

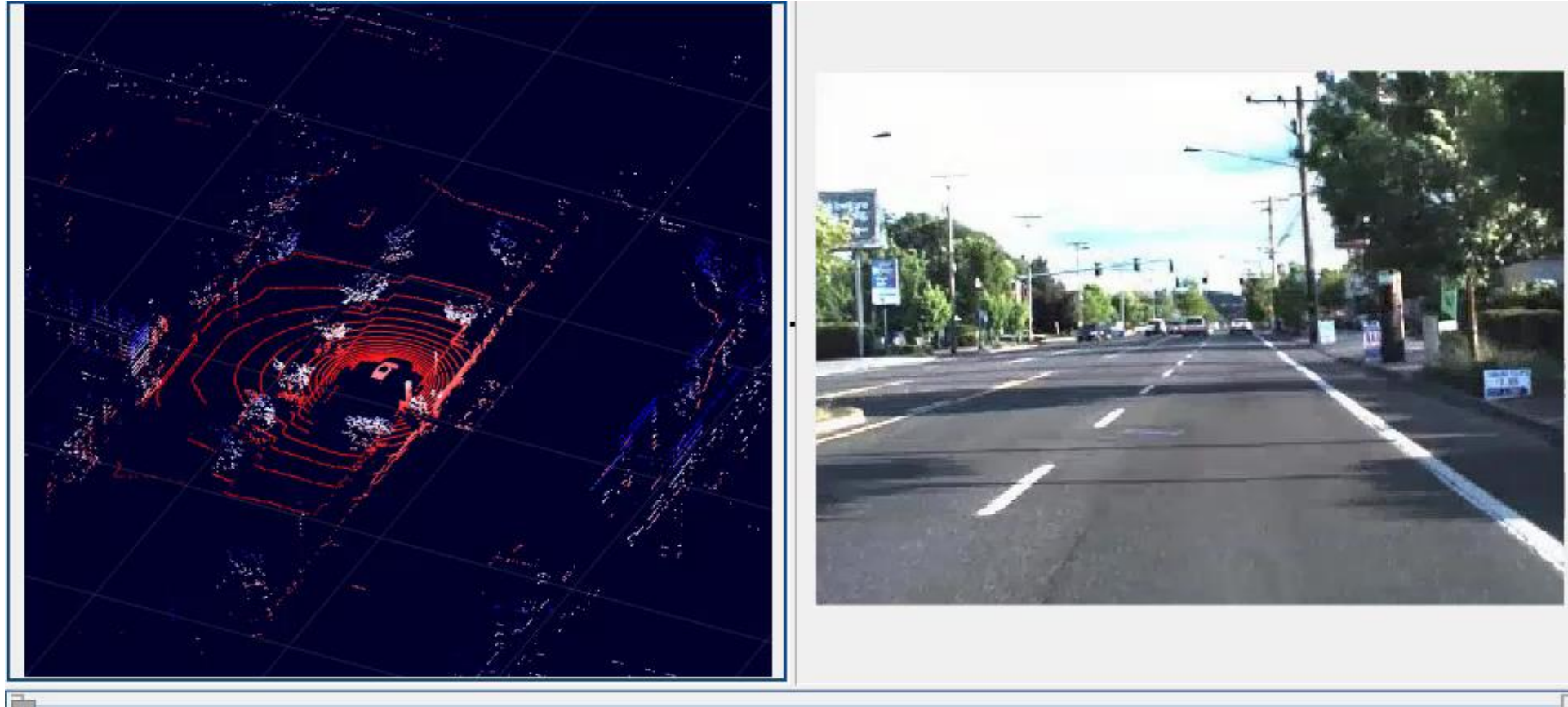
MATLAB EXPO 2021

Lidar点群処理と自律移動システムへの統合

MathWorks Japan
アプリケーションエンジニア
草野 駿一



自律システムにおけるLidarセンサ



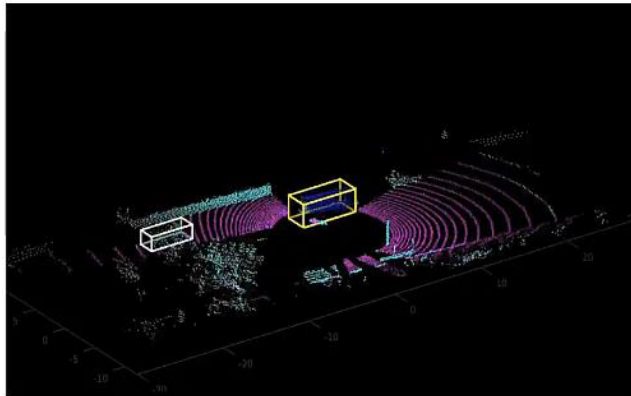
Accurate
Depth



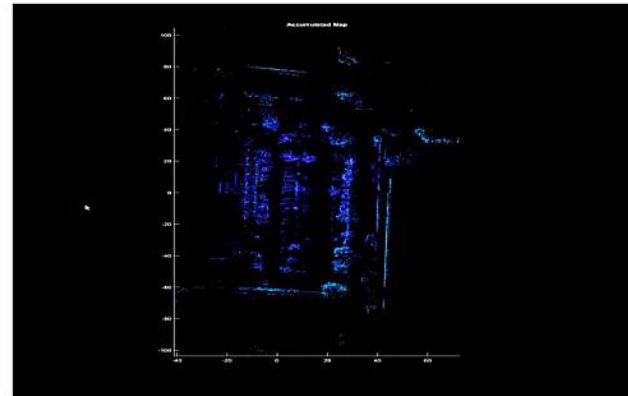
Lidar Toolbox

点群処理システムの設計、解析、テスト

Lidar Toolbox



Object Detection and Tracking



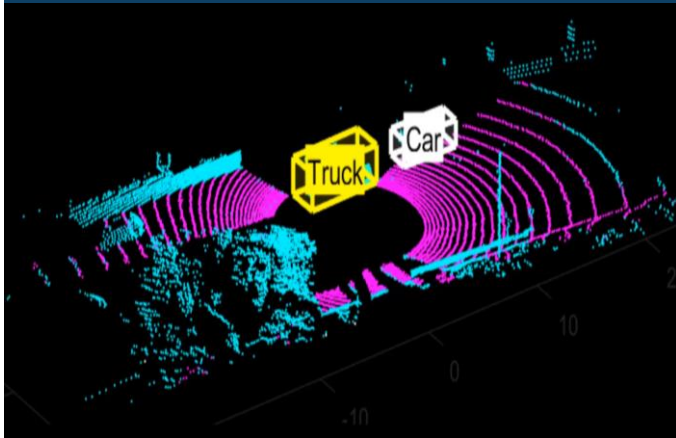
3D Map Building

R2020b

Lidar Toolbox

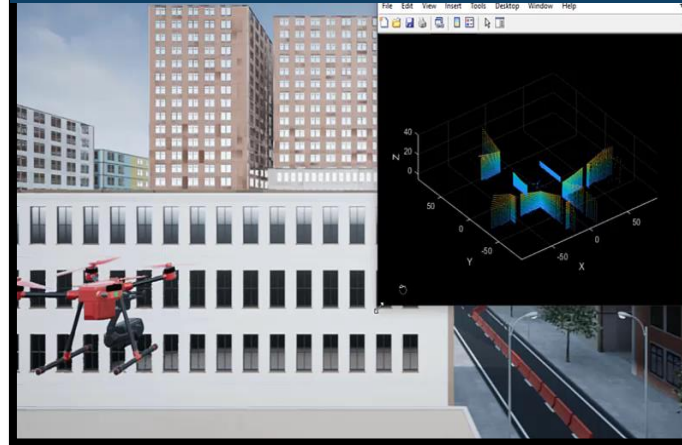
セッションの内容

オブジェクト検出と トラッキング (例題:自動運転)



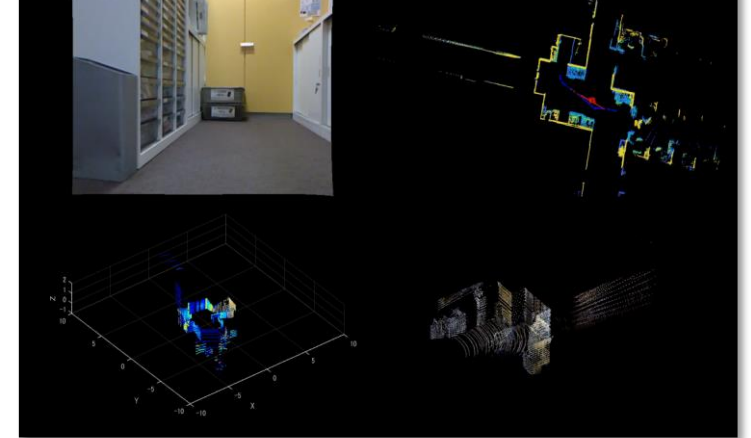
- ディープラーニングによる検出
- JPDA トラッカー
- コード生成

Lidar SLAM (例題:UAV)



- Lidar点群のシミュレーション
- 特徴量によるマッチング
- Lidar SLAM

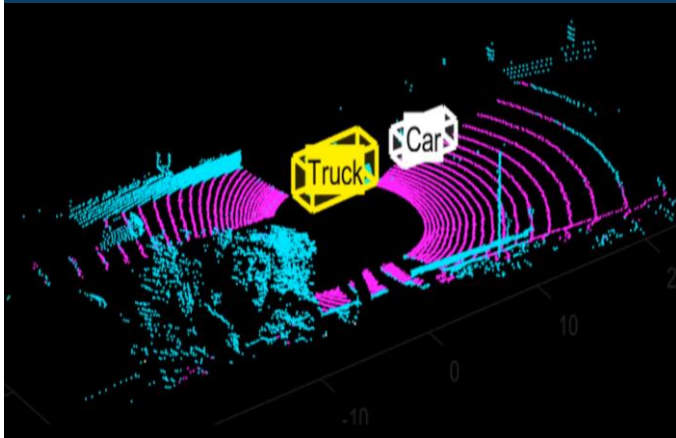
移動ロボットによる 色付き3次元マップ生成 (例題:移動ロボット)



- Lidar – カメラキャリブレーション (LCC)
- 室内のマッピング

セッションの内容

オブジェクト検出と トラッキング (例題:自動運転)



- ディープラーニングによる検出
- JPDA トラッカー
- コード生成

Lidar SLAM (例題:UAV)



- Lidar点群のシミュレーション
- 特徴量によるマッチング
- Lidar SLAM

移動ロボットによる 色付き3次元マップ生成 (例題:移動ロボット)



