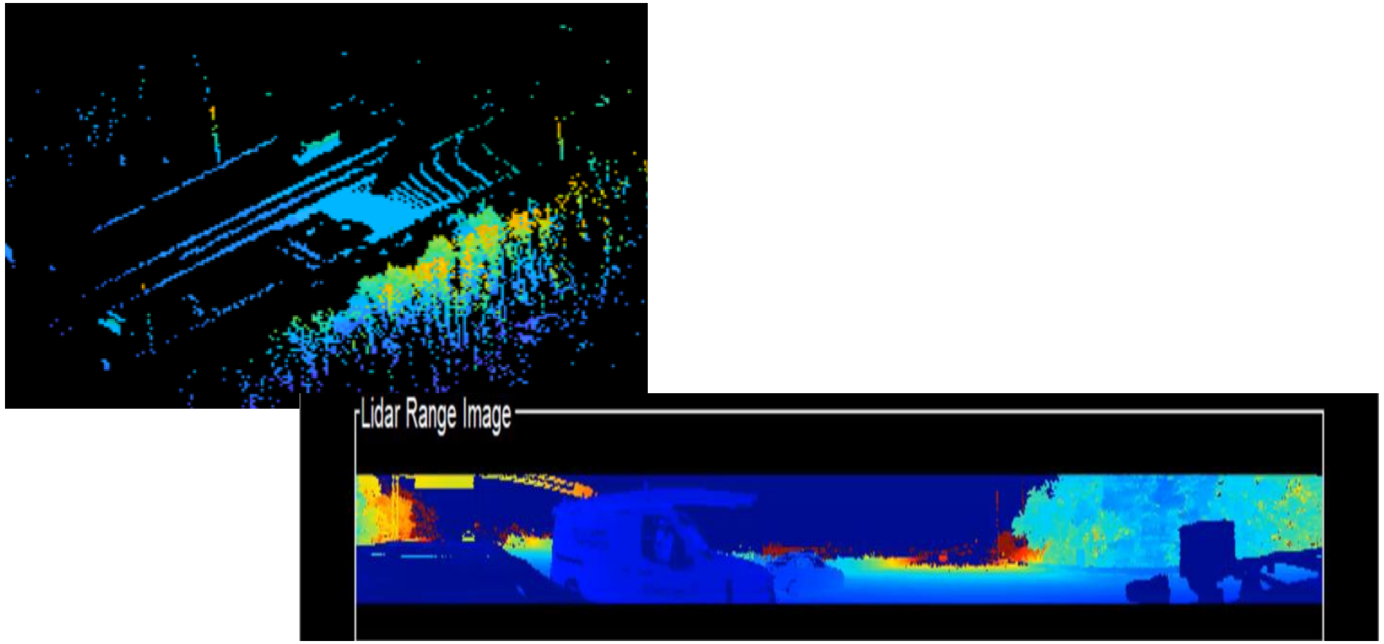
PointPillars: Fast Encoders for Object Detection from Point Clouds

Alex H. Lang Sourabh Vora Holger Caesar Lubing Zhou Jiong Yang Oscar Beijbom
nuTonomy: an APTIV company

arXiv:1812.05784v2 [cs.LG] 7 May 2019

Abstract

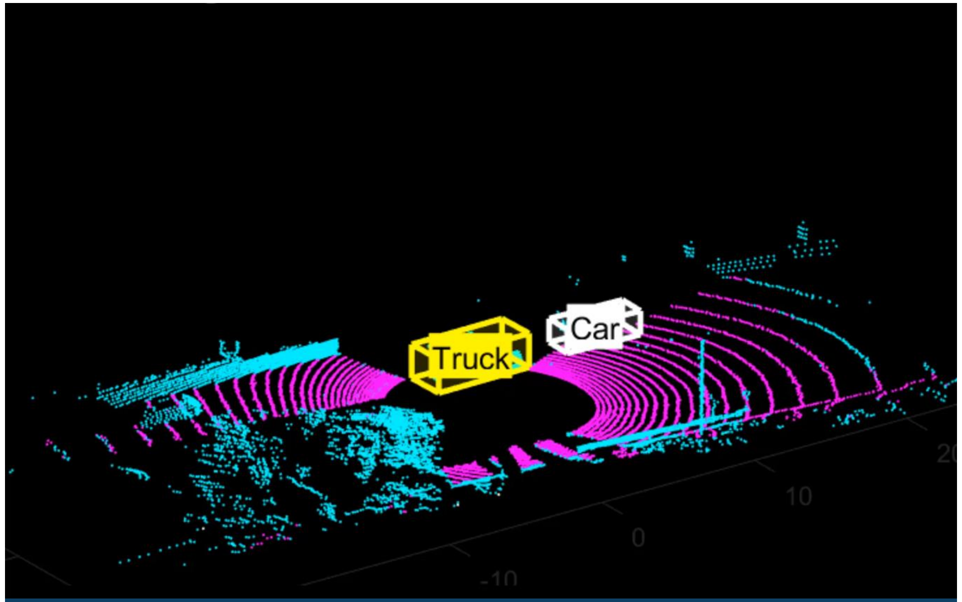
1. Introduction



What model do I use ?
There are so many research papers.

How do I train a model ?
Raw sensor data or transformed.

