RoboSub 2025 Technical Design Report

University of Alberta – Autonomous Robotic Vehicle Project

Daniel Faltaous (Project Co-Lead), Cole Dewis (Software Co-Lead),
Kendra Smart (Mechanical Co-Lead), Yousef Moussa (Electrical Team)

Abstract – The Autonomous Robotic Vehicle Project (ARVP) at the University of Alberta developed Kenai, a new Autonomous Underwater Vehicle (AUV) for RoboSub 2025 focused on reliability and reducing prior AUV failure points. Key updates include a simplified cylindrical hull, standardized electrical interfaces, subsystems, and improved control architecture. With Koda, a vision-based support robot, and a custom acoustic modem, the team has also expanded task coverage. A systems engineering approach defined clear system requirements and guided verification and integration, supported by bench tests, simulations, and over 30 hours of pool testing. Kenai achieved 100% success on dropper and torpedo tasks and halved maintenance downtime.

I. Introduction

The Autonomous Robotic Vehicle Project (ARVP) at the University of Alberta supports students in developing robotics skills through real-world projects across four sub-teams: Business, Electrical, Mechanical, and Software. ARVP also provides career development and a strong social environment for members.

The team's primary technical milestone each year is the RoboSub competition. In 2024, ARVP competed with Arctos, an AUV designed in 2019 and built in 2022, which completed the gate, buoy, dropper, and torpedo tasks but was inconsistent on the claw task. Despite these results, Arctos suffered from frequent electronics failures and hull sealing issues due to manufacturing defects.

After RoboSub 2024, only 20% of the team returned, leading to a loss of expertise and a large influx of new recruits. This transition challenged knowledge continuity but also presented an opportunity to re-evaluate ARVP's technical direction. After reviewing past performance and available resources, the team determined that continuing with Arctos would jeopardize future success. Instead, ARVP committed to developing Kenai, a new AUV designed for high reliability and maintainability, within an accelerated eight-month cycle as a minimum viable testing platform.

This paper outlines ARVP's strategic approach for RoboSub 2025, the design of Kenai to support that strategy, and the verification and validation conducted to ensure mission readiness.

II. Competition Strategy

Strategic Vision: A Reliable Testing Platform

The team's vision for RoboSub 2025 was to build a reliable AUV platform to improve task success rates and reduce pool downtime. Lessons from prior years showed a new vehicle was needed to meet these goals.

Kenai was developed with three main objectives:

- Simplify hull design and mechanical subsystems to improve reliability
- Eliminate hardware failure points through standardized electrical interfaces and cleaner power paths
- Improve control stability and object alignment for precise task execution

System requirements were derived from planned tasks, competition rules, and reliability targets. This ensured clear requirement ownership, traceability, and a defined integration and testing timeline.

A core principle was to limit complexity unless it significantly improved task performance. Systems or features without a clear performance benefit were simplified or removed to support overall reliability and ease of integration.

After a major recruitment push in September, team size grew to 60, exceeding initial scope. To manage this while maintaining reliability, ARVP formed a core team for critical systems and parallel tracks for projects like Koda and an acoustic modem. These expanded capabilities while managing risk to Kenai's timeline.

Course Approach

Task prioritization for RoboSub 2025 was based on the Historical success rates, Required time for implementation and testing, Robustness to sensor drift, and Point values per the RoboSub 2025 Handbook.

Appendix A summarizes the selected target task points. The gate, torpedo, and dropper tasks have high historical success rates and are retained with minor improvements to vision and mechanical subsystems. The slalom task occurs early in the course, making it a low-risk addition. The claw task, while not previously completed in competition, showed promise during testing; improved claw design and vision guidance are expected to increase its success rate. Basic inter-sub communication was developed via a custom acoustic modem.

Mechanical tasks were assigned to Kenai to leverage its robust subsystems, while the purely vision-based slalom task was assigned to Koda to reduce Kenai's workload and limit drift accumulation. Both robots use vision for navigation between tasks, with recovery behaviors integrated into motion planning to mitigate missed detections.

ARVP chose to forgo the go-home task for Kenai due to drift accumulation and time constraints. Inter-sub communication complexity was intentionally limited to ensure reliable execution, reflecting a balance between new functionality and overall system reliability.

