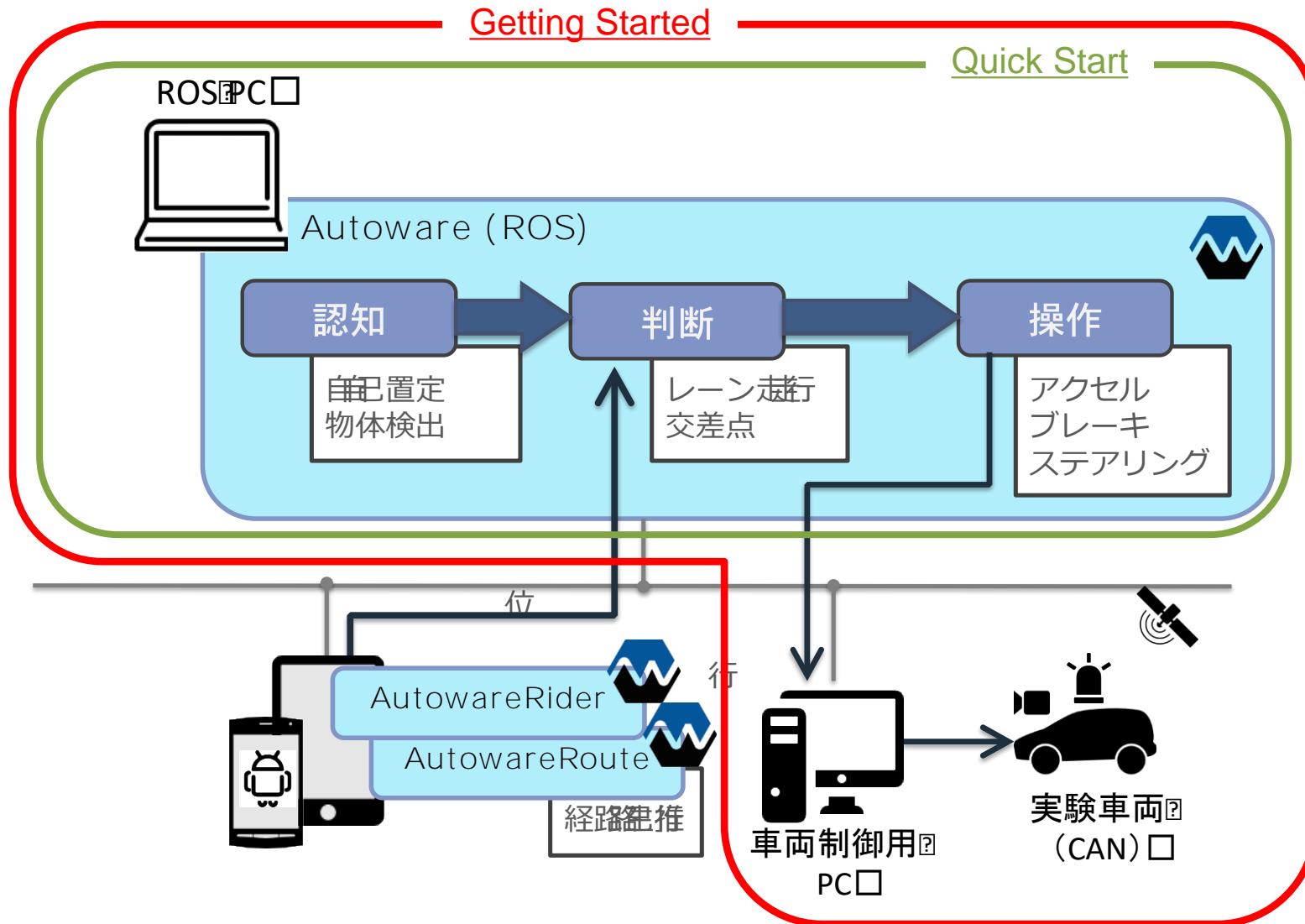


# Autoware – Getting Started



# 0. このマニュアルについて

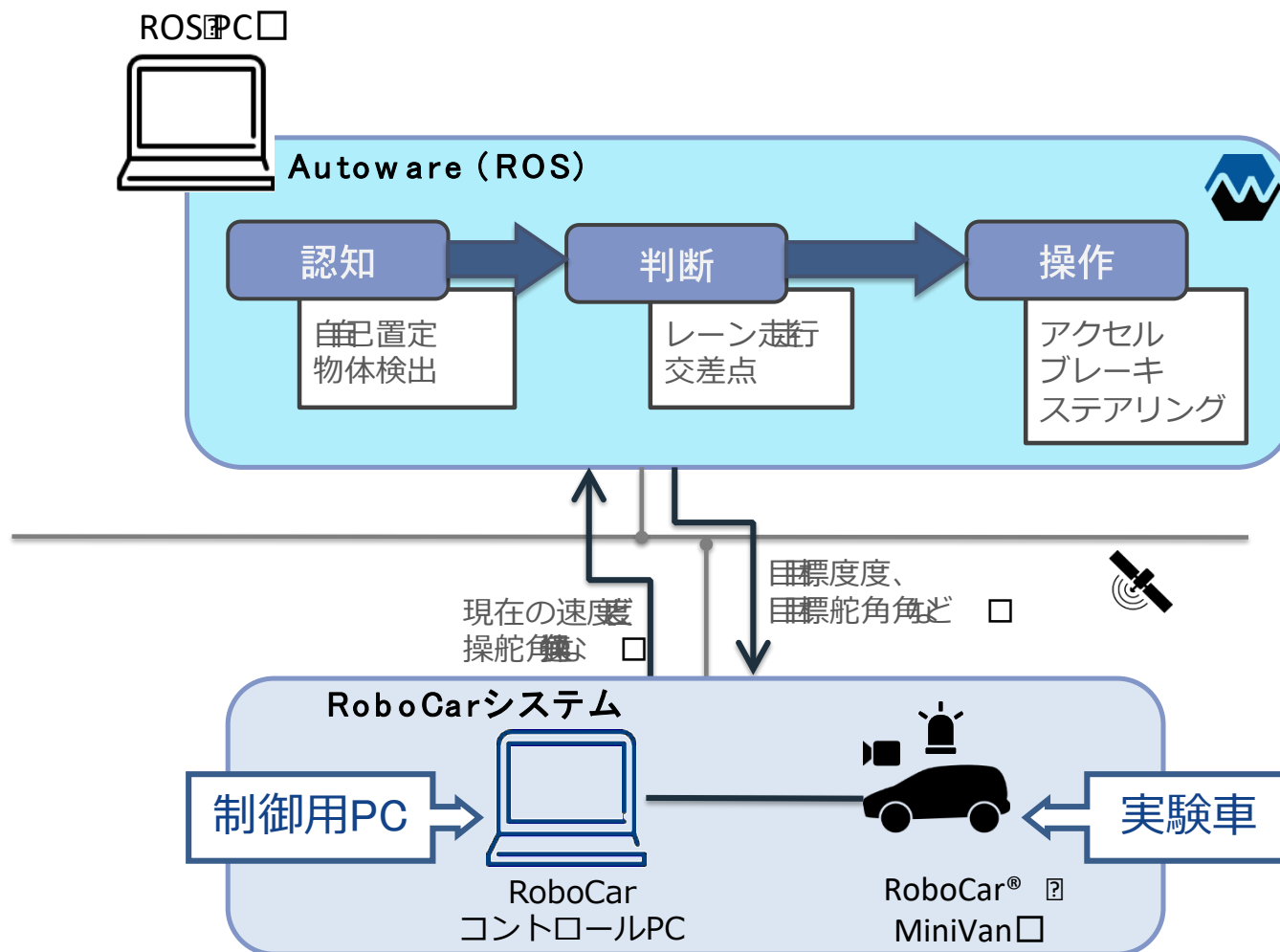
## 0.1. Quick Start と Getting Started の対象範囲