MATLAB provides a curated library of models with different inputs and styles

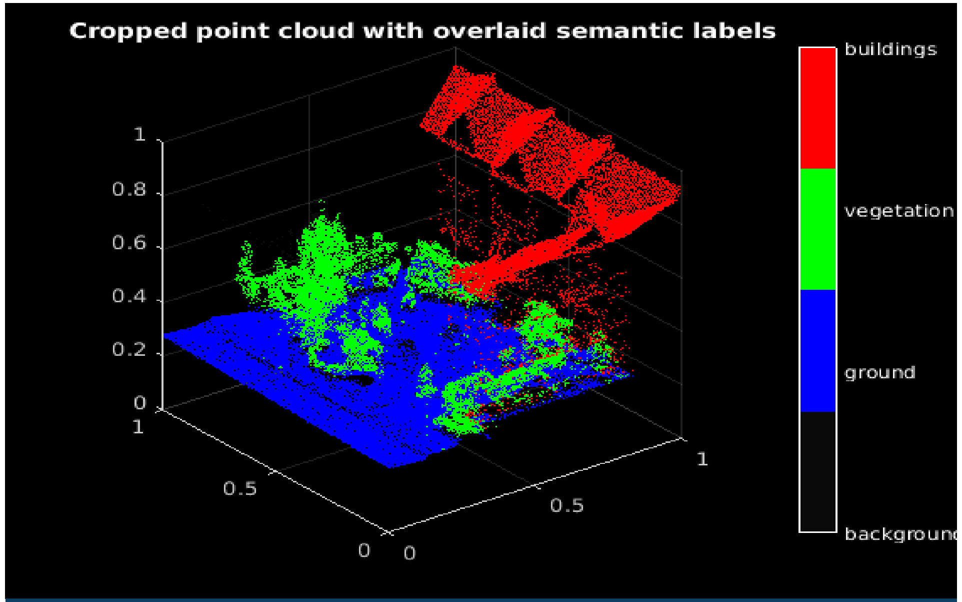


Object Detection
3D bounding box detection and classification

Curated Models

Raw Data

1. PointPillars



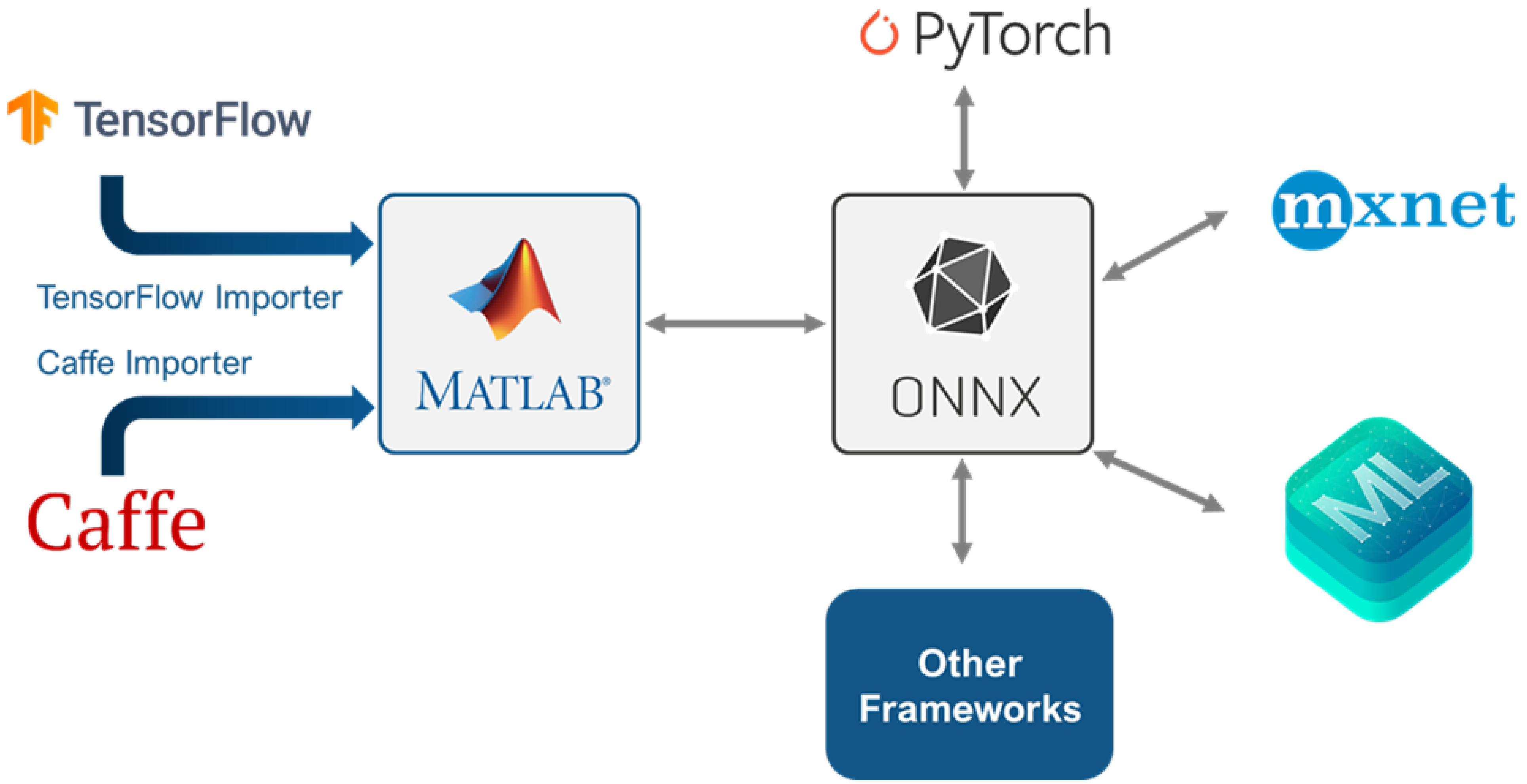
Semantic Segmentation
Classify each data point with label

Curated Models

Image Data

1. SqueezeSeg v2
2. PointSeg
3. SalsaNext
4. PointNet
5. PointNet++

Import models from open source AI frameworks



Lidar 3-D Object Detection Using PointPillars Deep Learning

Load Data

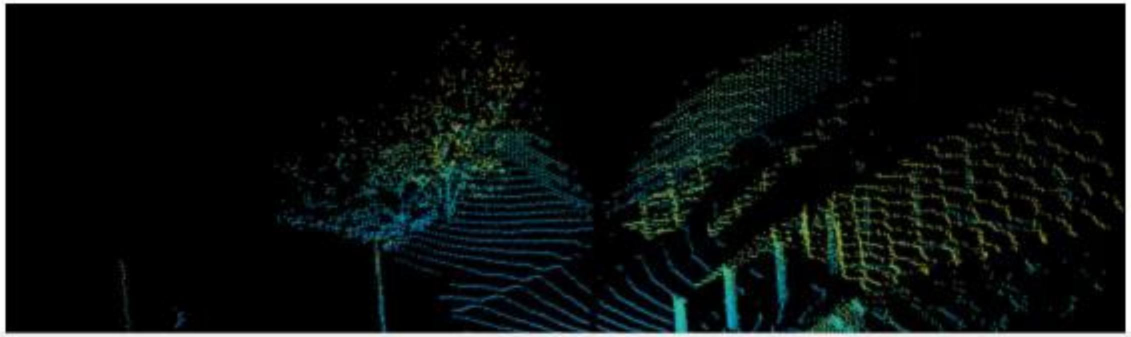
```
lidarURL = 'https://www.mathworks.com/supportfiles/lidar/data/WPI_LidarData.tar.gz';
lidarData = downloadWPIData(outputFolder, lidarURL);
```

Load the 3-D bounding box labels.

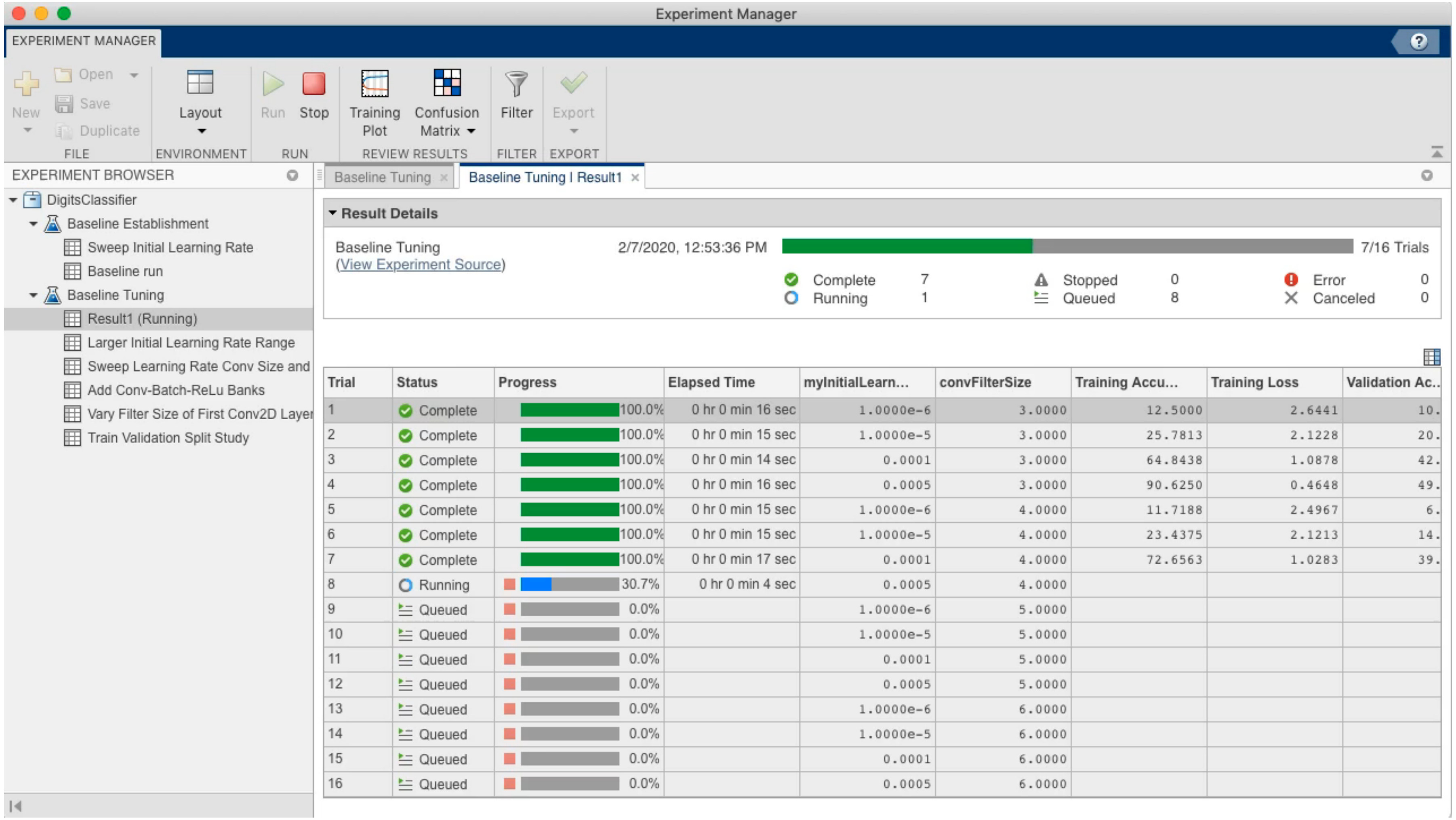
```
load('WPI_LidarGroundTruth.mat','bboxGroundTruth');
Labels = timetable2table(bboxGroundTruth);
Labels = Labels(:,2:end);
```