- Lidar – カメラキャリブレーション (LCC)
- 室内のマッピング

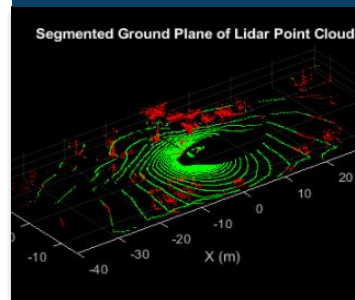
ワークフロー

自動運転に向けたオブジェクト検出とトラッキング

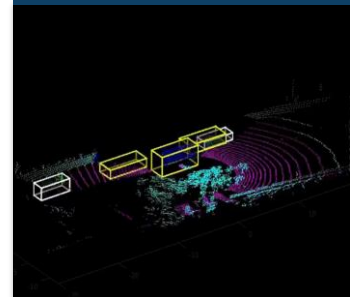
データの読み込み



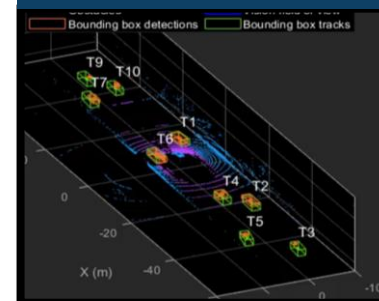
Lidar データの
前処理



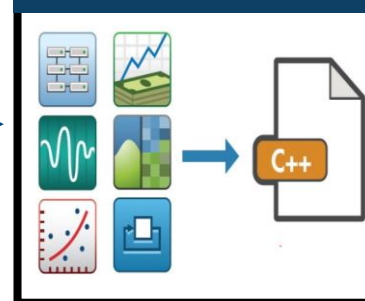
オブジェクト
検出



オブジェクト
トラッキング



コード生成



ファイルやセンサからのLidarデータ読み込み

Lidarデータの読み込み、書き込み

pcread

Read PCD and PLY files

pcwrite

Write PCD and PLY files

rosbag

ROS bag files

velodyneFileReader

Velodyne PCAP files

ibeoFileReader

Ibeo data container (IDC) file reader

lasReader

LAS or LAZ reader

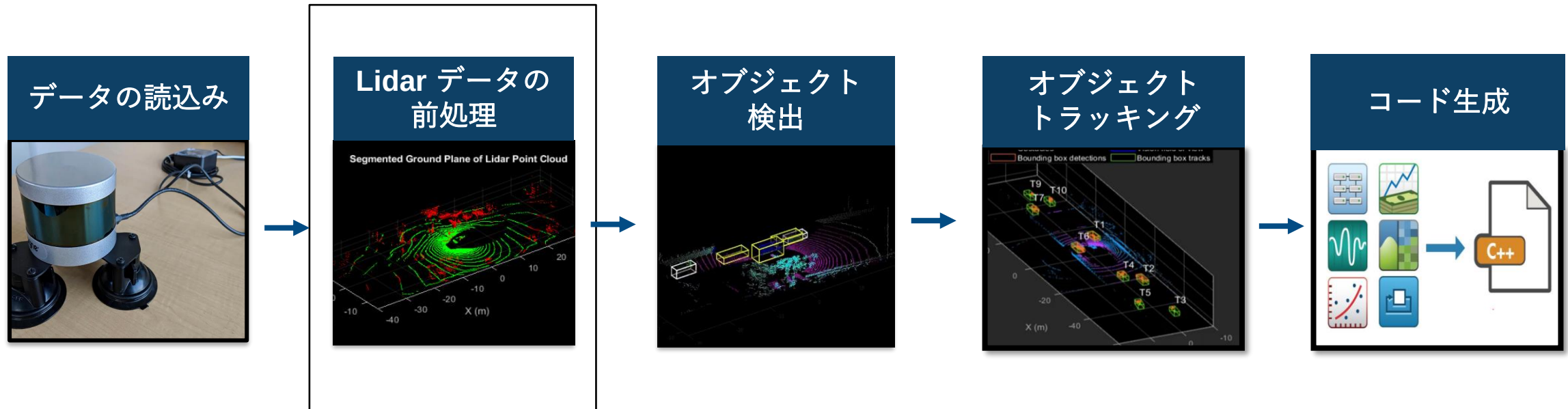
センサからの ライブストリーミング



velodynelidar

*Live stream from Velodyne
LiDAR®*

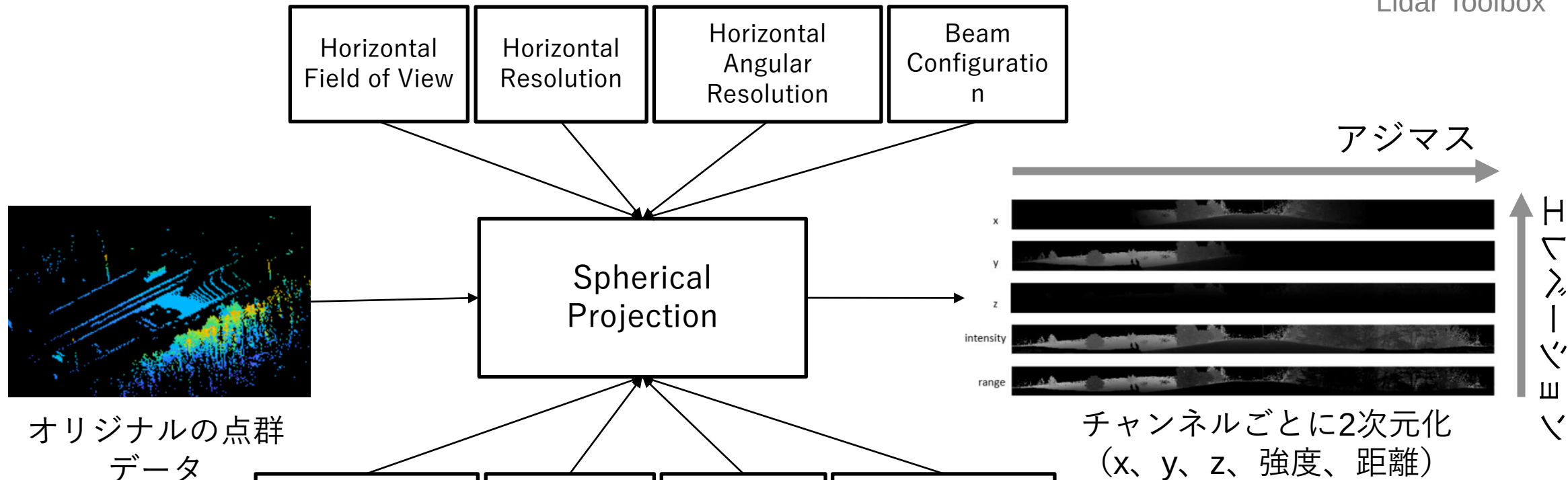
自動運転に向けたオブジェクト検出とトラッキング



点群データ処理効率化のための画像化 フォーマット変換

R2021a

Lidar Toolbox



チャンネルごとに2次元化
(x、y、z、強度、距離)

得られる効果

- 計算高速化
- CNNの利用

その他の様々な前処理関数

[pcfitplane](#) / [segmentGroundSMRF](#)
[segmentGroundFromLidarData](#)

地面の点群除去

[pcdownsample](#)

3-D 点群のダウンサンプル

[pcmedian](#)

3-D 点群データの中央値フィルタリング

[pcdenoise](#)

3-D 点群からノイズを除去する

[pcalign](#)

配列点群の位置合わせ

[pccat](#)

3-D 点群配列を連結する

[pcnormals](#)

点群の法線を推定する

[pctransform](#)

3-D 点群を変換する

[extractEigenFeatures](#)

点群セグメントから固有値ベースの特徴量を抽出する

[extractFPFHFeatures](#)

高速ポイント フィーチャ ヒストグラム(FPFH)記述子を点群から抽出する

[detectRectangularPlanePoints](#)

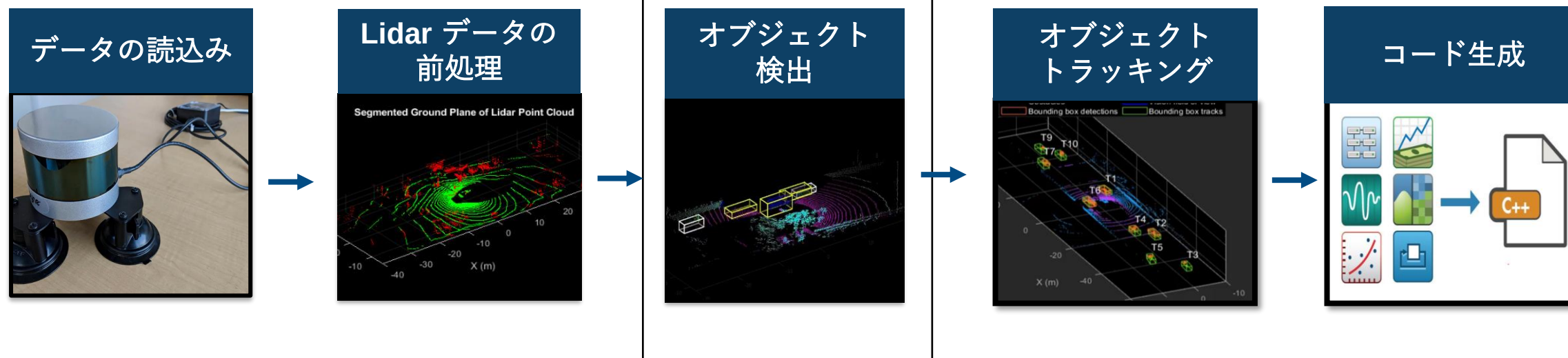
点群で指定された寸法の矩形平面を検出

<https://jp.mathworks.com/help/lidar/pointcloudprocessing.html>

Lidar Toolbox

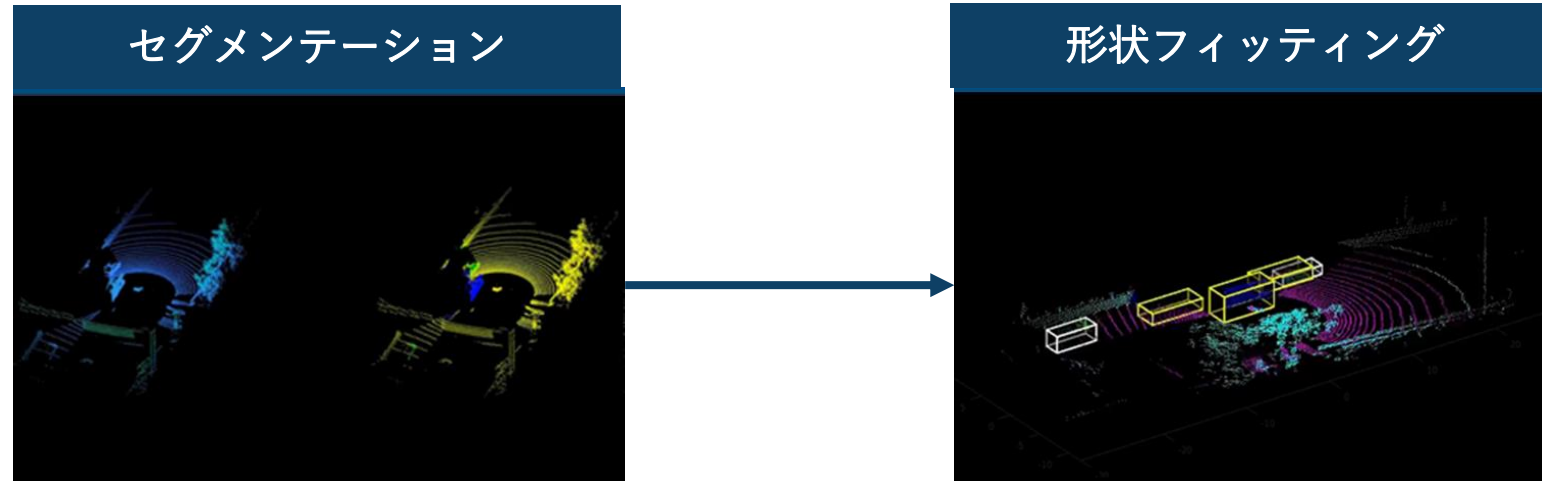
ワークフロー

自動運転に向けたオブジェクト検出とトラッキング

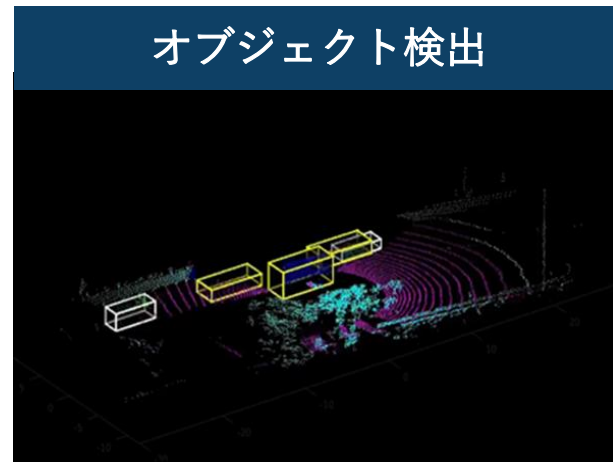


トラッキングのためのオブジェクト検出技術

1. セグメンテーション + 形状フィッティング

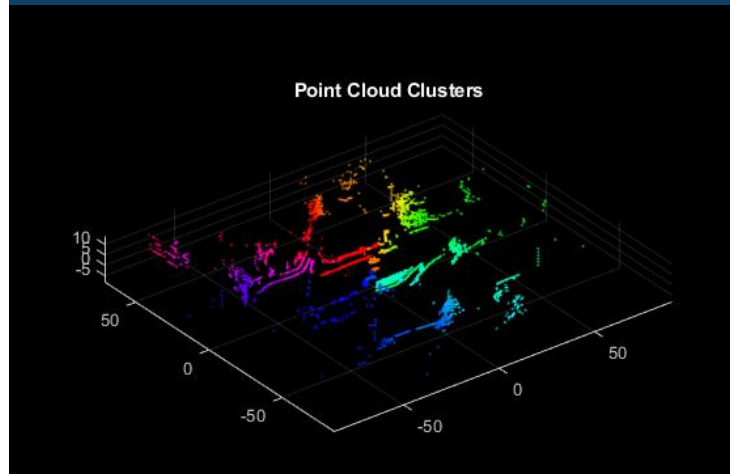


2. ディープラーニングによる 3次元オブジェクト検出



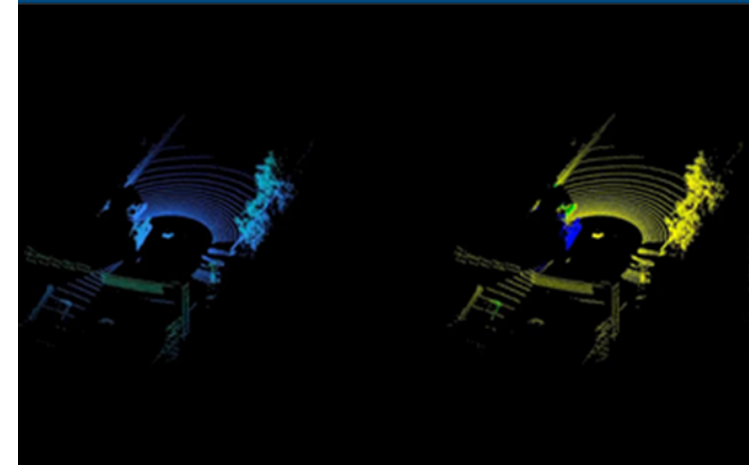
セグメンテーションに関する技術

クラスタリング



- 点間の距離に応じたクラスタリング
- 分類は別途
- 高速計算

セマンティックセグメンテーション



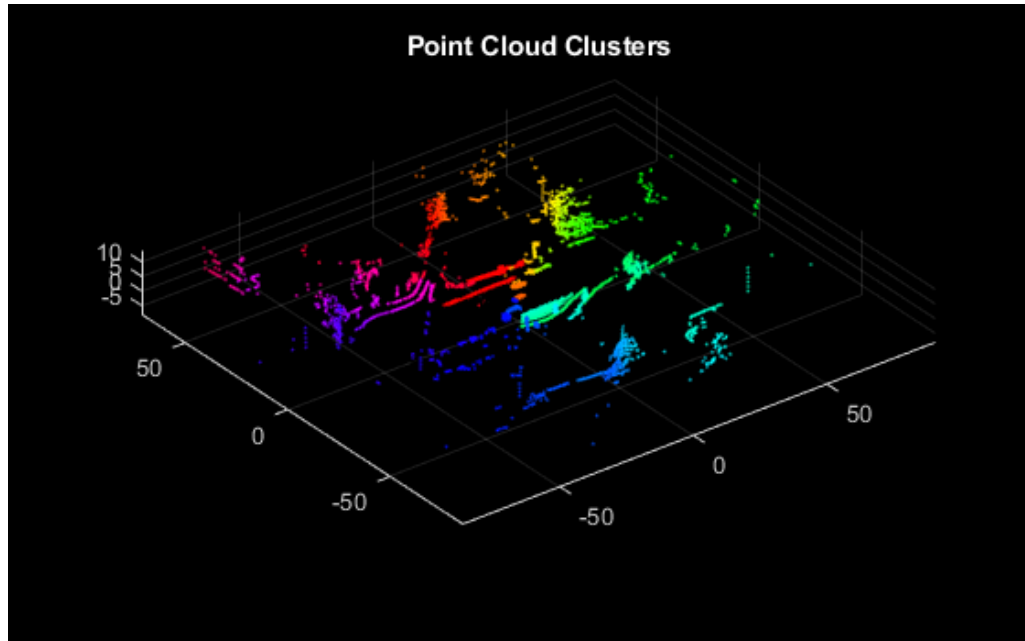
- ディープラーニングによる各点の分類
- 計算時間がかかる

ルールベースのクラスタリング

pcsegdist

ユークリッド距離ベースによるクラスタリング

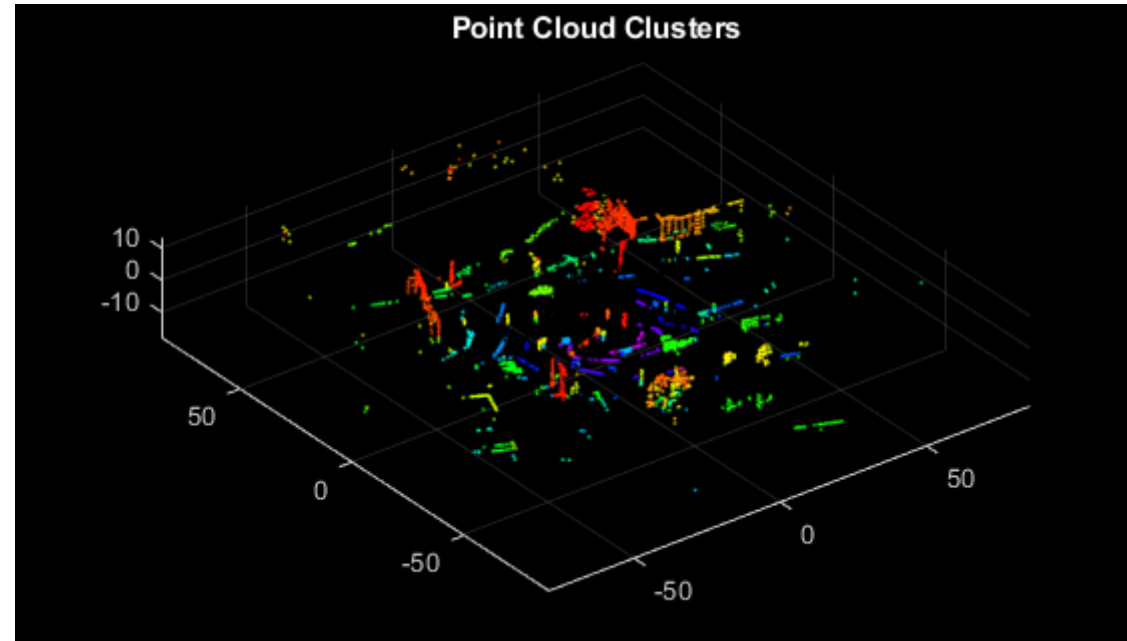
```
distThreshold = 0.5;  
[labels,numClusters] =  
pcsegdist(ptCloud,distThreshold);
```



