# Robosub 2026 Technical Design Report

*Bruin Underwater Robotics (University of California, Los Angeles)*

Lucas Choe[1], Stanley Wei[1], Mollie Naylor[2], Annabelle Stiger[2], Peter Boyd[2], Luis Prieto-Moreno Rubio[2], Julian Kim[2], Aaron Fan[2], Connor Tung, Asim Shariff, Tanisha Pidshetti, Joseph Claypool, Howard Ding, Henry Liu, Alex Lie, Matthew Duran Abeja, Dyllan Goldstein, Steven Hsu, Jackie Lei, Jared Ashman, Vidhu Venugopal, Aansh Singh, Nessa Lee, Daniel Koh, Kaden Do, Edward Sun

***Abstract*—Robosub 2026 marks Bruin Underwater Robotics' third year competing in the Robosub competition, as well as our third year designing autonomous robots as a whole. This year, we chose to reuse our 2025 AUV, Endurance. Although much of the overall design remains the same as last year, we have made several improvements to its end effectors, electronic hardware, and software systems. Additionally, we have conducted further testing of the AUV's various systems, which has helped make our AUV more robust than ever.**

## I. Competition Strategy

This year, our team had 2 primary goals: (i) improve our existing robot, and (ii) pursue new design challenges as a club. For the first objective, we chose to reuse our existing robot and software systems from the Robosub 2025. For the second objective, we opted to design & build a second robot this year, with the goal of entering it into the competition. This robot was intended to be smaller than our first AUV, with some key design differences to set it apart from our previous AUV. However, we were unable to finish it in time for competition; as such, we will only be competing with our previous robot.

[1]Project Lead
[2]Subteam Lead

### A. Course Strategy

Our primary goal for competition this year is to successfully complete the navigation objectives – crossing the gate with style, completing a partial slalom, and surfacing in the octagon.

Beyond these initial goals, we have a number of stretch objectives that we have developed some capability for, but which have not had said capabilities verified in practice. These objectives are the coin flip, complete slalom, and the torpedoes.

## II. Design Strategy

### A. Chassis

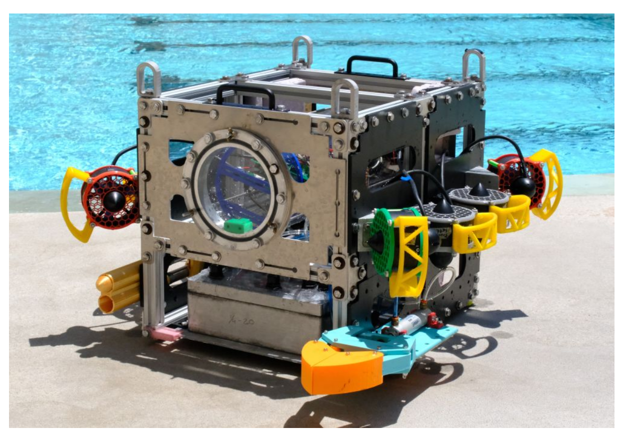

Fig. 1. Our AUV, Endurance

Last year, the chassis was designed with reliability and optimization as the top priorities, and as such it remained operable after the 2025 Robosub competition. This year, the primary change we made was renewing 3D printed parts in order to ensure reliability going forward.

*1) Frame:* Our AUV is made of an aluminum T-bar frame with high-density polyethylene panels (HDPE) and aluminum brackets, making the main shell of our AUV extremely rigid. Additionally, the frame has lots of internal space and external attachment points for different potential additions.

*2) Slide-Out Panels:* Our AUV utilizes battery and pneumatic boxes housed within the frame. These battery and pneumatic boxes are mounted on aluminum panels with 3D-printed rails that allow them to slide within the aluminum T-bar frame, giving us easy access to the boxes' contents with minimal disassembly.

