The Design and Implementation of Autonomous Driving Pallet Robot System using ROS

Ung-Gyo Lee[1] and Kyung-Jea Choi[1]
School of Robot and Smart System Engineering
Kyungpook National University
Deagu, South Korea
dndry123@naver.com, ckj01234@gmail.com

Soon-Yong Park[2]
School of Electronic and Electrical Engineering
Kyungpook National University
Deagu, South Korea
sypark@knu.ac.kr

***Abstract*— In this paper, we present the implementation of autonomous mobile pallet robot system using by ROS (Robot Operating System) and it shows that the packages provided by ROS are well loaded in our custom robot system. In session II, III, we will briefly introduce the robot's hardware and software system and then explain the process that how to implement the custom robot using by ROS and describe each require packages step by step. In session IV, we will experiment the autonomous navigation system with our handcraft pallet robot. In order to experiment, we built the map using by Google's Cartographer SLAM and the pallet robot successfully navigating on the grid map.**

***Keywords*— autonomous mobile robot; pallet robot; ROS; SLAM; Navigation; 2D LiDAR;**

I. INTRODUCTION

Robots are used in lots of research field not just traditional robotics research fields but also unexpected fields of research such as, architecture, ocean, medicine, graphics, games, etc. They often utilize the robots as their own research. In order to conduct the experiment, they purchase commercial robots as similar conditions as possible in the required research. But most cases, it is difficult to find a suitable commercial robot for the experiment. Designing robot for experiments is required a lot of expertise and it is a challenging works. In this paper, we explained how to implement custom robot system easily using by ROS.

In order to SLAM and Autonomous Navigation experiment, we designed the mobile pallet robot platform which we want to implement. Pallet robot is usually used at a factory and it is indoor transport vehicle robot. Pallet robot is effective way to saving the labor costs in factory, it moves stack of pallets and then transports it. It arrived target position and take off the stack on the floor autonomously. To make autonomous navigation pallet robot system requires many algorithms including Control, Mapping, Localization, Collision Avoidance, Path Planning, etc. In the ROS, these system easily loaded.

One of the most important thing for an autonomous mobile robot is localization which is recognizing own location in the work environment. In fact, for autonomous driving, the robot's own location must be known, before it can travel along the path. If the robot uses only dead-reckoning, it can't get accurate place of information due to the unexpected errors. When this situation goes on, errors are accumulated and it leads to drift of the robot. So our designed robot use Wheel Encoder and 2D LiDAR sensors to get the accurate location data. In this paper, we provide five sessions: session II gives a brief description to our pallet robot's system, session III is describe each package used for implementation, session IV experiments, we build a 2D grid map using cartographer SLAM and navigating on 2D grid map, session V discussions about conclusions.

II. SYSTEM OVERVIEW

A. Hardware and Sensors

For hardware system, we design pallet robot body frame. The pallet robot has two actuated wheels and four castors. Embedded board mounted with two motors, Wheel Encoders, Motor Drives and 2D LIDAR and it connected by USB. Fig. 1 shows our designed pallet robot system.

TABLE I. THE COMPONENTS OF PALLET ROBOT

	robot hardware component
Embedded board	NVIDIA Jetson-Nano
Wheel encoder	PHIDGET Optical Rotary Encoder
Motor	PHIDGET 42STH38 NEMA-17 Bipolar Stepper
Motor controller	PHIDGET Stepper Bipolar HC
2D LiDAR	SLAMTEC RPLIDAR-S1
Camera	Inter RealSense T265 Tracking Camera
Battery	NEXT-408PB-UPS (40800maAh)

Fig. 1. The hardware system of pallet robot we designed.

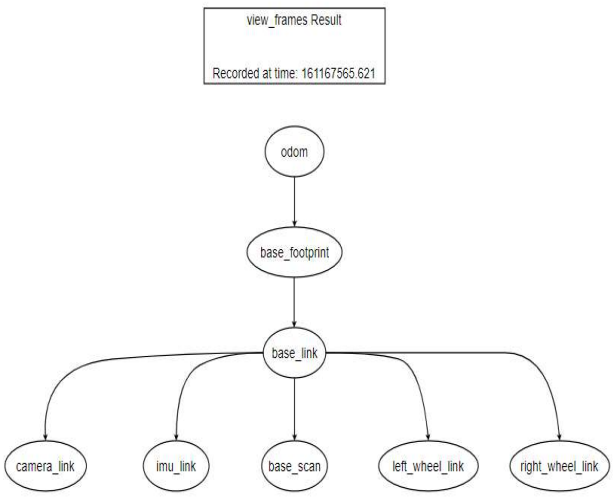

Fig. 2. The TF tree of our pallet robot system.

B. Environment and Software

The environment of system is Ubuntu 18.04 LTS (Bionic) and ROS Melodic version. In order to implement ROS based custom robot, we use these packages and briefly explain what these packages do.

