Amador Valley High School Robotics (AVBotz): Design of Marlin V2 AUV 2023

Justin Yu (President); Mechanical: Isabelle Lo (VP), Aadi Dahake, Aayush Gupta, Dylan Kwong, Kaustubh Lole, Meghana Kumar, Nathan Lin, Nikhil Barghava, Rachel Schmidt, Thomas Iordache, Tim Hsueh; Electrical: Eaton Huang (VP), Aarav Jain, Alex Vovenko, Allen Ding, David Li, Gaurav Gupta, Hunter Selkow, Raghav Pottabathini, Rajas Paranjpe, Samuel Wang, Sandilya Garimella, Sean Kadkhodayan, Shaurya Shrivastava, Shravani Kurapati, Tanusha Kolli, Zachary Cheng; Software: Craig Wang (VP), Aarav Singh, Abhinav Garg, Andrew Xiao, Angad Bhargav, Dhanvanth Rajesh, Erick Liu, Gregory Li, Jason Wu, Jerry Peng, Kush Nayak, Nikhil Sunkad, Ryan Fu, Ryan Yang, Vansh Chabra; Business: Richard Bai (VP), Arran Gill Gulati, Ayushi Deokule, Gyan Bhambhani, Janav Rakesh, Leland Chung, Nikash Ramakrishnan, Sarah Zhang

***Abstract*—Introducing Marlin v2, the second generation of our submarine. Building on last year's experience, we decided to focus on adding grabber and torpedo systems. Following our principles of creating modular, logical, and intuitive designs to mitigate the increase in complexity, we built new mechanical grabber and torpedo systems, more secure electrical backplane and connector systems, and software updates of computer vision, cascaded motor control, and simulation to test grabbing and shooting.**

I. Competition Strategy

Last year, we prioritized limiting complexity to core tasks such as the gate, buoy, and bins, but we were not able to complete the full range of the course. This year, our vision is to instead embrace complexity, by also attempting grabbing and torpedo tasks. The mechanical division this year then designed a grabber and torpedo shooter, adding complexity which we found requires a significant amount of work to become reliable. The software division implemented a cascaded position and velocity controller for precise movement necessary for torpedo and grabbing tasks, while the electrical division focused on creating reliable systems that connect mechanical and software.

A. Gate

The gate task was a high priority, as it is necessary to initiate the course. We plan to attack this task by relying on our machine learning pipeline, which inferences on live images and provides angular and distance information to our mission planner. With our new cascade controller, this also allows for more precise movements to add increased functionality, such as possibly saving time by spinning while moving forward.

B. Buoy

The buoy task is similar to the gate task, in that they both involve the submarine moving through a detected object. As a result, the navigation code for both tasks are nearly identical, as we can reuse fundamental vision pipeline and movement building blocks to limit complexity. We also prioritized detecting the path marker, by implementing a filter to restore underwater colors to improve reliability.

C. Bin

Because there is a path marker from the buoy to the bins, we can reuse our vision and navigation software for this task as well, limiting complexity. This year, with a new grabber from the mechanical subdivision, this combined with a software approach of continuously aligning to the bin lid with computer vision and offsetting the submarine to pick it up. This required immense testing in the simulator to develop, but the synergy between the mechanical and software designs provides a possible approach for us to attack this task.

D. Torpedo

The torpedo task was extremely exciting to our team because of the challenge of precise alignment and building an accurate launcher. Once again, this required intense collaboration between the mechanical and software divisions, with almost neutrally buoyant 3D printed torpedoes and a mechanical spring-based torpedo shooter designed to be functional, easy to use, and intuitive. This combined with the software subdivision, which prioritized developing new image processing techniques to calculate the orientation of the torpedo board, so that we could align perpendicular to the board to maximize our chances of success.