# 0. このマニュアルについて

## 0.2. RoboCar® を用いた操作

「Getting Started」では実験車として ZMP社のRoboCar® MiniVanを用いて説明します。  
車両の仕様およびZMPの免責事項については巻末にあるAppendixを参照してください。



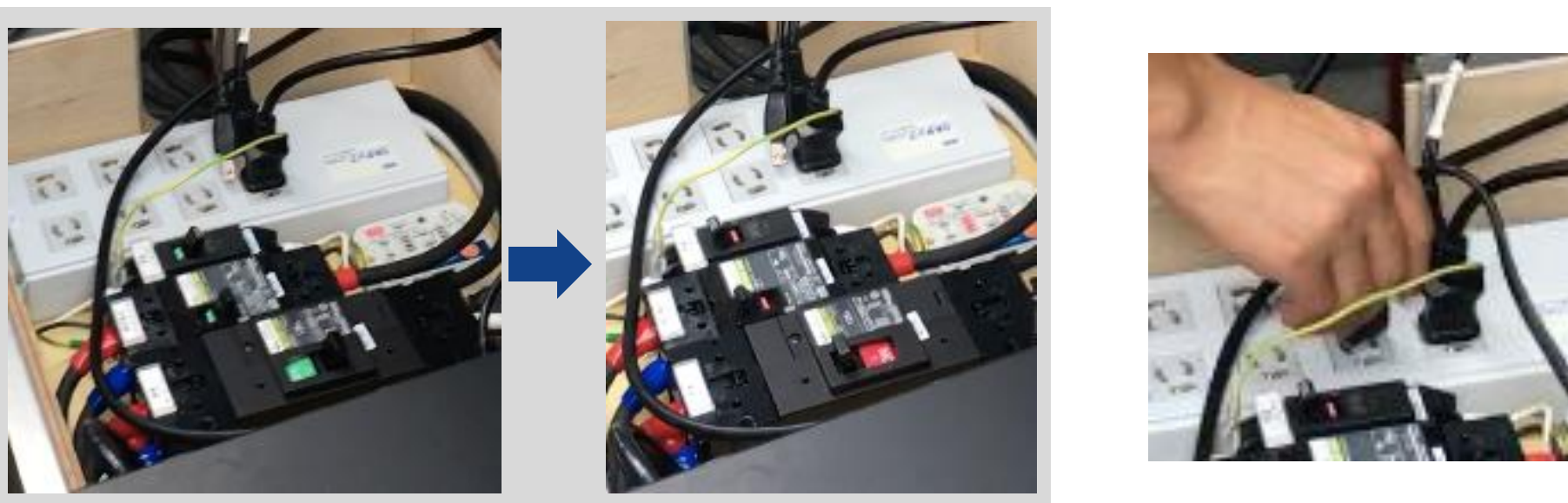
RoboCar®はZMPの登録商標です。

## 1. 走行準備

- 1.1. 実験車の準備
- 1.2. Autowareの起動
- 1.3. 地図と座標変換ファイルの読み込み
- 1.4. RVizの起動
- 1.5. キャリブレーション準備
- 1.6. キャリブレーション

# 1. 走行準備

## 1.1. 実験車の準備



1. 実験車 (RoboCar® MiniVan)で各種スイッチを ONにする。

- インバータ
- サブバッテリー
- タップ

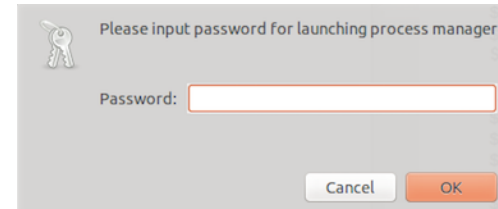
2. 実験車のエンジンをかける。

3. Velodyne のコンセントを挿す。

# 1. 走行準備

## 1.2. Autoware の起動

```
ando@ando-Diginnos-PC:~  
$ cd ~/Autoware/ros/  
ando@ando-Diginnos-PC:~/Autoware/ros  
$ ./run
```



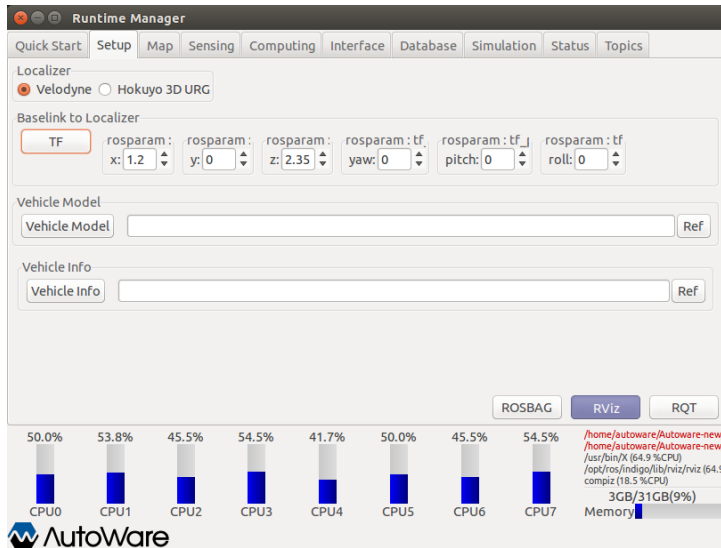
1. ROS PC を立ち上げる。
2. Autoware を起動する。
  - 2.1 端末で以下のように run を実行する。

```
$ cd ~/Autoware/ros/  
$ ./run
```

    - ✓ あるいはファイルマネージャから run を実行する。
3. ログインパスワードを入力し、[OK]ボタンをクリックする。
4. Runtime Manager が表示される。

# 1. 走行準備

## 1.3. 座標変換と地図の読み込み (1/2)



- ✓ ROS のターゲットであるロボットは複数のモジュールで構成されており、各モジュールごとに定義された座標系を相対座標系として計算する必要がある。
- ✓ 相対座標系を計算するために座標変換をする。
- ✓ TFは、車両の制御位置とVelodyneの取付位置の関係を表す。

### 1. 車両の制御位置から見たVelodyneの取付位置を設定する。

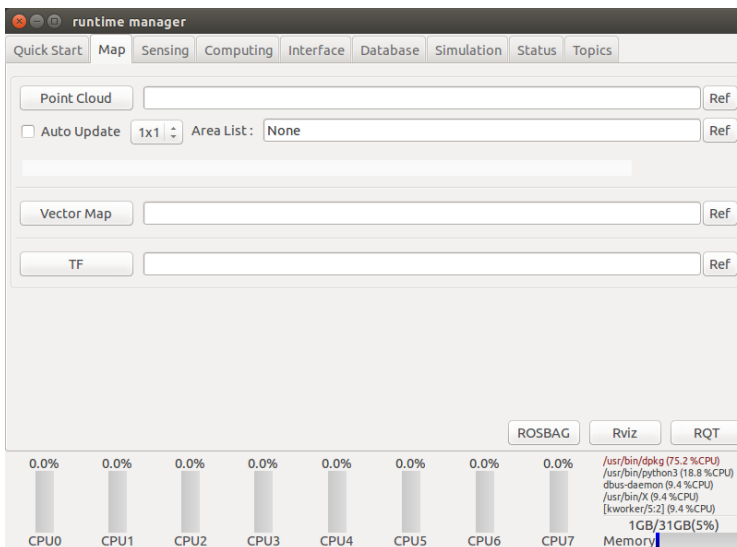
1.1 Runtime Manager の [Setup] タブで、[TF] にある[x][y][z][roll][pitch]および[yaw]の値を入力し、[TF]ボタンをクリックする。