- ROS URDF: Unified Robot Description Format can describes robot's modeling, kinematic and dynamics. It can modeling the robot with combine of base, link, joint. In this paper, we used it to easily build a coordinate system for custom robot.
- ROS control: These package include hardware interface, controller interface, control manager and controllers etc. To connecting real robot with ROS, use hardware interface and we can control the real robot using by controller, control manager. But when the robot is a special case, ROS control doesn't support special type of controller. In this case, create own robot's controller using by controller interface.
- ROS navigation: These packages are support the mobile robot's navigation system. It included a lot of system for navigation, for example, collision avoidance, path planning, localization, etc. In ROS, use these packages, navigation system can be easily configured.
- ROS cartographer slam: To build a map for autonomous navigation, we use cartographer slam.

URDF define robot's coordinate system for sensors and actuators. To make sure it registered in ROS system, we can check by ROS TF package. If it is registered well, TF tree will exist in ROS. Our pallet robot system's TF tree is shown in Fig.2, it is describe our pallet robot's fixed coordinate system.

C. Autonomous Navigation System

In order to implement the autonomous driving robot system, first we built a grid map with SLAM. In this paper, we built a grid map using by cartographer slam and the robot can use the navigation system on the grid map.

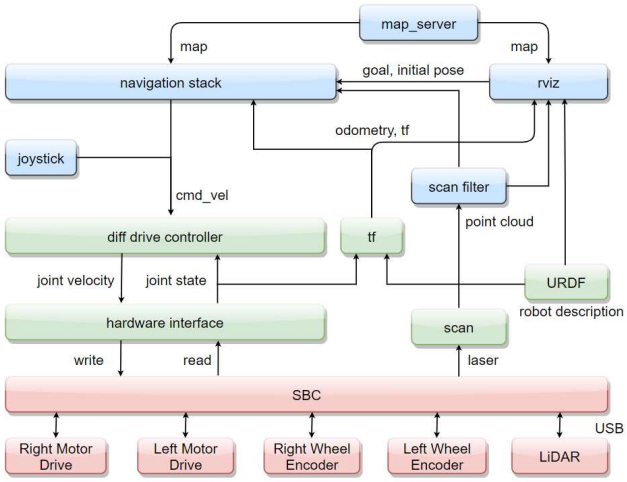

Fig. 3. System diagram of our pallet robot navigation system.

III. Implementation

In this section, we explain about process that how to implement custom robot using ROS. We will describe each necessary components sequentially. The system diagram of our designed navigation robot system is shown in Fig.3.

A. URDF

We need to define robot's coordinate system in ROS, it is called as TF tree system. The TF system provides information of the robot coordinate system. It gives the information to ROS system which make it available in ROS. In addition, this package has many functions for example, describe robot kinematic, dynamic model, collision model, and visual modeling.

B. ROS Hardware Interface

This step connects actual robot hardware with ROS system. The hardware interface connects sensor and actuator with ROS system. This package offer the sensors and motors information to controllable in ROS system. All we have to do is define read and write data interface format. This part need the developers ability because it depends on the hardware's characteristics of own robot. So the developer should make a code own their own.

C. ROS Controller

When this step is over, the custom robot development is completed. So, we can use a number of packages offered by ROS for our robot system. In this step, we set the controller manager and controller to make robot used by ROS.

- Controller Manager: Controller manager is connected to robot hardware interface and robot controller. It can coordinated between hardware interface and robot controller.
- Controller: Our pallet robot system is used in diff-drive-controller. ROS control provide a lot of controllers for arm, drone, mobile robot, gripper etc. To use these, we must provide the joint control type required by controller.