Display the full-view point cloud.

```
figure
ax = pcshow(lidarData{1,1}.Location);
set(ax,'XLim',[-50 50],'YLim',[-40 40]);
zoom(ax,2.5);
axis off;
```



Tune hyperparameters and reproduce training experiments

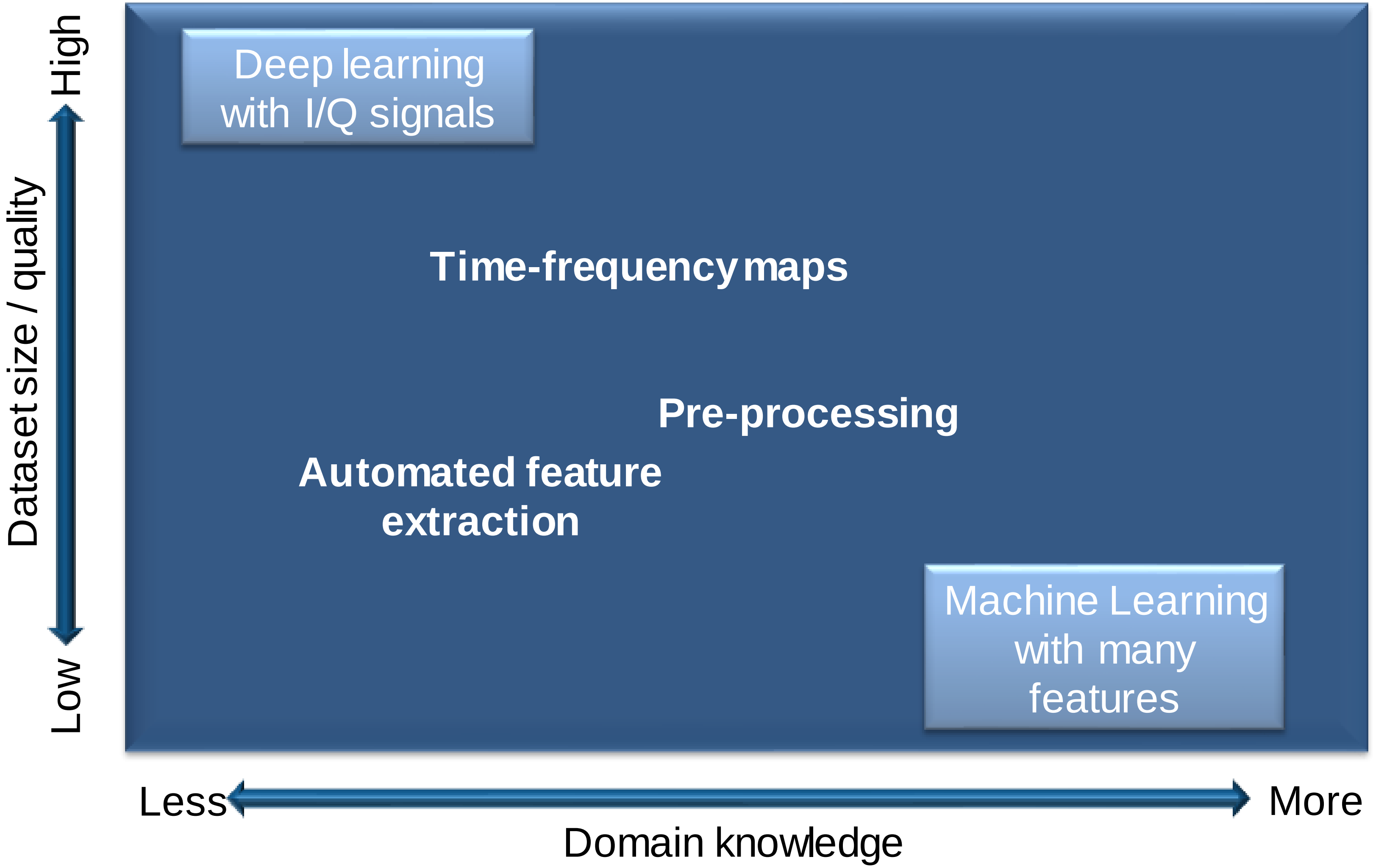


Trial	Status	Progress	Elapsed Time	myInitialLearn...	convFilterSize	Training Accu...	Training Loss	Validation Ac...
1	Complete	100.0%	0 hr 0 min 16 sec	1.0000e-6	3.0000	12.5000	2.6441	10.
2	Complete	100.0%	0 hr 0 min 15 sec	1.0000e-5	3.0000	25.7813	2.1228	20.
3	Complete	100.0%	0 hr 0 min 14 sec	0.0001	3.0000	64.8438	1.0878	42.
4	Complete	100.0%	0 hr 0 min 16 sec	0.0005	3.0000	90.6250	0.4648	49.
5	Complete	100.0%	0 hr 0 min 15 sec	1.0000e-6	4.0000	11.7188	2.4967	6.
6	Complete	100.0%	0 hr 0 min 15 sec	1.0000e-5	4.0000	23.4375	2.1213	14.
7	Complete	100.0%	0 hr 0 min 17 sec	0.0001	4.0000	72.6563	1.0283	39.
8	Running	30.7%	0 hr 0 min 4 sec	0.0005	4.0000			
9	Queued	0.0%		1.0000e-6	5.0000			
10	Queued	0.0%		1.0000e-5	5.0000			
11	Queued	0.0%		0.0001	5.0000			
12	Queued	0.0%		0.0005	5.0000			
13	Queued	0.0%		1.0000e-6	6.0000			
14	Queued	0.0%		1.0000e-5	6.0000			
15	Queued	0.0%		0.0001	6.0000			
16	Queued	0.0%		0.0005	6.0000			

What are the common challenges engineers face using AI with radar and lidar ?

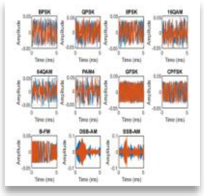
1. **Labeling recorded data** for AI training is manual and time consuming
2. Little-**no recorded data** to train models for safety-critical applications
3. **Unfamiliarity with AI models** for radar and lidar
4. Unclear how to **pre-process sensor signals** for best results
5. Real-world systems require **deployment of more than AI model**

Pre-processing radar data can improve performance of network



Dataset size vs. domain knowledge vs. compute resources

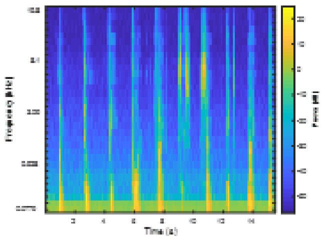
You can make the trade-off between pre-processing approaches



