Autoware User's Manual

2016 / JUN / 8

Nagoya University

Content

Introduction

Overview

This document was based on Linux and ROS (Robot OS), open for realizing automatic operation

Nsosu of the software package is the user's manual of "Autoware".

And Autoware, using a variety of sensor devices or data, of automatic operation or a part thereof

It describes the procedure to operate the function.

the term

- ROS (Robot Operating System)

Software framework for robot software development. hardware

Abstraction and low-level device control, commonly used implementation of the function, inter-process communication, package

It provides the functions such as the management.

- package (Package)

  Units of software to form the ROS. Node and library, a configuration file

Including the etc..

- node (Node)

  Process to provide a single function.

- Message (Message)

  Data structure when the node to each other to communicate.

- Topic (Topic)

  To which you want to send and receive messages. Sending the message "Publish" receives "Subscribe"

And call.

- OpenCV (Open source Computer Vision library)
  Image processing library for handling computer vision.
- Qt
  Application user interface framework.
- CUDA (Compute Unified Device Architecture)
  NVIDIA-supplied, general-purpose computing platform and programming using the GPU

model.

- FlyCapture SDK
  SDK to control the PointGrey's camera.
- FOT (Field Operation Test)
  Real road experiment.

- GNSS (Global Navigation Satellite System)
  Satellite positioning system.
- Ladybug SDK
  SDK to control the PointGrey's camera "Ladybug".
- LIDAR (Light Detection and Ranging or Laser Imaging Detection and Ranging)

  Device for measuring distance or the like using laser irradiation.

- DPM (Deformable Part Model)
  Object detection technique.
- KF (Kalman Filter)
  Method of estimating the future of the state on the basis of past observations.
- KLT (Kanade-Lucas-Tomasi feature tracker)
  Technique for extracting tracking feature points.
- NDT (Normal Distributions Transform)
  Position estimation technique.
- calibration

  For adjusting the position in the point and the 3-dimensional space projected on the camera, the camera parameters

The process of obtaining the over data.

- Sensor Fusion

  By combining a plurality of sensor information, such as to more accurately calculate the position and posture, advanced certification

Method to realize the identification function.

- TF (TransForm?)
  ROS coordinate conversion library?
- odometry (Odometry)

  Method for estimating the position by integrating the rotational angular velocity and the rotation angle of the wheel.
- SLAM (Simultaneous Localization and Mapping)

  It is carried out self-position estimation and environmental map created at the same time.
- CAN (Controller Area Network)

  Standards within the automobiles are used to transfer data between interconnected devices.
- IMU (Inertial Measurement Unit)

  Inertial measurement unit. Device for measuring the angular velocity and acceleration.
- DMI (Distance Measuring Instrument)

  Odometer.

related document

- Autoware

  http://www.pdsl.jp/fot/autoware/
- ROS

  http://www.ros.org/
- OpenCV

  http://opencv.org/
  http://opencv.jp/

- Qt

  http://www.qt.io/
  http://qt-users.jp/
- CUDA

  http://www.nvidia.com/object/cuda_home_new.html
  http://www.nvidia.co.jp/object/cuda-jp.html
- FlyCapture SDK

  https://www.ptgrey.com/flycapture-sdk
- Ladybug SDK

  https://www.ptgrey.com/ladybug-sdk

Contact

Autoware Developers (autoware@googlegroups.com)

Past posts of Autoware Developers mailing list:

https://groups.google.com/d/forum/autoware

How to participate in the Autoware Developers mailing list:

- If you have a Google account,

  https://groups.google.com/d/forum/autoware access to,

  Please click on the "sign up to join the group" button.
- If you do not have a Google account:

  autoware+subscribe@googlegroups.com please send an email to.

Overall structure

Autoware was based on Linux and the ROS, an open source to realize the automatic operation

All of the software package. Laser radar, cameras, environmental sensors, such as GNSS use

To, while recognizing the vehicle position and surrounding objects, autonomous traveling route on given from the car navigation

can.

Constitution

Function of automatic operation by Autoware, carried out such as self-position estimation and the surrounding objects of detection "cognitive",

Of running and stopping at the lane and intersection "judgment", "operation" of the actual vehicle, divided into three Rarema

It is.

- ros / src / computing / perception /
  Cognition and judgment
- ros / src / computing / planning /
  Judgment and operation
- ros / src / data /
  Reading of data such as a three-dimensional map (DB, file)
- ros / src / sensing /

Various sensors driver, calibration, fusion, etc.

- ros / src / socket /

  Interface with the smart phone for the application

- ros / src / util /

Runtime Manager, sample data, the pseudo-drivers, etc.

- ui / tablet /

  Smartphone application

- vehicle /

  Control of the vehicle, information acquisition, etc.

Main function

The Autoware has the following functions.

The user interface for performing these (Runtime Manager) also are available

You.

- self-position estimation

  As input a three-dimensional point cloud map and the three-dimensional LIDAR data, and based on the NDT algorithm
By performing the scan matching, the vehicle position can be estimated with an error of about 10cm
to come.

- 3-dimensional map generation

  Using SLAM technology, it is possible to generate a three-dimensional map in real time.

  The resulting three-dimensional map, it is also possible to add to the existing three-dimensional map. This feature
By, you can also achieve online update of the three-dimensional map.

  From three-dimensional map to extract the feature data, generates a three-dimensional map of vector format
You can also.

- signal detection

  The results and a high-precision three-dimensional map of the self-position estimation, and accurately calculate the position of the traffic signal, Shi
And projected onto the camera image by the sensor fusion of the three-dimensional position of the Unit. View from there
By color determined by the image processing, it is possible to detect the signal.

- object detection

  The camera image as an input, by performing the image recognition by DPM algorithm, vehicle

You can detect and pedestrians.

  It is also possible to perform the tracking using the KF and KLT. Guiding the tracking function

If you type, you can reduce the can track the individual object, and false recognition.

  Further, by fusion of the 3-dimensional LIDAR data, distance to the detected object

You can also calculate.