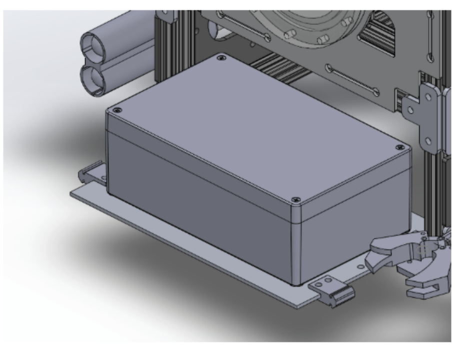

Fig. 2. An image of our slide-out panels and pneumatics box CAD

*3) Electronics Bay:* Our electronics bay utilizes an acrylic tube sealed with aluminum flanges fitted with O-rings. The electronics bay is mounted by its flange directly onto the frame using aluminum plates, resulting in a rigid and secure structure. All of our cables route outside the electronics bay through acrylic end caps and are sealed using cordgrips. This year, we replaced our 3D printed endcaps with new endcaps, printed with improved strength settings to limit future fatigue.

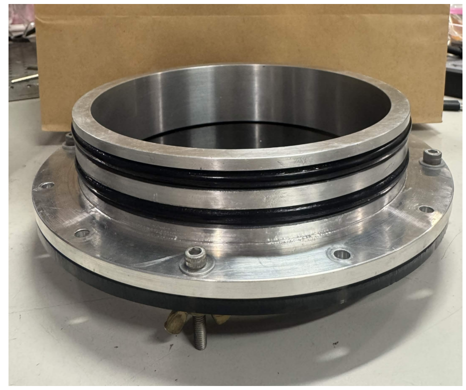

Fig. 3. Our electronics bay endcap

### B. Dive Operations

*1) Pneumatics:* Similar to last year, this year we decided to use pneumatics to operate the torpedoes, arm, and dropper. Our system consists of a paintball air tank connected to four solenoid valves that control the flow of air to the 4 actuators. To ensure that the system was safe, we incorporated a vented ball valve, 2 relief valves, and a pressure regulator to keep our operating pressure below the maximum pressure rating of the individual components. Our solenoids were housed in an adjacent aluminum box.

*2) Torpedo:* We used a double-barrel design for our torpedoes, utilizing two linear pneumatic actuators. Our 3D printed torpedoes were printed with 80% infill to match water density for neutral buoyancy. Additionally, the torpedoes feature knobs to slide along guiding rails inside the barrels, ensuring the the torpedo launches straight.

*3) Arm:* Our arm design features a completely redesigned four bar linkage system

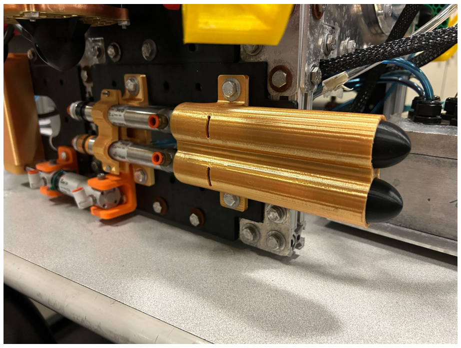

Fig. 4. A closeup of our pneumatic torpedoes

driven by a linear pneumatic actuator, designed to produce a much smoother and more reliable actuation compared to previous years' designs. We scaled our linkages to balance and offer greater mechanical advantage, a wider jaw opening, and more consistent torque at all positions. In order to achieve strength in the system, the linkages are mounted through the claws to better protect and house them.

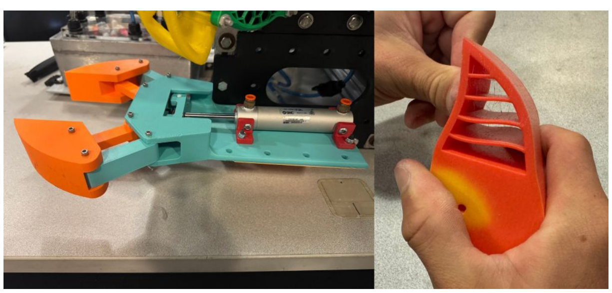

Fig. 5. Arm Assembly and Adaptable Claw

We redesigned our claw profiles to match the new mechanism, iterating through a number of shapes and increasing the surface area significantly. We printed our grippers using TPU so they can to deform around objects, thereby grasping them more reliably and tightly. As a result, our claw profiles are now much more adept for grabbing and holding a diverse variety of components, including irregular shapes and sizes.