E. DHD

The DHD task this year was the most challenging for us to attempt. Mechanical 3D printed a grabber with three fingers but modified it to four fingers after testing in the simulator, to account for different orientations that the grabber can hold the unique shape of the chevron. Software focused on intensive simulation testing of the computer vision and navigation code required to detect the chevron and align to it, which was an intensely frustrating yet engaging challenge. For this task, the software team developed new ways to orient the submarine to attempt to avoid accidentally crashing the grabber into an octagon divider while attempting to grab the chevron securely, adding significant complexity to our system. We are currently in the process of testing to ensure reliability.

II. DESIGN CREATIVITY

A. Mechanical Subsystem

1) Marker Dropper: To address many of the issues with last year's marker dropper–inconsistent releasing, drifting and an overall inefficiency–the mechanical team completely redesigned the marker dropper to be simpler and more effective. Rather than using a cam to push the markers out, which worked inconsistently, we switched to a much simpler method: gravity. The golf ball markers were switched out for a more hydrodynamic teardrop shape each with a heavy ball bearing at the bottom, greatly lowering the center of gravity and ensuring vertical travel. A double barrel houses both markers and helps with vertical alignment. This servo's arm is a sector shaped plate that rotates, allowing either marker drop through.

2) New Torpedo Launcher: In the past, we have experimented with air-propelled torpedoes which we found to be unpredictable and difficult to control. After a grueling research process, the team decided on mimicking a gun trigger mechanism. The system is composed of 3D printed housing, printed lever, printed torpedoes, springs, and a servo. The torpedoes are placed inside the 3D printed base and locked into place using the lever controlled by the servo. The servo locks the torpedoes in place, simultaneously creating an elastic force fowards as the springs are compressed. Grooves on the enclosure fit to the fins help ensure the straightest path of travel.

3) New Grabber: Another task we were not able to attempt last year were the grabbing tasks. In the past, we had a machined arm that was bulky and clumsy due to its size. Thus, we started on a complete redesign. After testing in simulations and with prototypes, we decided that a 2 finger, non-interlaced design was the most efficient given this year's grabbing tasks as it gave us enough surface area to hold on to objects without being too large and potentially obstructing the camera or DVL. Two vertical machined plates allow the grabber to reach the bottom of the frame without making the arms extremely long as the previous design had. The servo controls one arm which then opens the other arm through a 1:1 gear train, using herringbone gears that help the arms align perfectly.

4) Modular Frame: A noticeable flaw of the original Marlin's frame was its lack of flexibility and modularity with irregularly shaped holes permanently machined into the frame. This made it extremely difficult for us to find spots to mount updated components.. Therefore with Marlin's new frame, it became an important objective to increase modularity which we achieved through borrowing an idea from our previous sub's frame, modular panels. The top plane has five while the front and back legs each have one giving us a total of seven spots to mount components. With the 3D printed modular panels, all of our new parts like grabber and torpedos can easily be added and in multiple orientations and locations on the submarine creating more possibilities.

Another drawback was the inability of the frame to fit through standard doors which made transporting Marlin difficult and tedious. Marlin's new frame is thinner and has sufficient clearance to make moving the submarine a lot easier. The handles are also placed in a much more intuitive position.

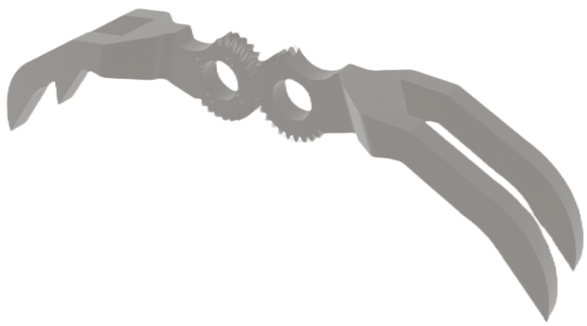

Fig. 1. New grabber with two fingers on each side.