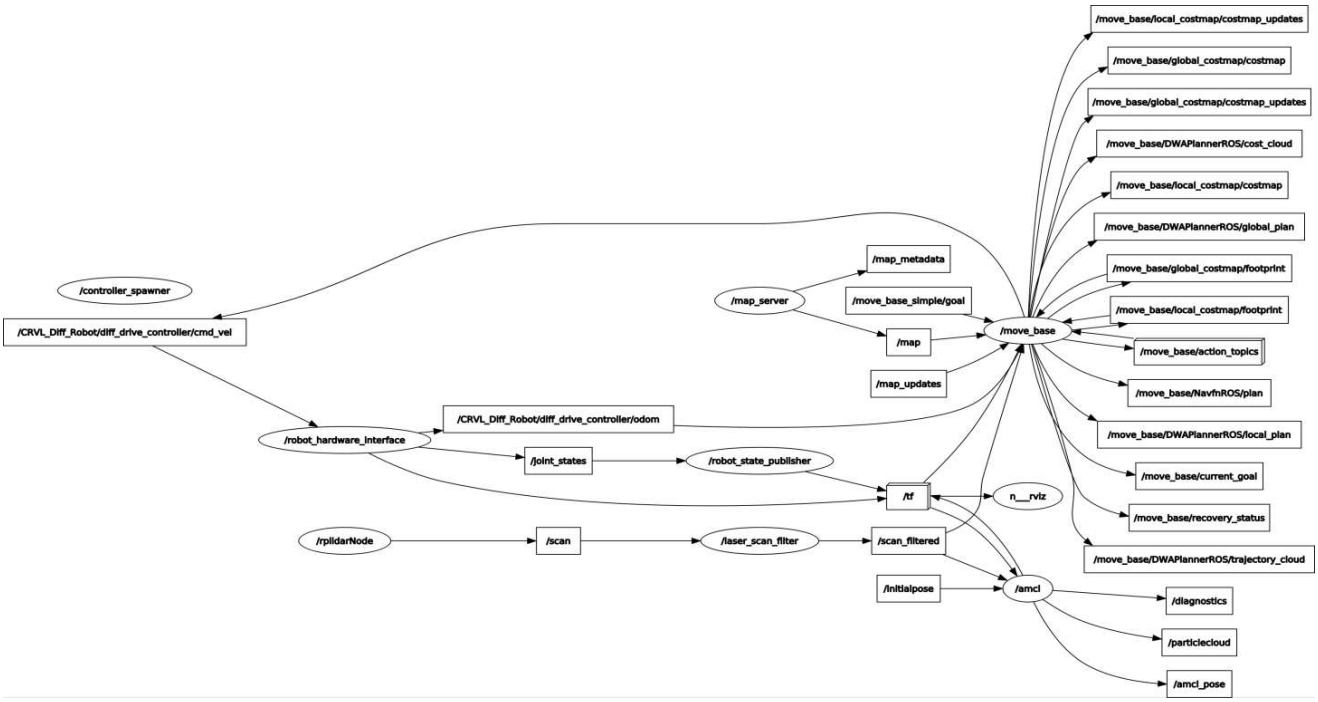

Fig. 4. The ROS rqt-graph of navigation system.

D. ROS Navigation

The navigation packages are perform to move the robot from the start position to the goal position without any collision on the given map. In ROS, these packages help easily implement autonomous driving system for mobile robot. But there are a few conditions to use:

- Robot must be running on ROS.
- Robot's coordinate system must be registered in ROS system.
- Robot has to communicating sensors output, actuators input data with ROS.

In this system, we loaded AMCL localization, DWAP local path planner and A-Star global path planner system. Fig.4 shows ROS rqt-graph of our pallet robot's navigation system.

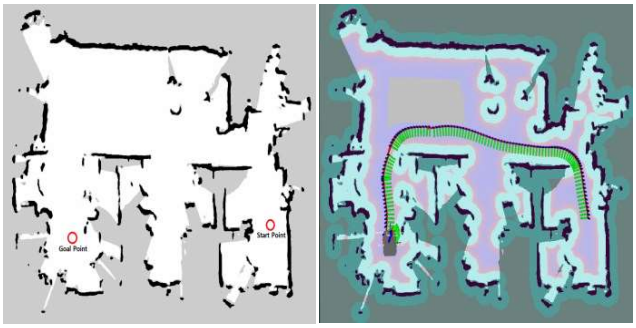

Fig. 5. Test map built with google's cartographer SLAM using by robot's wheel odometry, laserscan data and the pallet robot sucessfully reach the goal position using by ROS navigation system.

IV. Experiment Result

In this session, we experiment our pallet robot's navigation system. Fig.5 shows that the grid map built by cartographer SLAM and our pallet robot performs successfully reach the goal position on the grid map using ROS navigation packages.

V. Conclusions

In this paper, we explain about implementing the custom mobile robot using ROS. It shows that the ROS packages is well loaded in our custom system. So we can experiment with handcrafted robot equipment using ROS. Furthermore, the robot can adds other sensors, actuator or change shape suitable for our research.

Acknowledgment

This research is conducted with the support of the Korea Research Foundation's Nuclear Technology Development Project with the funding of the Ministry of Science and ICT in 2018 (NRF-2018M2A8A5083266).

References

[1] Li, Ruijao, et al. "Ros based multi-sensor navigation of intelligent wheelchair." 2013 Fourth International Conference on Emerging Security Technologies. IEEE, 2013.

[2] Gatesichapakorn, Sukkpranhachai, Jun Takamatsu, and Miti Ruchanurucks. "ROS based autonomous mobile robot navigation using 2D LiDAR and RGB-D camera." 2019 First international symposium on instrumentation, control, artificial intelligence, and robotics (ICA-SYMP). IEEE, 2019.

[3] Kim, Eung-Su, and Soon-Yong Park. "Extrinsic calibration of a camera-LIDAR multi sensor system using a planar chessboard." 2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN). IEEE, 2019.