Task Execution

1. **Gate**: Kenai starts via coin flip and passes through the shark side without style points, then signals Koda to begin. Koda also enters through the shark side and performs two barrel rolls for style points.
2. **Slalom**: Koda proceeds to the slalom, aligning using new visual servoing logic to navigate each set of three pipes.
3. **Bins**: Kenai goes directly to the bins, dropping both markers into the shark side. The dropper remains unchanged from last year due to its stable performance.
4. **Pinger:** Kenai uses the pinger to determine whether to attempt the table or torpedo first, backtracking as needed.
5. **Torpedo:** From 0.3 m, Kenai targets the shark and sawfish holes with an improved launcher designed for higher reliability and accuracy.
6. **Table:** Kenai surfaces in the octagon, detects the correct orientation, then identifies, grabs, and drops bottles into bins. A current-sensing claw confirms successful pickups, repeating as needed.
7. **Return Home:** After slalom, Koda returns using gate alignment and the same visual servoing approach.

III. Design Strategy

Overview

Kenai, shown in Figure 1, was designed to withstand the demanding testing and operational conditions at RoboSub. All previously unreliable mechanical systems were revisited to eliminate mid-pool maintenance. The torpedo and claw subsystems were redesigned to address periodic jamming at RoboSub 2024, with new grippers providing a larger pickup tolerance. The external battery pods were simplified with a reduced part count, cutting assembly time by half.

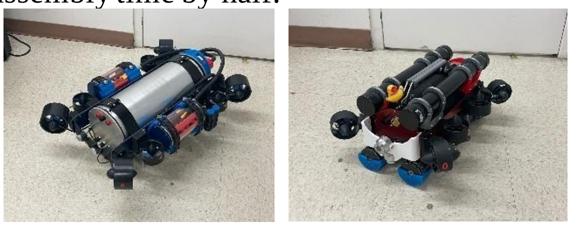

Figure 1. ARVP's AUVs Kenai (Left) and Koda (Right)

Kenai's electrical system was redesigned for reliability and easier troubleshooting. Legacy boards were consolidated, and interfaces standardized, using XT series connectors and consistent wire gauges throughout. A new battery monitoring system with an integrated kill switch and centralized power distribution board simplified wiring and improved system safety. CAN bus remains the communication backbone and is now supported by ESD protection. These upgrades minimize avoidable downtime and enable dependable performance, aligning with the team's reliability-focused strategy. Electrical Architecture diagrams can be found in Appendix B.

Most of the software stack transferred directly from Arctos to Kenai, allowing focus on node improvements. Updates to vision, motion planning stability, mission plans, and new visual servoing logic enhance task precision and execution reliability. Software architecture diagrams can be found in Appendix C.

Our second robot, Koda (Figure 1), was designed to reduce Kenai's operational load during competition. It uses a simplified electrical system with a Pixhawk and ArduSub, a lower-power Jetson Orin NX [1], [2, and no doppler velocity log or mechanical subsystems. It shares the same software stack as Kenai, except for a simplified control and motion planning node relying less on positional estimates. This design prioritizes simplicity to reduce potential failure points and streamline operation.

The following design highlights focus on the most critical new features developed this year, chosen to demonstrate creative design, rigorous integration, and systems thinking.

Mechanical Highlight: Hull & Frame

Following the RoboSub 2024 hull leak, integrity and reliability became top priorities to ensure a dependable test platform. The previous cubic hull maximized internal and external space efficiency but suffered from high stress concentrations at corners, contributing to failures.

A cylindrical hull machined from 6061-T6 aluminum pipe with Blue Robotics 8" O-ring flanges was selected to distribute stress more evenly and reduce failure points [3]. Sealing surfaces were verified to meet o-ring stretch (0–5%) and fill (75–85%) metrics. Inspired by Blue Robotics locking enclosures, thinner walls were used to simplify and reduce machining costs. Finite element analysis (FEA) validity was confirmed by first reproducing the 1000 m depth rating [3] of the Blue Robotics hull before evaluating Kenai's final design, which yielded a depth rating of 724 m with a safety factor of 2.

This design prioritizes reliability over space efficiency, supporting consistent testing and shifting pool time from mechanical repairs to software development. The cylindrical shape required creative internal and external space planning.