B. Electrical Subsystem

1) ESC Backplane: Since the Blue Robotic T200s no longer had built in ESCs like the VideoRay M5, we had to buy separate ESCS. We eventually settled on the Flipsky FSESC 6.7 Pro given its extremely short response time, high power rating, and multiple communication capabilities. Instead of using PWM to communicate with the ESC, we decided to use CAN, as it is less prone to magnetic interference, and is also more efficient than UART due to its ability to be daisy-chained. Last year, we discovered that when a thruster broke down, there had to be much downtime isolating the issue and getting it up and running again. We knew that with separate ESCs, this problem would only double, so we designed a backplane focused on efficiency and modularity in order to organize our thruster control system.

The backplane acts as a power and signal distributor for each ESC between the batteries and our microcontroller. The backplane consists of two sets of motherboards and four daughterboards for each. Each daughterboard holds an ESC along with on-board connections to power and communication. Through special high power and signal Samtec connectors, we plug the daughterboards individually into the motherboards. In this way, none of the thrusters are permanently connected to ESCs. Should a thruster or ESC go down, the daisy-chain of CAN and power distribution across the boards allow all other systems to continue working. The plug-and-play fashion of these boards allow testing to be streamlined and any broken component to be easily isolated and debugged.

2) Sensor Stack: We have a separate MCU consisting of the STM Blue Pill, which manages all the sensors and calculations for the sensor stack. The sensor stack is meant to check the vitals of our submarine during tethered runs. With humidity, temperature, and leak sensors, the sensor stack is able to relay key information of our sub's state-of-being during runs. By connecting to the submarine's ID, we can monitor the information given to us by the sensor stack. Along with a battery voltage monitor, the Sensor Stack is filled with an array of sensors that help us further our submarine's capabilities.

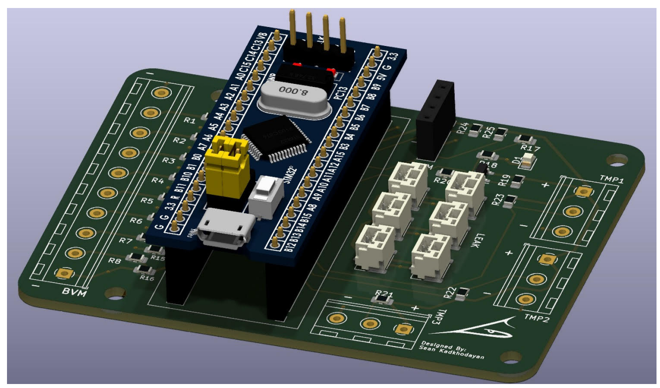

Fig. 2. Sensor stack built to monitor internal pressure, temperature, and voltage.

3) Microcontroller Hat: To create additional connections for our microcontrollers we designed a hat for our Nucleo-144, using KiCad PCB design software. The hat has multiple JST connections, for CAN (Controller Area Network), UART (Universal Asynchronous Receiver-Transmitter), PWM (Pulse Width Modulation), and SPI (Serial Peripheral Interface) communication protocols, as well as a connection for PNI's NaviGuider AHRS (Attitude and Heading Reference System). This hat sits directly on top of our Nucleo-144 and produces better connections since we can still use our JST connectors and are a better alternative to Dupont connectors which we discovered last year were prone to falling out during runs. The Hat connects to the ESC backplane system, Sensor

Stack, Nvidia Jetson, kill switch, as well as all our servos, pressure sensor, and DVL. Instead of a manual solder job, we were also able to incorporate a circuit for our kill switch to reliably relay the state of our thrusters power back to our MCU.