segmentLidarData

- 距離と角度ベース
- 2次元化したフォーマット用、高速計算

```
distThreshold = 0.5;  
[labels,numClusters] =  
segmentLidarData(ptCloud,distThreshold);
```

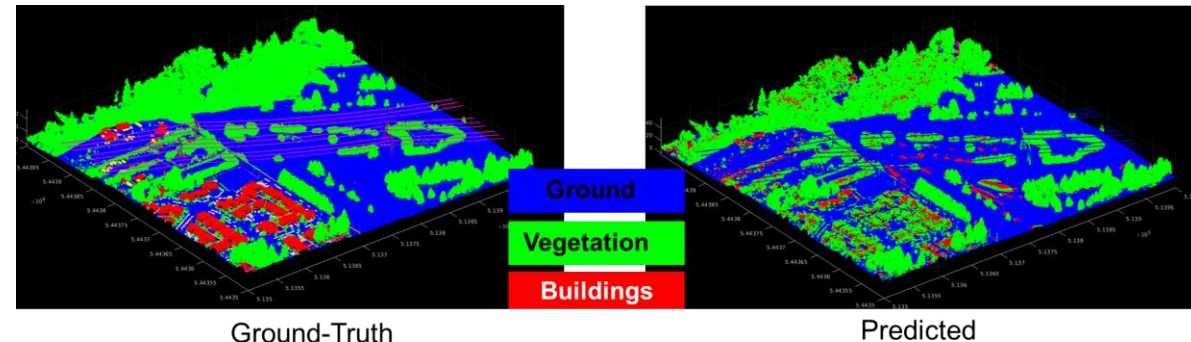
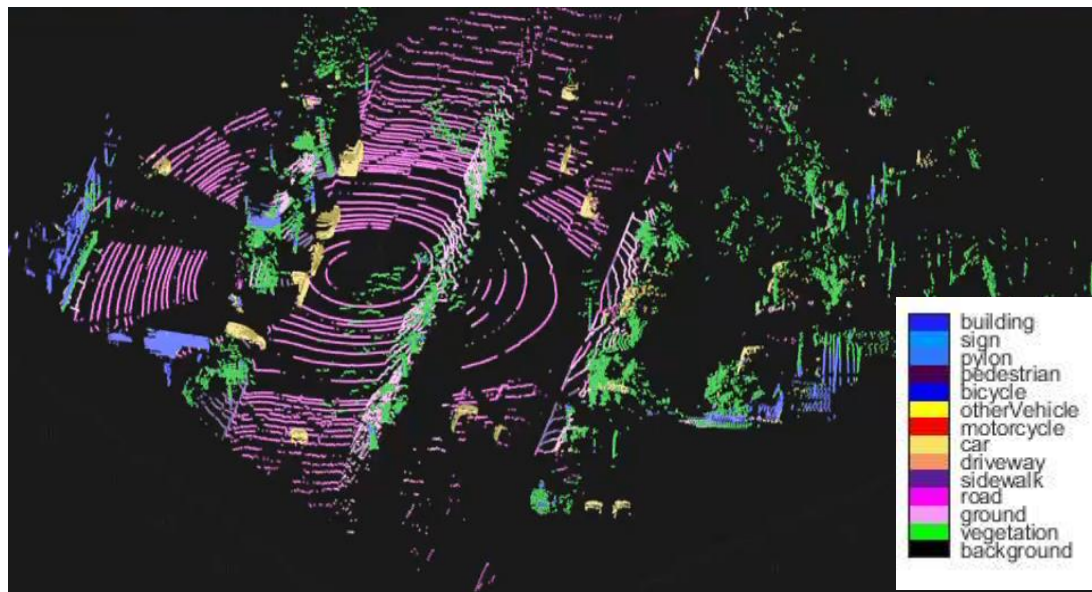


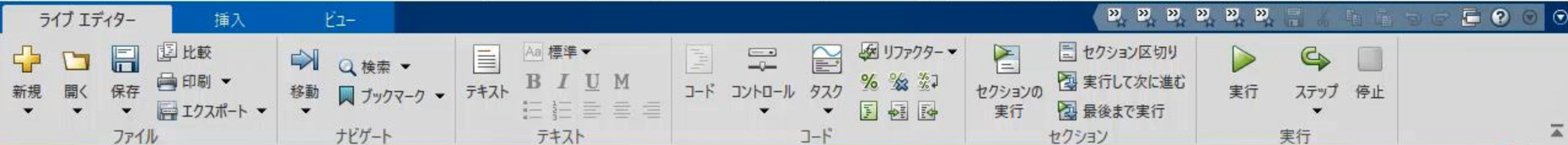
セマンティックセグメンテーション

Lidar Toolbox
Deep Learning Toolbox
Computer Vision Toolbox

セマンティックセグメンテーション用のネットワークを複数サポート

- 複数クラスのセグメンテーションをサポート: SqueezeSegv2, PointNet++ **R2021a**
- 学習済みのONNX モデルを使った推論のみのワークフローも
- 全ネットワークへのコード生成をサポート

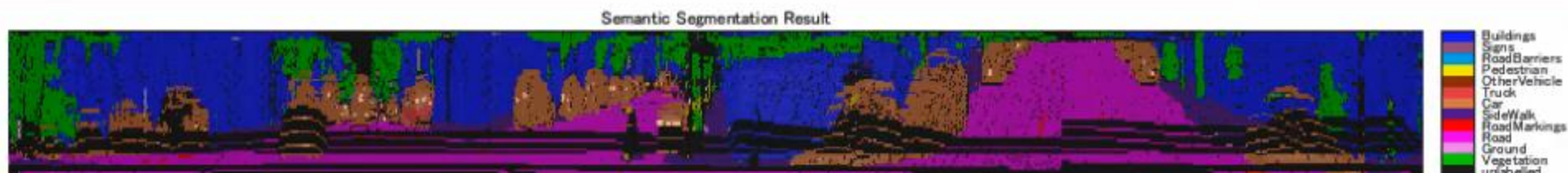




```

23 % セマンティックセグメンテーションの実行
24 predictedResult = semanticseg(I, net);
25 figure;
26 helperDisplayLidarOverlaidImage(I, predictedResult, classNames);
27 title('Semantic Segmentation Result');

```



サポート関数

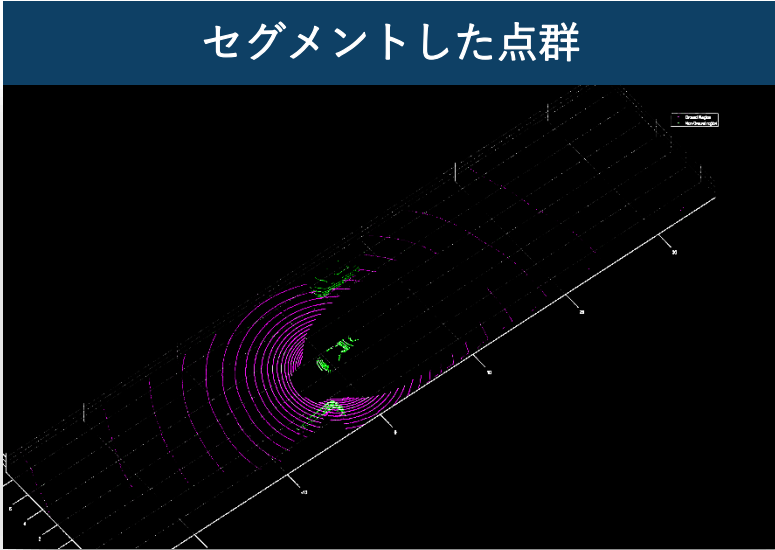
```

28 function helperDisplayImage(I)
29     figure;
30     tiledlayout(5,1,'Padding','compact','TileSpacing','compact');
31     nexttile;

```

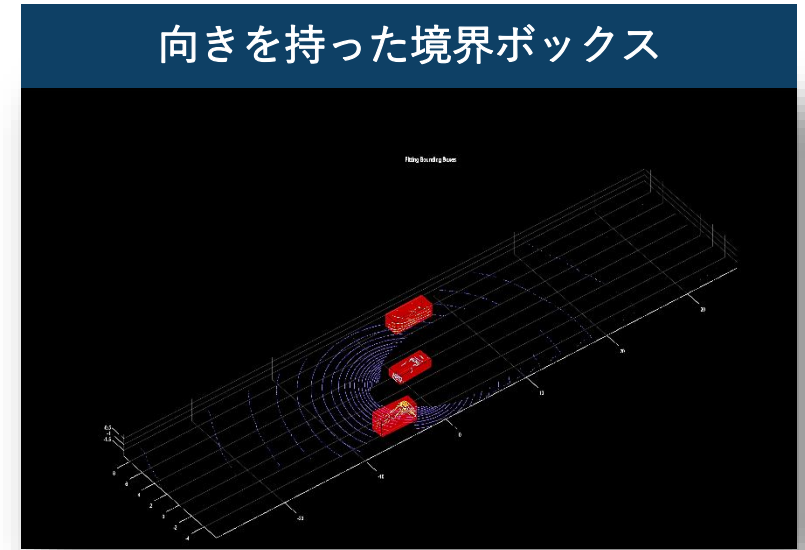
境界ボックスを点群にフィッティング

セグメントした点群



`pcfitcuboid`

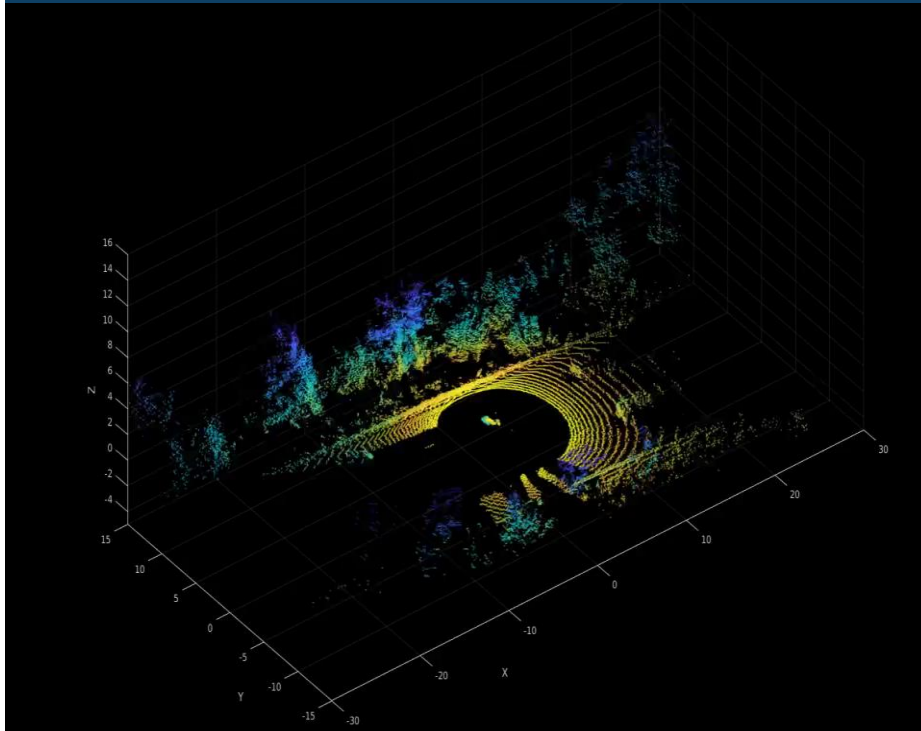
向きを持った境界ボックス



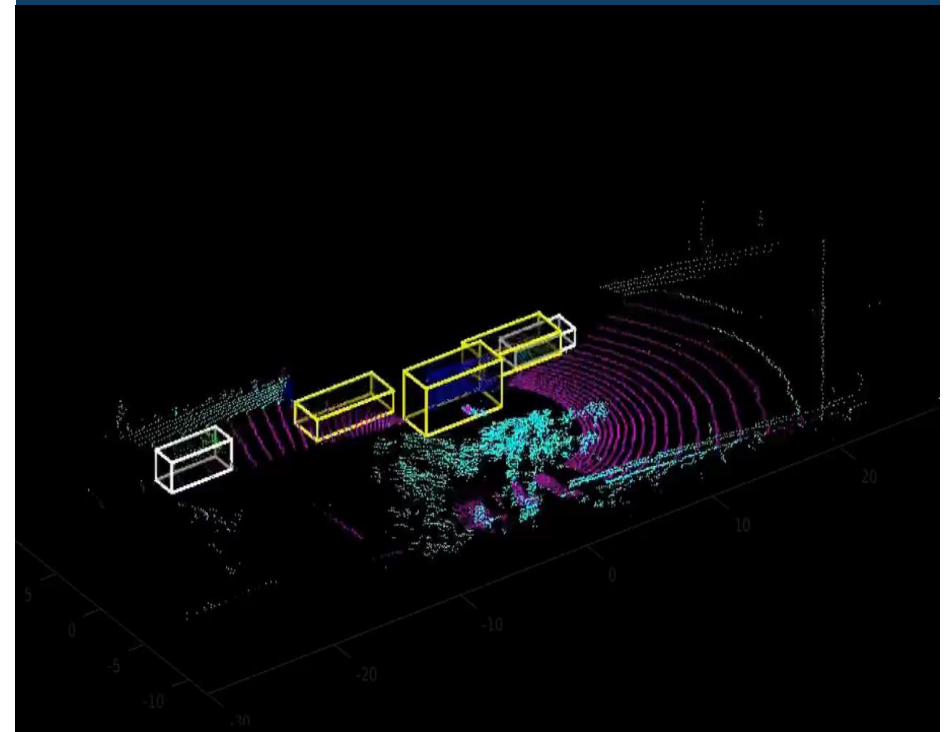
```
% 境界ボックスをセグメントした点群にフィッティング  
Model = pcfitcuboid(pc, label_id)
```