- path generation

The path of the automatic operation, use the smartphone of car navigation system application (MapFan You can enter from the route data generation application). Also included appropriate speed information to the route It is, and self-running its speed as a guide.

- path following

The generated route set mark of 1m intervals (way point), to go chasing the mark In make the path tracking. Referring to the way point of far in the vicinity of the way point, in a straight line curve Doing, we stabilize the self-traveling.

If you deviate from the path, and return to the path with the aim of way point in the vicinity.

Procedure of Environment

On your PC, the following procedure lists the steps to install Linux, ROS, Autoware the like.

CUDA, FlyCapture SDK and canlib is not mandatory.

If you do the calculation using the GPU, which is mounted on NVIDIA's graphics board, CUDA is必
Is required. Also, if you use the PointGrey's camera, you need a FlyCapture SDK.

Linux

At the moment, Linux distributions Autoware is compatible are as follows.

- Ubuntu 14.04
- Ubuntu 15.04

For the installation media and installation instructions, go to the following site to reference

Please.

- Ubuntu Japanese Team
  https://www.ubuntulinux.jp/
- Ubuntu
  http://www.ubuntu.com/

ROS

- In the case of Ubuntu14.04, install the ROS and the required packages in the following procedure

  And Le.

  ```
  $ Sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu trusty main"> \
    /etc/apt/sources.list.d/ros-latest.list '
  $ Wget http://packages.ros.org/ros.key -O - | sudo apt -key add -
  $ Sudo apt-get update
  $ Sudo apt-get install ros- indigo-desktop-full ros-indigo-nmea-msgs \
    ros-indigo-nmea-navsat- driver ros-indigo-sound-play
  $ Sudo apt-get install libnlopt- dev freeglut3-dev qtbase5-dev libqt5opengl5-dev \
    libssh2-1-dev libarmadillo-dev libpcap- dev gksu
  ```

- In the case of Ubuntu15.04, install the ROS and the required packages in the following procedure

  And Le.

  ```
  $ Sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $ (lsb_release -sc) main"
  > \
    /etc/apt/sources.list.d/ros-latest.list '
  $ Sudo apt-key adv --keyserver hkp : //pool.sks-keyservers.net: 80 \
    --recv-key 0xB01FA116
  $ Sudo apt-get install ros- jade-desktop-full ros-jade-nmea-msgs \
  ```



  ```
    ros-jade-nmea-navsat- driver ros-jade-sound-play
  $ Sudo apt-get install libnlopt- dev freeglut3-dev qt5-default libqt5opengl5-dev \
    libssh2-1-dev libarmadillo-dev libpcap- dev gksu
  ```

- Add the following such as /.bashrc.

  In the case of Ubuntu14.04:

  [-f /opt/ros/indigo/setup.bash] &&. /opt/ros/indigo/setup.bash
  In the case of Ubuntu15.04:

  [-f /opt/ros/jade/setup.bash] &&. /opt/ros/jade/setup.bash

CUDA

Note) If there is no video card compatible with CUDA, this task is not required.

http://docs.nvidia.com/cuda/cuda-getting-started-guide-for-linux/ in reference to, the following procedure
Install.

1. Check the environment
   $ Lspci | grep -i nvidia
   (Check that information of NVIDIA's board is output)

   $ Uname -m
   (make sure that it is a x86_64)

   $ Gcc --version
   (Sure it is installed)

2. Installation of the CUDA

   Http://Developer.Nvidia.Com/cuda-downloads CUDA Download from

   (Hereinafter, assumed cuda-repo-ubuntu1404_7.0-28_amd64.deb)

   $ Sudo dpkg -i cuda-repo- ubuntu1404_7.0-28_amd64.deb
   $ Sudo apt-get update
   $ Sudo apt-get install cuda

3. Restart the system (... it may not be necessary)

   $ Lsmod | grep nouveau
   (check that nouveau driver is not loaded)

4. Checks
   $ Cat / proc / driver / nvidia / version
   (Kernel module, version of gcc is displayed)

   $ Cuda-install-samples-7.0.sh
   $ Cd /NVIDIA_CUDA-7.0_Samples/1_Utilities/deviceQuery/

   $ Make
   $ ./deviceQuery

5. If you use CUDA from the usual, write the following settings, such as in .bashrc
   export PATH = "/ usr / local / cuda: $ PATH"
   export LD_LIBRARY_PATH = "/ usr / local / cuda / lib: $ LD_LIBRARY_PATH"

FlyCapture2

If you want to use the PointGray's camera, install the FlyCapture SDK in the following procedure
To do. (If you do not want to use, this task is not required.)

1. PointGrey's site (http://www.ptgrey.com/ From), downloading the FlyCapture SDK
   And over do. (User registration required.)
2. In the following procedure, pre-install the package.
   $ Sudo apt-get install libglademm- 2.4-1c2a libgtkglextmm-x11-1.2-dev libserial-dev
3. Expand the archive that you downloaded.
   $ Tar xvfz flycapture2-2.6.3.4-amd64-pkg.tgz
4. Start the installer.
   $ Cd flycapture2-2.6.3.4-amd64 /

$ Sudo sh install_flycapture.sh
This is a script to assist with installation of the FlyCapture2 SDK.

Would you like to continue and install all the FlyCapture2 SDK packages?

(y / n) $ y ← answer "y"

...

Preparing to unpack updatorgui-2.6.3.4_amd64.deb ...

Unpacking updatorgui (2.6.3.4) ...

It sets the updatorgui (2.6.3.4) ...

Processing triggers for man-db (2.6.7.1-1ubuntu1) ...

Would you like to add a udev entry to allow access to IEEE-1394 and USB hardware?

If this is not ran then your cameras may be only accessible by running flycap as sudo.

(y / n) $ y ← answer "y"

Autoware

Get the Autoware in the following steps to build and install.