C. Software

1) SolvePnP Orient Calculation: One issue we faced with shooting torpedoes was that if we shot at a slanted angle relative to the torpedo hole, there would be a lower chance of success because of the smaller hole opening relative to us. To address this, we implemented a SolvePnP [1] function to find the relative yaw orientation of the torpedo board, so that we could face the board head-on. After thresholding and drawing contours around the torpedo board in the image, we extract the 2D pixel coordinates of the corners of the torpedo board. Our function takes in the 2D pixel coordinates of the object as well as the dimensions of the object in 3D space. We then calculate the focal length and we identify the center of our image. With this, we are able to set up our camera matrix. Inputting these elements into our SolvePnP function, we calculate the rotation and translation vectors. Using the Rodrigues function, we generate a rotational matrix from the rotation vector we obtained from the SolvePnP function. After applying some math, we are left with the yaw, pitch, and roll orientation of the torpedo board, allowing us to translate to a position normal to the board.

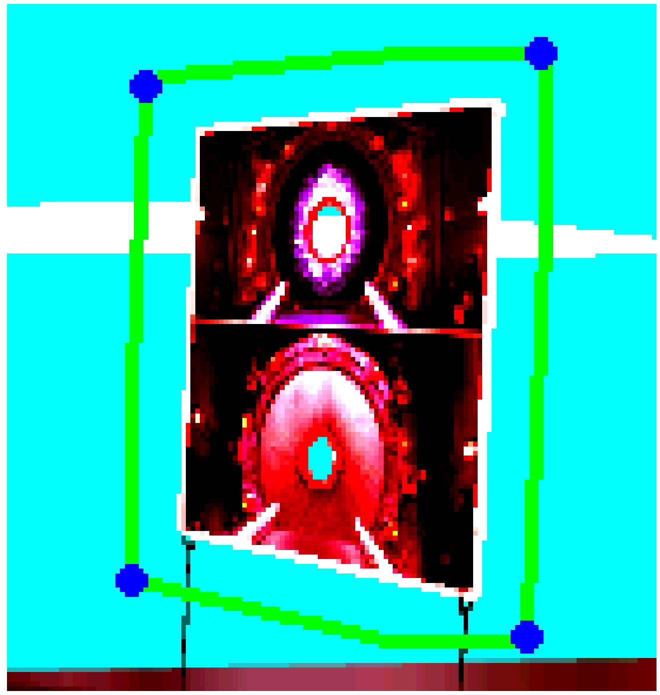

Fig. 3. Calculated Angle: 30 degrees. The sub would need to turn 30 degrees to the right to face the board head-on.

2) New Image Enhancement Filter: Last year, because we were in the UMD olympic swimming pool where the environment was extremely clear, our club was able to get away with not using an image enhancement filter for our down camera. However, this year, with being in the Transdec pool once again, we've looked into finding an efficient filter to separate objects in the image with the surrounding environment. From past competitions, we knew that our camera feeds were murky underwater, which made it extremely challenging for our vision models to pick up the path marker. This is because as light travels through water, high-frequency colors (like red, orange, and yellow) disappear. Furthermore, images taken underwater have an added blue-green cast and become distorted by backscatter. Our first approach was deciding to use our original filter from past years. We realized that, although this approach restored the image's original colors, the colors became a bit too vibrant and bright. As a result, we researched and looked into more efficient and mild methods of restoring the image object's colors, and came across the linear color transformation algorithm which we named mild_enhance. Firstly, our method takes the input image and splits it into three color channels: Red, Blue, and Green. After that, it calculates the minimum and maximum pixel values for each color and applies a linear transformation to each color channel individually to perform the enhancement. This transformation maps the original pixel values to the desired output range (0-255) through scaling and shifting them. Lastly, the transformed color channels are merged together once again to create a new enhanced image. The new image shows an improvement in visual quality with bringing out more colors and contrast, while also preserving the original image's general appearance.

