Technical Design Report

By Mason Klein, Margot Adam, and Jasmine Kuo

Abstract

As a new team to the RoboBoat competition, we have much to learn and many things to refine about our robot. Our main strategy is to keep our robot simple and reliable, and not worry about challenges we are not able to complete. In designing and refining our robot, we have come across many challenges, and have created some unique solutions, such as using a polarizing filter over the camera. However, one of the largest challenges for our team has been meeting and testing time, as we are all heavily invested in school at the moment. But, thanks to our mentors and the donations we have received, our team has been able develop a robot that can compete in the competition.

Competition strategy

From the start, our mission was to focus on what we could accomplish first. Thus, we naturally decided to start with the easier challenges and work our way up from there. This decision led to the omittance of the lidar system, which we deemed as too complex and time-consuming considering our lack of experience. While we do have a lidar module, and have fashioned a mount that is able to spin and tilt, we came across many issues while trying to implement the mount in our system. Not only is it not currently waterproof, we were not able to receive reliable data from the lidar itself. Instead, we decided to devote our time and resources to refining other systems, such as vision, GPS, and compass, which we have found to be considerably more reliable. With the use of OpenCV, our robot can get camera feed and identify buoys based on color and shape[1]. Combined with the use of GPS and compass modules, we can navigate to certain sections of the water in addition to avoiding buoys. We decided that this setup would give our team a good foothold to at least attempt the introductory course, if not the regular competition challenges.

To increase maneuverability in the water, we decided to restrict the size of our robot. We have found that the size of our final design, as well as the shape of our pontoons aids our computer vision system, mainly by keeping the boat steady while it is on the water. Since our recognition algorithms are still being developed and refined, it is crucial that our boat stays as stable as possible. This year being our first competition, we wanted to keep the robot simple, even if that hindered our robot's ability to fully complete certain challenges. Our main goal is to learn more about the competition and how other teams have overcome the obstacles they have encountered, rather than competing to gain as many points as possible.

Design creativity

A. Early Prototyping

The initial design that our team had for the robot proved to have many issues, resulting in changes that are seen in the final design of the robot. Our first prototype consisted of a monohull design, which was not stable enough for the computer vision system to function properly. Our second prototype implemented a catamaran design, with two 6 inch diameter PVC pipes, one of which began leaking once in the water. Our final design now consists of a catamaran design, but with insulation foam encased in fiberglass. With this design, it is impossible for the pontoons to leak, and the robot can remain relatively stable while on the water.

Our first prototype was controlled by a Raspberry Pi, a lead-acid motorcycle battery, and two RC plane motors. The Raspberry Pi was very appealing because of its small size, and we happened to already own a few, but installing OpenCV was difficult and the processing was slow. This led to our use of a Mac-Mini instead, because Mac-Minis are still relatively small, have much more computing power, and are easily powered by a battery. We also changed the operating system of the Mac-Mini from MacOS to Ubuntu Linux for improved compatibility with our software and future possible use of Robot Operating System. Once we upgraded to a Mac-Mini, there was less space in the electronics box, so we also decided to upgrade to Lithium Polymer batteries as opposed to our lead-acid motorcycle battery. Our original decision to use RC plane motors was made because they are light and not very expensive. One of our team members had extensive experience with model plane parts, which made implementing the RC plane motors easier. However, after a few months of testing and development, the RC plane motors began to wear out and gather more seaweed in their bearings. While the motors were quite weak to begin with, this made the motors significantly weaker. We decided to upgrade to two T200 Blue Robotics motors, which have proven to be much stronger and have less of an issue with collecting seaweed from the water.

B. Final Design Solutions

Our team implemented many features to allow our robot to function well in certain conditions and have an adjustable design. In the initial design phases for the robot, we were not sure what final size and shape we would want for the robot. To solve this, we had four aluminium bars connect the two pontoons together, and used angle brackets and t-nuts that could slide along the whole length of the bars for adjusting. This made it possible for our team to change the width between the pontoons in addition to the length of the electronics box that they support in between the pontoons. In terms of obstacles on the water, our team came across an issue with the camera, in which many small objects in the background of our camera feed were being recognized as false positives, so our robot had a harder time only identifying the buoys. Instead of changing our computer vision thresholding of the image, we decided to use a polarizing camera filter over the camera itself. Because of the filter, the camera images we receive are slightly darker and contain less scattered light from the reflections off the water and false positives in the background, so our computer vision system can function better without changing our actual algorithms.