- If you want to get the latest from github

  $ Git clone https://github.com/CPFL/Autoware.git
  $ Cd Autoware / ros / src
  $ Catkin_init_workspace
  $ Cd ../
  $ ./catkin_make_release
  $ Source devel / setup.bash

- If you want to use the archive

  $ Wget http://www.pdsl.jp/app/download/10394444574/Autoware-beta.zip
  $ Unzip Autoware-beta.zip
  $ Cd Autoware-beta / ros / src
  $ Catkin_init_workspace
  $ Cd ../
  $ ./catkin_make_release
  $ Source devel / setup.bash

AutowareRider

Note) Unless you are operating from the Android Tablet, this task is not required.

Get the APK file from the following URL, and to complete the installation.

- body
  - AutowareRider.apk
    https://github.com/CPFL/Autoware/blob/master/ui/tablet/AutowareRider/AutowareRider.apk
- route data generation application

  Note) If you do not want to generate a route data, installation is not required.
  - AutowareRoute.apk

https://github.com/CPFL/Autoware/blob/master/ui/tablet/AutowareRoute/AutowareRoute.apk

- CAN data collection applications

  Note) If you do not want to collect the CAN data, installation is not required.

  - CanDataSender.apk

    https://github.com/CPFL/Autoware/blob/master/vehicle/general/android/CanDataSender / bin / CanDataSender.apk
  - CanGather.apk

    https://github.com/CPFL/Autoware/blob/master/vehicle/general/android/CanGather / apk / CanGather.apk
  - CarLink_CAN-BT_LS.apk

    https://github.com/CPFL/Autoware/blob/master/vehicle/general/android/CarLink / apk / CarLink_CAN-BT_LS.apk

  - CarLink_CANusbAccessory_LS.apk

    https://github.com/CPFL/Autoware/blob/master/vehicle/general/android/CarLink / apk / CarLink_CANusbAccessory_LS.apk

  CanGather other than APK file, you must provide a configuration file.

  For more information, please refer to the following URL to reference.

  https://github.com/CPFL/Autoware/tree/master/vehicle/general/android#cangather-% E3% 81% AE% E5% A0% B4% E5% 90% 88

canlib

kvaser of site (Http: / /Www.Kvaser.Com/downloads/) Of "Kvaser LINUX Driver and SDK"

Get more source code linuxcan.tar.gz, perform the installation in the following procedure.

```
$ Tar xzf linuxcan.tar.gz
$ Cd linuxcan
$ Make
$ Sudo make install
```

Creation of the **SSH** public key

Note) If you do not want to use the dynamic map, this task is not required.
In addition, in the present circumstances only supports server of Nagoya.

pos_db, access to the database through SSH. At that time, without the passphrase
Use the SSH key.

Therefore, if you use the pos_db is, work in the following procedure the SSH key for the database server

Form, you need to register the SSH public key to the database server.

1. SSH key of how to create
   - by running the following command to create a key.
     - $ ssh-keygen -t rsa
   - In this case, the pass phrase is empty (press the Enter key without entering a string)

     Please create on.
   - Please specify the -t dsa if you want to use the DSA.

2. Register the SSH public key to the database server
    - the SSH public key that you created and copy it to the server with the following command.
        - $ ssh-copy-id -i / .ssh / id_rsa.pub posup@db3.ertl.jp

          (posup username, db3.ertl.jp the database server name)
    - Please enter as appropriate will be asked a password to that.

Instructions for use

Preparation

Data set required, /. Are described on the assumption that autoware / data / to some.
Sample data, please download from the following.

Script to generate a launch file for the demo

http://db3.ertl.jp/autoware/sample_data/my_launch.sh
Data used in the demo (such as Moriyama district of map calibration path)

http://db3.ertl.jp/autoware/sample_data/sample_moriyama_data.tar.gz
ROSBAG
http://db3.ertl.jp/autoware/sample_data/sample_moriyama_150324.tar.gz
Note) This order is to ROSBAG does not contain the image information, it can not object detection.

/.autoware/ The data of the downloaded demo to expand below.

$ Tar xfz sample_moriyama_data.tar.gz -C / .autoware /

Run the following script, to run the demo from Qutick Start tab
We would like to generate the launch file.

$ Cd /.autoware/
$ Sh my_launch.sh

When you run, following the launch file is generated.

```
my_launch /
        my_map.launch                   # Road Map
        my_sensing.launch                       Start-up of the # sensor driver
        my_localization.launch          # Self-position estimation
        my_detection.launch             # Object detection
        my_mission_planning.launch      # Path planning
        my_motion_planning.launch       # Path following
```

It should be noted that, if you want to place the data in a location other than the /.autoware/data is, pull the directory

Run the shell script to specify the number.

Example) If data has been placed in a /.autoware/data/quick_start/rosbag_sample/:

$ Sh my_launch.sh /.autoware/data/quick_start/rosbag_sample/

Starting the Runtime Manager

1. Start the Runtime Manager in ROS PC.
   $ Cd Autoware / ros /

   $ ./run

2. Click the "Ref" button in the Simulation tab upper right, a sample of
   Specify the ROSBAG (sample_moriyama_150324.bag), and click the "Play" button

   Once you click the "Pause" button immediately.

   (Done to the True parameter "/ use_sim_time".)

3. Click the "RViz" button at the bottom right of the Runtime Manager, to start the RViz.
4. Select the "Open Config" from the file menu of RViz,
   It specifies the Autoware / ros / src / .config / rviz / default.rviz.

Load of point cloud and vector map

1. Click the "Ref" button of the Map of the Quick Start tab, generated
   Was my_map.launch specify, and then click the "Map" button.
2. it will be displayed as "OK" on the right-hand side of the "Ref" button When you have finished loading the map.

Start-up of the sensor driver

1. Click the "Ref" button of Sensing of the Quick Start tab, generated
   Was my_sensing.launch specify, and then click the "Sensing" button.

Self-position estimation (NDT)

1. Click the "Ref" button of the Localization of the Quick Start tab, it was generated
   Specify the my_localization.launch, and then click the "Localization" button.
2. Click the "Pause" button in the Simulation tab, and then resume execution of the rosbag.