点群データからのオブジェクト検出

入力した 点群データ



出力した境界ボックス付き点群データ



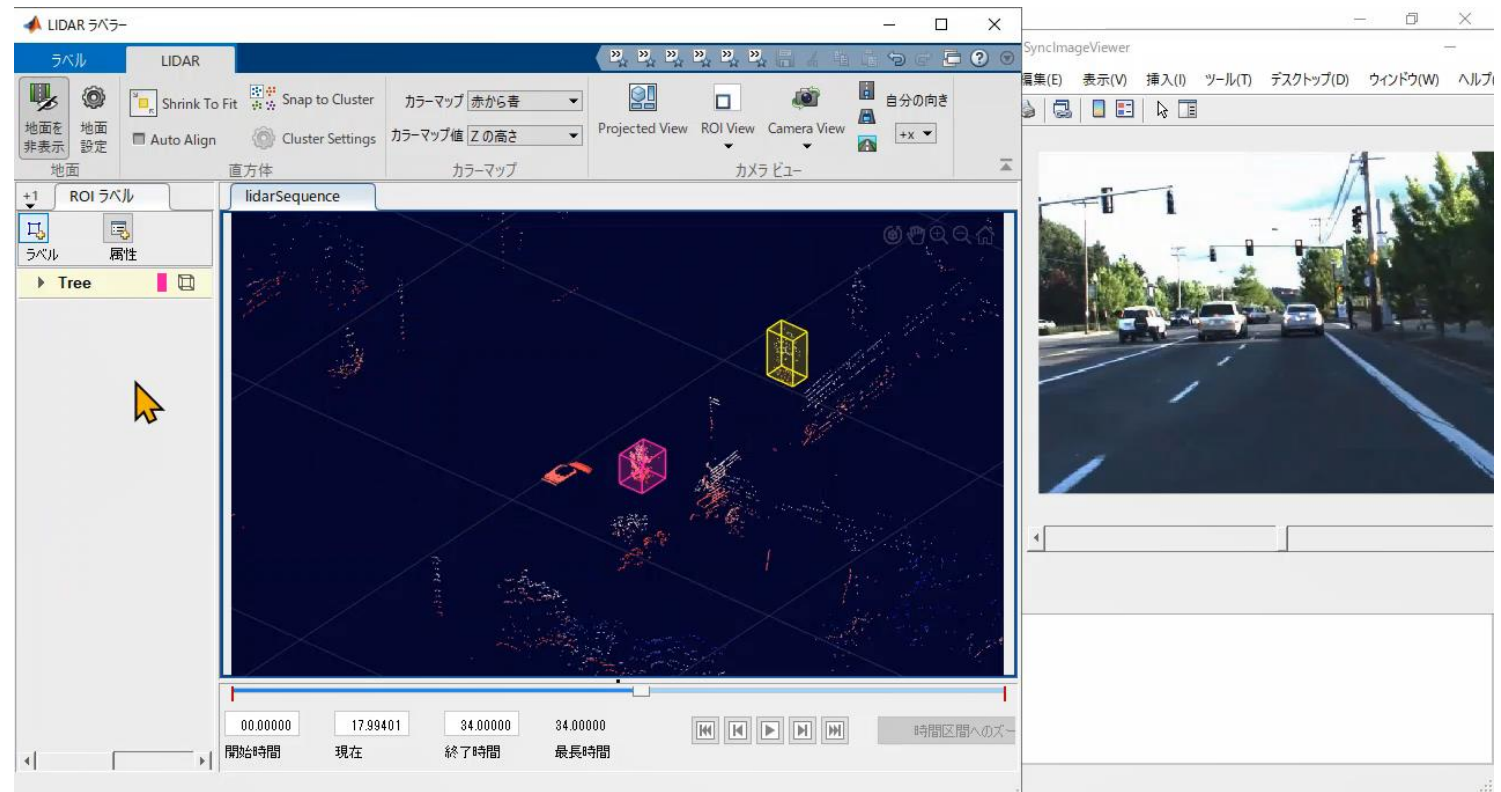
<https://www.mathworks.com/help/releases/R2021a/lidar/ug/object-detection-using-pointpillars-network.html>

Lidar ラベラー : Lidar ラベリングアプリ

- Lidarの点群や時系列の点群に対して真値ラベリング
- PLY, PCAP, PCD,カスタムフォーマットの読み込みサポート

機能

- カスタムファイルフォーマットの読み込み
R2021a
- キューボイドのROI定義
- ビルトインの関数
 - クラスタリング
 - 地面のセグメンテーション
 - PointPillars **R2021a**
- ビルトインもしくはカスタムの自動ラベリングアルゴリズム
- 自動ラベリング結果を視覚的に評価可能

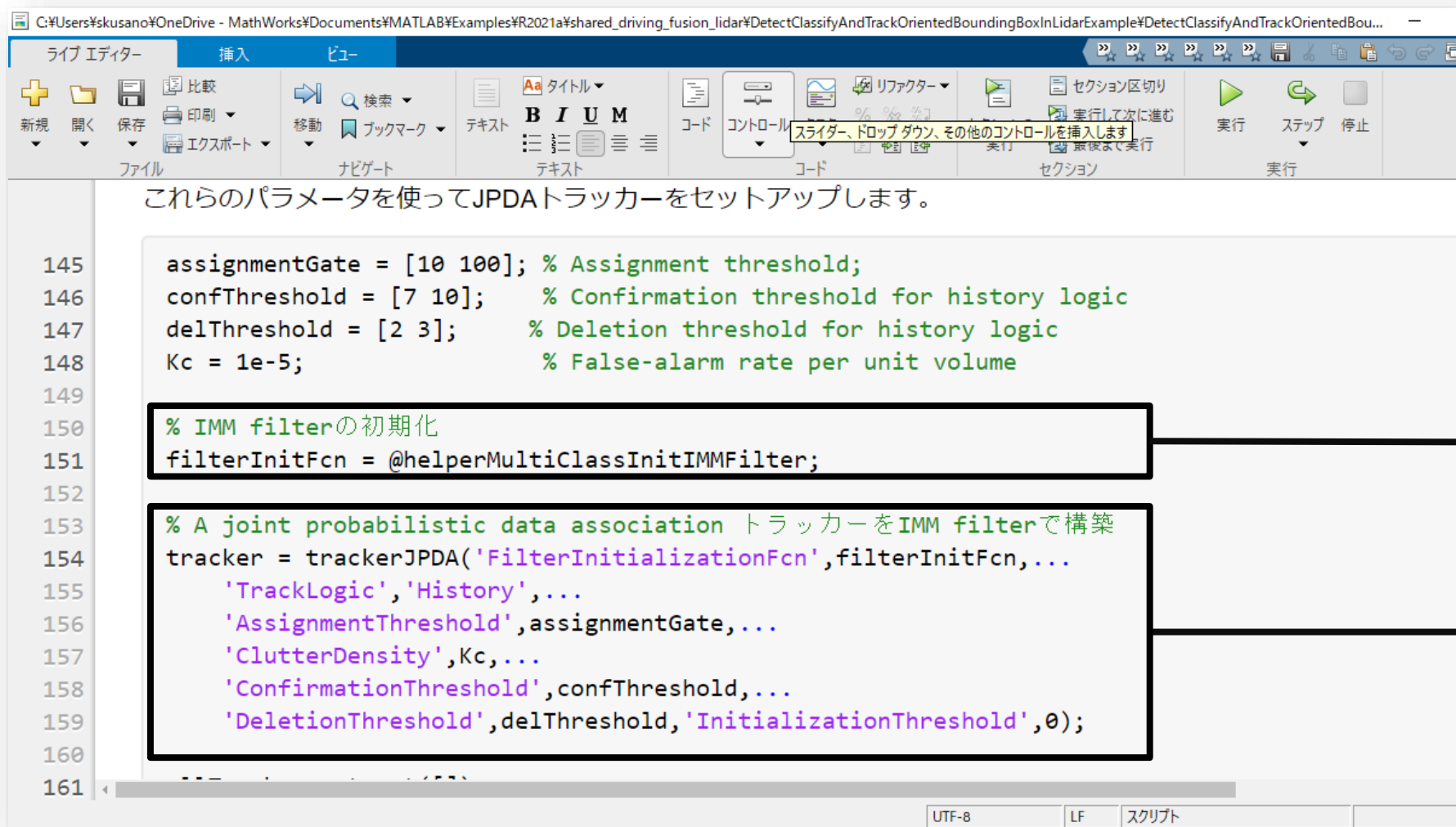


ワークフロー

自動運転に向けたオブジェクト検出とトラッキング



JPDA トラッカーの適用 (1)



トラッキング

複数のモーションモデルの
ためのIMMフィルタ

JPDA トラッカーの初期化

JPDA トラッカーの適用 (2)

```
176
177 rangeImage = input(:,:,5);
178
179 % Lidarデータから境界ボックスを検出
180 [boundingBox,coloredPtCloud,pointLabels] = detectBbox(lidarDetector,input);
181
182 % 境界ボックスをobjectDetectionsオブジェクトに
183 detections = helperAssembleDetections(boundingBox,measNoise,time);
184
185 % オブジェクトをトラッカーに渡してトラッキング
186 if ~isempty(detections)
187     % トラッカー更新
188     [confirmedTracks,tentativeTracks,allTracks,info] = tracker(detections,time);
189     numTracks(count,1) = numel(confirmedTracks);
190 end
191
192 % 点群の描画
193 updatePointCloud(display,coloredPtCloud);
194
```