Another challenge that our team came across was that the electronics box turned into a large entanglement of wires once all of our systems were implemented. We were also using a breadboard to hold all of the wires, frequently leading to many loose wires and broken connections when moved. To solve this issue, our team decided to design a PCB, which allowed for a much more organized system of wiring that also ensured solid connections. With the use of the PCB, we could also safely plug in multiple batteries at the same time (protected by diodes and circuit breakers), and could power the microcontroller, motors, and the Mac-Mini.

Experimental results

In order to test our computer vision programs, we downloaded videos of robot feed from past competitions (Thanks GeorgiaTech!)[2]. We were then able to see how well our robot could detect buoys of different sizes and colors without having to construct a similar obstacle course. However, actual tests done in Seattle on Lake Washington proved to be very different from videos of Florida because of the different lighting, which has always been a challenge for our team. Many of the weather conditions are unfavorable for testing in the water and testing vision in general. In Lake Washington, there are also many waves and slight currents that can throw the robot off-course and make it challenging to hold a certain compass heading, as well as mess up the computer vision. In addition, long strands of seaweed can make their way into the bearings of the motors, which can slow down the motors and damage them. However, the largest issue we have found with testing is finding the right time to do so. Since one member is still in high school for eight hours a day, and the other two are working on college classes, meeting times usually take place during the evening and around sunset, which is not an optimal time to test the robot.

Acknowledgements

It is nearly impossible to accomplish a project like this alone. We would like to thank those who have graciously donated to the team because without them we would not have been able to buy the necessary parts to develop the project. A very special thanks to the Margot and George Pearson foundation for helping to cover the cost of travel from Seattle to Florida. We would also like to thank Margaret Adam, for allowing the development and storage of the robot to happen in her garage and basement, as well as supporting the team with whatever extra financial support was needed (not to mention proofreading many of our documents, such as the Technical Design Report). Credit should be given to Olivia Escobedo for designing our team logo, and Aaron Adam for designing our team t-shirts and helping develop our team website. We would also like to thank Sachin Mehta of University of Washington, who was a helpful mentor in developing our computer vision strategies. And last but not least, we are all very thankful for

Johann Adam, who was our main mentor, who spent countless hours supporting the team, for standing knee deep in lake water while holding practice buoys, and for teaching all of us important life skills.

References

[1] OpenCV website, read-the-docs tutorials. http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html

[2] Georgia Tech Summary Video 2016
https://www.youtube.com/watch?v=fAYEVduLsGI&app=desktop

Appendix A

Component	**Vendor**	**Model/Type**	**Specs**	**Cost (if new)**
ASV Hull form/platform	self constructed			100
Propulsion	Blue Robotics	T 200		169
Power system	Tattu	5200mAh Lithium Polymer		25
Motor controls	Blue Robotics	Basic ESC		25
CPU	Apple	Mac-Mini		
GPS	Adafruit	Ultimate Breakout MTK3339		39.95
Compass	Adafruit	LSM303DLHC		14.95
Camera(s)	Sony	Playstation		7.98
Algorithms	self constructed			
Vision	OpenCV			
Team Size	3			
Expertise ratio	1:3			
Testing time: simulation	15 hrs			
Testing time: in-water	5 hrs			
Inter-vehicle communication	Cypress	PSoc 5		15

Programming Language(s)	Spyder	Python		

Appendix B

Our team attended both the John Rogers Elementary school STEM night and the Jane Addams Middle School STEM night. There, we met parents and students, and displayed the computer vision algorithms, using face detection as an example of the strategies we are implementing in detect buoys. We had students draw pictures of faces to learn the characteristics used to find faces within an image. Outreach is also a useful way to recruit more people who are interested in our projects. Since most of the team used to belong to the Nathan Hale Robotics team, we also attended one of their meetings and presented our robot, inspiring them to move on to more program-oriented projects that implemented Python.