3. On the right side of the screen RViz set in the "Target Frame" and "base_link" of the "Current View"
   Constant, and then click the "Zero" button. (Initially, the Type in "TopDownOrtho"

   Then it is easy to understand.)

4. If there is a map and a marker that does not appear is on the left side of the screen RViz "Displays"
   Please then turn on the respective check of.
5. In ROSBAG of the sample, the first is not self-position is stable because there is no map, parking
   Out of the car park is on until about the first stop will be stable. Still stable

   If it is not, click on the "2D Pose Estimate" at the top of the screen of RViz, actual

Please specify the position and direction of the vehicle.

Object detection

Note) sample in ROSBAG of, because the image information is not included, it can not object detection.

1. Click the "Ref" button in the Detection of Qutick Start tab, it was generated
   Specify the my_detection.launch, and then click the "Detection" button.

2. Start the "image_viewer" of Computing tab.
   The results of object detection is to the left side of the screen, to the right of the screen will display the tracking results.
3. If you make the object detected in the NDT run, on RViz, vehicle blue sphere, pedestrian green sphere
   In will be displayed.

Path planning

1. Click the "Ref" button of Qutick Start tab of the Mission Planning, was generated
   Specify the my_mission_planning.launch, clip the "Mission Planning" button
   And then click.
2. When you run, on RViz, path and speed will be displayed.

Path following

1. Click the "Ref" button of the Qutick Start tab of Motion Planning, generated
   Was to specify the my_motion_planning.launch, click the "Motiion Planning" button
   And Rick.
2. come until the path set by the path planning, blue sphere on the path (the nearest point of the waypoint)
   Etc. will be displayed.

Dynamic map

Note) In the current situation, only it supports server of Nagoya.

The car and the people of the information the vehicle and other vehicles has been recognized, and shared using the database.

1. the state of being delivered following topics.
   All does not mean that essential, if they are delivered hereinafter database the information

Register to over vinegar.

   current_pose (vehicle, ndt_matching delivery)
   obj_car_pose (another car, obj_fusion delivery)
   obj_person_pose (people, obj_fusion delivery)

2. Click the pos_uploader link of the Database tab (on the side in providing the information), de
   Enter the information for SSH access to over database server, and press the OK button or
   It is.
   (Procedure generation of SSH key, located on the steps of the Environment.)

3. Check in (providing at the side of the information) pos_uploader (to start the node).
4. Click on the link (on the side to browse the information) pos_downloader, database server
   Enter the information for accessing SSH into server, and press the OK button.
   (If you check to show my pose, and delivers the position of the vehicle.)

5. Check to pos_downloader (side by viewing the information).
6. (on the side to browse the information) in rviz, when you add a Marker of topics / mo_marker,
   Recognized the vehicle, it will be displayed other companies and people.

Database (VoltDB) "can" already created table in are as follows.

Column name	Type	Overview
id	varchar (32)	MAC address and terminal-specific information, such as
lon	float	Latitude (Android terminal)
lat	float	Longitude (Android terminal)
h	float	unused
x	float	Plane rectangular coordinates x (pos_uploader)
y	float	Plane rectangular coordinates y (pos_uploader)
z	float	Plane rectangular coordinates z (pos_uploader)
area	float	Plane rectangular coordinates of the system number (pos_uploader, 7 fixed)
dir	float	Direction (Android terminal)
acct_x	float	Acceleration x (pos_uploader)
acct_y	float	Acceleration y (pos_uploader)
acct_z	float	Acceleration z (pos_uploader)
vec	float	unused
type	smallint	1 = the vehicle, 2 = recognized the car, 3 = recognized person, 0 = position of the Android terminal Location
self	smallint	unused
tm	timestamp	Time stamp (GMT)

AutowareRider

Overview

AutowareRider is, for operating the Autoware to operate in ROS PC from the tablet terminal,

With a UI similar to Knight Rider, it is the Android application.

AutowareRoute has been implemented in MapFan SDK, Android appli- for the route data generation

It is the publication.

AutowareRider provides the following functions.

- sends the path data generated in AutowareRoute to ROS PC
- start the CAN data collection applications
- start the ROS PC of Launch files at the touch of a button
- reflect the CAN data received from the ROS PC to UI

This section describes the procedures for using these features.

starting method

1. Start the Runtime Manager in ROS PC.
2. Main tab [Network Connection] - pressing the Active button on the [Tablet UI], start the following
   To do.
   - tablet_receiver
   - tablet_sender
3. Computing tab [Planning] - from each of the anchors of the [Path], and set the following.
   - lane_navi
     - vector_map_directory
       Directory precision map is stored
   - lane_rule
     - vector_map_directory
       Directory precision map is stored
     - ruled_waypoint_csv
       File that waypoint is saved
     - Velocity
       Speed (Unit: km / h, the initial value: 40, range: 0 to 200)
     - Difference around Signal
       The rate of acceleration and deceleration at the front and rear of the signal (unit: km / h, the initial values: 2, range: 0 20)
   - lane_stop
     - Red Light
       Switch to the speed at the time of the red signal
     - Green Light
       Switch to the speed at the time of the green light
4. Computing tab [Planning] - to enable the check box of [Path], start the following
   You.
   - lane_navi

   - lane_rule
   - lane_stop
5. Start the AutowareRider from the application list screen of the Android tablet.
6. From the upper right menu] → [Settings], and set the following.
   - ROS PC
     - IP address
       ROS PC IPv4 address
     - instruction port number

tablet_receiver port number (initial value: 5666)

■ Information port number

tablet_sender port number (initial value: 5777)

7. press the [OK], then tries to connect to the ROS PC.

○ setting this time is automatically saved to a file, saved from next start

Attempts to connect the setting was.

8. The color of the center of the screen of the bar, it has if it is displayed in bright red the connection is successful.