### 2. 自車モデルを読み込む。

2.1 [Vehicle Model] ボタンをクリックする。

# 1. 走行準備

## 1.3. 座標変換と地図の読み込み (2/2)



### 1. 3次元点群地図を読み込む。

1.1 Runtime Manager の [Map] タブで [Point Cloud] の [Ref] ボタンをクリックし、3次元点群地図を指定して [Point Cloud] ボタンをクリックして読み込む。サンプルの場合、以下にあるファイルを全て指定する。

`~/autoware/data/map/pointcloud_map/`

### 2. ADASマップを読み込む。

2.1 [Vector Map] の [Ref] ボタンをクリックし、ADASマップを指定して [Vector Map] ボタンをクリックして読み込む。サンプルの場合、以下にあるファイルを全て指定する。

`~/autoware/data/map/vector_map/`

### 3. 座標変換ファイルを読み込む。

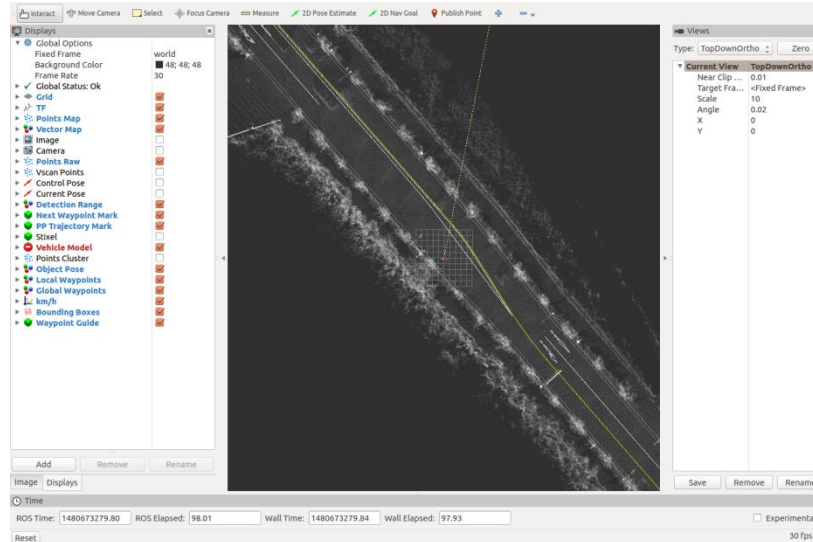
3.1 [TF] の [Ref] ボタンをクリックし、TFの launch ファイルを指定して [TF] ボタンをクリックして読み込む。サンプルの場合、以下のファイルを指定する。

`~/autoware/data/tf/tf.launch`



# 1. 走行準備

## 1.4. RViz の起動



### 1. RVizを起動する。

1.1 Runtime Manager の [RViz]ボタンをクリックする。

### 2. RVizのconfig ファイルを読み込む。

2.1 RVizのメニュー [File] - [Open Config] を選択し、以下のファイルを選択して [Open] ボタンをクリックする。

~/Autoware/ros/src/config/rviz/default.rviz

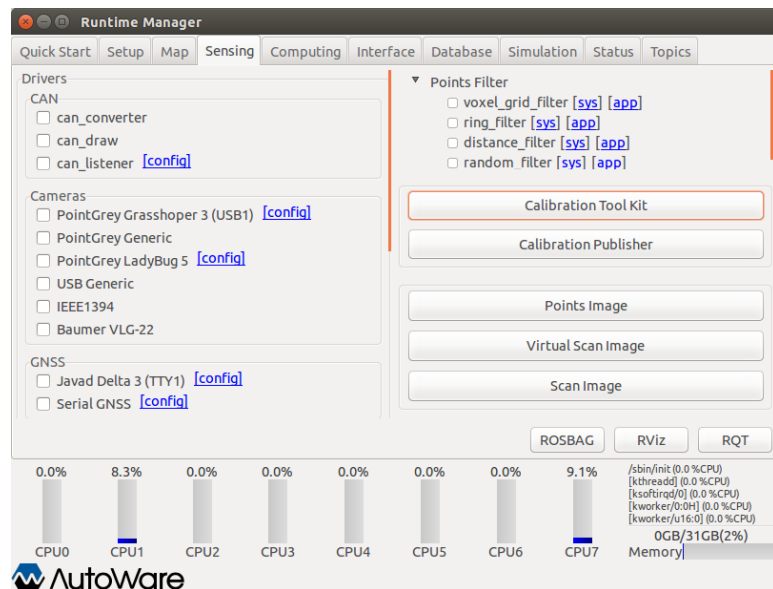
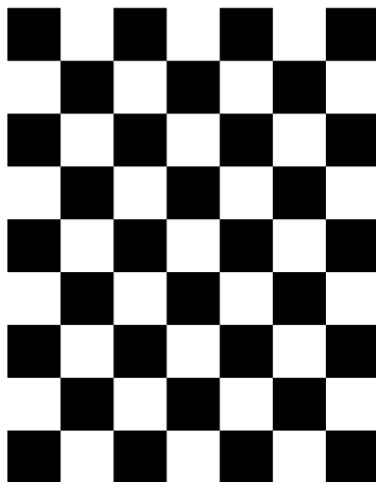
### 3. 3次元点群地図とADASマップを表示する。

3.1 RVizの左ペインで [Global Options] - [Fixed Frame]を "world" に設定する。

3.2 [Points Map] と [Vector Map]チェックボックスをONにすると3次元点群地図とADASマップが表示される。

✓ 表示されない場合、"Points Map" や "Vector map" のチェックボックスを OFF → ONして再ロードする。

## 1.5. キャリブレーション準備 (1/3)



1. 下記PDFをA0サイズで印刷したチェスボードを用意する。

[http://wiki.ros.org/camera\\_calibration/Tutorials/MonocularCalibration?action=AttachFile&do=view&target=check-108.pdf](http://wiki.ros.org/camera_calibration/Tutorials/MonocularCalibration?action=AttachFile&do=view&target=check-108.pdf)

✓ パネルは外枠なしで厚い方がよく、用紙はマット（光沢なし）が望ましい。

2. カメラのノードを起動する。

2.1 [Sensing] タブの [cameras] の [PointGrey Grasshopper 3 (USB1)] の [config] をクリックし、[Ref]ボタンをクリックして以下のファイルを指定してOpenする。