Lastly, this year we introduced a new mounting solution, pivoting the claw 90 degrees and allowing for easier access to the target objects. We emphasized a simple and secure design that was mounted closely onto the chassis, thereby maximizing stability and limiting the moments on the bot.

*4) Dropper:* This sub assembly consists of a release mechanism by a pneumatic actuator, two dropper items shaped like torpedoes, and a 3D printed housing. These dropper items are made of aluminum to ensure that they sink, and are designed such that they can be manufactured from a rectangular piece of stock using a CNC milling machine and a lathe.

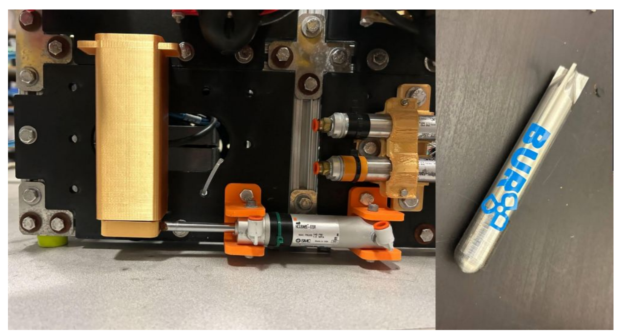

Fig. 6. Our dropper mechanism & item

A large portion of our work in this design cycle was to improve the dropper mounting mechanism and upgrade our torpedo geometry for increased accuracy. Our mounting was certified using ANSYS Mechanical and resisted double the stress factors as previous years. It included increased surface area, larger fillets, and direct chassis attachment.

### *C. Hardware Systems*

*1) Battery Management System:* In redesigning our onboard power management system this year, our goal was to address issues with our previous power supply system while maintaining existing modes of functionality and failure conditions. The key issues we attempted to address this year were a lack of emergency access and voltage monitoring.

To monitor battery voltage, each cell of our 4S batteries is connected using the existing voltage monitoring pins for balance charging to the ESP32. The system also has a voltage divider and low pass filter to modulate and smooth out the incoming battery voltage reading, as well as a zener diode to clamp the incoming signal. When a cell reading drops below the nominal 3.7V threshold, the system emergency shutoff is triggered.

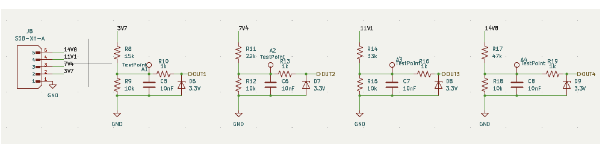

Fig. 7. A diagram of our battery monitoring system

*2) Emergency Shutoff:* Our other major modification this year was making our emergency shutoff much easier to access, as our previous switch was difficult to handle dexterously underwater in case of an emergency. Our solution to this issue was a magnetic contact sensor, backed by a panel of magnetic material, soldered in line with our existing switch structure to control our primary relay. This allowed us to attach a strong magnet to the exterior of our battery enclosure, actuating the switch across the barrier and attracting the magnetic panel material. With a 3D-printed handle attached to the magnet, we can now easily grasp and remove the magnet to deactivate the robot, increasing safety and efficiency in shutdown and startup.

## D. Software Systems

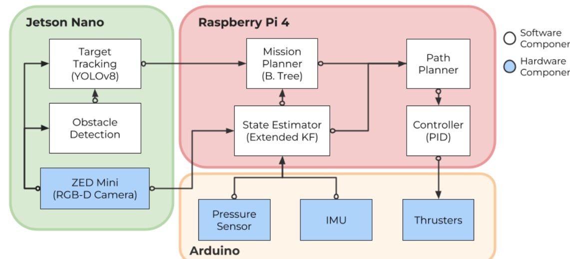


Fig. 8. A diagram of our software systems

For our onboard software, our focus this year was primarily on improving aspects of our existing subsystems, particularly our object detection models.