I/Q Signals

Requires less work on front-end

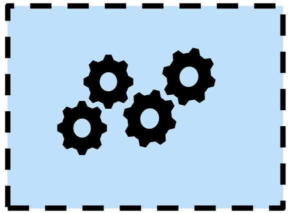
May require more network tuning



Pre-Processing
Time-Frequency Transformation

Features extracted automatically

Less complex training network

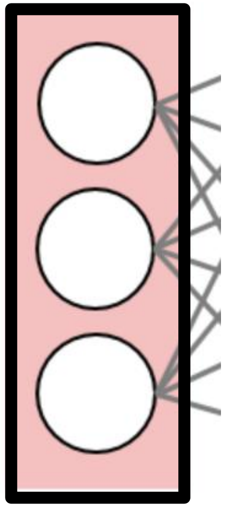


Feature Engineering

Leverage domain knowledge

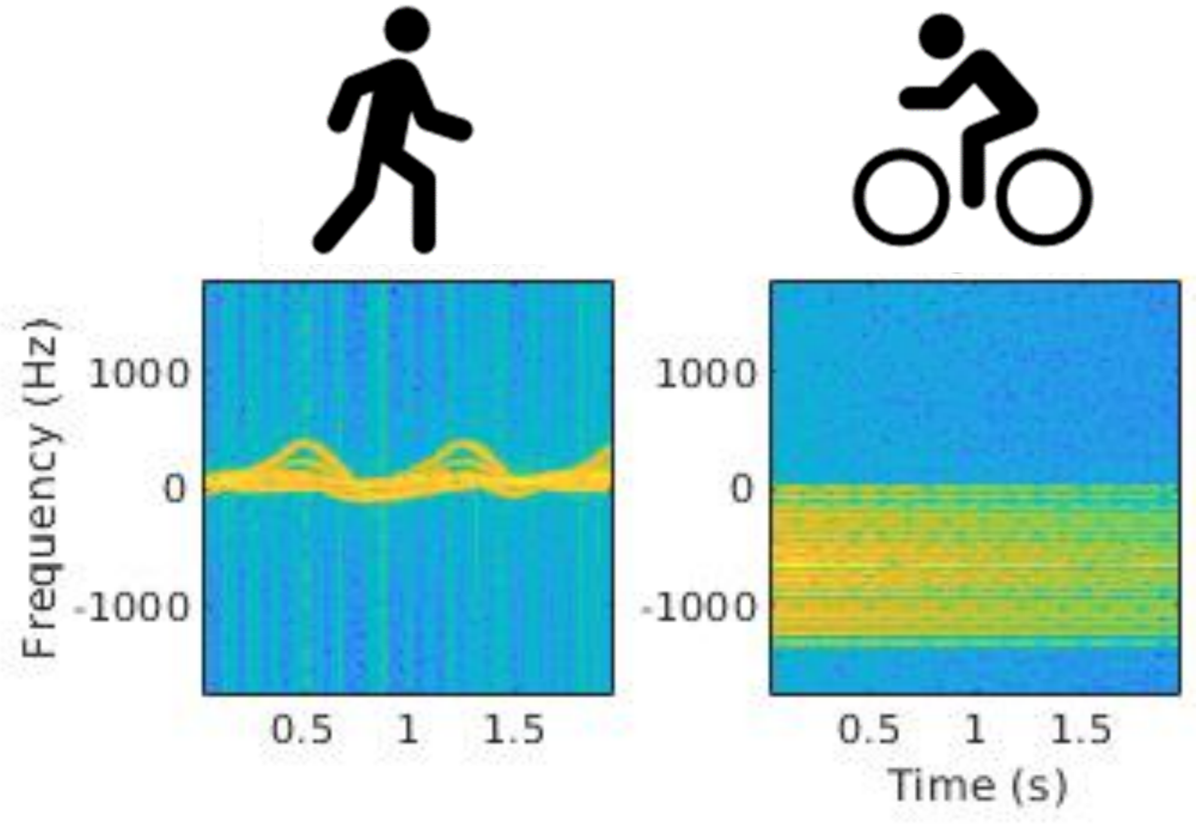
Less data at input layer

Less complex training network



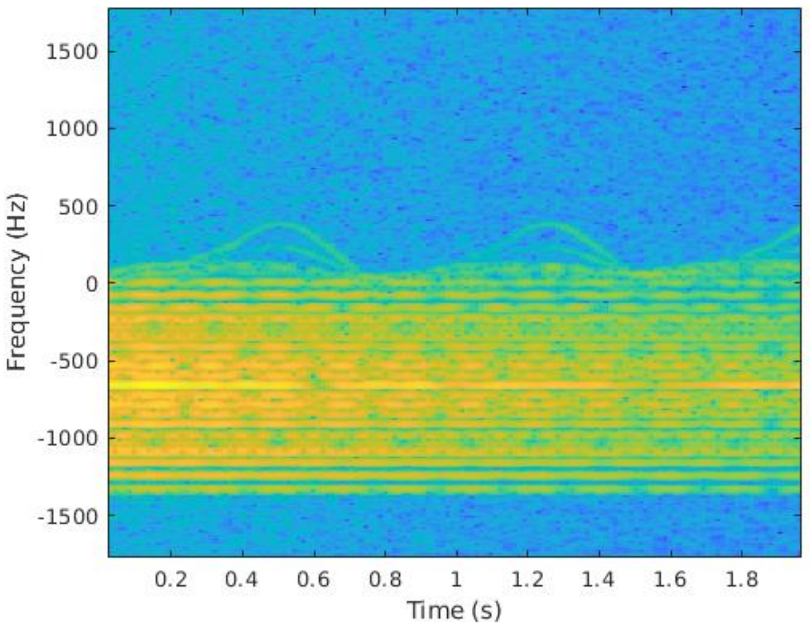
Input Layer

Time to test your ability to classify micro-Doppler returns …



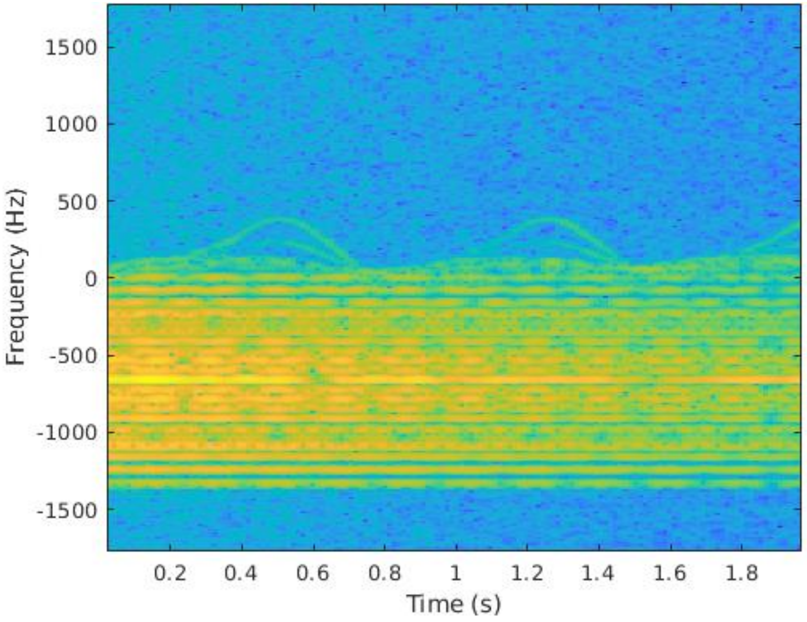
Ground truth – synthesized micro-Doppler

Is this a pedestrian or a bicyclist?



Poll

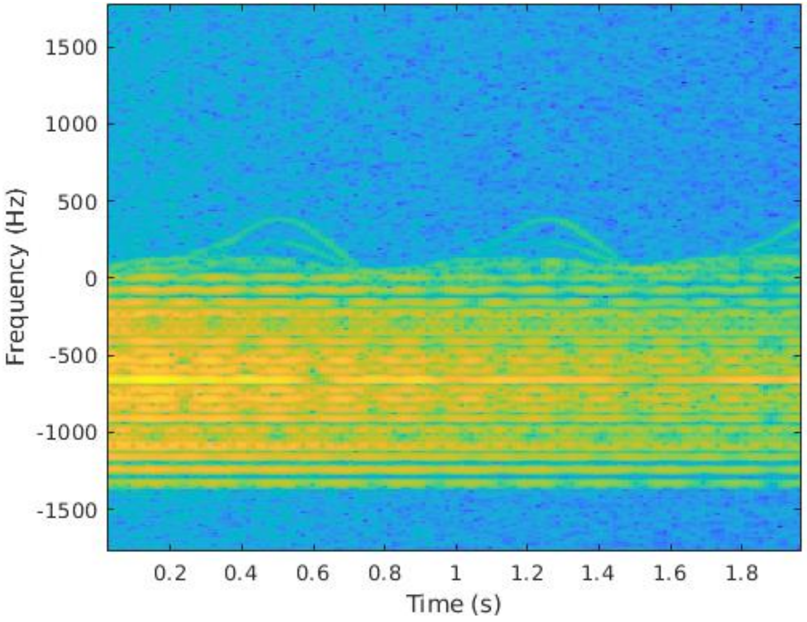
Is this a pedestrian or a bicyclist?