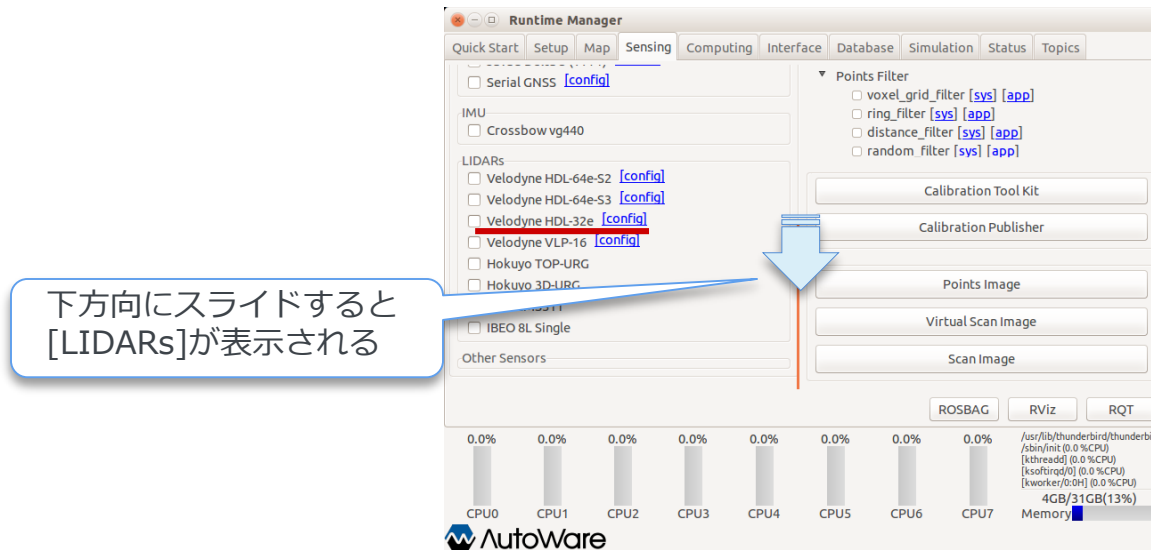
`~/autoware/data/calibration/camera_lidar_3d/prius/nic-150407.yml`

2.2 [PointGrey Grasshopper 3 (USB1)]チェックボックスをONにする。

3. カメラ画像を表示する。

3.1 RVizの左ペインで [Image]チェックボックスをONにするとカメラ画像が表示される。

## 1.5. キャリブレーション準備 (2/3)



### 4. LIDARのノードを起動する。

4.1 [Sensing] タブの [LIDARs] の [Velodyne HDL-32e] の [config] をクリックし、[Ref] ボタンをクリックして、以下を指定してOpenする。

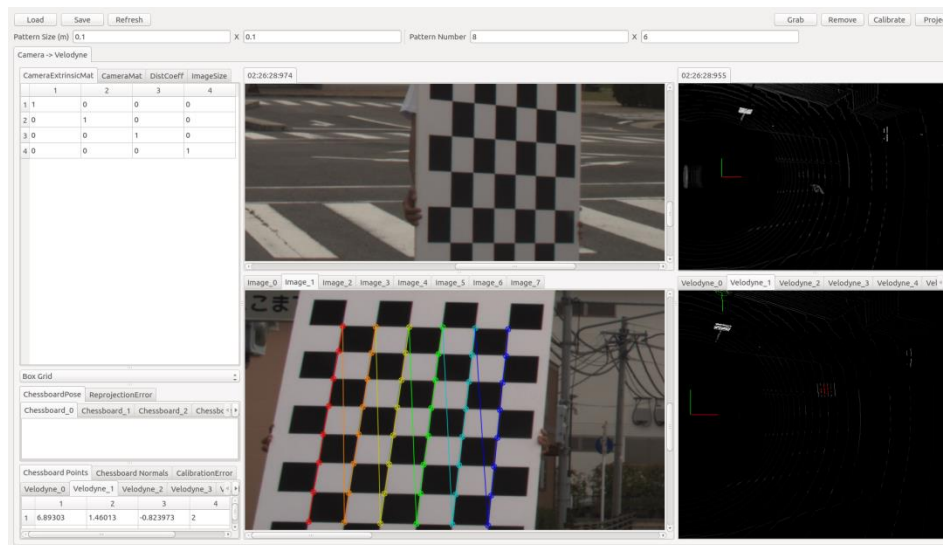
`~/Autoware/ros/src/sensing/drivers/lidar/packages/velodyne/velodyne_pointcloud/params/32db.yaml`

4.2 [Velodyne HDL-32e]チェックボックスをONにする。

### 5. LIDARのデータを確認する。

5.1 RVizの [Global Options] - [Fixed Frame] を [velodyne] に設定し、[Points Raw]チェックボックスをONにすると、LIDARのデータを確認できる。

## 1.5. キャリブレーション準備 (3/3)



### 6. rosbagの記録を開始する。

6.1 [ROSBAG]ボタンをクリックし、[Refresh]ボタンをクリックして、[/image\_raw] および [/points\_raw] チェックボックスをONにする。

6.2 記録するrosbagのファイル名を入力した後、[Start]ボタンをクリックして記録を開始する。

### 7. チェスボードを持っていくつかのパターンを撮影する。

(車両の近く, 遠く) x (車両の中央, 左, 右) x (下, 上) x (正面, 左, 右, 上向き, 下向き)であれば60パターンになる。

### 8. rosbagの記録を終了する。

[Stop]ボタンをクリックする。

## 1.6. キャリブレーション (1/2)

### 1. チェスボードに合わせて、[Pattern Size(m)]と[Pattern Number]を指定する。

#### 1.1 Pattern Numberは内側の格子数を指定する。

- ✓ check-108.pdfであれば変更不要。
- ✓ 格子数を変更した場合は、一度 Calibration Toolkit を終了して再度起動する。

### 2. チェスボードの点群が表示されるようにする。

#### 2.1 ウィンドウ右のVelodyneの枠内（黒画面）をクリック後、以下のキーを操作する。

- ✓ 移動: カーソルキー, PgUp, PgDn
- ✓ 回転: a, d, w, s, q, e
- ✓ 点のサイズ: o, p
- ✓ 背景色: b
- 参考: [~/Autoware/ros/src/sensing/fusion/packages/calibration\\_camera\\_lidar/CalibrationToolkit\\_Manual.pdf](#)

### 3. 各パターン毎に、[Grab]ボタンをクリックする。

- ✓ ウィンドウの下に[Image\_\*]タブおよび[Velodyne\_\*]タブが追加される。
- ✓ 十分な点群が得られない場合は追加されないため、別の近いパターンの取得を試みる。

