COST-EFFECTIVE REMOTE OPERATED VEHICLE

ZACHARY SMOLDER,
JINGANG YI (FACULTY ADVISOR)

ABSTRACT

Remote operated vehicles (ROVs) are robotic submersibles controlled typically by a person at the surface of a water body. ROVs can be applied to surveillance, environmental, and data recording jobs or tasks. The vehicle design may be modified to remove or add additional capabilities depending upon the specific purpose of the ROV. In this paper, we explore using remote operated vehicles as a cheap and affordable water exploration platform. ROV's high cost is a prohibitive barrier to entry, preventing widespread adoption of ROV for personal, research, and conservation uses. To address this problem, our paper explores a cost effective ROV with video capturing and directional control capabilities. Using state-of-the-art robotic technologies, a cost-effective competitive ROV is designed and constructed. This ROV was tested to a depth of 7 meters and has the potential to reach depths of up to 30 meters per its design.

1 INTRODUCTION

The field of robotics has always been known for high costs, especially when remotely operated vehicles (ROVs) are taken into context.[4] To reduce these costs, corners are typically cut. This can include implementing a ROV with a low resolution camera as well as building ROVs out of flimsy materials. Current ROVs on the market include Blue robotics, BlueROV2, and Geneinno's Tether Titan. These ROVs, similar to others, are expensive, typically costing more than one thousand dollars. Furthermore, much of the market is saturated at this high price point but with limited capabilities in the ROVs. Most of these ROVs are not able to be programmed and can only be controlled with the manufacturer's program or application. The idea of creating cost-effective robotics without compromising missions is one that we seek to explore throughout this research. Our target audience is the ROV community at large: specifically, our research is aimed at those who want an open-source, versatile, and quality ROVs sold at a cost-effective price. In addition to creating software to control the ROV, we seek to give the community the ability to add any water instruments and tools to the ROV, allowing the robot to be marketable towards research or conservation organizations. The goal of this project is to design and fabricate a cost-effective ROV system that has similar functionality and performance to its higher-priced counterparts. The process of designing and building the ROV as well as later design changes are detailed below.

2 METHODOLOGY

In order to facilitate the fabrication of the ROV, several criteria were specified to come up with the initial layout of the robot. Some of these initial ideas can be seen in the sketches in FIGURE 1. The ROV would have to have the following components:

1. Hull and outer structure;
2. Waterproof enclosure for electronics;
3. Thrusters; and
4. Electronics (including Electronic Speed Controller, ethernet connection, batteries, camera, central computer, and general signal cables)

After listing the initial specifications for the ROV, we explored prior work surrounding ROV designs. At first, the ROV design just had simple sketches to brainstorm potential candidates for what it could look like. The thought process used to create these simple designs revolved around past ROVs already created; these designs would be altered based upon technological specifications required for our purposes.

One similarity between the sketches and past designs was how the four thrusts were mounted on the side. This allowed for easy directional control and several industry ROVs use this design for this reason. A change that was made to the industry-inspired design was the easily accessible electronics compartment in the center of the ROV. In these sketches, the original idea was to have a hatch that opened to allow for easy access on the bottom. FIGURE 1 shows a rough sketch that includes preliminary thoughts based upon initial research. The vehicle was sketched as shown so that one could see side, front, and top-down views.

Numerous issues were present in some of these initial ideas. Some of these issues included the ROV's inability to fit in a waterproof enclosure. The hatch idea was scrapped since waterproofing a seal around the hatch is challenging. Additionally, these initial sketches had no clear method on how to secure the motors. At this time, the ROV was conceptualized to have five motors: the ROV would have four motors for depth and turning control and one motor for horizontal propulsion.

In FIGURE 2, there are several conceptualizations of the future designs that pertain to various parts created after extensive thought and research. The final design was one that incorporated several different ideas that we learned after much thought and deliberation.

This thought process included the fact that we reduced the motor count to four since the fifth motor was ruled nonessential. The rationale behind this decision was that it would just get in the way, making the overall ROV more expensive and heavier with no navigational benefit. This conclusion was met after realizing that the existing four thrusters could be arranged in a way to meet x y z navigation. Furthermore, a more cylindrical design was decided upon which revolved around the waterproof enclosure center. The research shows that creating a unique shape with materials other than cast acrylic would be more expensive than solely building from a cast acrylic cylinder.

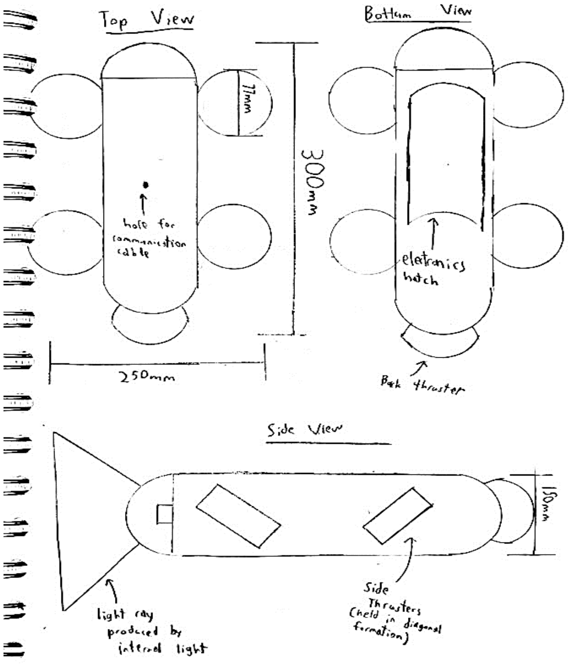

Top View
77mm
300mm
hole for communication cable
250mm
Bottom View
eletronics hatch
Back thruster
Side View
150mm
light ray produced by internal light
Side Thrusters (held in diagonal formation)