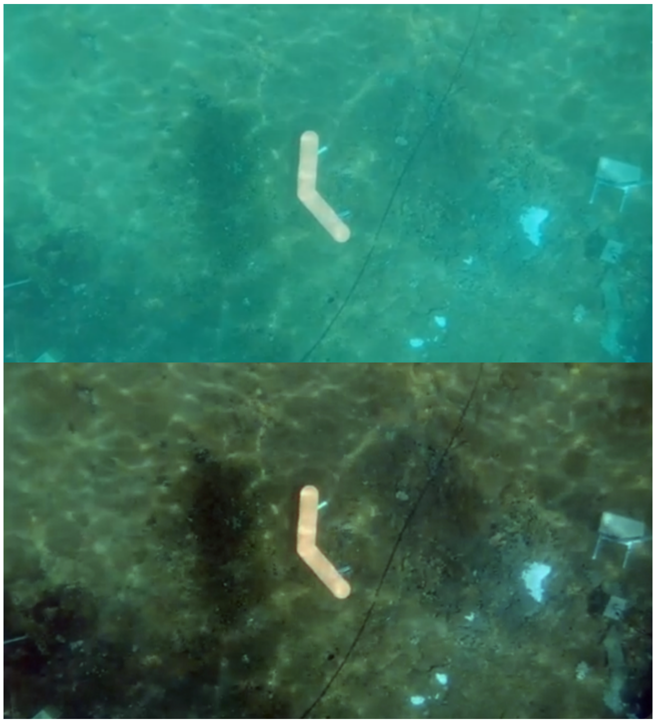

Fig. 4. The filter restores the object's original colors without overly distorting them.

3) Cascade PID: Because we are attempting tasks that require more positioning precision such as shooting torpedoes and grabbing objects, we sought to revamp our thruster control system to be more accurate. Drawing inspiration from other teams who utilize cascade PID controllers [2] to pristinely control their submarine, we implemented a cascade controller. Our high level position controller calculates the position error on six degrees of freedom, which feeds into a velocity controller. The velocity controller then calculates the error between our current velocity and our desired velocity, which becomes our force and torque setpoint to maintain a desired setpoint. By controlling velocity, this new controller allows for more precise movements. Although it adds another layer of complexity, our team deemed it worthwhile because of the massive upside of a controller with pinpoint precision.

III. Testing

A. Mechanical

For this year's goals, the mechanical team worked heavily on reducing, upgrading, and innovating on the current submarine. This year the team took on the task of developing more task based components and upgrading multiple components in the submarine. Our strategy for this year was to break the several tasks we had into smaller steps for us to complete. The first step taken included the designing and researching process, where we researched, brainstormed, and designed multiple solutions. Our second step was to build our prototypes, and our third step, which was often the most challenging, was to test these designs and make edits whenever necessary. The team knew it was crucial to have as much time and precision during the testing stage, to ensure, and test every possibility, for the most probability of success.

The mechanical team followed this procedure for developing a grabber, torpedo shooter, and ball dropper. After consulting with software's simulations that showed that the configuration with one finger on a side and two fingers on the other was unstable for certain grabbing orientations, we modified the grabber to have two fingers on each edge. We also added an extra inch of length, after discovering that our previous grabber's reach was too short to account for the height of the DHD dividers. For the torpedo shooter, we iterated over multiple torpedo printing settings, manually targeting infill levels that would bring the torpedo as close to neutrally buoyant as we could.

B. Electrical

This year, the electrical team's goals were to implement all of the component upgrades and create a modular and efficient systems for future generations to easily maintain and build upon. Electrical's roadmap was divided into four stages: plan, design, build, test. The first three stages were all preparation to build up for the testing stage where our work would be put to the test.

From day one, we were drafting our systems, using a lucid chart for power and signal distribution and mocking up the rack design with cardboard cutouts. Through the use of the open-source electronic design software KiCAD, the electrical team created a series of custom PCBs this year in order to streamline different levels of our power, sensor, and communication systems.

Through these PCBs, not only were we able to clean up connections and make the system easy to maintain, we were able to use them as great testing setups. Though we were unable to implement the ESC backplane into our sub this year, we continued to use them to test and program our ESCs, as they provided easy connection to power and communication. The MCU hat was also convenient to use for testing, as the breakoutboard provided not only the more robust JST connectors, but also pin headers that are much easier to debug and troubleshoot with. The sensor stack would aso prove instrumental during testing, being able to relay important information about our sub such as the temperature in different parts of the rack, the humidity of our MEB, potential leaks, and our battery voltage levels—all vital stats of our submarine during hours long testing runs.