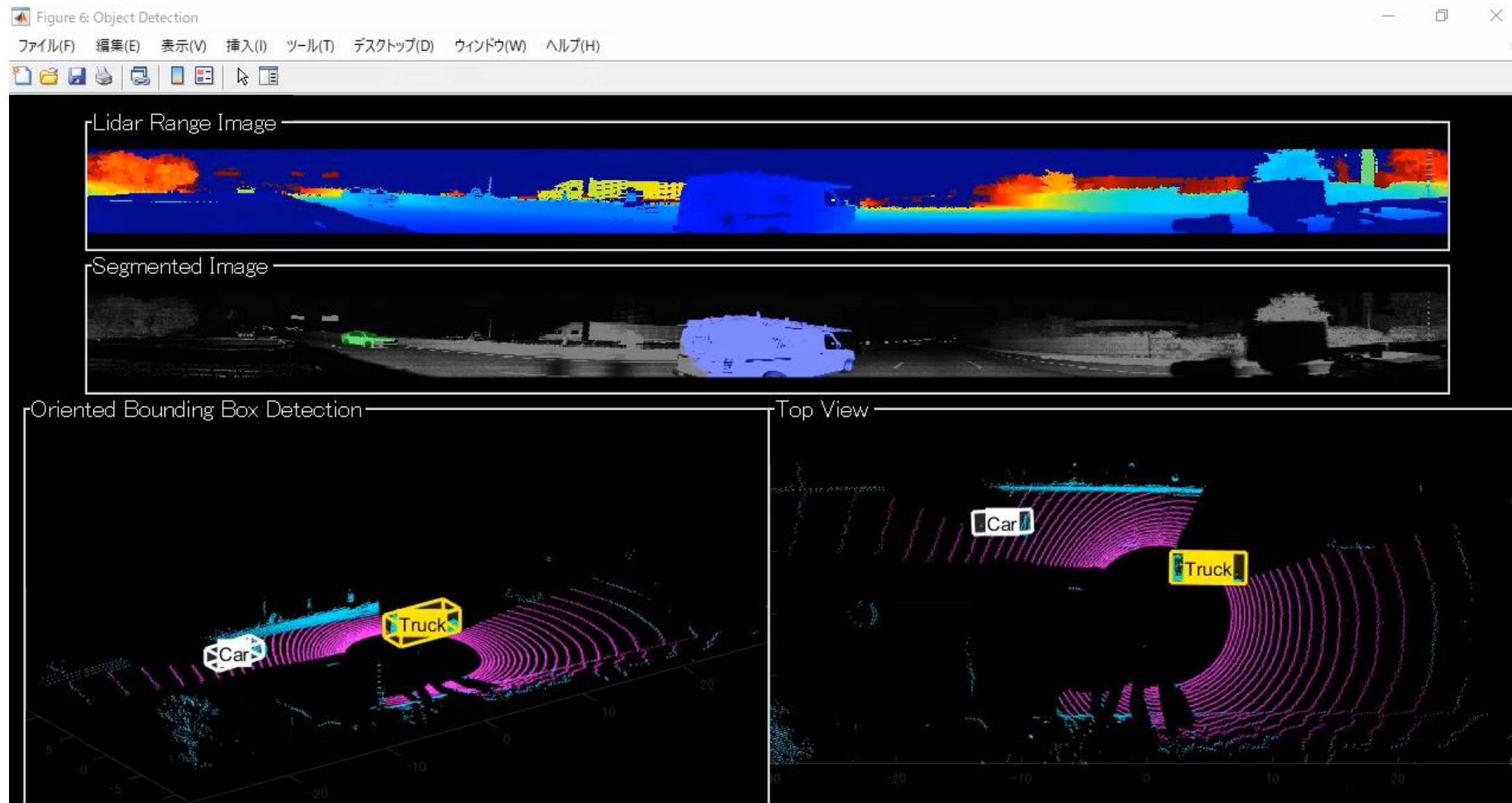
トラッキング

オブジェクト検出

検出をobjectDetections
オブジェクトに変換

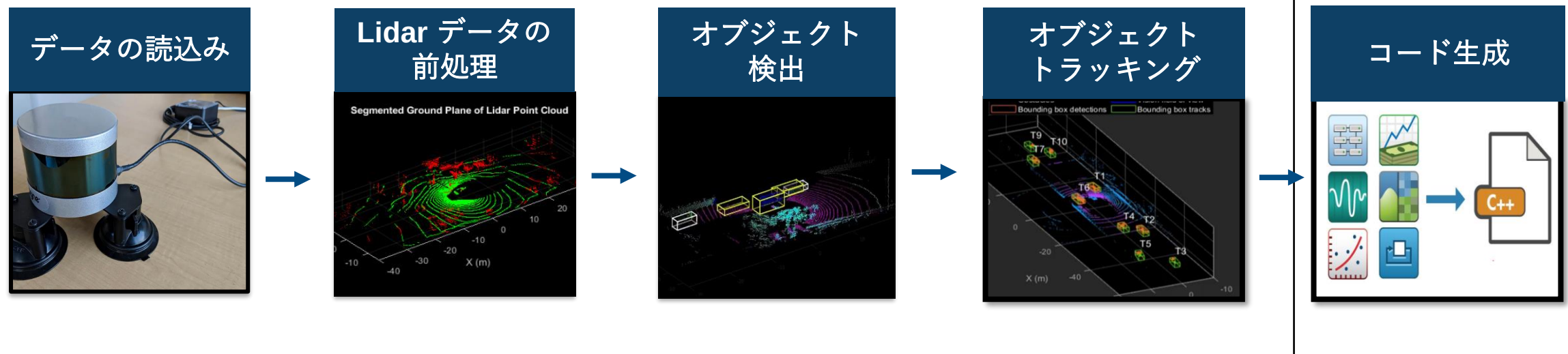
JPDAトラッカーの更新

オブジェクト検出とトラッキングの最終成果例

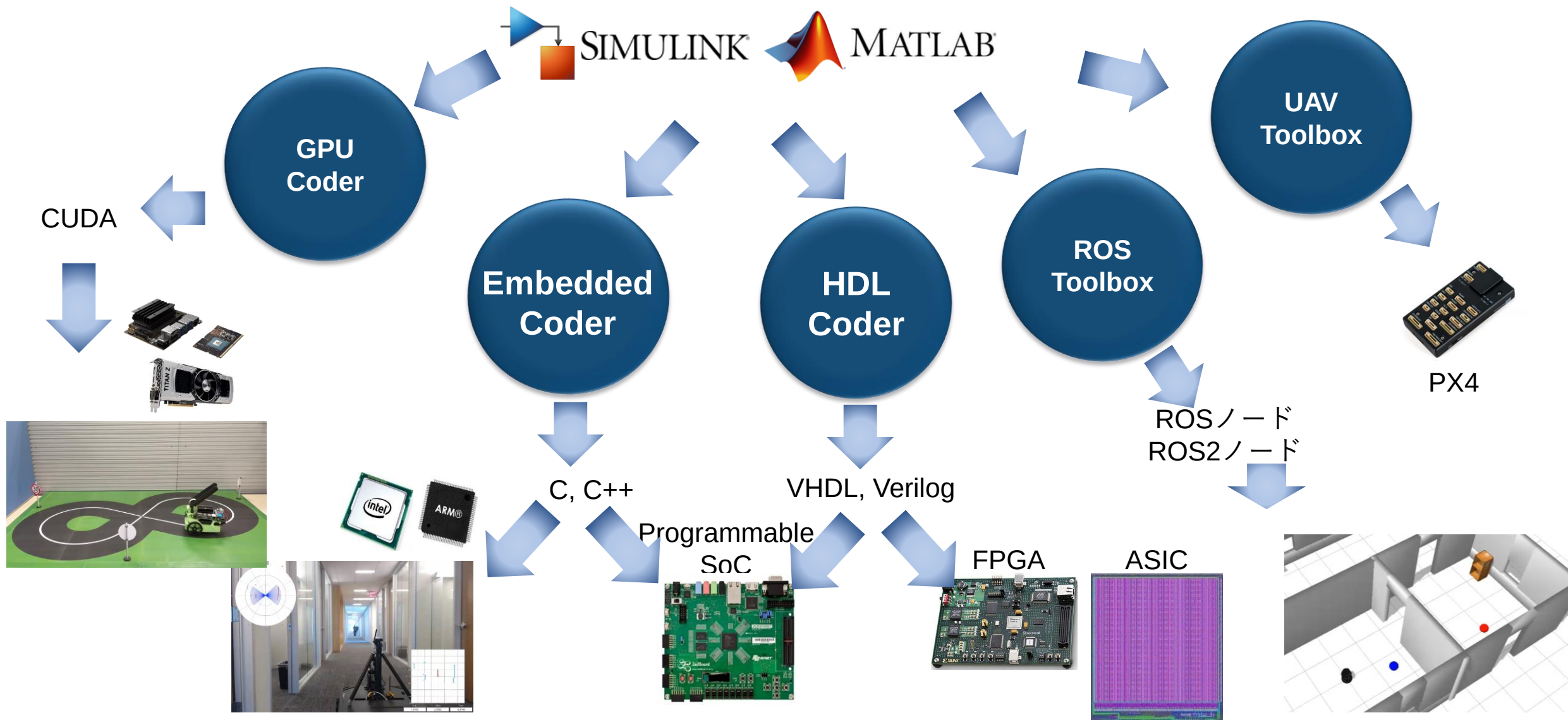


ワークフロー

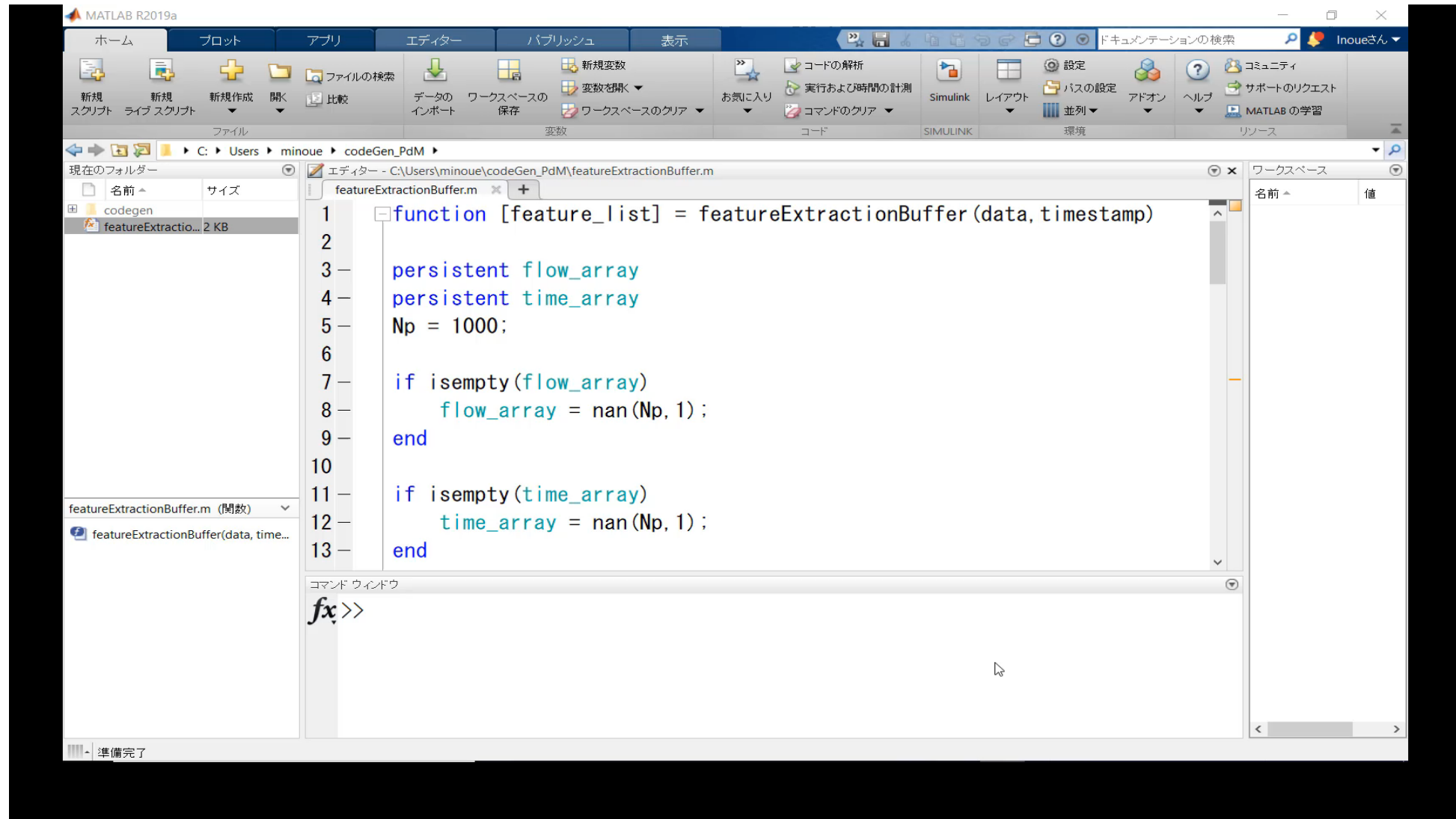
自動運転に向けたオブジェクト検出とトラッキング



コード生成で実機実装プロトタイピングも短時間で！



MATLAB CoderによるCコード生成



CUDAコード生成サンプル

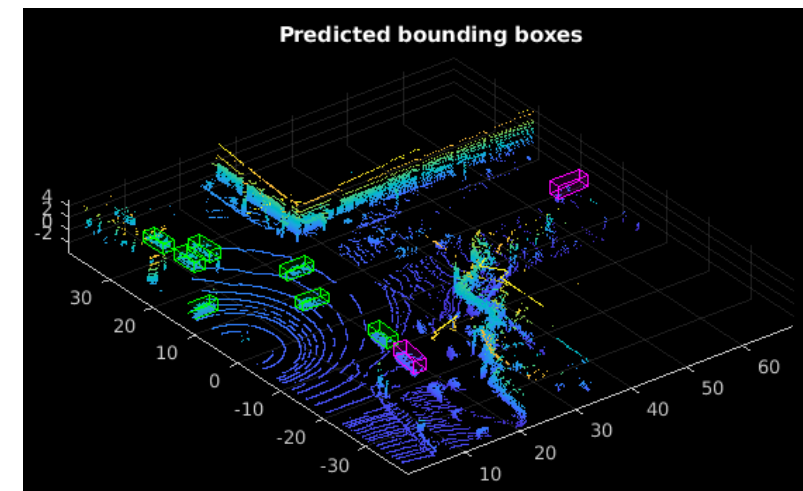
SqueezeSegV2



Code Generation for Lidar Point Cloud Segmentation Network

GPU Coder
Deep Learning Toolbox
Lidar Toolbox

PointPillars

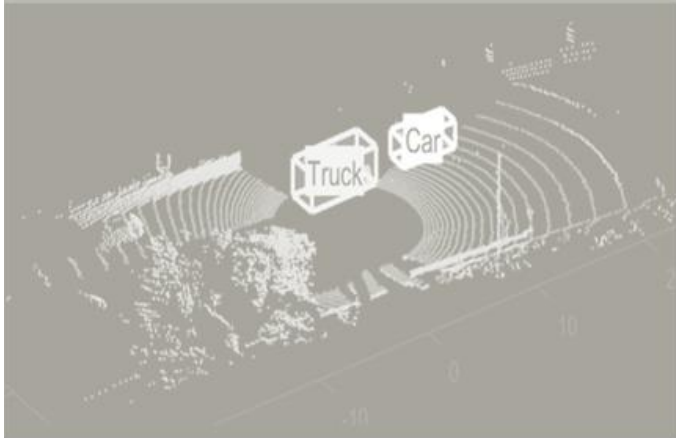


Generation For Lidar Object Detection Using PointPillars Deep Learning

GPU Coder
Lidar Toolbox

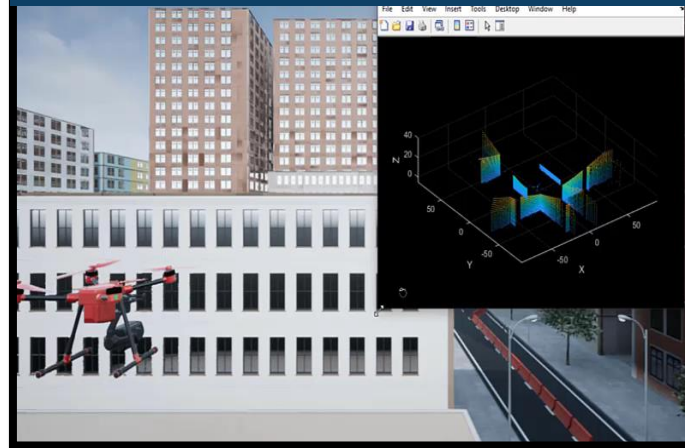
セッションの内容

オブジェクト検出と トラッキング (例題:自動運転)



- ディープラーニングによる検出
- JPDA トラッカー
- コード生成

Lidar SLAM (例題:UAV)



- Lidar点群のシミュレーション
- 特徴量によるマッチング
- Lidar SLAM

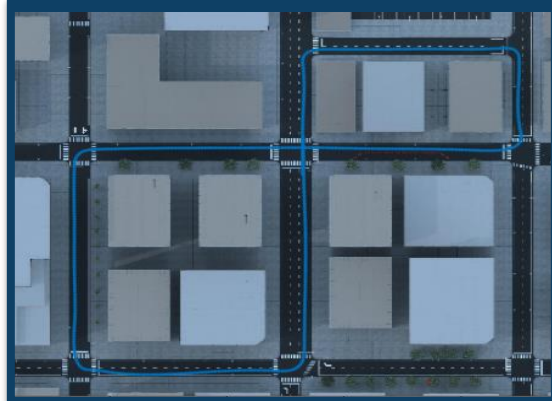
移動ロボットによる 色付き3次元マップ生成 (例題:移動ロボット)