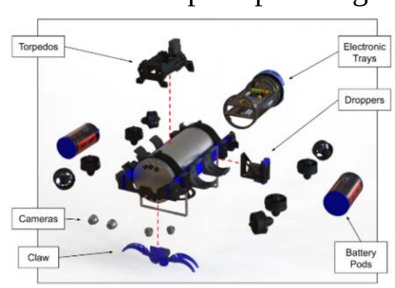

Figure 2. Kenai Labelled Exploded View

The frame was designed to improve subsystem accessibility, facilitate maintenance, and optimize placement relative to cameras and obstacles. For example, torpedoes are mounted on top and pushed forward to simplify removal and minimize shooting distance for bonus points. Internally, the electronics layout was co-developed with the electrical team to maximize board space and simplify wiring.

Electrical Highlight: Control and Monitoring

A central element of Kenai's electrical redesign was the new control board (Rev B), developed to improve reliability, simplify integration, and support critical actuation tasks. It consolidates actuator control, actuator monitoring, and internal environment sensing onto one PCB, reducing wiring complexity and failure points.

Problems in Rev A, including unregulated power rails, noise-prone PWM expansion, and fragile connectors, drove design changes. Rev B uses a single 12V input that is filtered and stepped down to stable 5V, 3.3V, and a selectable servo voltage (5V or 7.4V). Dedicated buck converters and onboard regulation reduce external conversion needs and improve noise immunity.

All servo PWM outputs are digitally buffered to improve signal integrity and each is paired with a dedicated current sensing circuit using INA219 modules and low-value shunt resistors [4]. This enables real-time stall detection, which is essential for tasks such as claw grasping and torpedo firing sequences.

Connectors were upgraded to keyed, locking Molex interfaces, replacing fragile Dupont and JST-GH types. This increases mechanical robustness, prevents miswiring, and streamlines maintenance. CAN bus remains the backbone of subsystem communication and now features onboard termination and ESD protection to improve reliability.

Environmental monitoring, including temperature, pressure, and humidity sensing, was also integrated to support system diagnostics and help prevent potential failures.

Overall, the control board Rev B reflects a systems engineering approach focused on reducing failure points, simplifying serviceability, and supporting precise, responsive control.

Software Highlight: Vision and Mapping System

The vision system is critical for detecting and localizing objects, enabling Kenai to navigate and execute tasks. At RoboSub 2024, ARVP relied on YOLOv8 bounding box models, which performed adequately for general tasks but lacked precise object orientation information [5]. This limitation significantly reduced success on tasks such as the claw pickup, where accurate alignment is essential for reliable grasping.

To address this, ARVP developed a new vision node capable of running YOLOv8 or YOLOv11 models in bounding box, oriented bounding box (OBB), or segmentation mask configurations. This flexibility allows advanced models to provide object orientation data, which is parsed by an upgraded mapping node to calculate more precise object poses. The enhanced mapping enables Kenai to align the claw more accurately, improving grasp success rates.

A key design constraint was maintaining compatibility with legacy tasks. The updated vision node supports task-specific model selection, allowing traditional bounding boxes to continue for stable tasks like gate navigation or bin dropping, while OBB and segmentation can be selectively used where higher precision is required. Figure 3 illustrates example detections from each model type. YOLOv11 integration ensures future adaptability to new model architectures.

Figure 3. Sample output from bounding box, oriented bounding box, and segmentation model.

This modular and extensible design supports higher task success while minimizing disruption to proven mission logic, aligning directly with ARVP's strategy of incremental capability improvements balanced against overall system reliability.

IV. Testing Strategy

Testing Philosophy

This year, with clear and detailed system requirements, ARVP verified subsystems directly against pass/fail criteria. Subsystem tests served as development checkpoints to confirm component functionality and prevent downstream rework. Once verified, subsystems were integrated into the full system and then validated at the vehicle level in operational environments to ensure the design met task objectives. Testing focused on cases without third-party inspection and where analysis alone was insufficient.

Electrical testing included board-level, load, integration, and end-to-end tests in the laboratory. Mechanical tests covered component, subsystem, accelerated life cycle, integration, and end-to-end evaluations, often using the University of Alberta Water Resources Lab for underwater and pressure scenarios. Software modules were initially tested in the custom Gazebo-based simulator, followed by lab-based sensor checks. Full system validation was conducted during three-hour bi-weekly pool tests in the City of Edmonton dive tank with mock obstacles, providing the most critical evaluation environment. Pool testing procedures and time allocations can be found in Appendices D and E.