FIGURE 1: *Initial basic sketches of the ROV design which includes a 5-thruster format.*

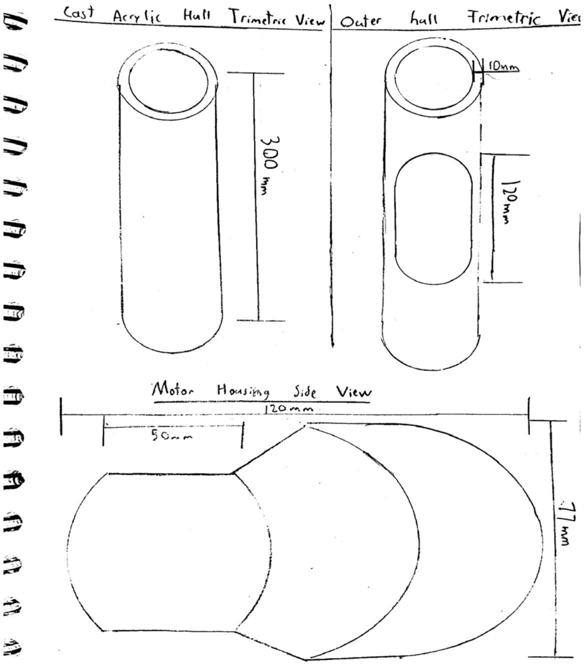


FIGURE 2: *Basic sketches of part designs including motor housings and the hull that would go over an acrylic cylinder.*

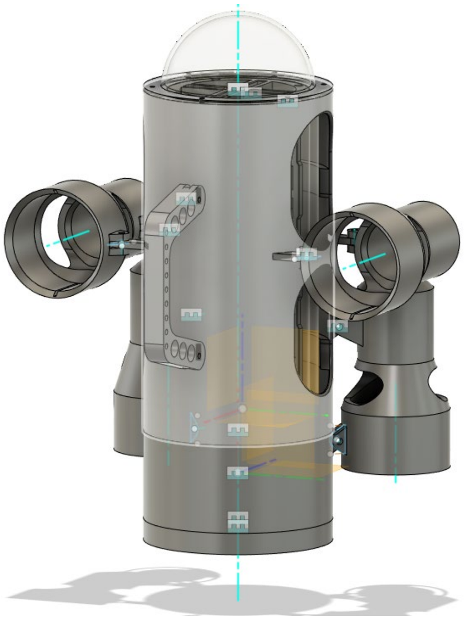


FIGURE 3: (ISOMETRIC VIEW) *Final CAD design of the ROV which uses a 4-motor design format.*

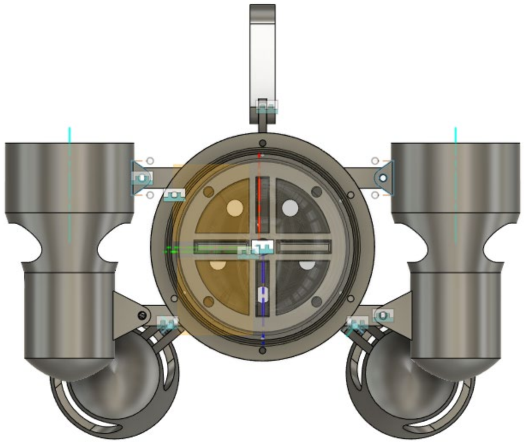


FIGURE 4: (FRONT VIEW) *This view shows the inside of the clear acrylic cylinder which will house the electronics.*

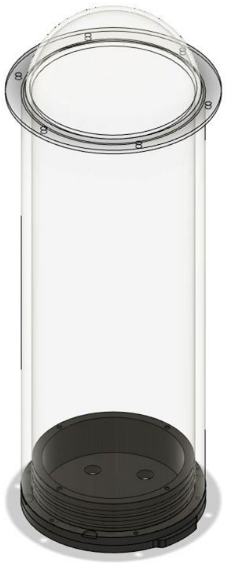
Based on this research, the simpler the design was, the more efficient the ROV would be. This observation was a design philosophy that supports the use of both the fewest thrusters and making the hull as easy as possible to manufacture in the name of cost.

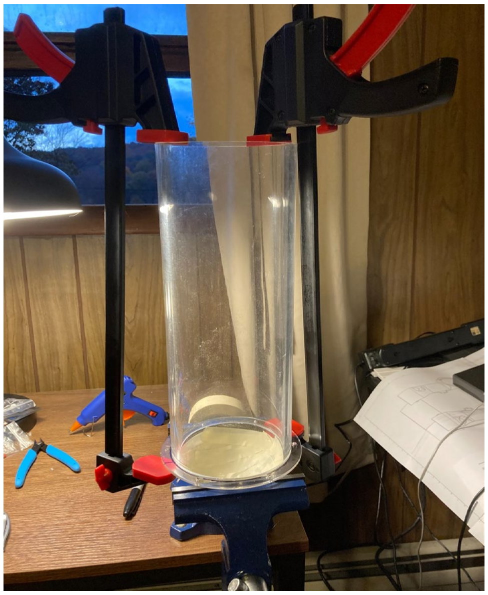
This final design incorporates, in addition to previously mentioned design concepts, an acrylic enclosure in the center of the ROV with a holding bracket for the electronics. Furthermore, the four thrusters are held in a two-part forward-backwards thrust-based system, in tandem with a rise-and-dive based system. Two of the motors are held vertically for rise and dive maneuvers. This is done alongside two motors held horizontally for forwards and backwards control. This final design stands to be efficient due to features like its cast acrylic center, as this reduces build time and stands to still be extremely strong. Furthermore, the motor configuration also allows for only four motors to be necessary while not compromising stability or navigational ability. FIGURE 3 and FIGURE 4 show CAD models of the final design based upon these constraints of motor configuration and a cast acrylic center.