A. One Pedestrian
B. One Bicyclist
C. One of each
D. Not sure

And the answer is ….

Is this a pedestrian or a bicyclist?

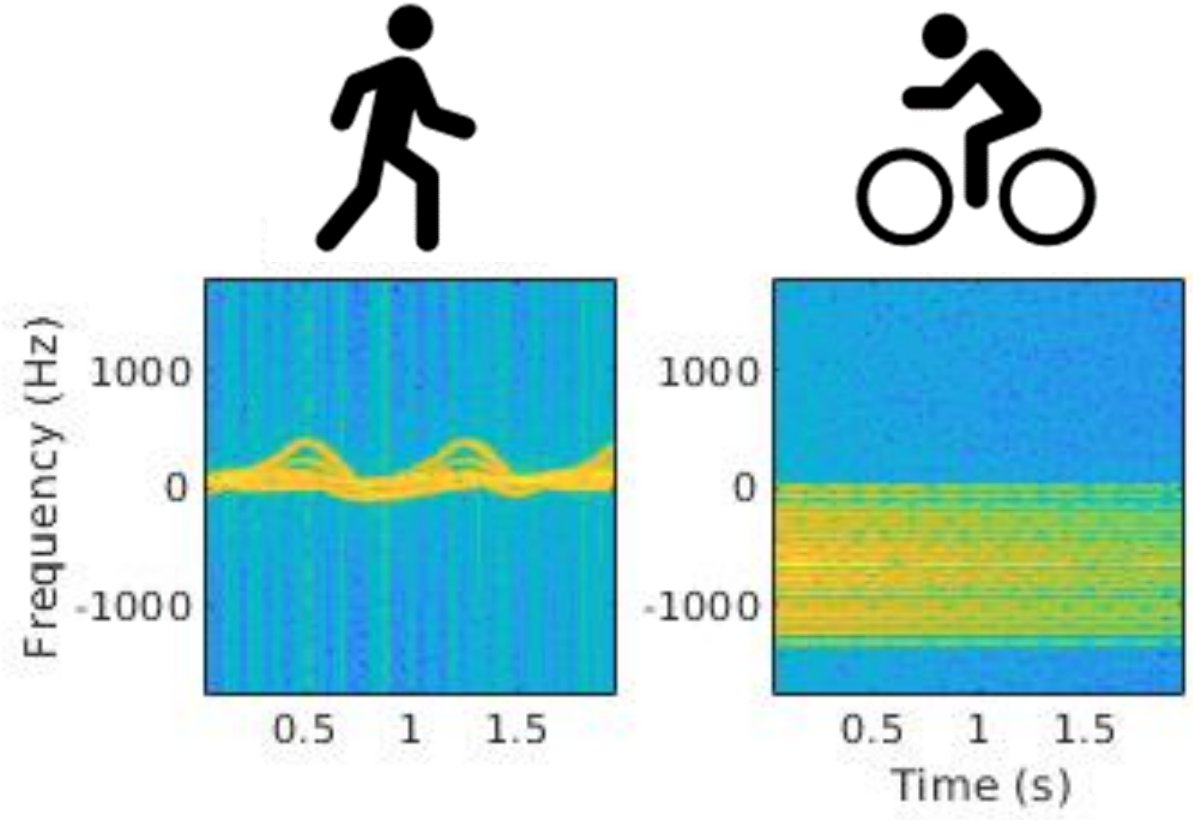


A. Pedestrian
B. Bicyclist
C. One of each
D. Not sure

This is a pedestrian and a bicyclist

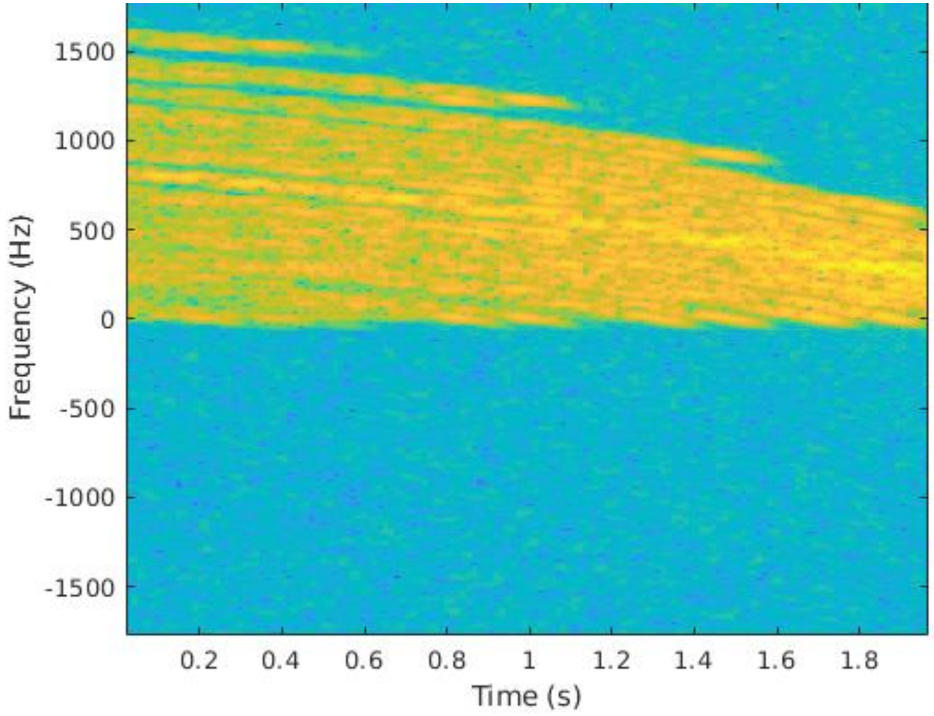
This one is a bit trickier. The network gets the correct answer

Example Link



Ground truth – synthesized micro-Doppler

Is this a pedestrian or a bicyclist?



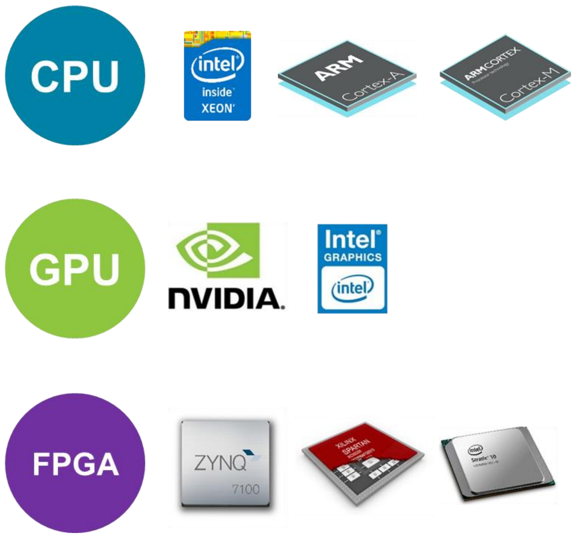
This is two bicyclists

What are the common challenges engineers face using AI with radar and lidar ?