Mechanical Highlight: Pressure Test Procedure & Hull Validation

The hull leak at RoboSub 2024 revealed critical gaps in preoperational seal verification, prompting a complete redesign of the pressure testing procedure.

Temperature fluctuations from electronics heating and sudden immersion in cold pool water affect internal pressure readings. To address this, the updated procedure uses the ideal gas law to calculate expected internal pressure from temperature measurements. This value is compared to actual measurements, with the difference monitored for stability to confirm a reliable seal. A detailed description and calculations are included in Appendix F.

This pressure monitoring approach is also applied during pool operation, with continuous logging of expected versus measured pressures. Additional safeguards include leak tape at seals and an internal camera for visual confirmation.

After machining, Kenai's sealing surfaces were measured to ensure tolerance compliance. The hull was then submerged in the University of Alberta Water Resources Lab tank under simulated competition conditions. Successful results validated the hull's integrity, allowing progression to regular AUV operations. Full test procedures are documented in Appendix G.

Electrical Highlight: Control and Monitoring Test Procedure

Control Board Rev B was rigorously verified to meet its design goals for reliability and subsystem integration. Initial benchtop tests used a custom C++ firmware suite on the Teensy microcontroller to exercise each subsystem: CAN bus communication was verified at 1 Mbps [6]; PWM outputs passed frequency sweeps for signal integrity; and all six INA219 current sensors were validated using adjustable loads from 200 mA to 2.7 A to simulate stall conditions. Environmental sensors (pressure, temperature, humidity) were checked against calibrated references.

These tests confirmed improvements over Rev A, including regulated voltage rails, buffered servo outputs, and robust connectors. All modules met specifications, enabling full system integration.

In system-level validation on Kenai, actuators and sensors performed reliably under mission conditions. The claw successfully used live current feedback to confirm object grasps and trigger retries when needed. Dropper and torpedo subsystems executed with correct timing, while intersub communication via RGB LEDs was verified through vision logs. CAN bus exchanges between Jetson and Teensy showed no dropped frames or signal issues.

Firmware verification took one week, and system-level validation was integrated into pool tests. Additional details appear in Appendix H.

Software Highlight: Vision and Mapping Test Procedure

Verification of the vision and mapping system began with bench-level tests to confirm each model type (bounding box, oriented bounding box, and segmentation) produced correct ROS topic outputs at acceptable frame rates.

Simulation tests followed to verify integration with the mapping node. Simulated oriented bounding boxes were passed to the mapper, and output orientations were compared against ground-truth values. Outputs within an acceptable threshold were considered successful.

System-level validation was conducted using a simplified claw task scenario. The robot detected and aligned above the ladle obstacle at the correct orientation, including cases with partial occlusion from other objects. Successful alignment and readiness to grasp were considered a pass.

Finally, full validation occurred during complete RoboSub 2025 course runs, confirming function in mission conditions.

Test results showed consistent frame rates across all models, accurate object orientation estimation in simulation, and a 65% success rate on 30 claw task trials. Further technical details and complete test logic are included in Appendix I.

V. Conclusion

While this year presented significant challenges, ARVP has emerged with Kenai: a robust, reliable testing platform designed to support full-course RoboSub 2025 task execution. Key improvements to the hull, control board, and vision system have enabled a new level of reliability and capability. Looking ahead, ARVP will continue to refine Kenai as a primary platform while expanding Koda's functionality as a complementary system.

VI. Acknowledgements

We thank our advisors, alumni, industry supporters, and sponsors; see Appendix J for detailed acknowledgements. ARVP's success is not possible without you.