The state of the connection with the ○ bar of color

Bar of color	State of the connection
Dark Red	ROS PC is not connected
Light red	ROS PC connection
Bright blue	Automatic operation (mode_info: 1)
Bright yellow	Abnormal occurrence (error_info: 1)

How to use the route data generation application

1. pressing the NAVI button of AutowareRider, to start the route search.
2. Press and hold the map, and then executed in order the following.

○ set to the departure point

Setting ○ to destination

○ route search execution

3. By the end of the route search after the execution of the route search, route data is transferred to the ROS PC

It will be.

○ route data at this time is automatically saved to a file, exploration route from the next time

You can transfer the route data to omit the search.

4. After the transfer, it returns the screen to again AutowareRider.

Route data transfer procedure to the ROS PC

Please refer to the How to use Step 3. of the above route data generation applications.

How to use the CAN data collection applications

Note) In the current situation, only it supports server of Nagoya.

1. From the upper right menu] → [Settings] of AutowareRider, set the following.

These settings have been launched from the AutowareRider, used by the CanDataSender.

○ data collection

■ table name

Data transfer destination table name

○ SSH

■ Host name

SSH destination host name

■ Port number

SSH connection destination port number (initial value: 22)

■ User Name

User name to log in with SSH

■ Password

Password to log in with SSH

○ port forwarding

■ Local port number

Local machine transfer source port number (initial value: 5558)

■ remote host name

Remote machine host name (default: 127.0.0.1)

■ remote port number

Remote machine destination port number (initial value: 5555)

2. By pressing the [OK], then save settings to a file.

○ However, SSH passwords are not saved to a file. The AutowareRider

Only while it is running, and holds only in memory.

3. From the upper right menu] → [data collection], to start one of the following.

○ CanGather
○ CarLink (Bluetooth)
○ CarLink (USB)

4. how to use after an application is started, the same as in the case where you start each alone

It is.

○ For more information, please refer to the following URL to reference.

https://github.com/CPFL/Autoware/blob/master/vehicle/general/android/README.md

CAN data transfer procedure to the ROS PC

Please refer to the How to Use Step 4. of the above CAN data collection applications.

How to Start Launch file

1. S1 button of AutowareRider, S2 button, to have each of the following Launch file

It supports.

○ Check.Launch
○ Set.Launch

2. By pressing a button, will launch the Launch file in ROS PC.

○ state of buttons and Launch file

button	The state of the Launch file
Pressed (character color: black)	Start-up ({ndt, lf} _stat: false)
Pressed (character color: red)	Start-up ({ndt, lf} _stat: true)

A description of each function

Node List

ros / src / computing / perception / detection

ros / src / computing / perception / localization

ros / src / computing / planning

ros / src / data

ros / src / sensing / drivers and ros / src / sensing / fusion

Runtime Manager

Overview
Runtime Python script Manager, which is contained in the runtime_manager package

(Scripts / runtime_manager_dialog.py) to start in rosrun command use.

$ Rosrun runtime_manager runtime_manager_dialog.py

When you start the Runtime Manager, the dialog is displayed on the screen.

The Runtime Manager of the dialog operation,

Start and end processing of various ROS node to be used in the Autoware,

Start the process of issuing topic for the parameters to the various ROS node, etc.
It can be carried out.

Runtime Manager dialog screen includes a plurality of tab screen.

Buttons for starting and ending a variety of ROS node,
By the node functions, it is classified and arranged on each tab screen.

Display of each tab screen, switched by tabs at the top of the screen.

Runtime Manager startup screen

Quick Start tab

Map toggle button

Start .launch script that is specified in the Map text box to the end.

Map text box

Specify the path of .launch script to start and end from the Map toggle button.

(Specified in the full path)

Map Ref button

File selection dialog is displayed.

The selected file is set to Map the text box.

Sensing toggle button

Start .launch script that is specified in the Sensing text box to the end.

Sensing text box

Specify the path of .launch script to start and end from Sensing toggle button.

(Specified in the full path)

Sensing Ref button

File selection dialog is displayed.

The selected file is set to Sensing the text box.

Localization toggle button

Start .launch script that is specified in the Localization text box to the end.

Localization text box

Specify the path of .launch script to start and end from Localization toggle button.

(Specified in the full path)

Localization Ref button

File selection dialog is displayed.

The selected file is set to Localization text box.

Detection toggle button

It is specified in the Detection text box start and to end the .launch script is.

Detection text box

Detection to specify the path of .launch script to start and end from the toggle button.

(Specified in the full path)

Detection Ref button

File selection dialog is displayed.

The selected file is set to the Detection textbox.

Mission Planning toggle button

To start and end the .launch script that is specified in the Mission Planning text box

That.

Mission Planning text box

To specify the path of .launch script to start and end from the Mission Planning toggle button

That.

(Specified in the full path)

Mission Planning Ref button

File selection dialog is displayed.

The selected file is set to the Mission the Planning text box.

Motion Planning toggle button

To start and end the .launch script that is specified in the Motion Planning text box

That.

Motion Planning text box

Specify the Motion Planning .launch script path to start and terminate from the toggle button

That.

(Specified in the full path)

Motion Planning Ref button

File selection dialog is displayed.

The selected file is set to the Motion the Planning text box.

Android Tablet toggle button

The runtime_manager / tablet_socket.launch script

Start and exit.

Oculus Rift toggle button

<Not implemented>

Vehicle Gatewat toggle button

The runtime_manager / vehicle_socket.launch script

Start and exit.

Clound Data toggle button

Start obj_db / obj_downloader node to the end.

Auto Pilot toggle button

To issue a mode_cmd topics in accordance with the button of the state.

ROSBAG button

To view the ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

ROSBAG Record dialog

ROSBAG Record dialog

The top text box
When you run the rosbag record command, specify the bag file.
(Specified in the full path)

Ref button
Save the file specification dialog is displayed.
The specified file is set to the upper text box.

Start button
Specify the set bag file in the top text box,
To start the rosbag record command.

Stop button

Start-up to exit the rosbag record commands are.