C. Software

The software team relied heavily on simulation [3] to test most aspects of our stack, including our PID controllers, vision processing, and navigation code. Our strategy was to update our simulator's 3D models as soon as possible when learning of the new competition tasks, which allowed us to then train machine learning models and write computer vision algorithms for the simulated environment. Inside the simulator, we tested gate, buoy, bins, torpedoes, octagon grabbing tasks, and path markers, allowing us to fine-tune our navigation code and debug detection issues before a single pool test.

One major challenge was adding a grabber model to the sim, which involved countless hours of debugging to create a stable grabber working in the sim that would not explode. With the grabber in the sim, we then spent significant time re-running grabbing missions while tweaking parameters each time, searching for the optimal navigational approach to grab bin lids and chevrons. The addition of torpedoes to the sim also provided another challenge, as we had to ensure the motion of torpedoes in the sim would closely mimic the hydrodynamics and buoyancy of our actual torpedoes, and this required a great deal of debugging and testing to get our coefficients just right.

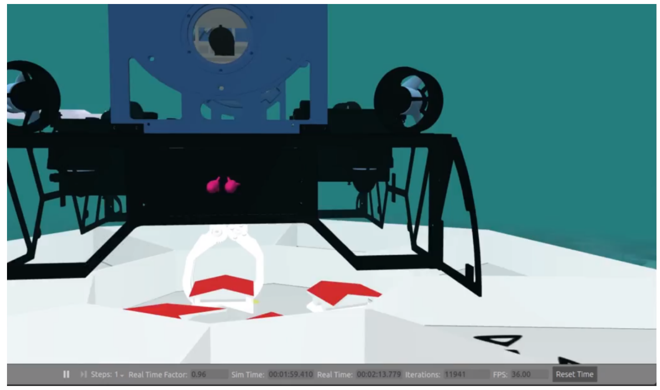

Fig. 5. Simulating grabbing chevrons in the sim.

D. Lessons Learned

The most important lesson we learned is that it is imperative to have clear communication between each subdivision of the club and between officers and members, to ensure that there is no confusion between timelines and goals. Furthermore, we learned that no matter how rigorously we prepare, we will inevitably face technical challenges. Then, the emphasis must be to stay calm even in the midst of a crisis, to think of all possible solutions to solve the problem. Usually, there is a solution if we look hard enough.

Acknowledgments

None of our work this year would be possible without a community of support. We would like to thank our advisor Mrs. Bree Barnett Dreyfuss for advising us on technical and organizational issues, Pleasanton Partnerships in Education Foundation and Parent Teacher Student Association for their financial support, NVIDIA for generously sponsoring a Jetson Orin computer, Waterlinked for graciously providing a compact and highly accurate DVL A50 model, PNI Sensor Corporation for kindly providing a user-friendly magnetically self-calibrating NaviGuider AHRS, True Value for providing sponsorships for screws and mechanical hardware, Amador Valley High School for providing us with pool facilities, and our parents for their endless patience and emotional support.

REFERENCES

[1] O. S. C. Vision, "Perspective-n-point (pnp) pose computation." [Online]. Available: https://docs.opencv.org/4.x/d5/d1f/calib3d_solvePnP.html

[2] Mathworks, "Designing cascade control system with pi controllers." [Online]. Available: https://www.mathworks.com/help/control/ug/designing-cascade-control-system-with-pi-controllers.html

[3] Liquid-ai, "Plankton: Open source simulator for marine robotics researchers." [Online]. Available: https://github.com/Liquid-ai/Plankton

