Area Scanning Control System for Back-packable Robot Boats

Report for CS39440 Major Project

Author: Elizabeth Stone (eas12)
Supervisor: Mark Neal (mjn)

Version: 1.0 - Draft
Monday 24th April, 2017

This report was submitted as partial fulfilment
of a BSc degree in Space Science and Robotics (FH56)

Acknowledgements

I thank my dissertation supervisor for helping me with this project...

Abstract

this project...

Contents

Background & Objectives

Background

previously was used manually using RC. Create a working controll system to autonoumously control the boat. [nasa]

Analysis

To autonomize the robotic boat - I learnt this... .

The use of a general control sytem non-specific to sonar would be useful. A system that can be easily used across simiar platforms.

Process

This project used Feature Driven Development (FDD) methodology for orginization. Upfront a desgin for the hardware and a software flow diagram was created, as can be seen in ***figures ??. A list of tasks to be completed was then kept, with an order of priority (the first tasks on the list needed to be completed before the next tasks). This list was backed up on github, as with the rest of the project, to prevent loss of data if one machine was broken or corrupt.

Design

Overall Architecture

Pi and arduino

Hardware

Rasperry Pi 3 and Moteino (can be replaced with any arduino, as the wireless function of the moteino is not used), with GPS connected to the Pi, and compass, motors and rudders attached to the moteino.

Different designs were considered for the hardware - the gps could be on the arduino or the Pi. On the arduino this has the advantage that PID control could be done faster as the PID loop wouldn't be waiting on comms from the Pi (which would take longer than getting comms straight from the gps), but in this aragement there would be a much larger delay in getting location information to the Pi (which it needs for navigation and matching with gathered data) as the gps location is given as lattitude and longitude in degrees to 6 decimal points of accuracy, so 8 numbers would have to be transmitted rather than 2 as would be the case with speed. Thus it was decided to place the gps on the Pi, and for now have the speed not be PID controlled. This is because the use of PID control of the speed would be vastly complex and may not give any advantage (other than to know the boat has travelled at the same pace in different conditions - which may be useful for scientists to know that experiments have be kept under the same conditions, but this is generally of limited value***).

Also, for updating gps, this takes like 1 second at least, but is variable, so pid-ing speed would be hard. - this would cause iterations on arduino to be slow, or rely on multithreading***? which would be difficult/impossible on arduino.

High Level

Dei

Low Level

The low level control is put on the moteino to control the I/O (except for the gps, as explained above***). This is to prevent higher level computations from slowing down the updates of the rudder positions and motor speeds, as PID relies on fast iteration times to be effective***.

The design is to have a pid controller on the arduino which updates the rudder positions to ensure a smooth path is followed (rather than oscillating).

Comms

Implementation

This project is stored on github github.com/Tebazil12/AqASS for version control and to prevent loss of data in the event of a broken machine (other information relating to the disseration was kept at github.com/Tebazil12/diss-docs). The control system produced is called the Aquatic Area Scanning System (AqASS).

Arduino Code

The first section to implement was the low-level. This is because limitiations of the hardware can be better discovered when working at the lower levels than at the higher levels, and so starting with the higher level code would most likely result in incorrect assumptions being made about the movement of the boat.

Drivers

PID Controller

Pi-Meteino Communications

Python Code

Obstacle representation

Testing

this is some words hello world?

Overall Approach

Stuff

Unit tests

more stuff

Integration Testing

More more stuff

Evaluation

this is some words hello world?

diss