Indoor real-time navigation for robot vehicles

Final year project for the specialization program in Embedded Systems Engineering

The goal of the project is to design a real-time navigation system for a robot vehicle (figure 1). The acquisition device is based on a portable autonomous laser scanner (Lidar) controlled by a STM32 MCU and an Exynos board that can both communicate through a serial / USB connection. The Lidar data is streamed to a PC base station running SLAM and visualization softwares. The acquisition device and the base station communicate using Robot Operating System (ROS) and a Wifi connection. The SLAM algorithm is based on Google Cartographer and a 3D visualization tool called RVIZ, both integrated in ROS.

A typical use case is the following: we want to be able to move the laser scanner around and collect data with it all while watching the map get created/updated in real time on a remote basestation.

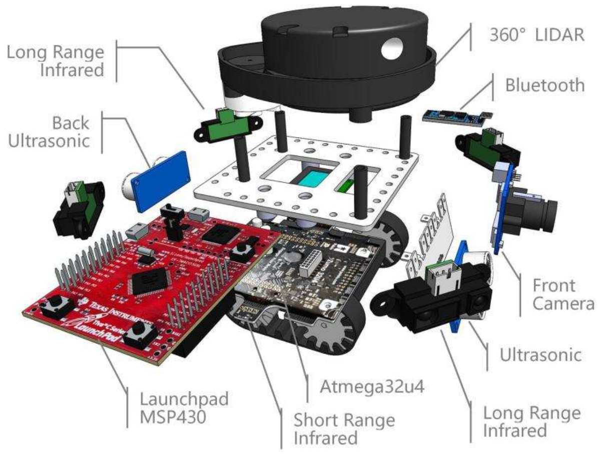

Figure 1. LIDAR / SLAM (general principle)

The project is decomposed in the following milestones and delivrables (which will run from week 37 to week 51), with an indicative timetable.

Milestone 1 ROS Cartographer
D1.1 ROS cartographer demo on PC workstation (week 38). Install ROS and Google cartographer. Run and identify the most suited demo that can be used as a starting point for the development of a 2D LIDAR / SLAM system. Analyze the key parameters of the example (number of laser scan per sample, angular resolution, number of samples per second).
D1.2 ROS cartographer demo on Exynos (weeks 39 - 40). Install Ubuntu 16.04 on a 32GB micro SD card for the Exynos board. Make sure to use a GCC 5.4 compiler in all following developments. Install ROS and Google Cartographer and run the demo on Exynos.
D1.3 ROS Topics local (week 41). Communication of Lidar data between the acquisition device and the PC base station will be based on ROS messages and Topics. Setup an example where (fake) laserscan data can be published and read between two nodes.Try first on a single PC workstation first, try then on the Exynos alone, but do not setup Wifi remote communication between Exynos and PC basestation for the moment (this will be made in D3.2).

Milestone LIDAR 2D 有问题，用uart代替
D2.1 USB CDC communication (week 42). Setup STM32 project to realize a USB CDC communication with Exynos, and a C program on the Exynos to read data on the USB link (using termios library).
D2.2. Lidar control with STM32 (week 43). Setup STM32 project to configure Lidar device via I2C and check with a single distance sample.
D2.3 Lidar 2D (week 44). Extend previous project to control the stepper motor and scan 360°. Compare and adapt the characteristics of the Lidar to previously identified parameters (D1.1).
D2.4 Lidar device demo (week 45). Extend previous projects to stream Lidar data from Lidar/STM32 to Exynos. Setup a demo such that the Exynos automatically runs a waiting loop at

startup and STM32 starts streaming data when the start button is pushed. Use the LCD screen to display useful debug messages.

Milestone 3 SLAM (weeks 46 - 47)
D3.1: Lidar 2D bagfile. Generate a bagfile on the Exynos from previous Lidar device demo, using the record bag file feature of ROS. Check that the bagfile generated works as an input of Google Cartographer.
D3.2: ROS Topics remote. Extend previous ROS Topic communication (D1.3) to Wifi remote communication between Exynos and PC basestation. Display the data as text (printf) for a simple test.
D3.3: 2D Map display. Extend previous D3.2 such that the Lidar data can be displayed now directly in RVIZ (without the need of a bagfile).

Milestone 4 Integration, demo & reporting (weeks 48 - 50)
D4.1 SLAM demo. Integrate the Lidar device with a battery such that the system can be fully autonomous and mobile. Setup a demo where a map is created, displays and update as the Lidar moves.
D4.2 Polytech bagfile. Produce a bagfile with a cartography of Polytech buildings.
D4.3 Video of demonstration. Produce a high quality video of previous Polytech cartography, where you can visualize the evolution of an observer holding the Lidar, and the display of the map. There will be no other report except the demo movie which will then be the base of your evaluation (together with the validation of deliverables on time). At the end, you must also provide a clean version of your projects correctly commented, documented and packaged.

Organization, attendance and evaluation

1) The work is organized in a two person team, each team work independently. Excessive similarity between two projects will suffer penalties in the final evaluation.
2) Each binom has access to a PC workstation and Exynos board with some needed peripherals. There will be only one Lidar available, so the agenda for Lidar sessions will be the following:
   Morning:
   GR1 08:00 – 08:30 GR2 08:30 – 09:00 GR3 09:00 – 09:30,
   GR4 09:30 – 10:00 GR5 10:00 – 10:30 GR6 10:30 – 11:00
   Afternoon:
   GR1 13:30 – 14:00 GR2 14:00 – 14:30 GR3 14:30 – 15:00,
   GR4 15:00 – 15:30 GR5 15:30 – 16:00 GR6 16:00 – 16:30

3) Project advancement is monitored with partial evaluations (at each Milestone/Delivrable), final validation and demonstration (December 11) and final report (due to December 11). Final report is 5 pages maximum and structured as follows: Introduction / Specification / System overview, High-Level design / Hardware / Software, Test and results, Conclusion (objective critique, possible improvements), References (articles, links, web), Apendices (code). As a reminder, the overall coefficient for this project is **12**.
4) Presence to all project sessions is mandatory and will be controlled. Absence to non supervised sessions will suffer penalties in the final evaluation.