All check box

If the check box is ON, when you start the rosbag record command,

-a option is specified.

Other check box group

When you start the rosbag record command,

Check box to specify the topic of ON.

(However, Enable All check box is a case of OFF only)

Refresh button

Run the rostopic list command, examine the current valid topic,

To update the other check box group.

Information display of the bottom line

To view the load status and memory usage of CPU (core).

Map tab

Map tab

Point Cloud toggle button

Start map_file / points_map_loader node to the end.

Point Cloud text box

Passed in the argument when you start the map_file / points_map_loader in Point Cloud toggle button,

To specify the pcd file group of the path.

(The full path ',' be separated specified in)

Point Cloud Ref button

File selection dialog is displayed.

Multiple files can be selected. (But limited to the same directory)
The selected files are set to Point, Cloud textbox.

Auto Update check box

When you start the map_file / points_map_loader in Point Cloud toggle button,

To specify the presence or absence of the automatic update

Auto Update menu

When you start the map_file / points_map_loader in Point Cloud toggle button,

At the time of automatic updates enabled, to specify the number of scenes.
(Valid only when ON in the Auto Update check box has been specified)

Area List text box

Passed in the argument when you start the map_file / points_map_loader in Point Cloud toggle button,

To specify the path to the area list file.

(Specified in the full path)

Area List Ref button

File selection dialog is displayed.

The selected file is set to Area List text box.

Vector Map toggle button

Start map_file / vector_map_loader node to the end.

Vector Map text box

Passed in the argument when you start the map_file / vector_map_loader in Vector Map toggle button,

To specify the path to the csv file group.

(The full path ',' be separated specified in)

Vector Map Ref button

File selection dialog is displayed.

Multiple files can be selected. (But limited to the same directory)
The selected files are set to this Vector the Map text box.

TF toggle button

Start the lauch files that are set to TF text box to the end.

If the TF textbox launch file is not set,

Start the launch files in the following path to the end.
/.autoware/data/tf/tf.launch

TF text box

To specify the path to launch the file to boot and end by TF toggle button.

(Specified in the full path)

TF Ref button

File selection dialog is displayed.

The selected file is set to the TF text box.

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

Sensing tab

Sensing tab

Drivers / CAN column

can_converter item
Start kvaser / can_converter node to the end.

can_draw item
Start kvaser / can_draw node to the end.

can_listener item
Start kvaser / can_listener node to the end.

config link

To view the can_listener dialog.

To set the channel to be specified in the node startup.

Drivers / Cameras column

PointGrey Grasshoper 3 (USB1) Item
pointgrey / a grasshopper3.launch script starts and ends.

config link

calibration_path_grasshopper3 to display the dialog.

To set the path of CalibrationFile be specified in the script at startup.

PointGrey Grasshoper 3 (USB2) item
<Not implemented>

PointGray LadyBug 5 item
<Not implemented>

USB Generic items
Start uvc_camera / uvc_camera_node node to the end.

IEEE1394 item
<Not implemented>

Drivers / GNSS field

Javad Delta 3 (TTY1) item
javad / a gnss.sh script starts and ends.

Drivers / IMU column

Crossbow vg440 item
<Not implemented>

Drivers / LIDARs column

Velodyne HDL-64e item
velodyne / a velodyne_hdl64e.launch script starts and ends.

config link

To view the calibration_path dialog.

To set the path of the calibration to be specified in the script at startup.

Velodyne HDL-32e item
velodyne / a velodyne_hdl32e.launch script starts and ends.

config link

To view the calibration_path dialog.

To set the value of the calibration_path be specified in the script at startup.

Hokuyo TOP-URG item
hokuyo / top_urg script

Hokuyo 3D-URG item
Start hokuyo / hokuyo_3d node to the end.

SICK LMS511 item
<Not implemented>

IBEO 8L Single item
<Not implemented>

Drivers / OtherSensors column
<No items>

Calibration Tool Kti toggle button

Start camera_lidar3d / camera_lidar3d_offline_calib node to the end.

Calibration Publisher toggle button

Start calibration_camera_lidar / calibrtion_publisher node to the end.

At start-up, so to display the calibration_publiher dialog,
To set the path for the YAML file that specifies the node startup.

(Specified in the full path)

Points Image toggle button

Start points2image / points2image node to the end.

Scan Image toggle button

Start scan2image / scan2image node to the end.

Virtual Scan Image toggle button

runtime_manager / a vscan.launch script starts and ends.

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

Computing tab

Computing tab

Localization / gnss_localiser column

fix2tfpose item
Start gnss_localizer / fix2tfpose node to the end.

nmea2tfpose item
gnss_localizer / a nmea2tfpose.launch script starts and ends.

Localization / ndt_localiser column

ndt_mapping item
ndt_localizer / a ndt_mapping.launch script starts and ends.

Link

To view the ndt_mapping dialog.

After changing parameters

/ To issue the config / ndt_mapping topic.

ndt_matching item
ndt_localizer / a ndt_matching.launch script starts and ends.

Link

To view the ndt dialog.

After changing parameters

/ To issue the config / ndt topic.

Detection / cv_detector column

dpm_ocv item

runtime_manager / a dpm_ocv.launch script starts and ends.

dpm_ocv dialog is displayed at startup.

After setting the parameters, the script starts in Detection Start button.

Link

When you select the type of tuning parameters the (Car or Pedestrian),

To view the car_dpm dialog or pedestrian_dpm dialog.

After changing parameters

To issue the / config / car_dpm or / config / pedestrian_dpm topic.

dpm_ttic item

cv_tracker / a dpm_ttic.launch script starts and ends.

dpm_ttic dialog is displayed at startup.

After setting the parameters, the script starts in Detection Start button.

Link

When you select the type of tuning parameters the (Car or Pedestrian),

To view the car_dpm dialog or pedestrian_dpm dialog.

After changing parameters

To issue the / config / car_dpm or / config / pedestrian_dpm topic.

fusion_ranging item

cv_tracker / a ranging.launch script starts and ends.

car_fusion dialog is displayed at startup.