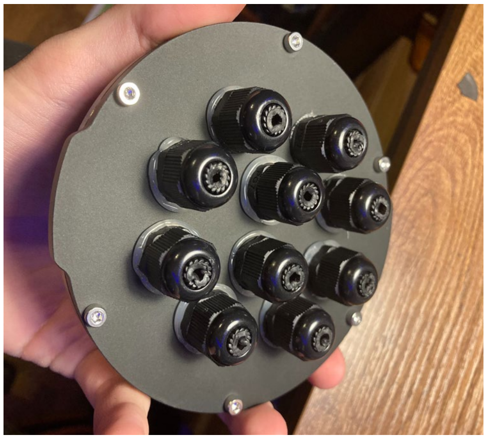
i. Hull Design / Waterproof Enclosure

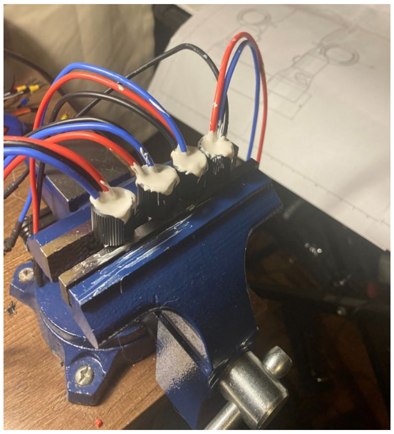
The inner acrylic enclosure was the priority for manufacturing. This was because all other parts of the ROV would revolve around this central design element. In the end, cast acrylic was chosen to be used for the physical cylinder and half sphere cap. These two parts are the objects shown in FIGURE 5, and in the case of this specific ROV design, they are responsible for holding the electronics. The cast acrylic was chosen due to its strength and ability to be welded together to make a final product. Weld-On 4 was used to properly seal the cylinder to the cap. A metal cap, manufactured by an outside party due to time constraints, was attached to the opposite side of the waterproof enclosure. FIGURES 5-7 show the design, fabrication, and welding process of the Water Proof Enclosure.

The metal cap on the end opposite the dome of the waterproof enclosure was manufactured to be removable and enable cable wires to pass through to outside components of the ROV. Several ideas were considered to allow a waterproof fitting around the cables to prevent water leakage. Originally, we planned to use standard weatherproof wire covers with hot glue. This was a failure









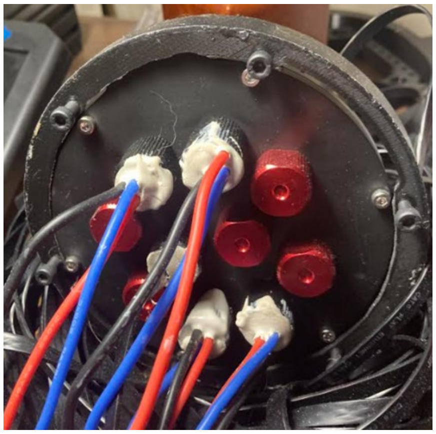


FIGURE 5: (TOP LEFT) *This view showcases the CAD model of the acrylic cylinder and dome when assembled.*

FIGURE 6: (TOP CENTER) *The process of welding the acrylic dome to the acrylic cylinder required clamping and the use of* Weld-On 4.

FIGURE 7: (TOP RIGHT) *The two halves of the acrylic are now welded together and ready for a hull to be attached.*

FIGURE 8: (BOTTOM LEFT) *Initially weatherproof wire covers were used and they were found to not work adequately for the pressure cables would experience at desired testing depths.*

FIGURE 9: (BOTTOM CENTER) *The now potted cables need to dry for 48 hours in order to form a hardened seal around the cables and in the glands.*

FIGURE 10: (BOTTOM RIGHT) *The dry glands were inserted into the endcap which separates the water from the rest of the electronics in the electronics enclosure.*

upon testing, which can be seen in FIGURE 8. The hot glue leaked water after multiple tests and the weatherproof wire covers did not form a decent seal with the cables with which they interfaced. We instead decided to use the conventional approach of "potting", as shown in FIGURE 9. "Potting" is when one sits cables in manufactured glands and fills the glands with epoxy to completely waterproof them. This allows for a permanent, waterproof bond around all the cables. FIGURE 10 shows these glands put into the end cap.

During the beginning phases of deciding upon materials and the fabrication processes of the hull, cost-effectiveness was kept in mind. This was to continue the overall pursuit of a cost-effective ROV product. 3D printing was used to create the outer shell of the ROV. Due to its cheap yet sturdy nature, Polylactic Acid plastic (PLA) made a great candidate for materials. FIGURE 11 shows a 3D-printed hull over the acrylic electronics enclosure.

PLA was used to 3D print the main hull and motor housings. Then, epoxy resin was used to make the PLA stronger and create an even plane on which the water can glide, as shown in FIGURE 12. This figure shows parts of the hull after having a fresh coat of polyester resin applied as well as the paint rollers used to apply this resin. After this process, the resin was sanded and polished evenly. A body filler coat was applied to fill in wherever the resin was not properly leveled. Finally, black spray paint was used to cover the finished product. FIGURE 13 shows all parts of the hull to be assembled.

ii. Propulsion Design

Propulsion was a serious challenge both in terms of cost and power. The ROV required a lot of motors based upon the design. The power constraints were also a serious issue. The decision to use 64mm 5-rotor 4500 KV brushless motors was made. The rationale behind this choice was that these motors had to be brushless to function underwater. These motors provided enough thrust with balanced power consumption to be wise choices (each motor is using power efficiently). We designed and fabricated a 3D-printed motor housing that allowed the motors to function perfectly in their desired locations. This housing can be seen in the middle of being 3D printed in FIGURE 14.