APPENDIX A: COMPONENT SPECIFICATIONS

Component	Vendor	Model/Type	Specs	Custom / Purchased	Cost	Year of Purchase
Frame	Custom	Aluminum 6061 - T6	90.50cm x 63.50cm x 33.34cm	Custom	Sponsored	2023
Main Waterproof Enclosure	In-House	Acrylic Hull Sealed with Two Rubber O-Rings	Diameter: 24 cm	Custom	$200	2016
Waterproof Connectors	SubConn	Circular Series SubConns	(Varies Based on Series) Micro-Circular Series, Power Series	Purchased	$1500	2015
Thrusters	Blue Robotics	T200 Thrusters	113 mm Length	Purchased	$200 ea.	2022
Electronic Speed Controllers	Flipsky	FSESC 6.7 Pro	14-60 V, CAN Communication, 67 mm x 39 mm x 18.7 mm	Purchased	$100 ea.	2022
Microcontroller: Motor Control	DigiKey	Nucleo F767ZI	2 MB Flash Memory, 512 KB SRAM	Purchased	$30	2023
Batteries	ZEEE Power	4S	9000mAh, 14.8V	Purchased	$195	2022
DC to DC Converter	Cincon	CHB200W12-72S12	200W, 16V to 12V	Purchased	$185	2023
Computer	Nvidia	Jetson AGX Orin Developer Kit	414mm x 311mm x 182mm, 275 TOPS, 2048-core GPU, 12-core CPU	Purchased	Sponsored	2022
Internal Comm Network	ROS	ROS2 Foxy	Ubuntu 20.04	Custom	Free	2022
External Comm Interface	-	Ethernet	1 GB/s	Purchased	Included with SubConn	2015
Doppler Velocity Log (DVL)	Waterlinked	A50	5cm–50m altitude range, 600m depth rated, Ethernet and Serial communication, 1 MHz frequency	Purchased	Free	2022
Altitude Heading and Reference System (AHRS)	PNI Sensor	NaviGuider	Heading Accuracy: 2° rms, UART Communication	Purchased	Sponsored	2023

Pressure Sensor	Blue Robotics	Bar-30	Accuracy: ±2.9psi, I2C Communication, Supply Voltage: 2.5-5.5V	Purchased	$85	2023
Front Camera	FLIR	BFS-U3-200S6	Frame Rate: 30 fps, Resolution: 5472x3645, Megapixel: 20MP, Sensor Type: CMOS	Purchased	$750	2015
Front Camera Lens	Computar	VO828-MPY	8mm fixed lens, Resolution: 12MP, Horizontal Angle: 77.3°, Vertical Angle: 61.7°	Purchased	Sponsored	2015
Down Camera	FLIR	BFS-U3-13Y3C-C	Resolution: 1280x1024, Megapixel: 1.3MP, Frame Rate: 170FPS, Sensor Type: CMOS	Purchased	$540	2015
Down Camera Lens	Theia	SY125M	Focal Length: 1.3mm, Resolution: 5MP, Horizontal Angle: 125°, Vertical Angle: 119°	Purchased	Sponsored	2015
Signal Processing	Diligent	Nexys 4 DDR Artix-7	Block RAM: 4,860 Kbits	Purchased	$250	2019
Algorithms: Vision	Ultralytics	YOLOv8s, RGB equalizing filter	5 FPS	Open Source	Free	2023
Algorithms: Acoustics	In-House	MUSIC	Hydrophones	Custom	Free	2018
Algorithms: localization, mapping	In-House	DVL data, image calculations	DVL, IMU, CV	Custom	Free	2017
Algorithms: Autonomy	In-House	Linear instructions	ROS2 nodes	Custom	Free	2022
Open source software	Open source	ROS2, YOLOv8s, OpenCV	Node management, computer vision	Custom	Free	2023
Team Size (number of people)	44					
Expertise ratio (HW vs. SW)	23:13 + 8 Business					
Testing time: simulation	125 hours					
Testing time: in-water	90 hours					

Programming Languages	C, C++, Python 3