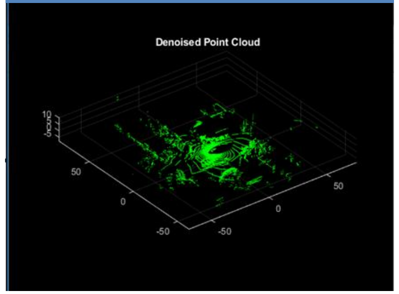
1. **Labeling recorded data** for AI training is manual and time consuming
2. Little-**no recorded data** to train models for safety-critical applications
3. **Unfamiliarity with AI models** for radar and lidar
4. Unclear how to **pre-process sensor signals** for best results
5. Real-world systems require **deployment of more than AI model**

Challenge

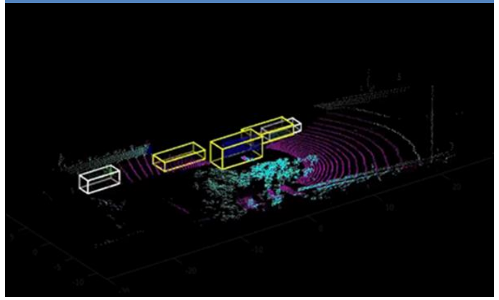
Deploying AI model and application code prototype to a larger system



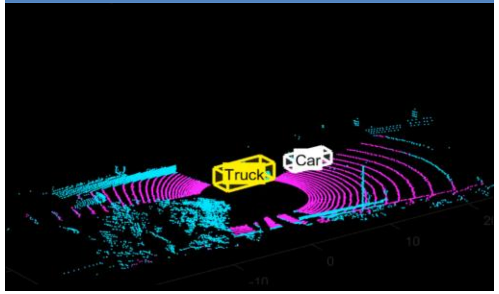
Pre-processing
Denoise, Ground Removal



AI Model
Object Detection

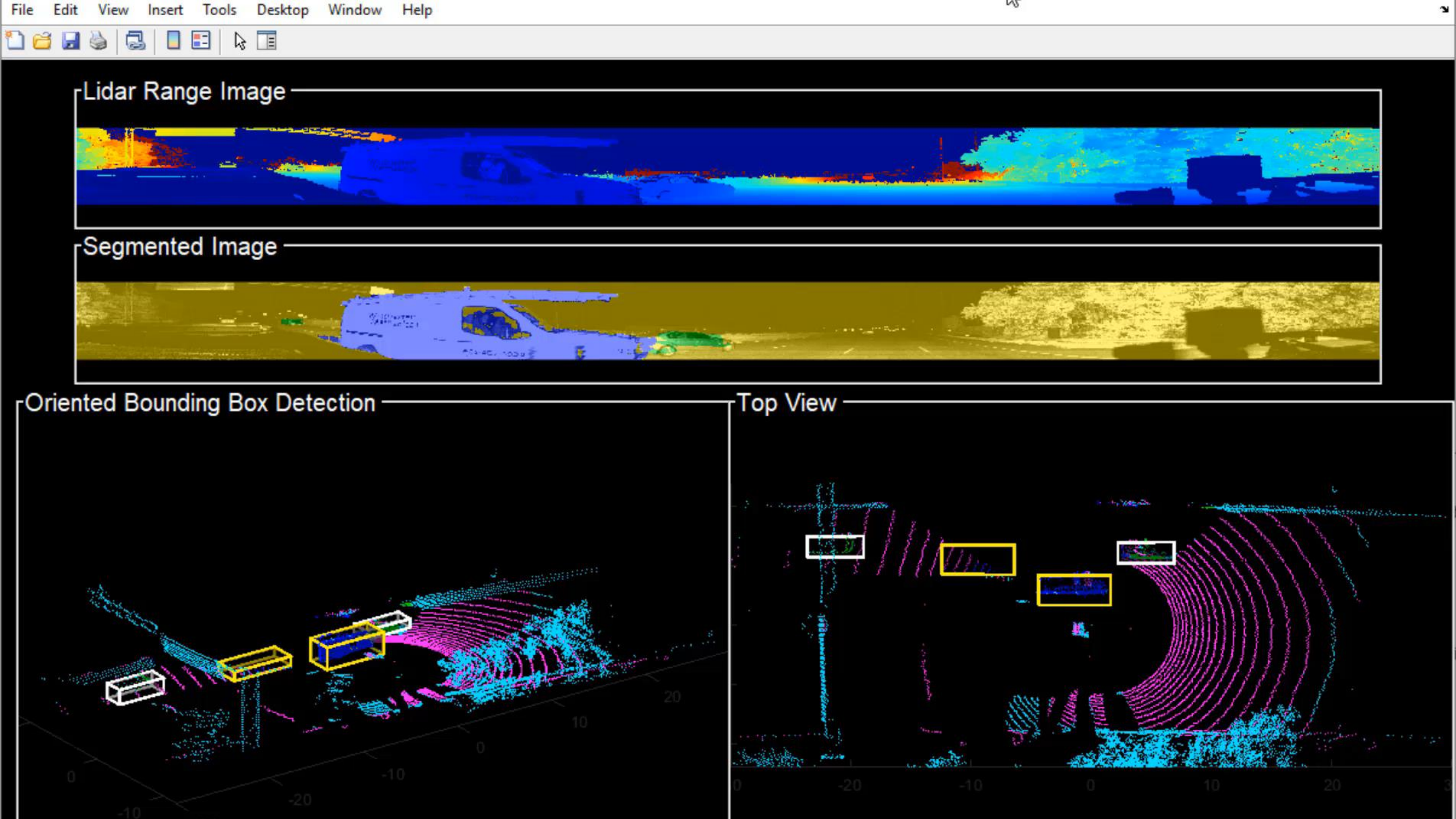


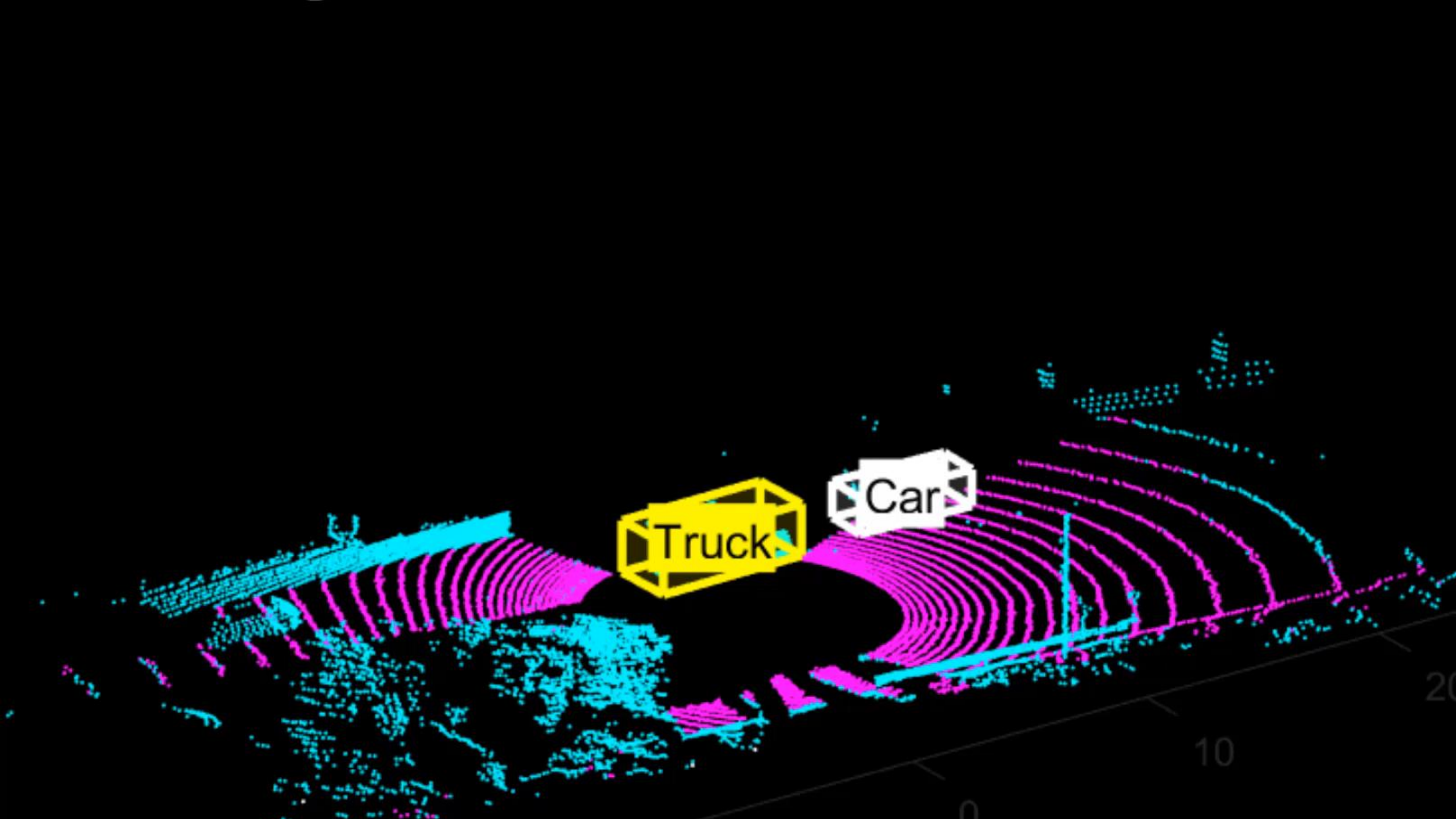
Post-processing
JPDA object tracker



Multiple options for deployment platform
CPU/GPU/FPGA

System requires AI model + pre and post processing


File Edit View Insert Tools Desktop Window Help
Lidar Range Image
Segmented Image
Oriented Bounding Box Detection
Top View
-20
-10
0
10
20


Truck
Car
0
10
20

MATLAB R2020b

HOME PLOTS APPS LIVE EDITOR INSERT VIEW

Search Documentation

Minhaj

Design App | Get More Apps | Install App | Package App

FILE