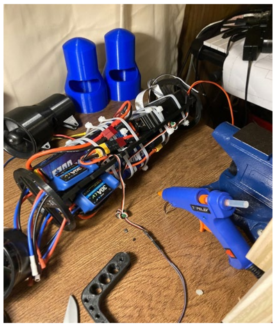
iii. Electronics Design

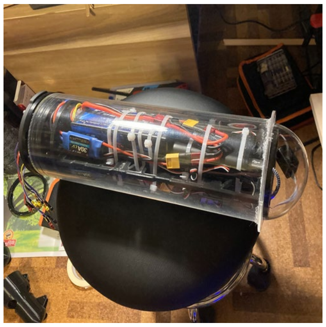
Several design decisions had to be made during the electronics and electronics tray fabrication process. The electronics tray had to be created so that it would fit in the electronics enclosure and put the computer, batteries, and electronics speed controllers (escs) in reasonable and safe places repetitively. In the end, a 3D-printed form factor or ergonomic design proved to be successful. This tray had holes to allow for zip-tying to keep cables from being loose and threading places to put wire through. FIGURE 15 shows how the tray looked with electronics included on it.

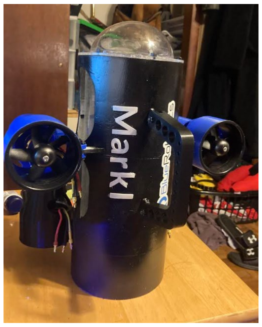
For the electronics, a Raspberry Pi 4 was used as the brain of the whole ROV. This computer oversees running code, sending signals to parts of the robot, and, thus, controlling the robot's movement and the use of electronic components. The Raspberry Pi 4 is a cheap yet powerful computer that worked well with the robot's limited internal space while still allowing for uncompromising computing performance. In addition to these strengths, the Raspberry Pi 4 allows for users to connect any additional instruments to the ROV easily by just connecting them to the Pi and writing code. Two 5200 mAh LiPo batteries were used to power the four motors. Finally, one 1000 mAh lithium-ion battery was used for the Raspberry Pi 4. Motor bullet connectors were soldered in-house. Batteries, motor leads, and the ROV computer were wired. This was then finally put on the tray and into the enclosure.

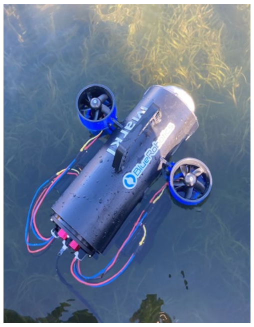
iv. Coding

The script used for the control of the ROV was coded on Python due to the power and simplicity of the language. The ROV was controlled by connecting a Raspberry Pi 4 in the ROV to a Raspberry Pi 4 on land by an ethernet cable. After a connection was established, software was used to see the desktop environment of the ROV computer. Once in this environment, the control script was run on the ROV's computer. This script functioned by sending PWM signals to the motors by branching if-else command









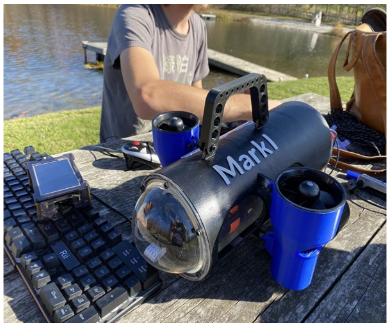


Figure 11: (TOP LEFT) *The hull was covered in car filler to properly smooth out any imperfections from the polyester resin process, hence its gray color. This hull is soon ready to be attached to the acrylic cylinder.*

Figure 12: (TOP CENTER) *The parts of the hull were given polyester resin and left to dry for days to create a strengthened and smooth surface.*

Figure 13: (TOP RIGHT) *All the parts can be seen ready to put together with metric screws.*

Figure 14: (MIDDLE LEFT) *A motor housing is in the middle of being 3D-printed.*

Figure 15: (MIDDLE CENTER) *The electronics tray is fully loaded up with batteries, computer, camera, and electronic speed controllers.*

Figure 16: (MIDDLE RIGHT) *The electronics tray has been put together inside of the electronics enclosure.*

Figure 17: (BOTTOM LEFT) *The ROV is fully assembled and ready for testing and implementation.*

Figure 18: (BOTTOM CENTER) *The ROV's buoyancy is very positive but the robot exemplifies adequate horizontal control.*

Figure 19: (BOTTOM RIGHT) *The ROV is being connected to initially with a separate control computer in order to begin operation.*

structures. Once a forward, left, right, or dive command was sent via the control computer, the ROV would respond by sending PWM signals to each of its respective motors. This code proved to be effective and can be seen in the appendix.

3 WATER TEST RESULTS

Once the fabrication was complete, testing took place. Four testing iterations were conducted at two locations. All of these tests had sunny weather conditions with low water motion. During the first of these tests, buoyancy was measured on the ROV. The observations centered on whether or not the ROV was positively or negatively buoyant and by how much was the ROV buoyant.

The ROV was found during this first test to be 500 grams positively buoyant. This issue was resolved through the attachment of a diver belt to the ROV with 500 grams on the belt.

During the second test, horizontal thrust and navigation was tested at a shallow location. Now that the buoyancy was fixed, the ROV stayed at a specific depth underwater. The robot during this test was able to perform fantastic left, right, and forward movement operations. This test was deemed successful because the ROV was able to triumphantly navigate underwater and maintain a waterproof electronics enclosure while doing so. The ROV was designed to function for about a half hour and at depths of 30 meters; however, the ROV was never fully tested at depths of 30 meters. It was successfully tested at depths of about 7 meters due to the unavailability of a deeper body of water within time constraints. With increased depth comes increased pressure, which is why we feel it is necessary to physically test at 30 meters before deeming the ROV fully capable in its depth-traversal abilities.