### 4. 各[Velodyne\_\*]タブ毎に、チェスボードの点群をクリックする。

- ✓ チェスボードの部分が赤色でマークされる。
- ✓ 緑色の線は、チェスボードの平面に対する平行線と法線となる。
- ✓ 右クリックでやり直すことができる。

## 1.6. キャリブレーション (2/2)

### 5. 4 x 4 のマトリックスを生成する。

5.1 [Calibrate]ボタンをクリックする。

### 6. カメラ画像に点群を投影する。

6.1 [Project]ボタンをクリックする。

6.2 各「Velodyne\_\*」タブ毎に、投影された点群を確認する。

✓ 位置がずれている場合は、右クリックして再度クリックする。

6.3 [Calibrate]ボタンをクリックして再計算し、[Project]ボタンをクリックして再確認する。

### 7. マトリックスを保存する。

7.1 [Save]ボタンをクリックする。

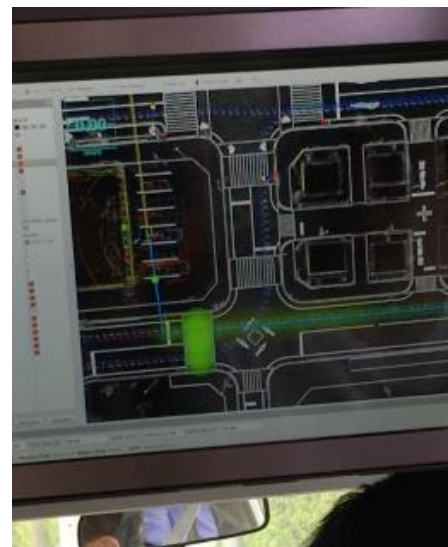
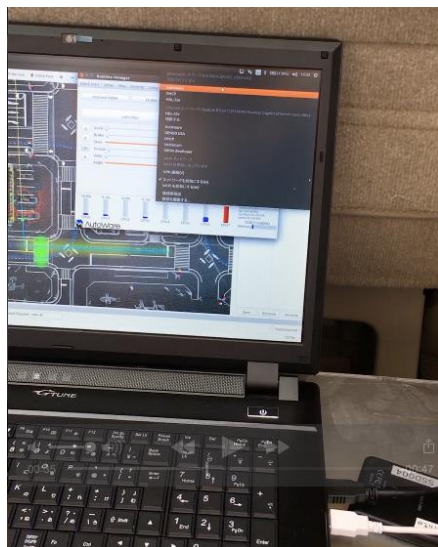
7.2 “Save Camera Calibration Data?” および “Save Velodyne Calibration Data?” には[No]を選択する。

## 2. 経路生成

2.1. 車両制御用 PC との接続

2.2. 実走行による経路生成

### 2.1. 車両制御用 PC との接続



1. 運転席に人が着座していることを確認する。

✓ 急にプログラムが動作することもあるので、制御を入れるときは必ず運転席に人が着座していること。

2. ROS PC でネットワーク接続を開始する。

2.1 [Interface]タブの[Vehicle gateway]ボタンをクリックして起動する。

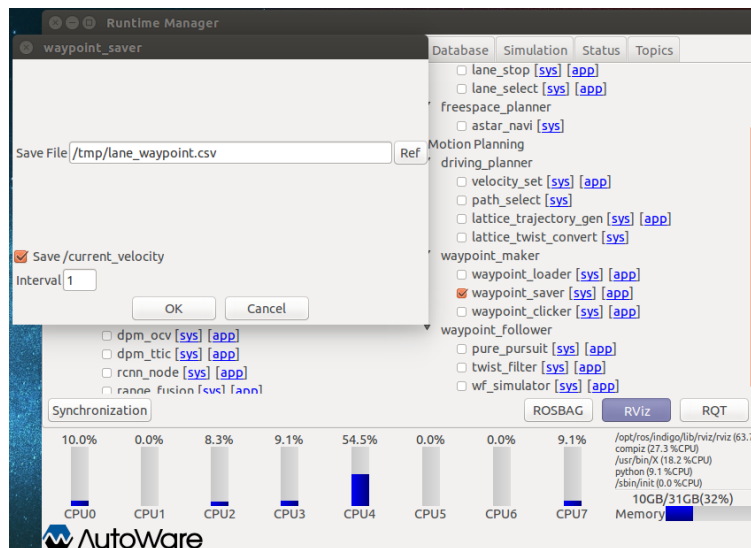
3. 車両制御用 PC (RoboCar®コントロールPC) を起動し、車両制御アプリケーションを起動する。

4. 車両制御アプリケーションのログを起動する。

5. RViz の画面に速度が表示されることを確認する。



## 2.2. 実走行による経路生成 (1/2)



- ✓ Vector Mapからも経路を生成できるが、Vector Mapが無い場合、実際に走行した時のセンサーのログデータを用いて経路を生成することができる。
- ✓ 経路は 座標・方向・速度を持つデータ列として csvファイルに出力される。

### 1. 保存するファイル名と保存場所を指定する (.csv)。

1.1 [app]をクリックして「waypoint\_saver」ウィンドウを立ち上げ、[Ref]ボタンをクリックしてファイル名と場所を指定する。

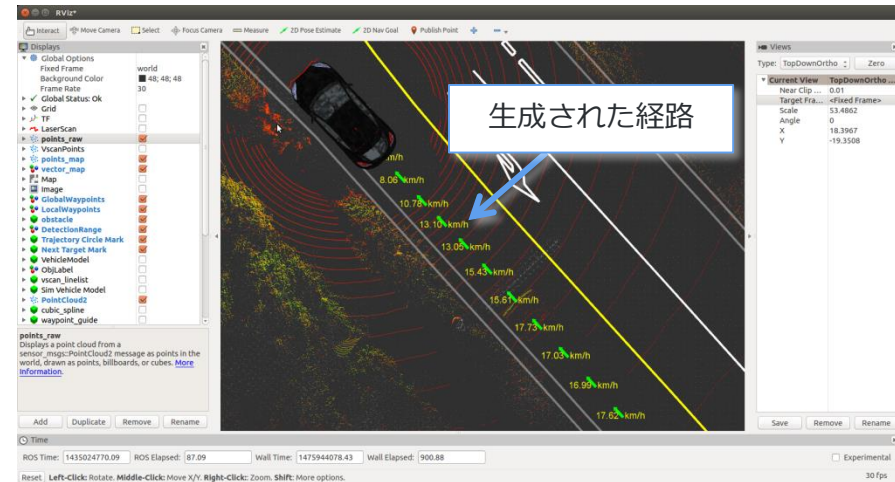
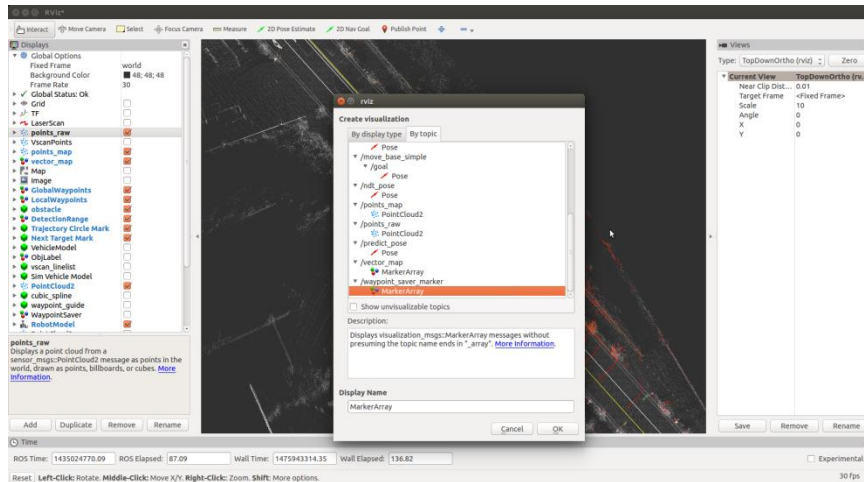
1.2 [save current velocity]チェックボックスをONにすると走行時の速度が記録される。

- ✓ チェックをOFFにするとwaypointに速度が0km/hと記録されるので後から手動による設定が必要になる。

### 2. 経路生成を開始する。

2.1 [Computing]タブの[waypoint\_maker]の [waypoint\_saver]チェックボックスをONにしてwaypoint saverを起動する。