Curve Fitting | PID Tuner | Signal Analyzer | Image Acquisition | MATLAB Coder | Distribution Fitter | Control System Desi... | Control System Tuner | Flight Log Analyzer | Linear System Analyzer | Model Reducer

APPS

C: ▸ Users ▸ mpalakka ▸ OneDrive - MathWorks ▸ Documents ▸ MATLAB ▸ Examples ▸ R2020b ▸ shared_driving_fusion_lidar ▸ TrackVehiclesUsingLidarExample ▸

Live Editor - C:\Users\mpalakka\OneDrive - MathWorks\Documents\Demos\DetectClassifyAndTrackOrientedBoundingBoxInLidarExample\DetectClassifyAndTrackOrientedBoundingBoxInLidarExample.mlx

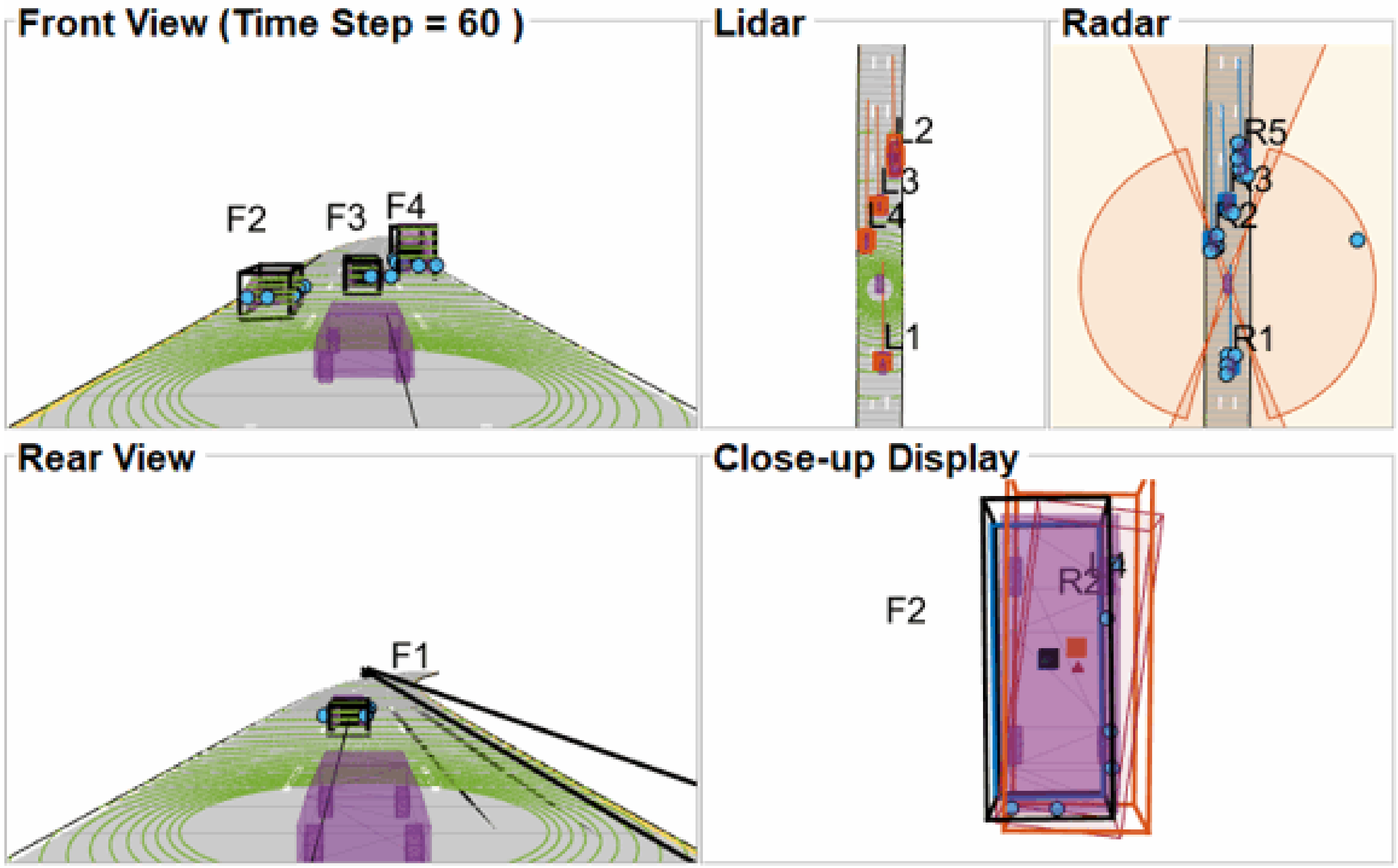
DetectClassifyAndTrackOrientedBoundingBoxInLidarExample.mlx | TrackVehiclesUsingLidarExample.m | +

```
filterInitFcn = @helperMultiClassInitIMMFilter;

% A joint probabilistic data association tracker with IMM filter
tracker = trackerJPDA('FilterInitializationFcn',filterInitFcn,...
    'TrackLogic','History',...
    'AssignmentThreshold',assignmentGate,...
    'ClutterDensity',Kc,...
    'ConfirmationThreshold',confThreshold,...
    'DeletionThreshold',delThreshold,'InitializationThreshold',0);

allTracks = struct([]);
time = 0;
dt = 0.1;

% Define Measurement Noise
measNoise = blkdiag(0.25*eye(3),25,eye(3));

numTracks = zeros(numFrames, 2);
```