During the third and fourth tests, the full capabilities of the ROV were tested. Depth as well as horizontal maneuverability was observed and analyzed. The ROV during both these tests was able to exemplify left, right, forward, rise and dive capabilities at depths of 7 meters. At this depth, there were no traces of water found in the robot's electronics enclosure, and the camera used for vision was still in complete working order. Other ROVs on the market have been rated to go depths of 7 meters or more, making the robot unsuccessfully validated to beat the depth rating of other ROVs for a competitive price. However, we still deemed these tests a success because the robot was able to strongly navigate and operate for about 100 minutes at a time, beating our original 30-minute estimate. The reasoning behind these results was the robot's ability to control the speed of its motors instead of running full throttle for the full thirty minutes. This operation time is comparable to several similarly priced ROVs like the ThorRobotics' TenchRover, which costs $1,380 and can operate for 120 minutes during a single session.

During the tests, several aspects of the product were both challenged and validated. In regard to the buoyancy issue, we addressed this by attaching a diver belt to the ROV each testing time, this was an effective solution to a serious problem. It did, however, have room to improve as the belt can shift during operation. In the future, weights should be implemented into the hull physically, such as in extra pockets in the hull for weights to be inserted or making the hull thicker. This will make the buoyancy not shiftable during operation and result in a cleaner design. Alternatively, if we were to keep the original hull design, we could make a form factor weight to secure in the electronics enclosure. This could accomplish similar results to redesigning the hull.

In addition to the buoyancy issue, propulsion could have been improved to be more powerful. This extra power could help the ROV navigate better in turbulent waters. Communication could have been made better in the sense of cost. Currently, the ROV requires a second computer for control, but this could be solved through the use of a smartphone instead.

One validating aspect of the tests was the ability of the ROV to stay completely waterproof during operation. All electronics were left dry after post-test inspection in the waterproof enclosure. This inspection included visually and electronically testing each of the electronic components inside of the ROV after each test. Additionally, the ROV was able to showcase operating times of up to 100 minutes which was similar to slightly higher priced ROVs. Furthermore, the ROV was able to navigate at distances of 10 meters horizontally from shore. The biggest limiting factor in the ROV's ability to navigate at large distances is its ethernet cable length, which, for more money, could be adjusted for farther expeditions. The ROV was recorded being able to display uninterrupted video feed while underwater with its camera. Navigation at the speed of 1 knot or .5 m/s was recorded. Due to the fact that this speed was recorded all in favorable environmental conditions, it should be noted that ROV performance in rough waves would result in lower speed measurements. Altogether, the robot was able to exemplify competitive battery life, navigational control, and waterproof capabilities, deeming it to be an outstanding ROV.

4 Discussion & Conclusion

Looking back at the work over the past several months, we felt that the project has done a lot in terms of cost-effective robotics solutions. On average, ROVs cost about between $1000 and $4000. This robot costing $900 means that it is cheaper than even the lowest priced commercial solution. This low cost allows for a profitable yet cost-effective sale price. This project has been successful in creating a cheap and capable ROV. Due to the ROV's open-source design, adding additional instruments would be as easy as just wiring these tools to the ROV's internal Raspberry Pi 4 and writing code to use these tools.

Several ideas can be focused on as examples of creating cost-effective ROVs. Design methods, such as that of designing around a cylinder-shaped piece of cast acrylic, proved fruitful in being cost effective while not sacrificing waterproof capabilities. Additionally, the idea of 3D printing a hull structure around the cast acrylic proved to be good in allowing for a cheap and versatile platform to which components can attach. Also, 3D printing was found to be a promising new technology that, when treated, was an inexpensive and strong waterproof material for parts attached to the hull. Simplicity was the main guide for design decisions and was found to be a helpful philosophy when keeping costs low while keeping uncompromised capability.

SIMPLICITY WAS THE MAIN GUIDE FOR DESIGN DECISIONS AND WAS FOUND TO BE A HELPFUL PHILOSOPHY WHEN KEEPING COSTS LOW WHILE KEEPING UNCOMPROMISED CAPABILITY.

Altogether, the project validates that it is possible to cut costs while still delivering a competitive ROV. The result of the efforts of this project culminated into a robot that was able to have battery life comparable to other ROVs with functional directional control. This was done at a fraction of the cost compared to many other ROVs and, therefore, has implications to the ROV community. We have shown that ROVs can be produced and sold with decent functionality at competitive costs.

The project was successful in meeting our original criteria. The ROV was able to properly navigate waters with decent battery life and functional watertight compartments all at a cost lower than other ROVs in the market. The project has several limitations that could be improved and changed in the future. This includes propulsion, communication, and buoyancy, which all have room for improvement in the current design. Propulsion could be modified so that the thrusters are held more effectively and are more powerful. Communication could have been improved on the robot, as the ROV could have been made to be controlled with a phone, and, assuming the user already has a phone, would further reduce costs. Furthermore, if the buoyancy was manufactured into the hull instead of being corrected with a dive belt, the ROV would operate more efficiently since there would be no chance of shifting weight. All these places for improvement show several branches for further research and work to be done to explore these ideas■

5 ACKNOWLEDGEMENTS

We would like to express appreciation for the project sponsor Blue Robotics Inc. for giving a discount on the water enclosure cap. Furthermore, we would like to thank the faculty mentor professor Jingang Yi for editing the report and for guidance on the project.