## 2.2. 実走行による経路生成 (2/2)



## 3. RViz で waypoint を記録しているマーカーを表示する。

3.1 RViz の左のトピックリストの下にある[Add]ボタンをクリックする。

3.2 表示されたウィンドウの [By Topic]タブから[/waypoint\_saver\_marker] の「MarkerArray」を選択し、クリックする。

✓ RVizの車両の位置に0km/hと表示される。

## 4. 実験車を手動で運転する。

## 5. 実験者の手動運転を終了する。

## 6. 経路生成 を終了する。

6.1 [waypoint\_saver] チェックボックスをOFFにする。

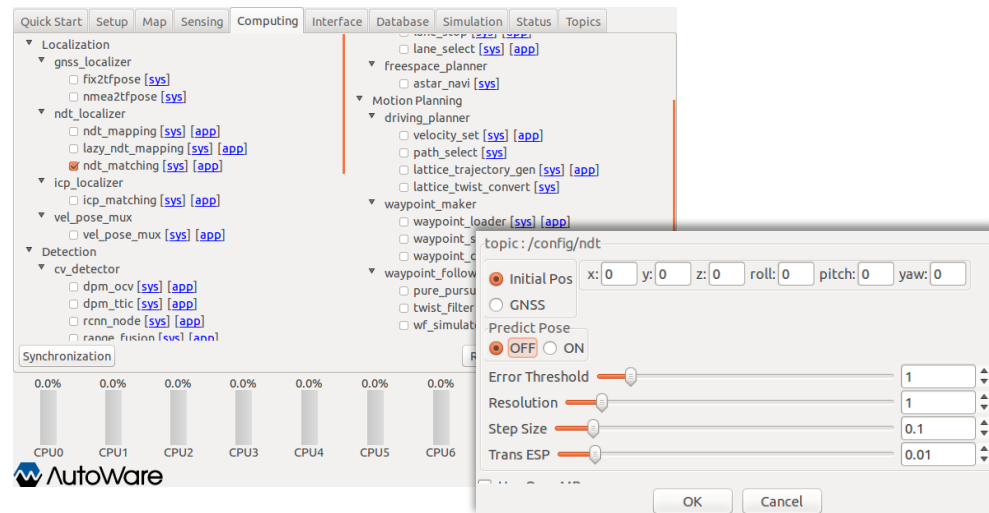
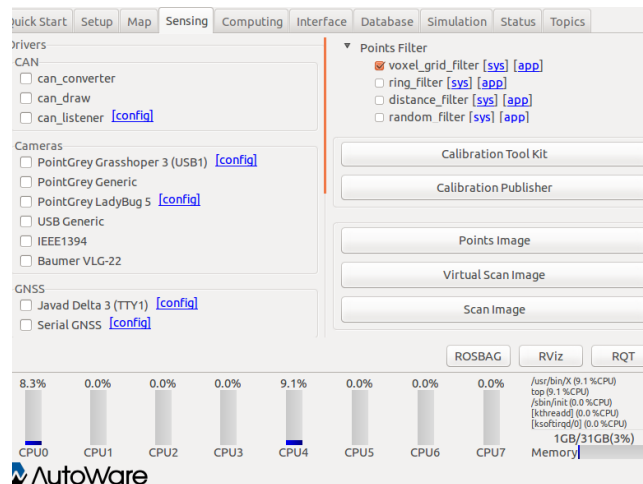
6.2 指定したディレクトリに経路ファイルが保存される。

## 3. 自動走行

- 3.1. 自車位置推定の設定
- 3.2. 自車位置の設定
- 3.3. 経路の読み込み
- 3.4. 経路計画および経路追従と車両位置制御の設定
- 3.5. 走行開始
- 3.6. 経路追従時の信号での停止・走行

# 3. 自動走行

## 3.1. 自転車位置推定の設定



### 1. スキャンデータのダウンサンプリングを開始する。

1.1 [Sensing]タブの[Points Downsampler] - [Voxel\_grid\_filter] チェックボックスをONにする。

### 2. NDT (Normal Distributions Transform) の設定をする。

2.1 [Computing]タブの[Localization] - [ndt\_localizer] - [ndt\_matching] の [app] リンクをクリックして「ndt」ウィンドウを立ち上げる。

- ✓ [Initial Pos] ラジオボタンをONにする。
- ✓ [Predict Pose] が OFF になっていることを確認する。

2.2 「ndt」ウィンドウを [OK]ボタンをクリックして閉じる。

### 3. NDT を開始する。

3.1 [ndt\_matching] チェックボックスをONにする。

### 3.2. 自車位置の設定



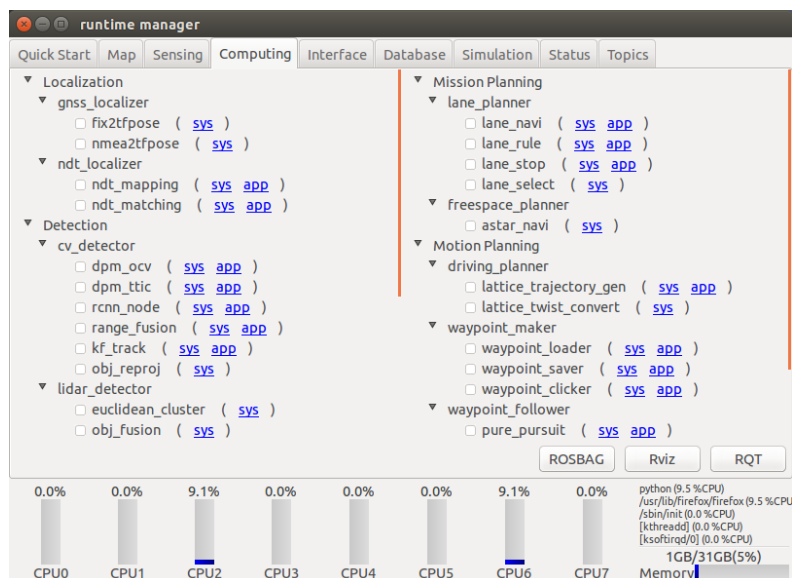
#### 1. RVizで自車位置を設定する。

1.1 [2D Pose Estimate]ボタンをクリックし、自車の現在位置と方向をマウスで指定する。

1.2 点群や自車のモデルが表示される。

- ✓ ボタンをクリックした時にモデルが飛んでいってしまうときは位置が推定できていない。
- ✓ ほぼ正確な位置を与えていれば、モデルは飛んでいかない。
- ✓ RVizを3D表示にして([ThirdPersonFollower]にして)、高さが合っているか、念のため確認する。
- ✓ 止まってもずれていることがあるため、一度車両を動かして推定できているのかを見た方がよい。
- ✓ 車両は50mくらい動かすとよい。

## 3.3. 経路の読み込み



## 1. 経路を指定する。

1.1 [Computing]タブの[Motion Planning] - [waypoint\_maker] - [waypoint\_loader] - [app] をクリックして「waypoint\_loader」ウィンドウを立ち上げる。

1.2 “Driving”と “Passing”をそれぞれロードして[OK]ボタンをクリックしてウィンドウを閉じる。

## 2. 指定した経路を読み込む。

2.1 [waypoint\_maker] - [waypoint\_loader] のチェックボックスをONにする。

## 3. RVizで経路が表示される。

3.1 読み込まれた経路が青い点で表示される。

### 3.4. 経路計画および経路追従と車両位置制御の設定

#### 1. 経路計画の設定

##### 1.1 [Mission Planning] - [lane\_planner]

- [lane\_rule] チェックボックスをONにする。
- [lane\_select] チェックボックスをONにする。
- [lane\_select] - [app]リンクをクリックして「lane\_select」ウィンドウを立ち上げて、[Driving] ラジオボタンが選択されていることを確認する。