After setting the parameters, the script is started in the Start button.

Link

When you select the type of tuning parameters the (Car or Pedestrian),

To view the car_fusion dialog or pedestrian_fusion dialog.

After changing parameters

To issue the / config / car_fusion or / config / pedestrian_fusion topic.

kf_tracking item

cv_tracker / a kf_tracking.launch script starts and ends.

car_kf dialog is displayed at startup.

After setting the parameters, the script is started in the Start button.

Link

When you select the type of tuning parameters the (Car or Pedestrian),

To view the car_kf dialog or pedestrian_kf dialog.

After changing parameters

To issue the / config / car_kf or / config / pedestrian_kf topic.

obj_reproj item

cv_tracker / a reprojection.launch script starts and ends.

obj_reproj dialog is displayed at startup.

After setting the parameters, the script is started in the Start button.

Detection / lidar_detector column

euclidean_clustering item

lidar_tracker / a euclidean_clustering.launch script starts and ends.

obj_fusion item

lidar_tracker / a obj_fusion.launch script starts and ends.

obj_fusion dialog is displayed at startup.

After setting the parameters, the script is started in the Start button.

Detection / road_wizard column

feat_proj item

Start road_wizard / feat_proj node to the end.

Link

To view the feat_proj dialog.

After changing parameters

/ To issue the config / adjust_xy topic.

region_tlr item

road_wizard / a traffic_light_recognition.launch script starts and ends.

Link

To view the region_tlr dialog.

After changing parameters

/ To issue the config / superimpose topic.

Detection / viewers column

image_viewer item

Start the viewers / image_viewer node to the end.

image_d_viewer item
Start the viewers / image_d_viewer node to the end.

points_image_viewer item
Start the viewers / points_image_viewer node to the end.

points_image_d_viewer item
Start the viewers / points_image_d_viewer node to the end.

vscan_image_viewer item
Start the viewers / vscan_image_viewer node to the end.

vscan_image_d_viewer item
Start the viewers / vscan_image_d_viewer node to the end.

traffic_light_viewer item
Start the viewers / traffic_light_viewer node to the end.

Mission Planning / lane_planner column

lane_change item
Start lane_planner / lane_chabge node to the end.

lane_navi item
Start lane_planner / lane_navi node to the end.

Link

To view the lane_navi dialog.

After changing parameters

rosparam / lane_navi / velocity,
Setting the rosparam / lane_navi / output_file.

lane_rule item
Start lane_planner / lane_rule node to the end.

Link

To view the lane_rule dialog.

After changing parameters

rosparam / lane_rule / vector_map_directory,
Set the rosparam / lane_rule / ruled_waypoint_csv,

/ To issue the config / lane_rule topic.

lane_stop item
Start lane_planner / lane_stop node to the end.

Link

To view the lane_stop dialog.

After changing parameters

/ To issue a traffic_light topic.

Mission Planning / freespace_planner column

astar_navi item
Start freespace_planner / astar_navi node to the end.

Motion Planning / driving_planner column

lattice_trajectory_gen item
driving_planner / a lattice_trajectory_gen.launch script starts and ends.

Link

To view the lane_follower_trajgen dialog.

After changing parameters

/ To issue the config / lane_follower topic.

lattice_twist_convert item
Start driving_planner / lattice_twist_convert node to the end.

Motion Planning / waypoint_maker column

waypoint_loader item
waypoint_maker / a waypoint_loader.launch script starts and ends.

Link

To view the waypoint_loader dialog.

After changing parameters

/ Issued the waypoint_loader / vector_map_directory topic,

Set the rosparam / waypoint_loader / ruled_waypoint_csv,

/ To issue the config / waypoint_loader topic.

waypoint_saver item
waypoint_maker / a waypoint_saver.launch script starts and ends.

Link

To view the waypoint_saver dialog.

To set the value of the save_filename and Interval specified in the script at startup.

waypoint_clicker item
Start waypoint_maker / waypoint_clicker node to the end.

Link

To view the waypoint_clicker dialog.

After changing parameters

rosparam / waypoint_clicker / velocity,
Setting the rosparam / waypoint_clicker / output_file.

Motion Planning / waypoint_follower column

pure pursuit item
waypoint_follower / a pure_pursuit_sim.launch script starts and ends.

Link

To view the waypoint_follower dialog.

After changing parameters

/ To issue the config / waypoint_follower topic.

velocity_set item
waypoint_follower / a velocity_set.launch script starts and ends.

collision_avoid item
waypoint_follower / a collision_avoid.launch script starts and ends.

twist_through item
Start waypoint_follower / twist_through node to the end.

car_simulator item
lane_follower / a car_simulator.launch script starts and ends.

Link

To view the car_simulator dialog.

After changing parameters

rosparam / odom_gen / use_pose
rosparam / odom_gen / initial_pos_x
rosparam / odom_gen / initial_pos_y
rosparam / odom_gen / initial_pos_z
rosparam / odom_gen / initial_pos_roll
rosparam / odom_gen / initial_pos_pitch
rosparam / odom_gen / initial_pos_yaw
To set.

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

Interface tab

Interface tab

Android Tablet toggle button

The runtime_manager / tablet_socket.launch script

Start and exit.

Oculus Rift toggle button

<Not implemented>

Vehicle Gatewat toggle button

The runtime_manager / vehicle_socket.launch script

Start and exit.

Sound check box

sound_player / a sound_player.py script starts and ends.

Auto Pilot toggle button

To issue a mode_cmd topics in accordance with the button of the state.

Lamp L, R toggle button

To issue a lamp_cmd topics in accordance with the button of the state.

Indicator L, R toggle button

To issue a indicator_cmd topics in accordance with the button of the state.

D, R, B, N button

To issue a gear_cmd topics in accordance with the button that was ON operation.

Accel slider

To issue a accel_cmd topic.

Brake slider

To issue a brake_cmd topic.