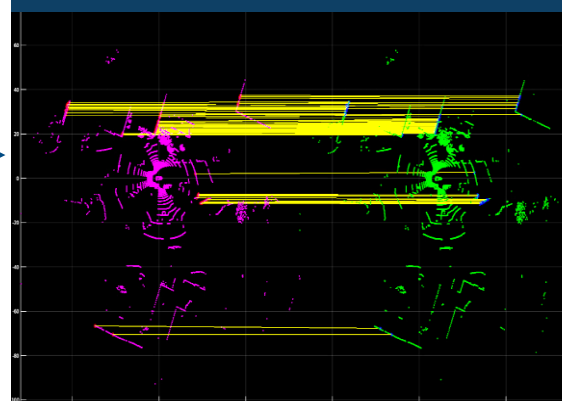
- Lidar - カメラキャリブレーション (LCC)
- 室内のマッピング

Lidar SLAMの航空分野での適用

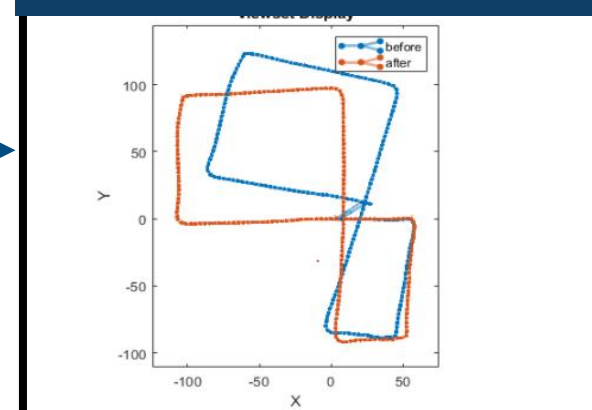
シナリオシミュレーション



点群 レジストレーション



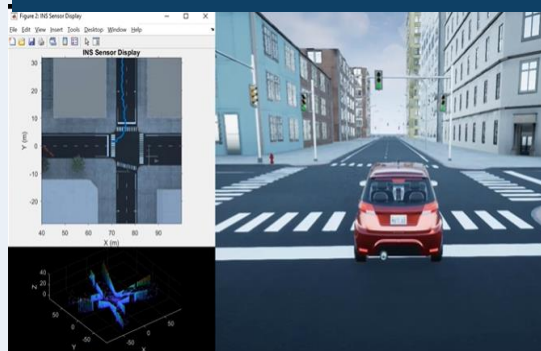
ポーズグラフ最適化



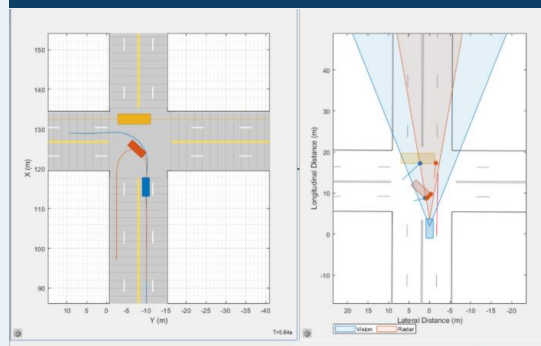
シミュレーション環境

Automated Driving Toolbox

Unreal Engine®



Cuboid Environment



3D Scene 生成

RoadRunner

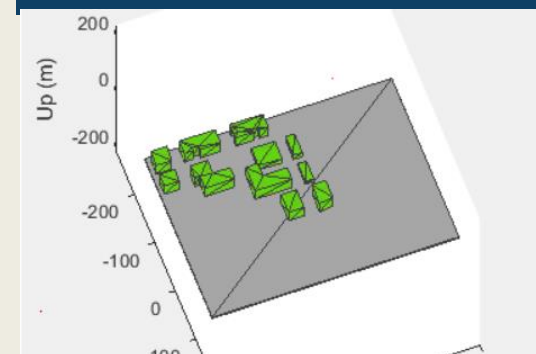


UAV Toolbox

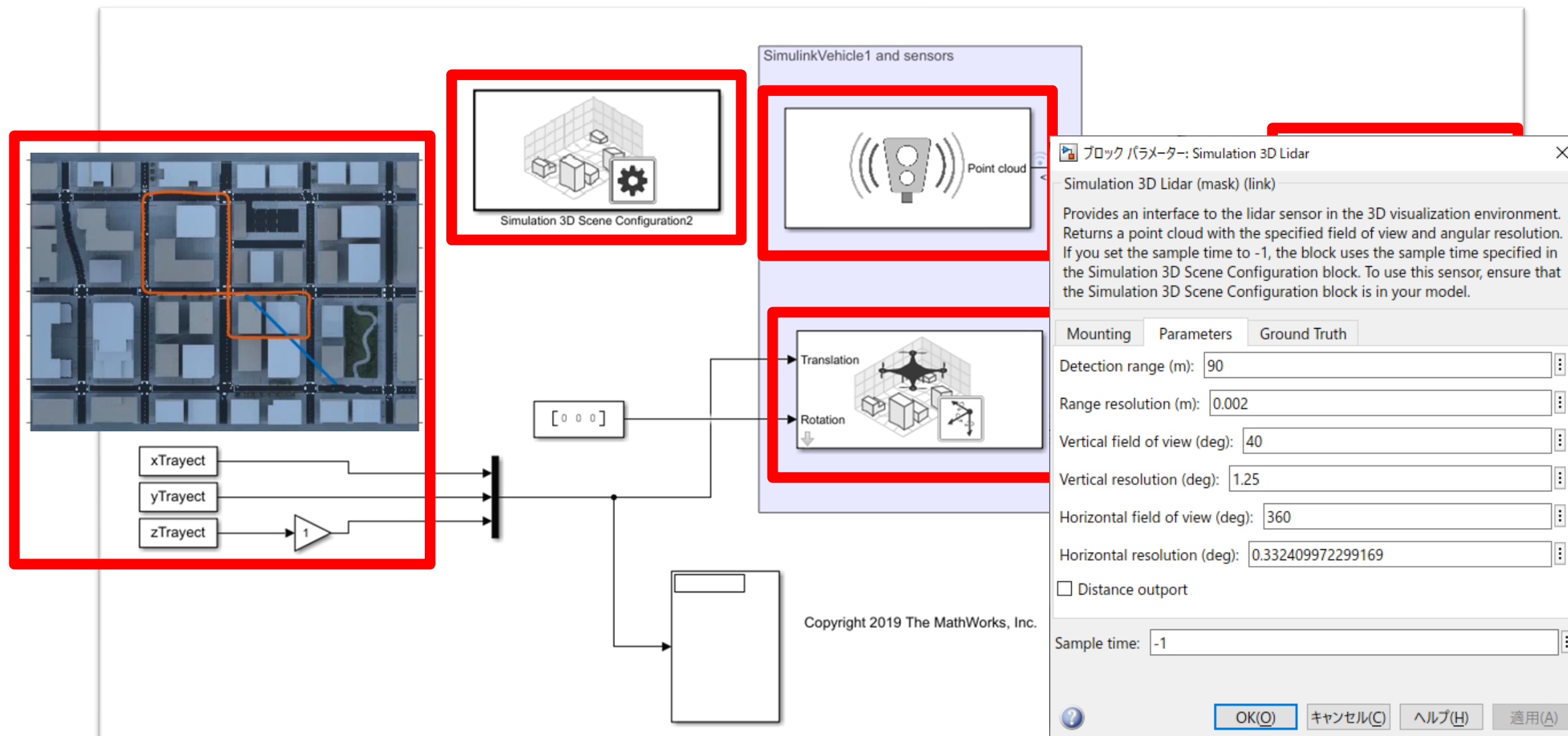
Unreal Engine®



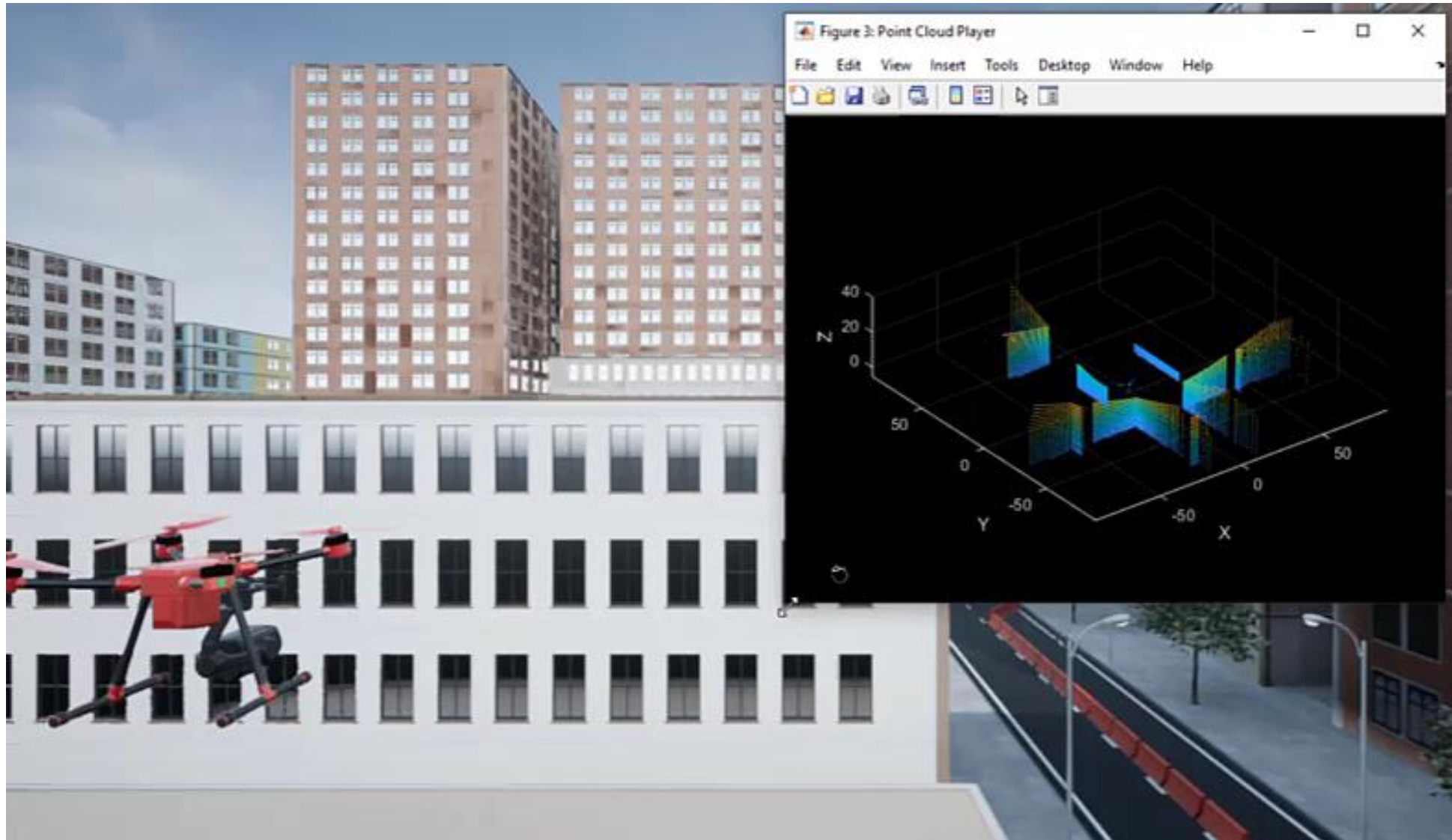
Cuboid Environment



UAV-シナリオシミュレーションモデル



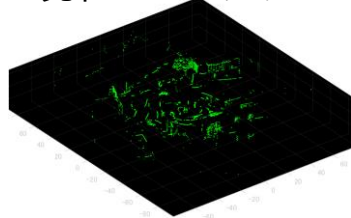
シミュレーションと得られる点群データ



FPFH特徴量ベースの点群マッチング

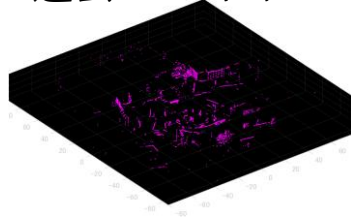
- FPFH特徴量の抽出 + マッチング
- マッチングした点を比較し、点群同士の相対姿勢を推定
- 点群すべてを使うより高速