For our robot controller, we had previously been relying on velocity control, which required additional complexity in other subsystems to achieve proper positioning. This year, we moved to using cascaded position-velocity PID controllers [1], keeping in place our existing design of assigning an independent controller to each degree of freedom of our robot.

For our vision software, building on last year's simulation pipeline, this year's efforts focused on aligning our synthetic training data with real-life conditions to better prepare the model for real-life competition conditions. On the rendering side, the Unity environment was enhanced with reverse projection caustics to produce the flickering light patterns characteristic of real underwater environments [3]. Water clarity was also increased to better reflect competition water conditions.

To further close the gap between synthetic and real data, we experimented with

Fast Fourier Transform (FFT)-based domain adaptation. By replacing the lower-order frequencies in our synthetic images with the lower-order frequencies of real competition images, we can better align the underlying color distribution of our synthetic images with the actual underwater footage [2]. By aligning our synthetic data with real data, we thereby reduce the visual shift the model would otherwise experience at competition.

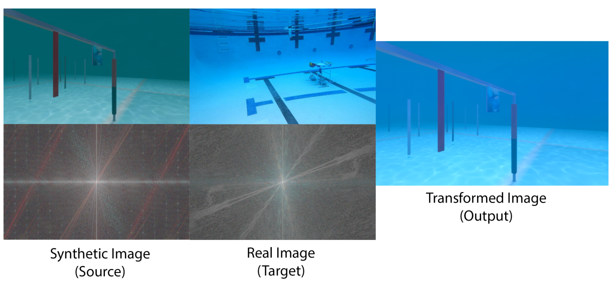


Fig. 9. Fourier domain adaptation (incl. amplitude spectrums)

This year, we trained our main vision model to identify the gate, bin, slalom, and tagging elements. In addition, we are developing a dedicated secondary model specifically for the torpedo task. This model operates exclusively within the torpedo target's bounding box as identified by the primary tagging detector, treating the cropped region as its input. This way, the secondary model can focus on the fine-grained spatial detail needed to reliably localize individual holes.

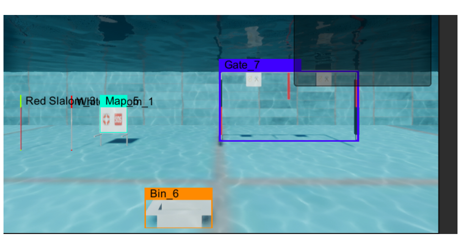


Fig. 10. A sample from our synthetic training dataset (with labels)

## III. Testing Strategy

### A. Dive Operations

*1) Torpedo:* The torpedo designs were optimized using our torpedo test rig, which mounts an exact replica of our launcher onto a variable platform. A series of underwater test fires allowed us to optimize the proper launch angle and positioning of our torpedo launcher on our AUV.

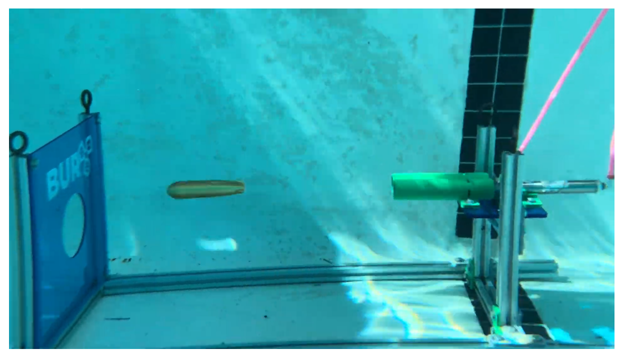

Fig. 11. Our torpedo test rig, underwater & mid-launch

In addition, we simulated our torpedo designs using ANSYS to see if our in person findings agreed with the stresses and velocities shown using simulation. Several different nose and fin shapes were proposed. For each design, CFD analysis was performed to analyze the velocity profile and drag coefficient, then a simulation was run to represent a torpedo being shot into still water. One torpedo design appeared to have the best results and was selected as a final design.