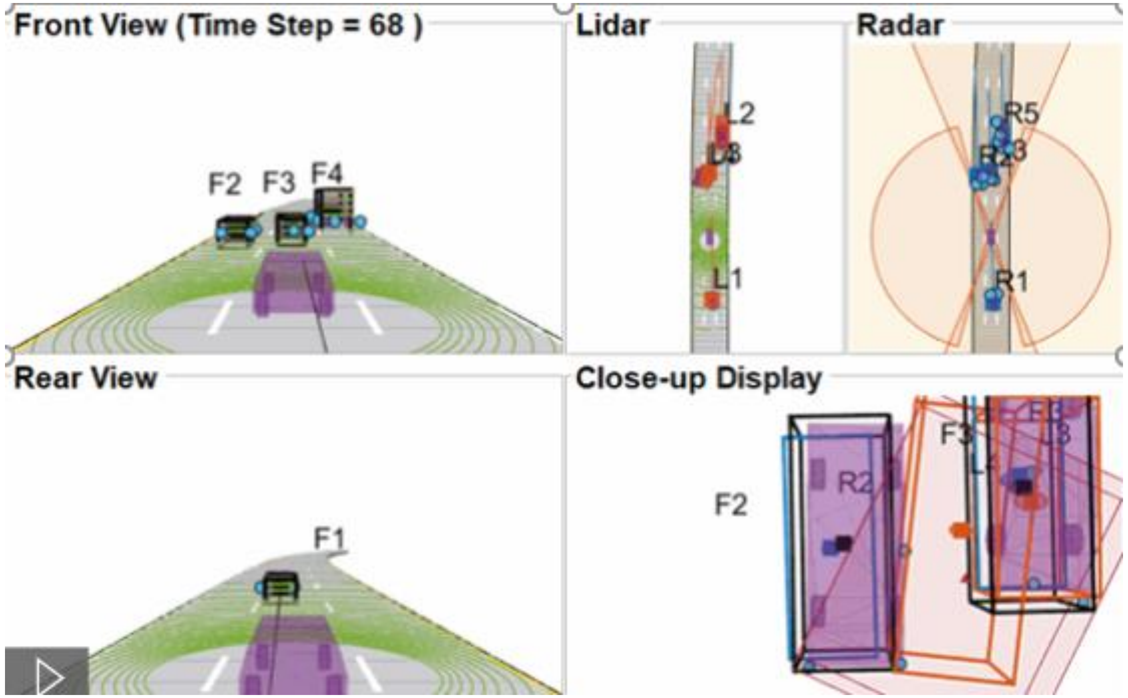
The detected objects are assembled as a cell array of objectDetection objects using the `helperAssembleDetections` function.

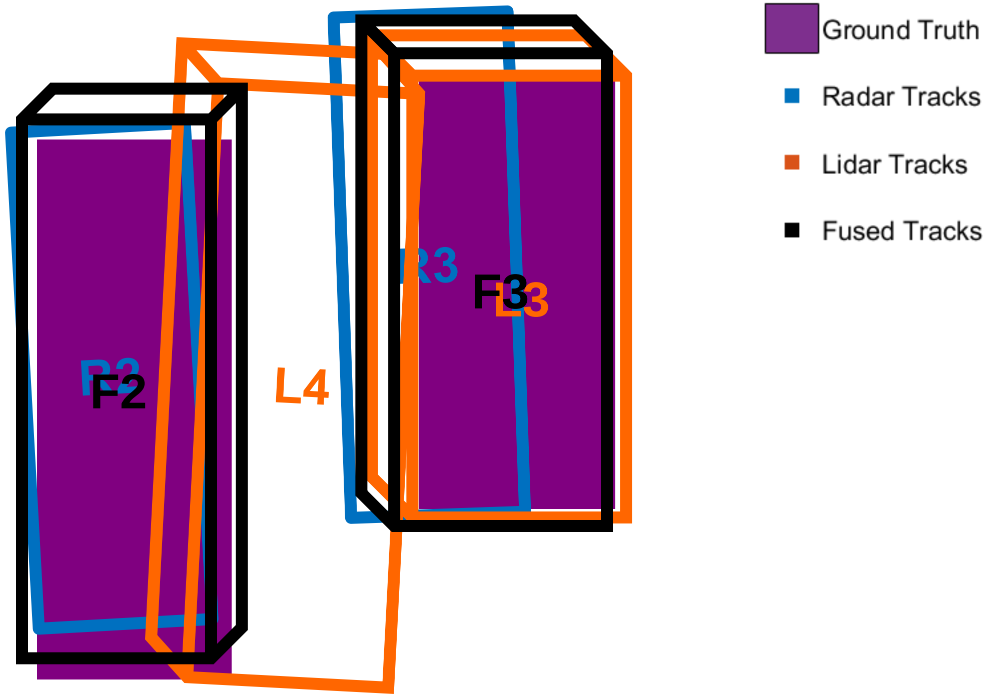
```
display = helperLidarObjectDetectionDisplay;
initializeDisplay(display);

for count = 1:numFrames
    time = time + dt;
    % Get current data
```

We can improve our results when we fuse the two sensors



Let's take a closer look …

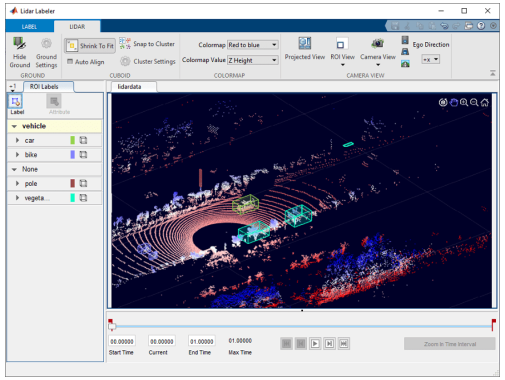




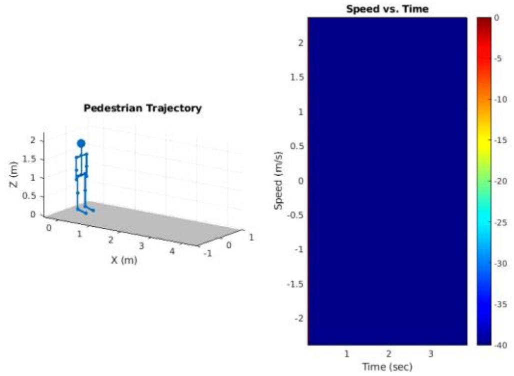
Fused tracks more accurate than individual sensor tracks

[Sensor Fusion and Tracking Toolbox ™](): designing, simulating and testing systems that fuse data from multiple sensors

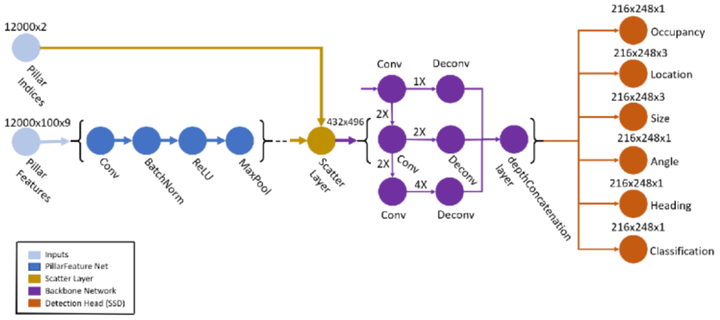
How MATLAB and Simulink help create AI-driven radar and lidar processing systems



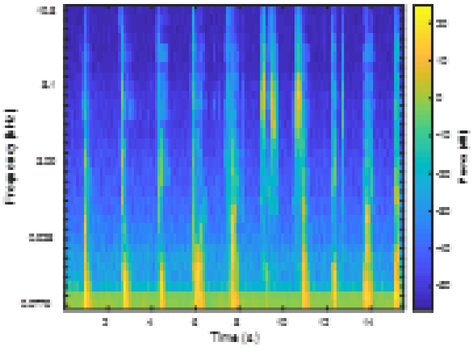
Labeling Automation



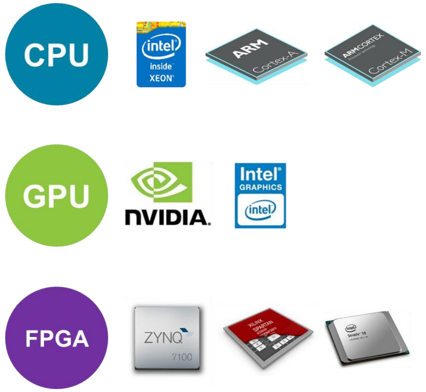
Data Synthesis



AI Workflow
Pre-trained models, training, evaluation, validation



Pre-processing



Full Application Deployment

MATLAB EXPO
2021

감사합니다

MathWorks®