6 REFERENCES

[1] Aguirre-Castro, O. A., Inzunza-González, E., García-Guerrero, E. E., Tlelo-Cuautle, E., López-Bonilla, O. R., Olguín-Tiznado, J. E., & Cárdenas-Valdez, J. R. (2019). Design and construction of an rov for underwater exploration. Sensors (Basel, Switzerland), 19(24), 5387–.

[2] Ahmed, Y., Yaakob, O., & Sun, B. (2014). Design of a new low cost ROV vehicle. *Jurnal Teknologi*. 69.

[3] Aniwaa. (2021) "The 11 Best Underwater Drones in 2021." *Aniwaa.com*, retrieved 12 Jan. 2021 www.aniwaa.com/buyers-guide/drones/best-underwater-drones/

[4] Cherdo, L. (2020) "Best Underwater Drones 2020: the 13 Best ROVs This Year." *Aniwaa*

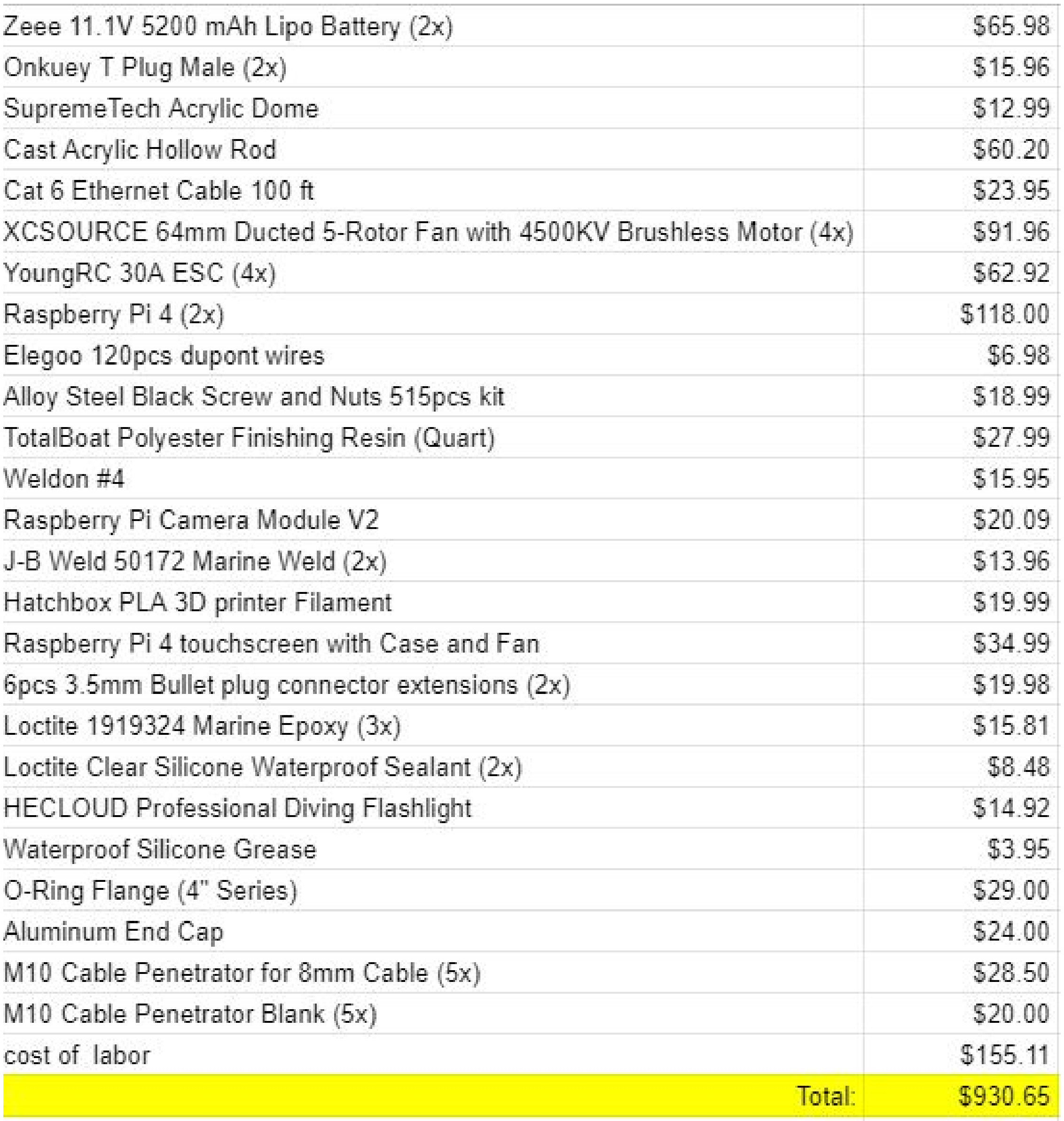
[5] Martos, G., Abreu, A., Gonzalez, S., & Tremante, A. (2013). *Remotely Operated Underwater Vehicle (ROV) 100% Report* (Unpublished B.S. Thesis). Retrieved from https://mme.fiu.edu/wp-content/uploads/2013/12/F13-SR-T-1.pdf

[6] Rahimuddin et al. (2018) J. Phys: Conf. Ser. 962 012017

Zachary Smolder is a Rutgers School of Engineering sophomore. He is majoring in mechanical engineering and plans to pursue a career in entrepreneurship and film. He is very passionate about robotics and their multivariable applications. Another interest of Zachary's is the great issue of microplastics in the world's oceans which he hopes to help out in during his career. Throughout quarantine, Zachary decided that due to flexibility and freedom in his schedule he would do independent research during the Fall 2020 semester. Under the guidance of Professor Chigang Yi, Zachary was able to complete his project while at home and working remotely. Zachary is glad for the opportunity to be able to do research during COVID and looks forward to continuing his research endeavors while on campus.

