VALE: Indoor Guide

TAMU Team 2

SICK Lidar Competition 2020

Team Members:
Shanley Mullen
Jesse Rosart-Brodnitz
Jacob Ladigo

Faculty Advisors:
Dr. Xingyong Song
Dr. Garth V. Crosby

Introduction/Theory

VALE, is an autonomous guide robot designed for use within large office buildings or retirement homes that can lead people towards unknown rooms. Its main purpose is to help the disabled and elderly that might have cognitive issues that limit them from being able to freely navigate indoor buildings. With the SICK lidar VALE continually scans while driving and then generates a list of points to create a blueprint. After this, room coordinates can be extracted from the blueprint. From this it can guide the user to that destination. This is done through the use of the lidar for localization and obstacle avoidance.

To begin, the process was broken up into modular sections allowing individuals to work on specific aspects of the project simultaneously. The robot base, known as the S.C.U.T.T.L.E. (Sensing, Connectected Utility Transport Taxi for Level Environments), was provided by the MXET Department. The VALE system was mounted on the S.C.U.T.T.L.E. base which allowed the project to focus more on software instead of hardware. The software was broken up into 7 components.

Hardware Analysis

The first major task was cleaning up the hardware. This involved redoing the wiring of the robot, upgrading 3D printed parts, and putting new sensors on the robot. This included mounting the TiM 781 Lidar. Another task was setting up a computer on board. We chose the BeagleBone Blue with Debian to act as the controller. This system would allow us to set up a remote secure shell to use the computer from afar. The robot also uses two magnetic encoders mounted by the motors to track motor spin. These are not directly attached to the wheels but instead use a sprocket chain setup to increase the power on each wheel. The robot itself is a rear wheel drive system with two caster wheels in the front. The lidar was mounted in the rear center of the robot, positioned between the two wheels. The Lidar's location on the robot is strategic as it actually makes the math simpler.

Software Analysis

Our first major task for the software was creating the map. To start off, the robot takes a scan with 27 data points and reorients the 0 positions to be pointed straight ahead. At the point, the data is split into a two-column array containing the distance and angle. Any data with a distance that is less than the width of the robot is filtered out. Next the entire array is passed through a function that re-localizes the data with the robot's y-axis based on whether or not the reading is positive or negative. All the data is then converted into global coordinates from where the robot is currently situated. From here, the odometry data is taken through calculations of the wheel rotations. The odometry program returns an array containing, the global x position, the global y position, and the orientation of the robot calculated through odometry. The global x and y and the local x and y can be seen below in *Figure1 Local and Global Axes*. As the robot moves, additional scans are taken and the position of the robot is updated with the odometry. This, in turn, causes all the future data points to register based on the robot's new position.

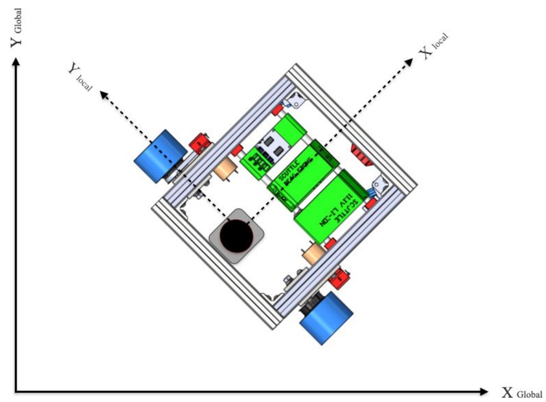

Figure1 Local and Global Axes

$$x_{p2} = d * sin(\theta)$$

$$y_{p2} = d * cos(\theta)$$

Equation 1 - Finds Local

$$xp2' = cos(\theta') * xp2 - yp2sin(\theta') + x'$$

$$yp2' = sin(\theta') * xp2 + yp2sin(\theta') + y'$$

Equation 2 - Finds Global

The odometry program pulls data straight from the encoders in order to calculate the current position and orientation of the robot. This is achieved by first taking two encoder readings separated by a small time-step. The difference between these encoder readings is taken and the amount of rotation on each wheel is calculated. The updates in the wheel position are then averaged to give the distance the center of the robot traveled and the change in the angle of the center. The robot's X and Y locations are then calculated based on the old position of the robot, the change in distance traveled of the center and the change in angle. This is then formatted and output as an array with the angle of the robot and the magnetometer reading. The magnetometer is used to give the initial reading for which direction the robot is facing and then used to correct large deviances through an accumulated error in the angle.

VALE also excels at obstacle avoidance. Using the lidar, a scan is taken and the distance and angle are recorded in an array. How close the obstacle is determines the severity of the reaction. The angle is used to determine if the obstacle is to the left or right of the robot. A level based system is put in place for this. The levels are the following ranges, beyond 2 m, between 1 m and 2 m, and between 1 m and .5 m and less than .5 m. The robot will undergo no action if the 2 m range is exceeded. If the obstacle is between the 2 m and 1 m range, the robot will commence a gradual curve. If the obstacle is between 1 m and .5 m, a medium turn will be conducted. If it is less than .5 meters, block motion will be conducted.

Accessibility

VALE exists to be accessible to those who need it. With this in mind, VALE uses commands to remind you of your destination. The robot will also check in with you regularly and make sure you are still following. Built with bright and contrasting colors, VALE is always able to be seen. The original idea of VALE was to act as means for those with memory issues to be able to freely navigate their environments on their own to provide an extended period of freedom that they might not have had otherwise. We were able to achieve this goal on the small scale and hope to be able to implement this on the larger scale once businesses begin to reopen.

Main Obstacles

A major obstacle our team faced towards the beginning of the design was correctly and accurately reading in our Lidar data to then be used to generate a map. Primarily, this was due to us needing to familiarize ourselves with how the structure of the data needed to be. Moving on from there we needed to work through how the wheel odometry of the robot would factor into finding the overall position to be used in conjunction with the lidar scans.

Conclusion

VALE is an autonomous robot that can lead you to your destination. Built to illustrate possible uses for the TiM781 SICK Lidar, VALE now serves as a guide for those who need it.
Designed with the disability and elderly communities in mind first, VALE exists to offer some independence back to those communities in the form of a friendly guide robot.

Note:

Due to the COVID-19 pandemic, our team was forced to relocate and work remotely. This caused a delay in our time schedule and prevented us from operating in an office environment and from testing in a real world environment. We wanted to test on a larger scale, but due to the closure of campus, larger buildings, and any other public building, we were forced to demonstrate in our apartments.