現在のスキャン



特徴量抽出

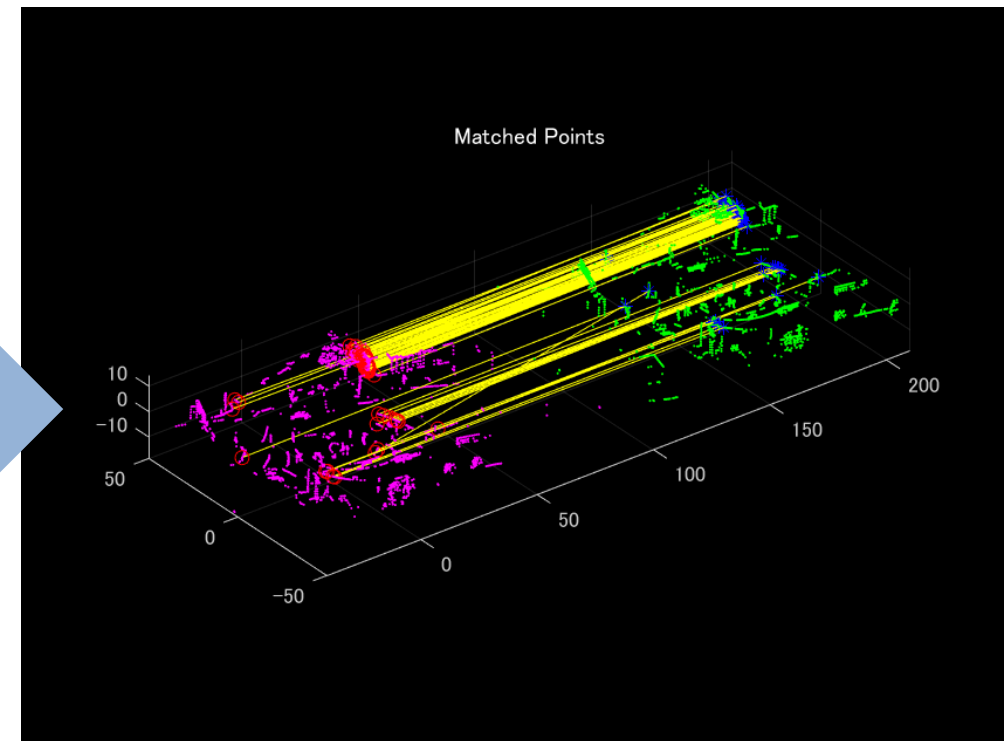
過去のスキャン



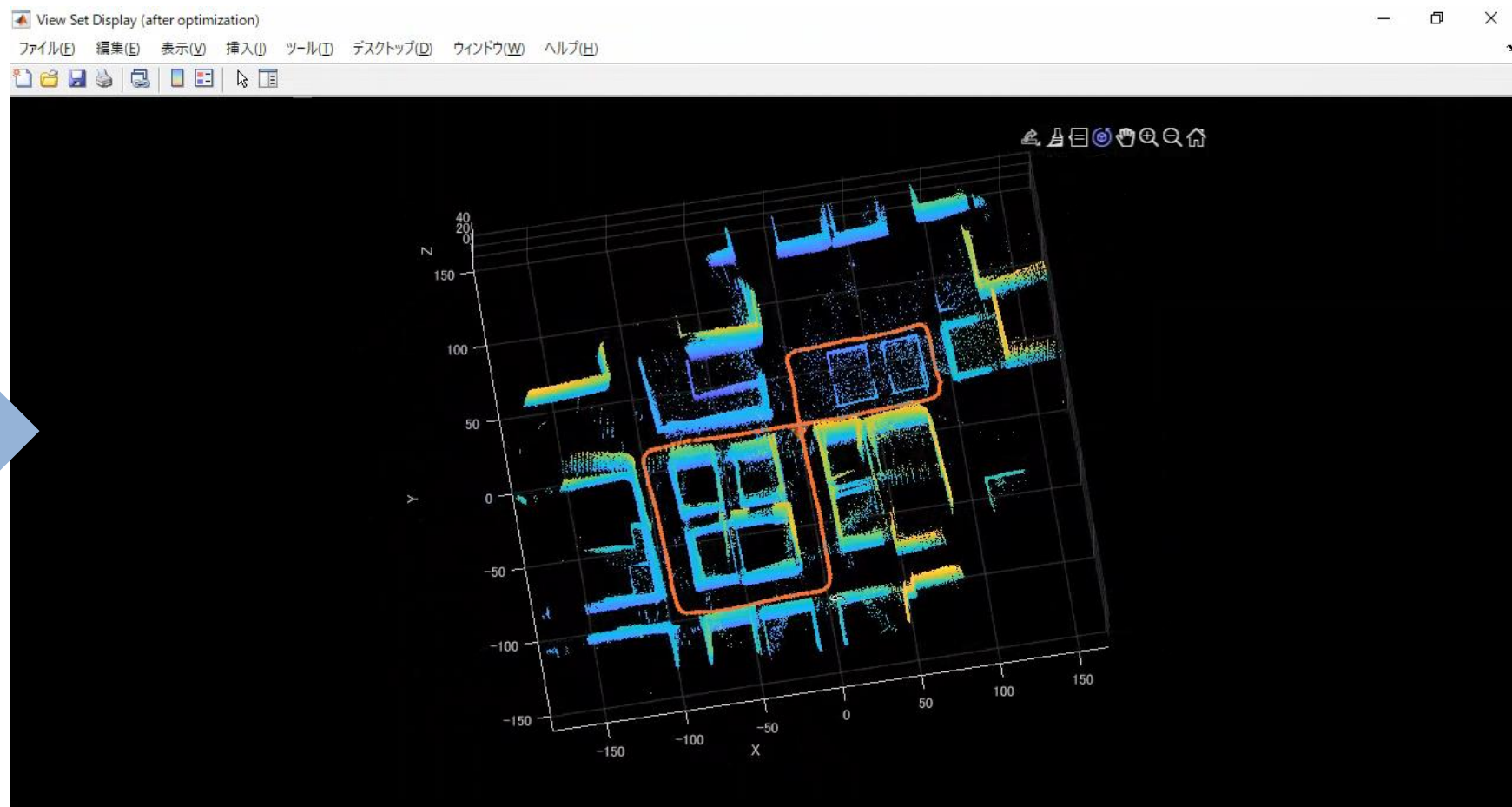
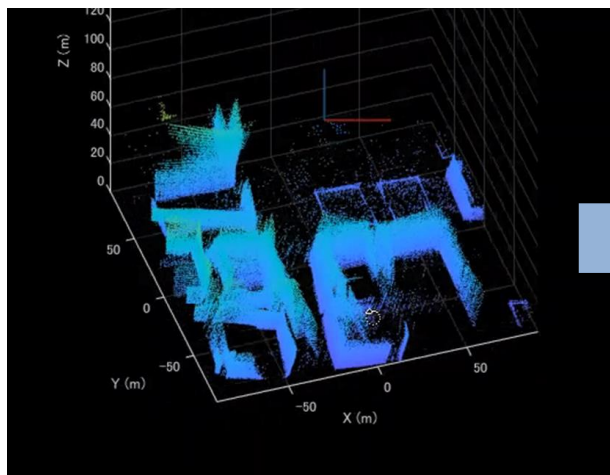
特徴量抽出

マッチング

マッチング結果

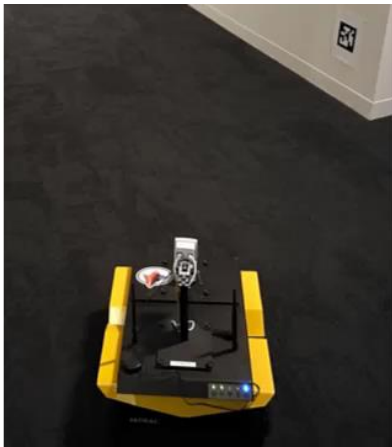


ポーズグラフ最適化による高精度な地図生成



その他のSLAM関連のアルゴリズム

ランドマーク SLAM



[Landmark SLAM Using AprilTag Markers](#)

Navigation Toolbox

Computer Vision Toolbox

R2021a

ステレオ Visual SLAM



[Develop Visual SLAM Algorithm Using Unreal Engine Simulation](#)

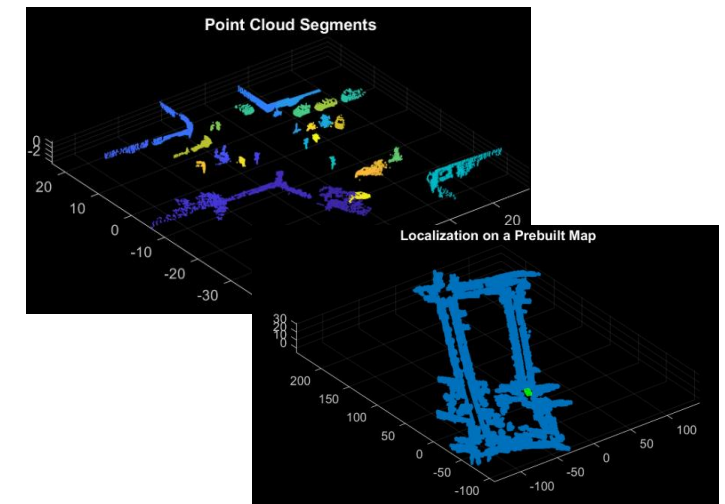
Automated Driving Toolbox

Navigation Toolbox

Computer Vision Toolbox

R2021a

セグメントマッチング



[Build Map and Localize Using Segment Matching](#)

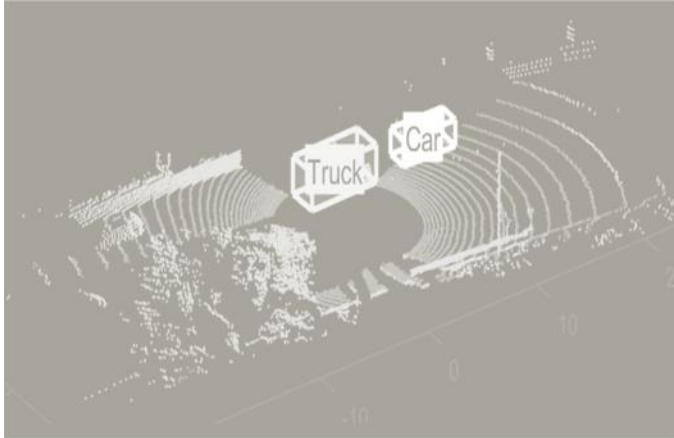
Lidar Toolbox

Navigation Toolbox

R2021a

Session Outline

オブジェクト検出と トラッキング (例題:自動運転)



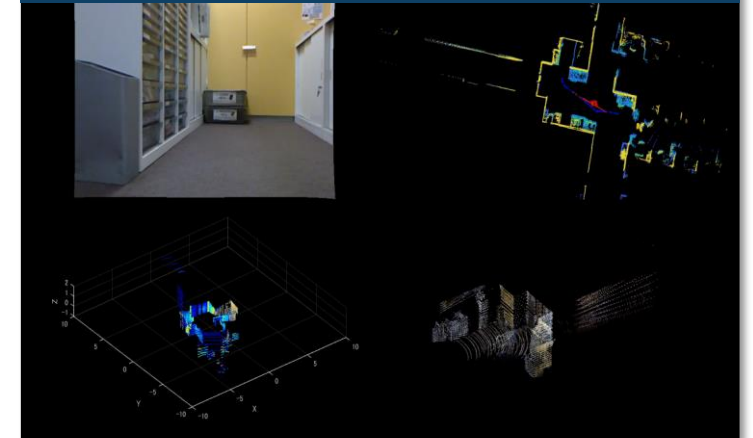
- ディープラーニングによる検出
- JPDA トラッカー
- コード生成

Lidar SLAM (例題:UAV)



- Lidar点群のシミュレーション
- 特徴量によるマッチング
- Lidar SLAM

移動ロボットによる 色付き3次元マップ生成 (例題:移動ロボット)