#### 2. 経路追従の設定

##### 2.1 [Motion Planning] - [driving\_plannner]

- [velocity\_set] チェックボックスをONにする。
- [path\_select] チェックボックスをONにする。

##### 2.2 [Motion Planning] - [waypoint\_follower]

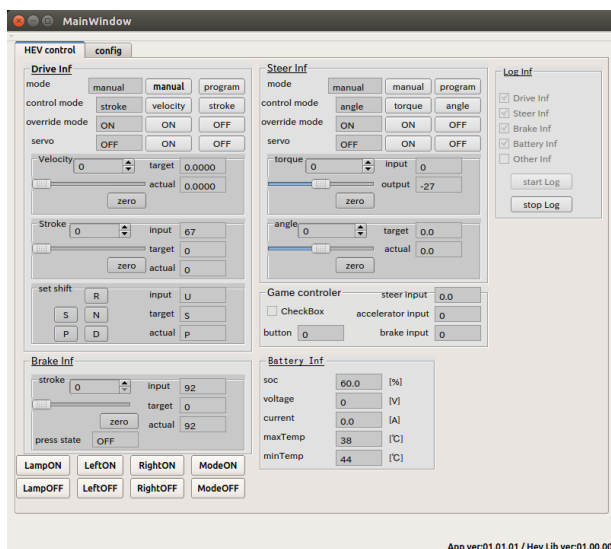
- [pure\_pursuit] - [app]リンクをクリックして「waypoint\_follower」ウィンドウを立ち上げて、[Linear\_Interpolate] チェックボックスがONになっていることを確認する。
- [pure\_pursuit] チェックボックスをONにする。
- [twist\_filter] - [app]リンクをクリックして「twist\_filter」ウィンドウを立ち上げて、[lateral\_accell Limit] が "5"になっていることを確認する。
- [twist\_filter] チェックボックスをONにする。

#### 3. 車両の位置制御の設定

##### 3.1 NDTの後「Localization」 - [vel\_pose\_mux] - [vel\_pose\_mux]チェックボックスをONにする。



## 3.5. 走行開始



## 1. 車両を開始位置に移動する。

- 1.1 フットブレーキを踏みこむ。
- 1.2 車両のギアを [D]（ドライブ）にする。

## 2. モードを切り替える。

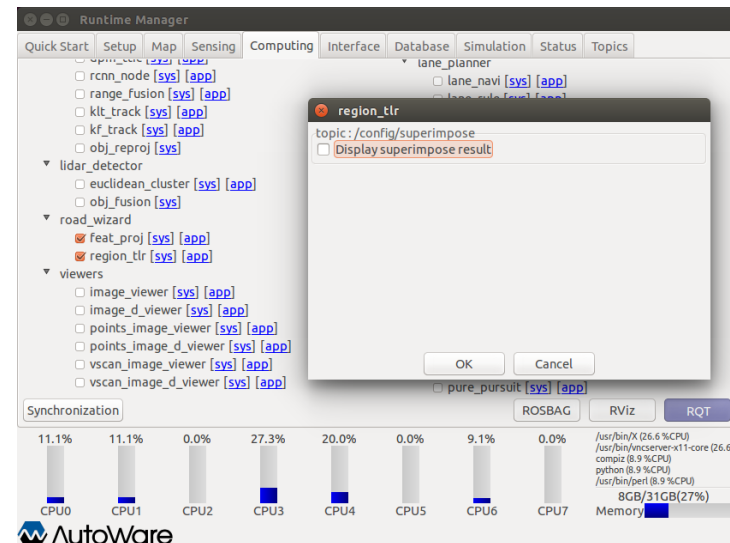
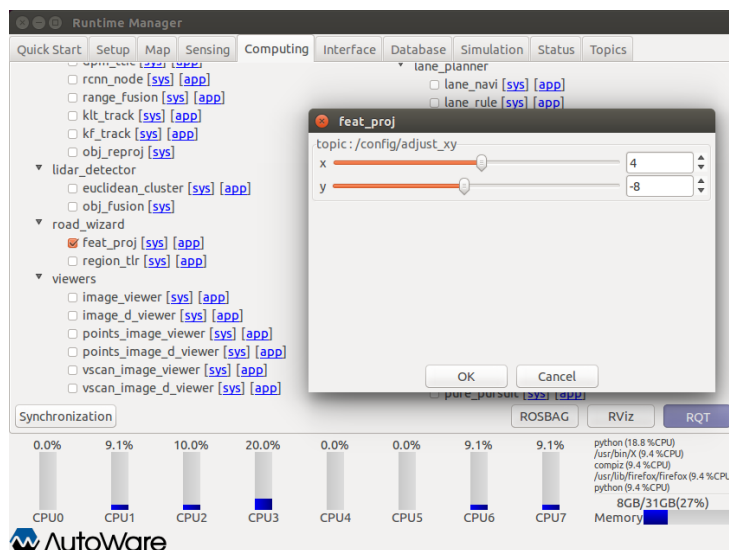
- 2.1 車両制御用 PC で マニュアルモードからプログラムモードへ切り替える。
  - ✓ モード切替により、Autowareからのデータ（目標速度・目標位置）で実験車が走行する。

## 3. 走行開始

- 3.1 フットブレーキを解除すると動き出す。
- 3.2 RVizでは車両の向かう目標点となる、黄緑の丸が表示される。
  - ✓ 最初は車両の現在位置と異なる場所に示されていて、経路に乗ると表示と位置が合致して移動していく。



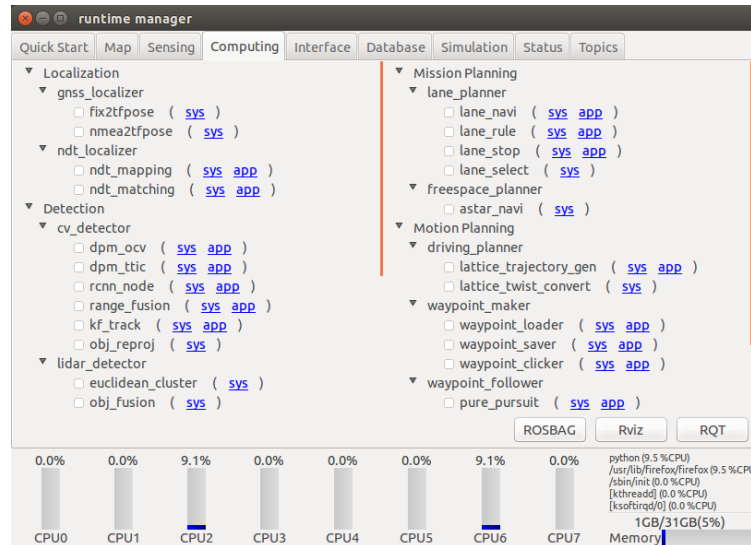
## 3.6. 経路追従時の信号での停止・走行（1/2）



## 1. 信号機の場合および色を認識する設定を行う。

- 1.1 [Computing]タブの[Detection]-[road\_wizard]で、[feat\_proj]および[region\_tlr]チェックボックスをONにする。
- 1.2 [region\_tlr]の[app]をクリックして、[superimpose result]ウィンドウを立ち上げる。
- 1.3 「Display superimpose result」を チェックする。
  - ✓ 信号の位置に丸が表示されるので、実際の信号の位置とずれがある場合は、[feat\_proj]の[app]をクリックし、[x]と[y]のスライダーを操作して、位置を合わせる。
  - ✓ [Interface]タブの[Sound]チェックボックスをONにすると、信号の状態を音声で知らせる。