7 APPENDIX

Below is the total cost and expenses for the project. It is important to note that the ROV only cost $930.65, seeing as many ROVs in the modern day cost thousands of dollars.[3] This expense chart sums up the true innovation and benefit of the design. Note that in order to calculate the cost of labor, a rate of $12.95 was used for a total labor time of 12 hours which would be enough in tandem with 3D printings and other tools to assemble the robot.



Zeee 11.1V 5200 mAh Lipo Battery (2x)	$65.98
Onkuey T Plug Male (2x)	$15.96
SupremeTech Acrylic Dome	$12.99
Cast Acrylic Hollow Rod	$60.20
Cat 6 Ethernet Cable 100 ft	$23.95
XCSOURCE 64mm Ducted 5-Rotor Fan with 4500KV Brushless Motor (4x)	$91.96
YoungRC 30A ESC (4x)	$62.92
Raspberry Pi 4 (2x)	$118.00
Elegoo 120pcs dupont wires	$6.98
Alloy Steel Black Screw and Nuts 515pcs kit	$18.99
TotalBoat Polyester Finishing Resin (Quart)	$27.99
Weldon #4	$15.95
Raspberry Pi Camera Module V2	$20.09
J-B Weld 50172 Marine Weld (2x)	$13.96
Hatchbox PLA 3D printer Filament	$19.99
Raspberry Pi 4 touchscreen with Case and Fan	$34.99
6pcs 3.5mm Bullet plug connector extensions (2x)	$19.98
Loctite 1919324 Marine Epoxy (3x)	$15.81
Loctite Clear Silicone Waterproof Sealant (2x)	$8.48
HECLOUD Professional Diving Flashlight	$14.92
Waterproof Silicone Grease	$3.95
O-Ring Flange (4" Series)	$29.00
Aluminum End Cap	$24.00
M10 Cable Penetrator for 8mm Cable (5x)	$28.50
M10 Cable Penetrator Blank (5x)	$20.00
cost of labor	$155.11
Total:	$930.65

FIGURE 20: *Detailed expense and cost report showing the total price of the ROV's construction.*

Here is a comparison chart further showing the great value our ROV has over more expensive competition. While the ROVs below have better specifications than our ROV in some ways, their ROVs are not open source and cost significantly more than ours for minimal additional capabilities. The only exception to this is Blue Robotics' BlueROV2, which has open-source abilities but at a price four times higher than that of our ROV.

Notilo Plus iBubble (cost is $4,499)

- Not open source
- Battery life of 60 minutes
- Diving depth of 60 meters

Geneinno T1 (cost is $2,499)

- Not open source
- Battery life of 240 minutes
- Diving depth of 150 meters

Aquarobotman Nemo (cost is $1,799)

- Not open source
- Battery life of 180 minutes
- Diving depth of 100 meters

BlueROV2 (cost is $4,268)

