Autoware Introduction

Theodore Grey
<theodore.grey@linaro.org>

Servando German
<servando.german.serrano@linaro.org>
Introduction

Theodore Grey
Software Engineer, Linaro
96Boards Software Engineer for Linaro. Recent Graduate of Kings College London, with a passion for machine learning and autonomous vehicles.

Servando German Serrano
Software Engineer, Linaro
Autonomous systems engineer with experience working in different domains: air, underwater and ground based platforms. Past experience involves working at Ixion Industry & Aerospace, a highly innovative SME in Spain, Airbus Defence and Space, in Stevenage, and most recently at Transport Systems Catapult, based in Milton Keynes.
Launched in December 2018.
Aimed at supporting open source projects to enable self-driving mobility.
Currently, three main projects:
  - Autoware.AI
  - Autoware.Auto
  - Autoware.IO

Members:
Based on ROS 2.
Clean re-write of Autoware.AI.
Applying best-in-class software engineering practices.
Clearly defined APIs and interfaces for the different modules.
Software developed at production level for MISRA compliance.
Autoware.AI was started in 2015 by Shinpei Kato at Nagoya University.

Today, Autoware.AI is supported by the largest autonomous driving open source community with 2300+ stars on GitHub and 500+ accounts on Slack.

Autoware.AI has found widespread and international adoption as it is used by more than 100 companies and runs on more than 30 vehicles in more than 20 different countries.

Courses using Autoware are offered in 5 countries.

Automotive OEMs are using Autoware for Mobility as a Service (MaaS) development.

Autoware is qualified to run on driverless vehicles on public roads in Japan since 2017.
Developed in ROS 1 and available under the Apache 2 license.

Presented at ROSCon 2017.

Used for Research and Development applications.

First “All-in-One” open source software for autonomous driving technology.

Contains the following modules:

- **Localization**: achieved by 3D maps and SLAM algorithms combined with GNSS/INS sensors.
- **Object Detection**: camera and LiDAR data used for sensor fusion algorithms and deep neural networks.
- **Prediction and Planning**: based on probabilistic robotics and rule-based systems.
- **Control and actuation**: path following such as pure-pursuit or MPC and vehicle-dependent actuation.
AV Sw Modules

- Sensing:
  - LIDAR
  - GNSS
  - Camera
  - IMU
  - Radar
  - Etc.

- Localisation:
  - Absolute (i.e. lat, long, etc)
  - Relative to other agents

- Perception / Situation Awareness:
  - Computer Vision
  - ML / DL
  - KFs, PFs
  - Obstacle detection, object classification (traffic lights, signals), lane detection

- Navigation:
  - Global planner (route A-B)
  - Local planner (obstacle free path)
  - Decision making

- Actuation:
  - Path following
  - PID(s), etc: vehicle dependent

- External:
  - Map

- V2X:
  - CAM
  - DENM
  - Future specs

- Simulation
Autoware.AI: sensors
Autoware.AI - Localisation

- Mainly LiDAR based and 3D pointcloud maps.
- Based on NDT matching (ICP also implemented).
- GPU and CPU implementation of ROS nodes.

From The NDT: A new approach to laser scan matching
AV Sw Modules

Sensing
- LIDAR
- GNSS
- Camera
- IMU
- Radar
- Etc.

Localisation
- Absolute (i.e. lat, long, etc)
- Relative to other agents

External
- Map

Perception / Situation Awareness
- Computer Vision
- ML / DL
- KFs, PFs

Obstacle detection, object classification (traffic lights, signals), lane detection

Navigation
- Global planner (route A-B)
- Local planner (obstacle free path)
- Decision making

Actuation
- Path following
- PIDs, etc: vehicle dependent

Simulation

V2X
- CAM
- DENM
- Future specs
Object detection via:

- Euclidean Clustering for LiDAR pointclouds and DNN algorithms for cluster classification.
- DNN, such as SSD and YOLO, for real time performance on camera data.

Object tracking using Kalman filters or Particle filters.
Autoware.AI - Pred. & Planning

Vector map (sets of waypoints and features)
- Lane Info.
- Intersection Info.
- Traffic lights and signs

OpenPlanner
- Global Planner (way_planner)
  Global Path
- Local Planner (dp_planner)
  Rollout Trajectory Generator
  Behavior State Generator

Goal(s) Pose (rviz, file)
Start Pose (rviz, current_pose)
Detected Obstacles
Traffic Light Color

Smooth, Obstacle-Free Trajectory
Vehicle controllers

Pure pursuit and MPC
- Autoware Foundation: https://www.autoware.org/
- Autoware.Ai Github repository: https://github.com/CPFL/Autoware
- Join the Autoware community discussion forum on Slack: https://autoware.herokuapp.com
- Discourse - Autoware project specific discussions: https://discourse.ros.org/c/autoware/
Thank you

Join Linaro to accelerate deployment of your Arm-based solutions through collaboration

contact@linaro.org