*2) Dropper:* Our dropper test was simple, consisting primarily of actuating the pneumatics and making sure tolerances worked as planned. Torpedo shapes were tested using the ANSYS and test rig duo, as mentioned above.

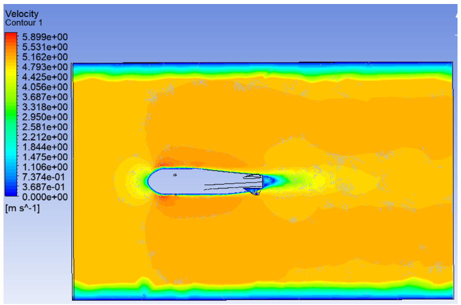


Fig. 12. Velocity profile of fluid around the torpedo

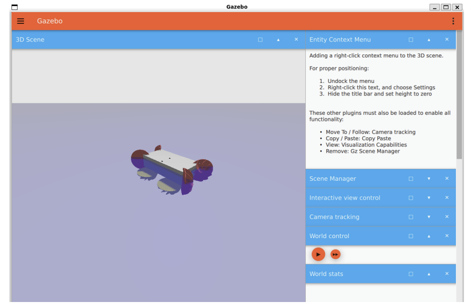


Fig. 13. An image of our simulated robot in Gazebo

### B. *Software*

To test the real-life performance of our controller, we did pool testing in order to gather performance data and tune our controller parameters accordingly. To test, we would disturb our robot (either via a physical push, or by attaching additional buoyant material), then record our robot's position and orientation error as it attempted to return to steady state.

To test our behavior tree-based mission manager, we used a series of prespecified inputs to ensure that the output behaviors matched our expectations. We tested several tasks/missions – for each mission, a prewritten set of inputs were provided, and the resulting commands from the mission manager were compared to our expectation. We also did some exploratory work integrating the Gazebo simulation suite into our testing process for proper physical simulation, though we have not yet incorporated simulated game objects.

For vision testing, our testing set was kept entirely separate from both training and validation data, drawing primarily from video footage and images captured at last year's competition. Given that the core game elements have remained largely consistent between years (with exceptions), this footage provided us a strong real-world benchmark for evaluating model performance.

Finally, another new addition this year was the integration of Docker, a software virtualization platform, into our testing framework. This has helped us greatly by allowing us to conduct any purely-software tests within a highly reproducible closed development environment.

## IV. Acknowledgments

The work achieved by Bruin Underwater Robotics was only possible due the support of a number of individuals and organizations. We would like to first give special thanks to Professor Veronica Santos for her unwavering support and drive for our team to succeed, as well as to ASME at UCLA for their allocation of resources and space for our project. Furthermore, we would like to acknowledge UCLA Recreation for providing a location for water tests.

Bruin Underwater Robotics would also give thanks to our generous sponsors, whose support provided the opportunity for our team to work on this incredible project. We would like to provide special thanks to SMC Pneumatics, Industrial Plastics Supply, and Protolabs for their generous donations to our team.

## References

[1] A. M. El-Dalatony, T. Attia, H. Ragheb, and A. M. Sharaf, "Cascaded pid trajectory tracking control for quadruped robotic leg," International Journal of Mechanical Engineering and Robotics Research, vol. 12, no. 1, pp. 40–47, Jan. 2023.

[2] Y. Yang and S. Soatto, "FDA: Fourier Domain Adaptation for Semantic Segmentation," presented at the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 4085–4095. Accessed: May 26, 2026. [Online].

[3] "Chapter 2. Rendering Water Caustics," NVIDIA Developer. Accessed: May 27, 2026. [Online]. Available: https://developer.nvidia.com/gpugems/gpugems/part-i-natural-effects/chapter-2-rendering-water-caustics