- Is Open Source
- Diving depth of 100 meters

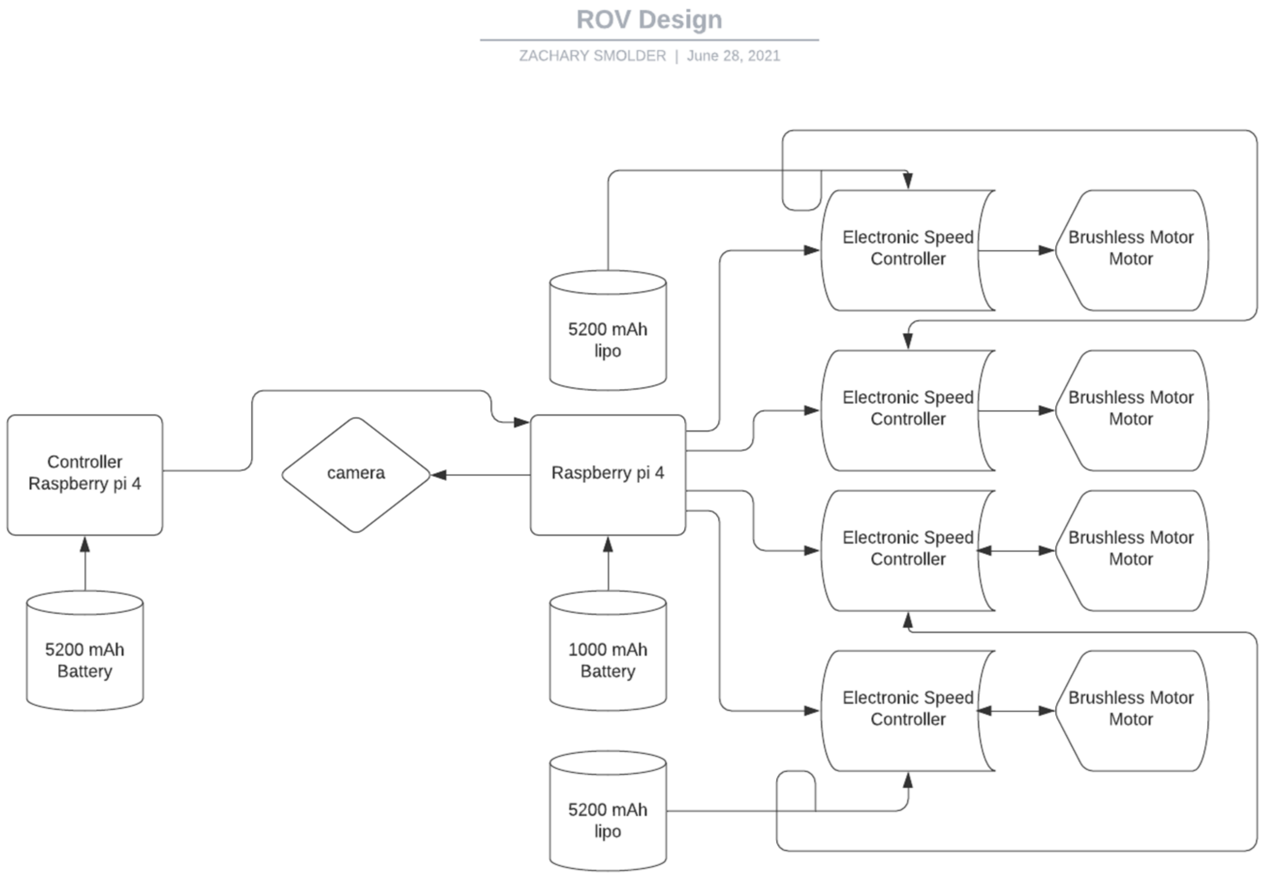


FIGURE 21: *Schematic showing the wiring of the ROV by component. This was used to have a better overview of how to wire the ROV during construction.*

PYTHON CONTROL SCRIPT

```
import os
import time
os.system ("sudo pigpiod")
time.sleep(1)
import pigpio
import picamera
import pygame

camera = picamera.PiCamera()

esc1 = 27
esc2 = 17
esc3 = 5
esc4 = 6

pi=pigpio.pi()
pi.set_servo_pulsewidth(esc1,0)
pi.set_servo_pulsewidth(esc2,0)
pi.set_servo_pulsewidth(esc3,0)
pi.set_servo_pulsewidth(esc4,0)

max_value = 1600
min_value = 500
print ("initiating launch sequence")

def calibrationsequence():
    pi.set_servo_pulsewidth(esc1,0)
    pi.set_servo_pulsewidth(esc2,0)
    pi.set_servo_pulsewidth(esc3,0)
    pi.set_servo_pulsewidth(esc4,0)
    print("press enter to confirm calibration")
    inp = input()
    if inp == '':
        pi.set_servo_pulsewidth(esc1,max_value)
        pi.set_servo_pulsewidth(esc2,max_value)
        pi.set_servo_pulsewidth(esc3,max_value)
        pi.set_servo_pulsewidth(esc4,max_value)
        print("press enter again")
        inp = input()
        if inp == '':
            pi.set_servo_pulsewidth(esc1,min_value)
            pi.set_servo_pulsewidth(esc2,min_value)
            pi.set_servo_pulsewidth(esc3,min_value)
            pi.set_servo_pulsewidth(esc4,min_value)
            print ("One moment")
            time.sleep(10)
            pi.set_servo_pulsewidth(esc1,0)
            pi.set_servo_pulsewidth(esc2,0)
            pi.set_servo_pulsewidth(esc3,0)
            pi.set_servo_pulsewidth(esc4,0)
            time.sleep(2)
            print("arming")
            pi.set_servo_pulsewidth(esc1,min_value)
            pi.set_servo_pulsewidth(esc2,min_value)
            pi.set_servo_pulsewidth(esc3,min_value)
            pi.set_servo_pulsewidth(esc4,min_value)
            controlsequence()

def controlsequence():
    print ("thrusters engaging")
    time.sleep(1)
    speed = 500

    print ("+ for increase, - for decrease")
    while True:
        pi.set_servo_pulsewidth(esc1,speed)
        pi.set_servo_pulsewidth(esc2,speed)
        pi.set_servo_pulsewidth(esc3,speed)
        pi.set_servo_pulsewidth(esc4,speed)
        inp = input()
        if inp == "-":
            speed -= 100
        elif inp == "+":
            speed += 100
        elif inp == "0":
            stop()
        else:
            print ("invalid control")

def forward():
    pi.set_servo_pulsewidth(esc1,max_value)
    pi.set_servo_pulsewidth(esc2,max_value)
def left():
     pi.set_servo_pulsewidth(esc1,max_value)
     pi.set_servo_pulsewidth(esc2,0)
     pi.set_servo_pulsewidth(esc3,0)
     pi.set_servo_pulsewidth(esc4,0)
def right():
    pi.set_servo_pulsewidth(esc1,0)
    pi.set_servo_pulsewidth(esc2,max_value)
    pi.set_servo_pulsewidth(esc3,0)
    pi.set_servo_pulsewidth(esc4,0)
def dive():
    pi.set_servo_pulsewidth(esc1,800)
    pi.set_servo_pulsewidth(esc2,800)
    pi.set_servo_pulsewidth(esc3,max_value)
    pi.set_servo_pulsewidth(esc4,max_value)

def launch():
    print ("beginning underwater exploration")
    print ("control with wasd")
    camera.start_preview(fullscreen=False,
window=(100,20,640,480))
    pi.set_servo_pulsewidth(esc1,0)
    pi.set_servo_pulsewidth(esc2,0)
    pi.set_servo_pulsewidth(esc3,0)
    pi.set_servo_pulsewidth(esc4,0)
    while True:
        inp = input()
    if inp == "w":
        print ("forward")
        forward()
    elif inp == "a":
        print ("left")
        left()
    elif inp == "d":
        print ("right")
        right()
    elif inp == ("s"):
        print ("dive")
        dive()
    else:
        print ("finish launch")
        stop()

def stop():
    pi.set_servo_pulsewidth(esc1,0)
    pi.set_servo_pulsewidth(esc2,0)
    pi.set_servo_pulsewidth(esc3,0)
    pi.set_servo_pulsewidth(esc4,0)
    pi.stop()

def startup():
    print ("launch sequence...press 1 to start, press 2
to begin system control, and press 0 to abort")
    inp = input()
    if inp == "1":
        calibrationsequence()
    elif inp == "2":
        launch()
    elif inp == "0":
        stop()
    else:
        print("abort")

startup()
```