Members: Connor Strang, Adam Grosse, Brandon Guerrero, Igor Savchenko, Daniel Lee, Logan Courtright, Kevin Wegner

1. Abstract

UMBC Robosub is a new team aiming to complete 1 or 2 tasks reliably and focusing only on tasks we can reasonably complete for the competition. Our goal is to make a sub that performs reliably with a high degree of modularity. This Technical Design Report will discuss our design choices made to adhere to our goal and how it affected our design strategy and strategic vision.

2. Competition Strategy

As a new team, our main priority is to build a reliable and modular foundation for this and future competitions. We are focusing on developing a system with longevity in mind. Considering that this is the first year the sub will be competing, we have decided to set ourselves a bare minimum goal of qualifying, passing through the starting gate and following the path. We think these tasks will for the most part allow us to reuse successful computer vision code and therefore minimize time spent on developing involved solutions for the more complex tasks. This time can then be dedicated to testing the sub to ensure reliability. Our top priority is to make sure we maximize probability of the sub completing the qualifier and two tasks we have chosen. Whilst implementing anything, we also keep in consideration that modularity is another priority for us. This often is not easily done and will lead to varying increases in complexity. Our current capability is not just ready for qualification yet so time is being put into maximizing qualification success. Highly modular behavior software on main computer, allows rapid iteration of behavior methods. A second controller is implemented to handle low level sensor access allowing flexibility in selection of main computer based on increasing memory, processing needs, tight motor control loops, and data preprocessing if necessary. This is a first time barebones sub where successes, failures, and experience are all meant to be gained and used for future competitions. We seek to stray away from previous iterations of teams who have hacked together various methodology in a loose manner. We plan on coming back to the competitions year after year and also creating a system that can be passed down to new people. Our first real obstacle is qualifying and we hope to achieve that. Our next goal is to take anything we can from the competition to bring back and improve on our system. By achieving one goal at a time we aim to improve each iteration.

3. Design Creativity

The sub has a straightforward design allowing for a moderate level of modularity in terms of attachments. With this design it is also more manageable to fix and/or replace parts due to damage or updated architectural designs. It implements dual controllers to allow tasks to be processed in parallel depending on real-time usage.

4. Experimental Results

Our testing sessions are dependent on the recreational pool hours available and our ability to get enough team members together to move the sub to the pool. Both of these have presented challenges, limiting us to one 2 hour in-pool testing session for the sub each week. To fit into these constraints, we limit the mechanical and software features added each week to the number we think can be effectively tested in that short in-water testing session. However, the use of a separate sealed camera case and laptop webcams have allowed the software team to test computer vision without having to transport the entire sub or depend on it to be functioning.

5. Acknowledgements

This would not be possible without the support of various faculty and financial supporters. This would include department heads, the capstone system, IEEE support, and other donors to the team.

6. References

Adrian Rosebrock

Pyimagesearch

https://www.pyimagesearch.com/author/adrian/

Open Source Computer Vision Library Documentation

https://docs.opencv.org/3.4/index.html

Appendix A

Component	Vendor	Model/Type	Specs	Cost (if new)
Buoyancy Control				
Frame				
Waterproof Housing				
Waterproof Connectors				
Motor Control				
Propellers				
Battery				
CPU				
Programming Language 1		Python 3		
Programming		C++11		

Language 2				
Inertial Measurement Unit (IMU)				
Camera(s)		Logitech c310		
Algorithms: Vision				
Team Size		7		