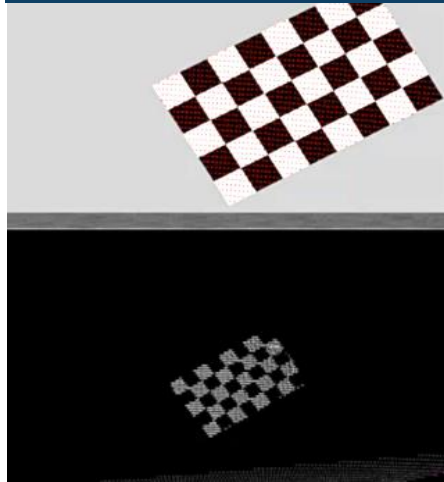
- Lidar – カメラキャリブレーション (LCC)
- 室内のマッピング

移動ロボットにおける3次元Lidar SLAM

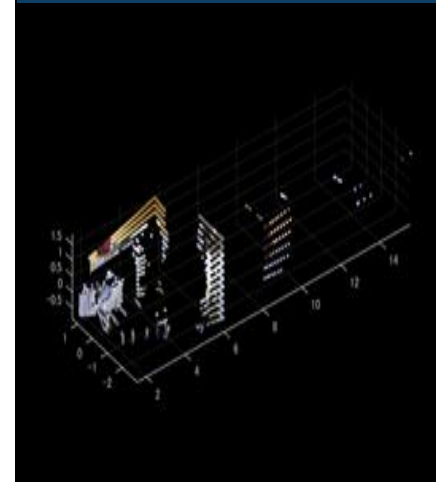
Rosbag
データの
読み込み



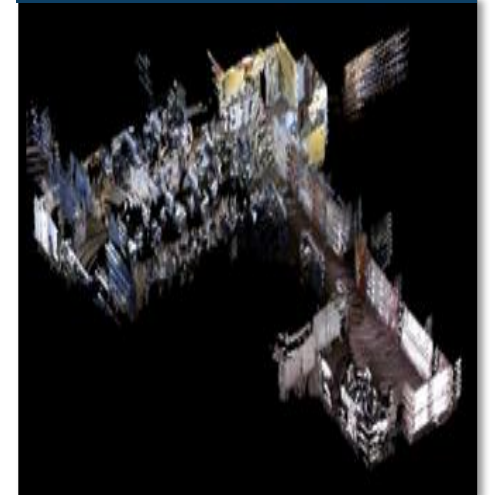
Lidar カメラ
キャリブレーション



Lidarデータへの
カメラ画像の投影



3次元地図生成



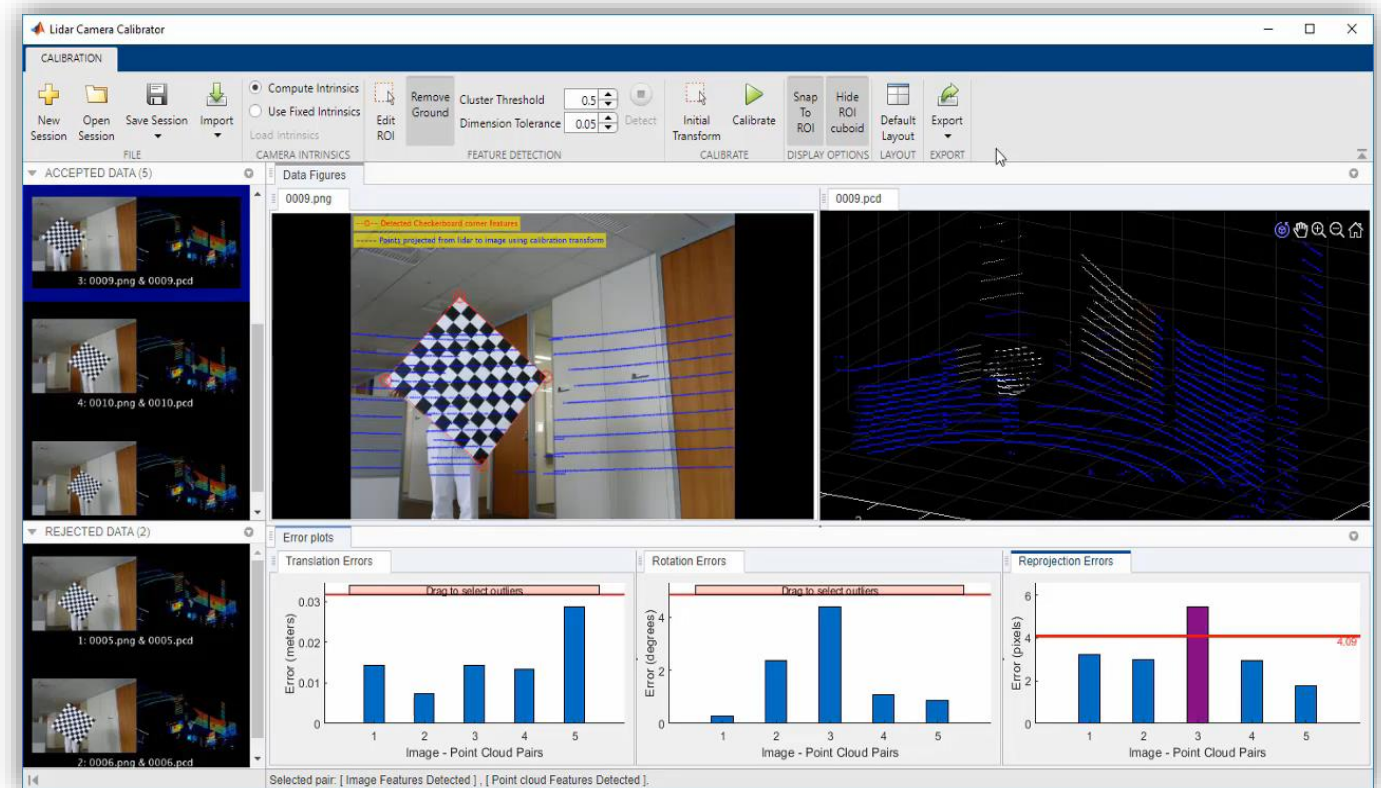
Lidar-カメラキャリブレーションApp

R2021a
Lidar Toolbox

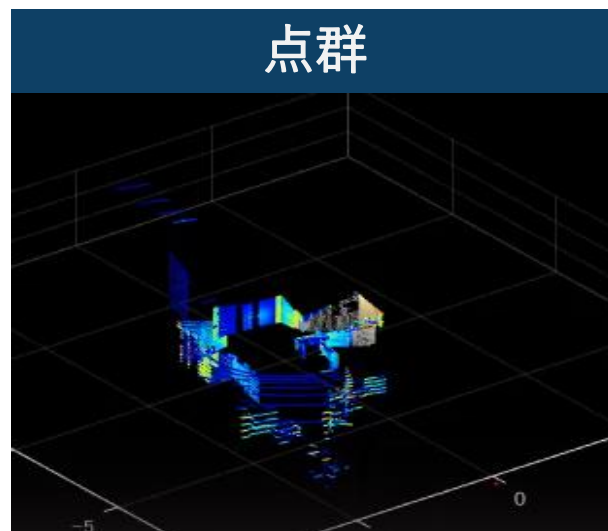
Lidarからカメラへの変換行列をインタラクティブに計算

機能

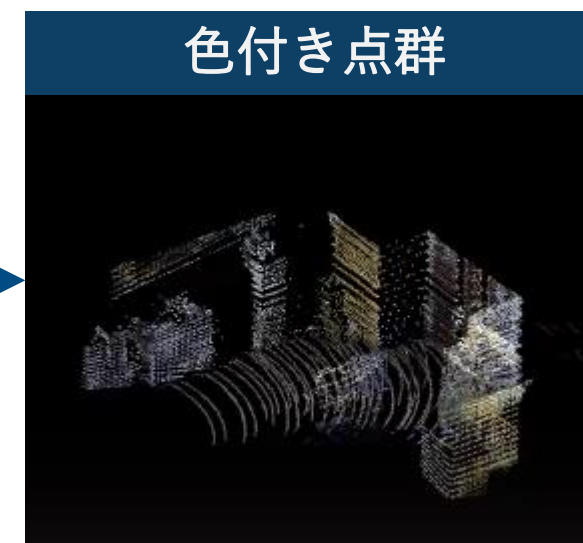
- 点群データとカメラ画像の読み込み
- カメラの内部カメラの読み込み
- 変換推定
- 特徴抽出設定の調整
- Lidar-カメラ間のキャリブレーション
- 結果の可視化と解析



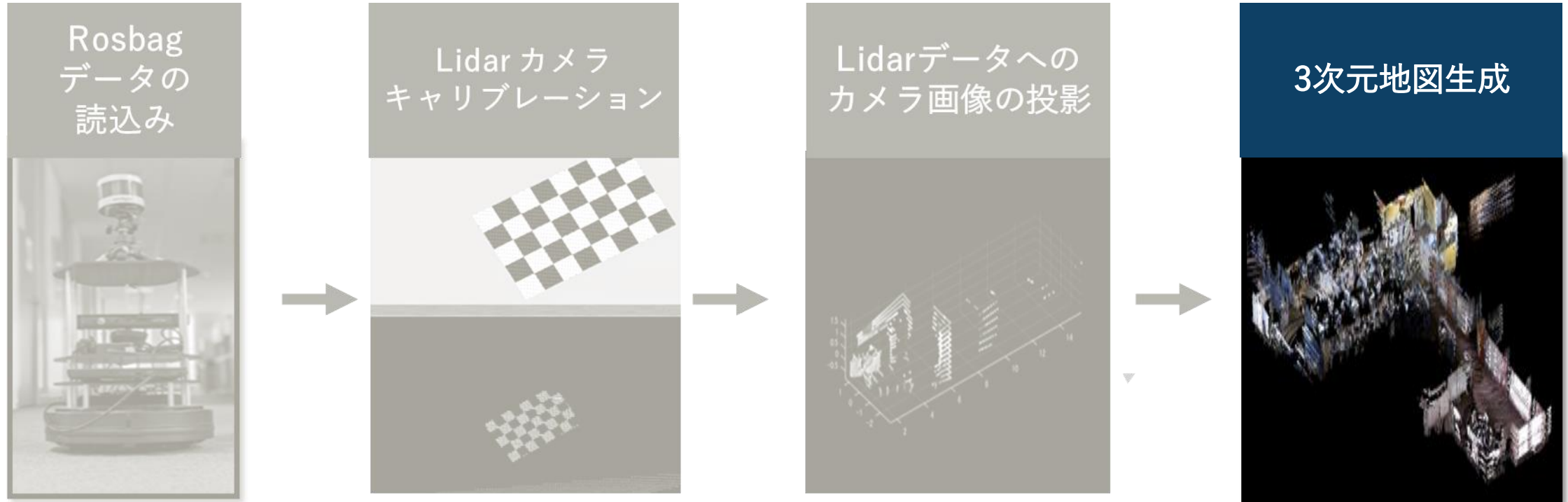
画像データと点群データのフュージョン

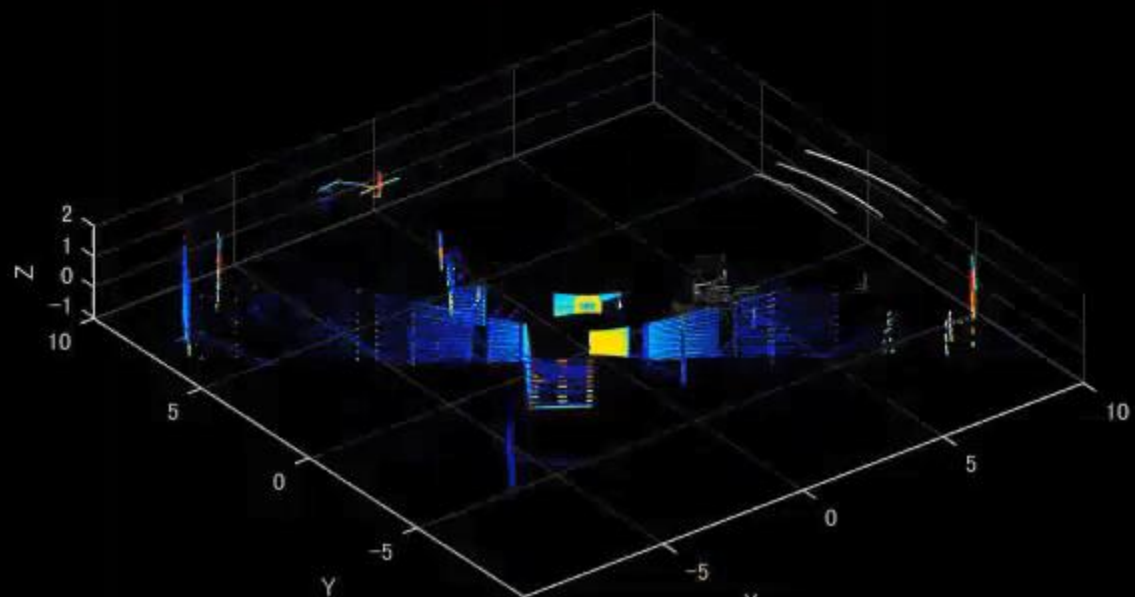
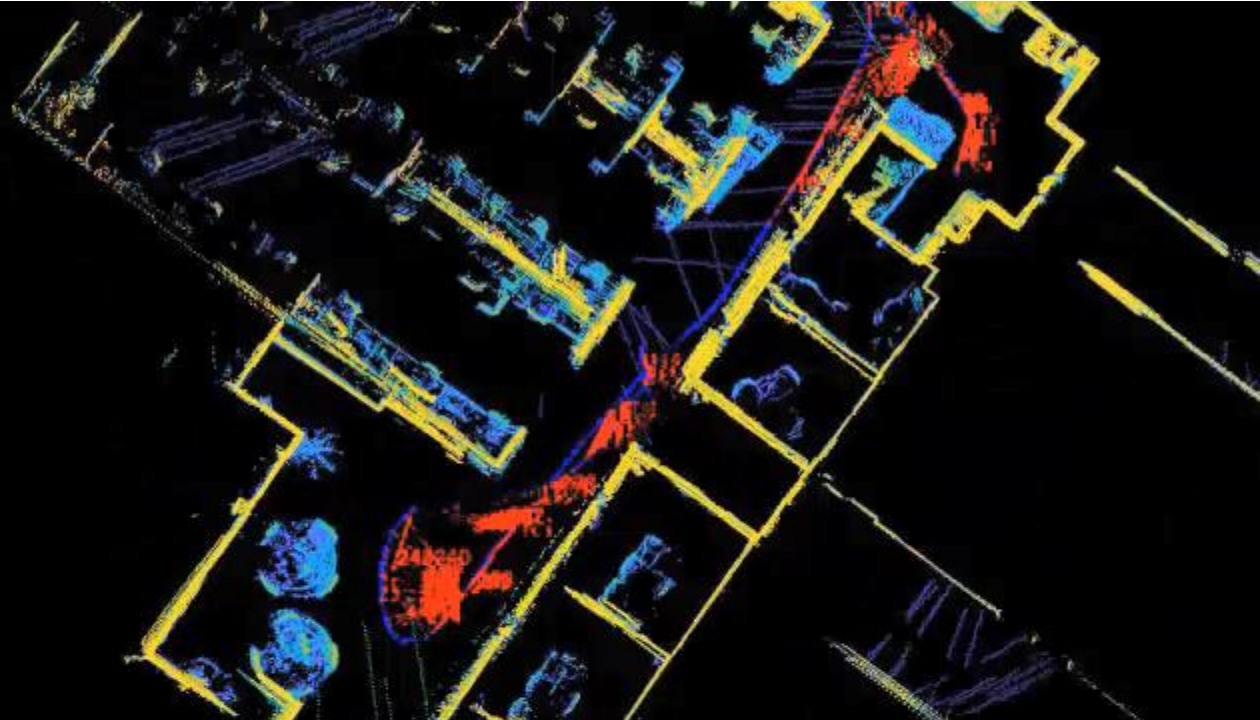


`fuseCameraToLidar`



移動ロボットにおける3次元Lidar SLAM

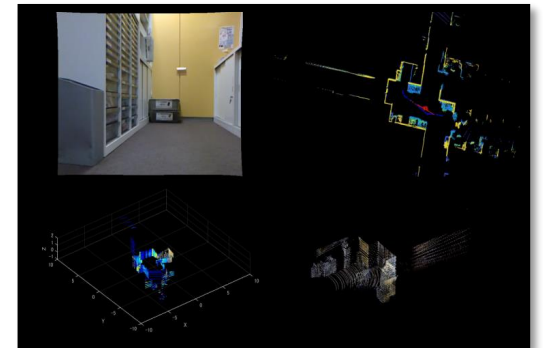
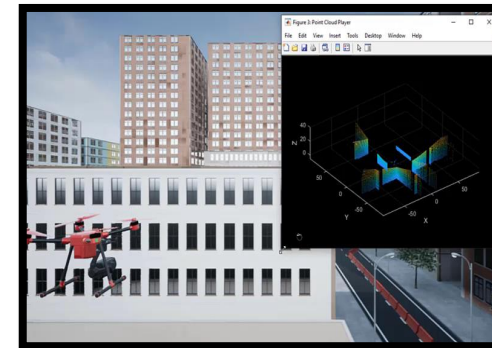
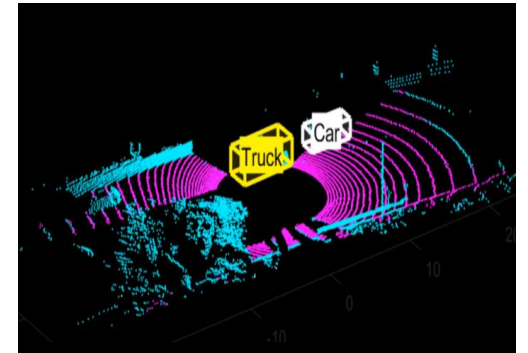




まとめ

Lidar点群データ処理のためのMATLAB

- ディープラーニング適用のためのワークフロー
- 自律移動システム開発のためのLidarデータのシミュレーション
- アルゴリズムのサポート：
 - 前処理
 - オブジェクトトラッキング
 - 地図生成 (SLAM)
 - Lidar カメラキャリブレーション
- コード生成 (C/C++, CUDA)



各種リソース



Lidar Toolbox
Design, analyze, and test lidar processing systems

[Download a free trial](#) [Watch video](#)

Lidar Toolbox™ provides algorithms, functions, and apps for designing, analyzing, and testing lidar processing systems. You can perform object detection and tracking, semantic segmentation, shape fitting, lidar registration, and obstacle detection. Lidar Toolbox supports lidar-camera cross calibration for workflows that combine computer vision and lidar processing.

[Lidar Toolbox - MATLAB](#)

Other Toolboxes

[Computer Vision Toolbox - MATLAB & Simulink](#)

[Deep Learning Toolbox - MATLAB](#)

[Automated Driving Toolbox - MATLAB](#)

[UAV Toolbox - MATLAB](#)

[Navigation Toolbox - MATLAB](#)

リファレンス例題

[Detect, Classify, and Track Vehicles Using Lidar](#)

[Code Generation for Lidar Point Cloud Segmentation Network](#)

[Lidar and Camera Calibration](#)

[Aerial Lidar SLAM Using FPFH Descriptors](#)

MATLAB EXPO 2021

Thank you



© 2021 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [mathworks.com/trademarks](https://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.