Steer slider

To issue a steer_cmd topic.

Torque slider

<Not implemented>

Veloc slider

To issue a twist_cmd topic.

(The value of the slider is reflected in the twist.linear.x field of the message)

Angle slider

To issue a twist_cmd topic.

(The value of the slider is reflected in the twist.angular.z field of the message)

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

Database tab

Database tab

CAN column

can_uploader item
Start obj_db / can_uploader node to the end.

Link

To view the other dialog.

Map column

map_downloader item
<Not implemented>

Link

To view the map_file dialog.

Position column

pos_downloader item
Start pos_db / pos_downloader node to the end.

Link

To view the pos_db dialog.

pos_db dialog (pos_downloader)

pos_uploader item
Start pos_db / pos_uploader node to the end.

Link

To view the pos_db dialog.

pos_db dialog (pos_uploader)

Sensors column

image_upload item
<Not implemented>

Link

To view the other dialog.

pointcloud_upload item
<Not implemented>

Link

To view the other dialog.

Query text box

<Not implemented>

Query button

<Not implemented>

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

Simulation tab

ROSBAG text box

When you run the rosbag play command in the Play button, specify a bag file.

(Specified in the full path)

ROSBAG Ref button

File selection dialog is displayed.

The selected file is set to ROSBAG text box.

Rate text box

To specify a number to specify the -r option of when to start the rosbag play command.

If you do not specify the -r option in the case of not set.

Start Time (s) text box

Specify the number of seconds of the start position that you specify in the --start option of when to start the rosbag play command To.

If you do not specify a --start option if it is not set.

Repeat check box

If the check box is ON, when you start the rosbag play command,

--loop option is specified.

Clock check box

If the check box is ON, when you start the rosbag play command,

--clock option is specified.

(This setting is not saved at the end)

Sim Time check box

rosparam / use_sim_time of setting value (true, false) to display.

When you operate the check box, the value is set to rosparam / usr_sim_time.

(This setting is not saved at the end)

Play button

ROSBAG specify the set bag file into a text box,

To start the rosbag play command.

Stop button

Start-up to exit the rosbag play commands are.

Pause button

Start to pause the rosbag play commands are.

Center

For .bag file ROSBAG is set to text box,

To view the execution result of rosbag info command.

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

Status tab

Status tab

Display of the upper

To view the execution result of the top command running inside.

Display in the lower left-hand side

Related node to display the cycle execution time to be issued.

Display in the lower right-hand side

Start node, the standard output of the script, to display the contents of the standard error output.
(But not visible in some of the nodes performing the progress bar display)

Information display of the bottom line

To view the load status and memory usage of CPU (core).

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

Topics tab

Topics tab

Display of the left-hand side

To view a list of topic name.

When you click the link,

Run the rostopic echo <target topic> command, and displays the results on the right side of the upper stage,
Run the rostopic info <target topic> command to display the result in the lower right.

Display of the right upper

To view the execution result of rostopic echo command.

Display of the right lower

To view the execution result of rostopic info command.

Refresh button

To update again to get a list of topic name.

In addition, to stop if rostopic echo command running by a link click.

Information display of the bottom line

To view the load status and memory usage of CPU (core).

ROSBAG button

To view the ROSBAG Record dialog.

See the Quick Start tab for details of ROSBAG Record dialog.

Rviz toggle button

Start rviz / rivz node to the end.

RQT button

Start rqt · to end.

User interface

Overview

AutowareRider is, for operating the Autoware to operate in ROS PC from the tablet terminal,

With a UI similar to Knight Rider, it is the Android application.

AutowareRoute has been implemented in MapFan SDK, Android appli- for the route data generation

It is the publication.

This section describes the functions of these UI.

AutowareRider

The following is the screen at startup.

The function of each button in the figure are below.

- NAVI

  Starting ○ AutowareRoute.apk

- MAP

  ○ unimplemented

- S1

  Start ○ the check.launch in ROS PC

- S2

  Start ○ the set.launch in ROS PC
- B

  ○ send the gear information B to ROS PC
- N

  ○ send the gear information N to ROS PC
- D

  ○ send the gear information D to ROS PC
- R

  ○ send the gear information R to ROS PC
- AUTO CRUISE
  ○ unimplemented
- NORMAL CRUISE
  ○ unimplemented
- PURSUIT
  ○ unimplemented (current situation is the end of the application)

The following can be selected from the upper right menu.

- [Settings]
- [data collection]

The following is the screen of the Settings.

A description of each item in the figure is the following.

- ROS PC
  - IP address

    ROS PC IPv4 address
  - instruction port number

    tablet_receiver port number (initial value: 5666)
  - information port number

    tablet_sender port number (initial value: 5777)
- data collection
  - table name

    Data transfer destination table name
- SSH
  - host name

    SSH destination host name

  - port number

SSH connection destination port number (initial value: 22)

- ○ user name

  User name to log in with SSH

- ○ password

  Password to log in with SSH

- port forwarding
  - ○ local port number

    Local machine transfer source port number (initial value: 5558)

  - ○ remote host name

    Remote machine host name (default: 127.0.0.1)

  - ○ remote port number

    Remote machine destination port number (initial value: 5555)

The following is the screen of the [data collection].

The function of each button in the figure are below.

- CanGather

  Starting ○ CanGather.apk

- CarLink (Bluetooth)

  Starting ○ CarLink_CAN-BT_LS.apk

- CarLink (USB)

  Starting ○ CarLink_CANusbAccessory_LS.apk

AutowareRoute

The following is the screen at startup.

By pressing the length map, you will see the following dialog.

The function of each button in the figure are below.

- set to the departure point

  Setting ○ long press was the point as the starting point of the route data

- set to the stop-off place

  Setting ○ long press was the point as a stop-off areas of the route data

- set to destination

  Setting ○ long press was the point as the destination of the route data

- route erasure

  ○ erasing route data generated by the route search execution

- route search execution

  ○ starting point, a stop-off land, generation of route data in accordance with the destination