## 3.6. 経路追従時の信号での停止・走行 (2/2)



## 2. 赤信号で車両を停止する設定を行う。

2.1 [Computing]タブの[lane\_stop]をチェックする。

2.2 [lane\_stop]の[app]をクリックし、[Use traffic light recognition result] をチェックする。

- ✓ 経路追従が動作している状態を前提とする。
- ✓ これにより、信号認識結果が赤信号の際に停止するようになる。

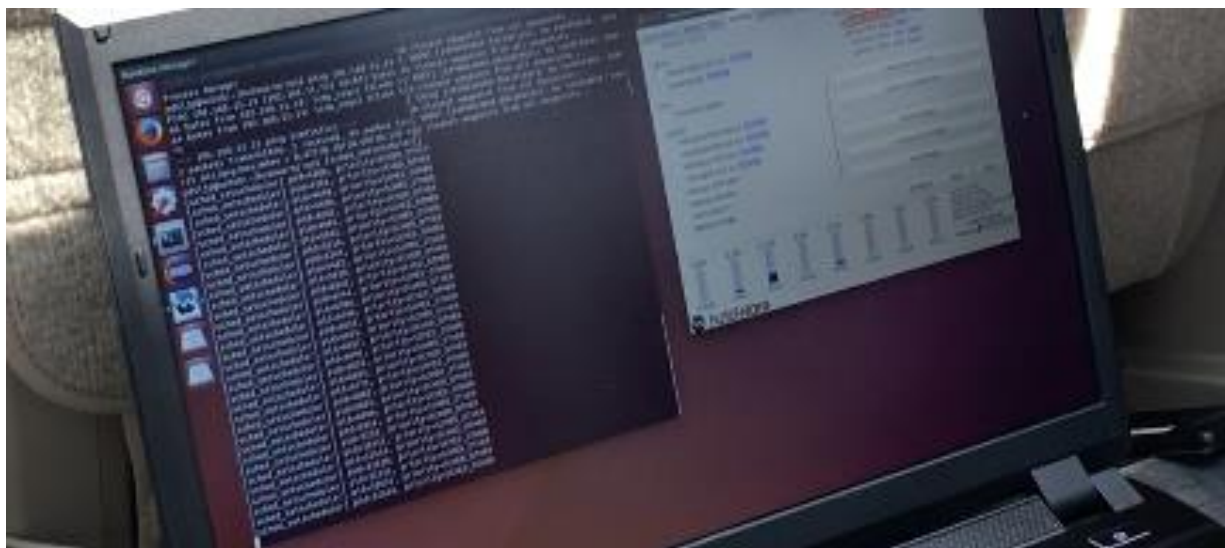
## 4. 終了作業

4.1. 各 PC の終了

4.2. 車両および Velodyneの終了

## 4. 終了作業

### 4.1. 各PC のシャットダウン



---

#### 1. 車両制御用PC をシャットダウンする。

##### 1.1 車両制御用アプリケーションを終了する。

- ✓ 起動時と同様に、必ず運転席に人が着座している状態で行う。

#### 2. ROS PCをシャットダウンする。

##### 2.1 RVizを終了させる。

##### 2.2 Runtime ManagerでチェックボックスがONになっている全て項目のチェックをOFFにする。

##### 2.3 Runtime Manager の 左上の x ボタンをクリックして終了させる。

- ✓ これによって動いているプロセスは全て終了する。
- ✓ チェックボックスをOFFにしなくても終了できるが、OFFにしてから終了させる方が安定する。

## 4. 終了作業

### 4.2. 車両および Velodyneの終了



1. Velodyne の電源をコンセントから抜く。
2. 車のエンジンを切る。
3. スイッチを切る
  - 「タップ」の電源をOFFにする。
  - 「サブバッテリー」の電源をOFFにする。
  - 「インバータ」の電源をOFFにする。

## Appendix

- RoboCar® MiniVanの概要

## RoboCar® MiniVanの概要 1

「RoboCar® MiniVan」は、市販ハイブリッドミニバンをベース車両とし、ユーザが作成したプログラムから「走る」「曲がる」「止まる」を制御することが可能な実験用車両です。

カメラやレーザセンサ、GPS、IMUなどを搭載し、プログラムを開発することにより、自動運転技術の研究開発等にご利用頂けます。

車両仕様概要	
項目	仕様
種別	普通自動車
定員	7名
全長	4,800 [mm]
全幅	1,820 [mm]
全高	1,760 [mm]
車両重量	1,970 [kg]
ホイールベース	2,950 [mm]
トレッド F/R	1,545/1,550 [mm]
駆動方式	4WD
トランスミッション	電気式無段変速機

## RoboCar® MiniVanの概要 2

### 1.特徴

- ✓ ソフトウェア開発環境（Linux搭載コントローラ付属、ライブラリ/ドキュメントなど）が付属
- ✓ ユーザプログラムを実行可能
- ✓ CANインターフェイス/CANプロトコルを公開
- ✓ CAN情報取得が可能（速度、ステアリング、アクセル、ブレーキ、シフトポジションなど）
- ✓ オプションでステレオカメラやレーザレンジセンサ等の外界センサを搭載可能
- ✓ ステアリング、アクセル、ブレーキを制御可能
- ✓ 自動制御モード（プログラムモード）とマニュアルモードの切り替え可能

### 2.機能

- ✓ 車両性能
  - ベース車両に自動コントロール機能をアドオンする形で提供するもので、車両の仕様、運動性能の諸元はベース車両による
- ✓ CAN情報の取得
  - 車速、バッテリー情報、ドライブモード、アクセル、ブレーキ情報、など
- ✓ ステアリング制御
  - ステアリングホイールのトルク入力、出力
  - 角度制御
- ✓ アクセル、ブレーキ制御
  - アクセルペダル、ブレーキペダル入力、出力
  - 速度制御
  - シフトレバー制御
- ✓ UI



## RoboCar® MiniVanの概要 3

### 1. 免責事項

RoboCar®の使用は、研究用途に限られます。一般の旅客、輸送には使用できません。

RoboCar®を利用する場合には、RoboCar®について知識のある者が搭乗し、その者の監督の下安全を確認の上、利用してください。

搭乗の際は、安全装備(ヘルメット・グローブなど)を着用し、シートベルトを必ず締めてください。

弊社にて保証する範囲はRoboCar®付属の仕様書記載の機能および性能のみです。

RoboCar®を走行させる場所は、一般の人が立ち入らない制限された敷地を用意し、その敷地内のみで走行させてください。

RoboCar®と、RoboCar®の一部もしくは搭載の技術は、人命に関わるような用途(例えば、医療、航空、宇宙などきわめて高い信頼性が要求される)でのご使用はできません。

### 2. 問合せ

ZMP RoboCar®シリーズについての問合せは下記まで。

株式会社ZMP 営業部 (Tel:03-5802-6901 / Mail:info@zmp.co.jp)



**Intelligent Vehicle**

[www.tier4.jp](http://www.tier4.jp)