VII. References

[1] Pixhawk Project, "Pixhawk Autopilot Hardware and Open Standards," Accessed: June 2025. [Online]. Available: https://pixhawk.org/

[2] NVIDIA, "Jetson Orin NX Module," NVIDIA Developer, Accessed: June 2025. [Online]. Available: https://developer.nvidia.com/embedded/jetson-orin

[3] Blue Robotics, "8″ Series Enclosures and Flanges Technical Data," Blue Robotics, Accessed: June 2025. [Online]. Available: https://bluerobotics.com/store/watertight-enclosures/series-8

[4] Texas Instruments, "INA219 Zero-Drift, Bi-Directional Current/Power Monitor Datasheet," Rev. SLYS154, Mar. 2021. [Online]. Available: https://www.ti.com/lit/ds/symlink/ina219.pdf?ts=1750624737016&ref_url=https%253A%252F%252Fwww.google.com%252F

[5] J. Redmon et al., "YOLOv8: Ultralytics Real-Time Object Detection," Ultralytics, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[6] PJRC, "Teensy 4.1 Technical Specifications," PJRC, Accessed: June 2025. [Online]. Available: https://www.pjrc.com/store/teensy41.html

VIII. Appendices

Appendix A: Percentage of Competition Points for Various Robosub Tasks Between 2024 to 2025

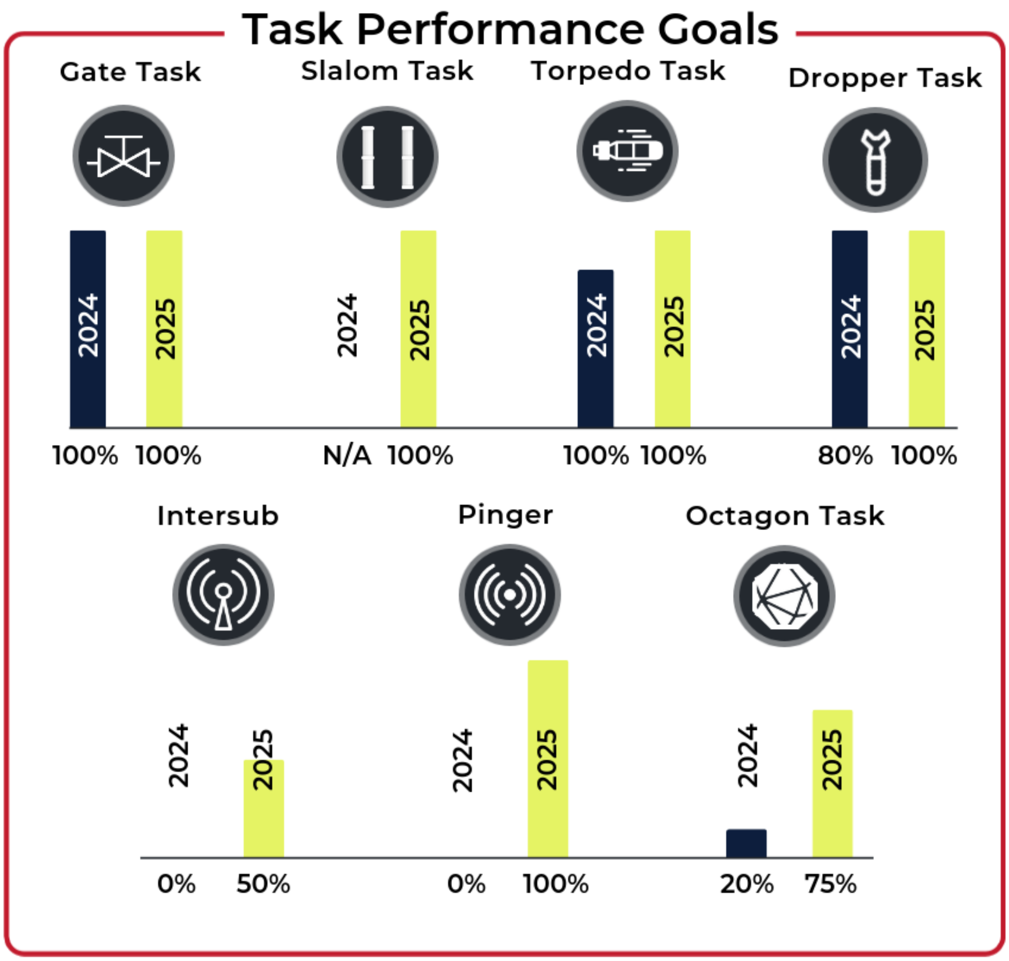

Figure 4. RoboSub 2025 Task Performance Goals

Appendix B: Hardware Architecture

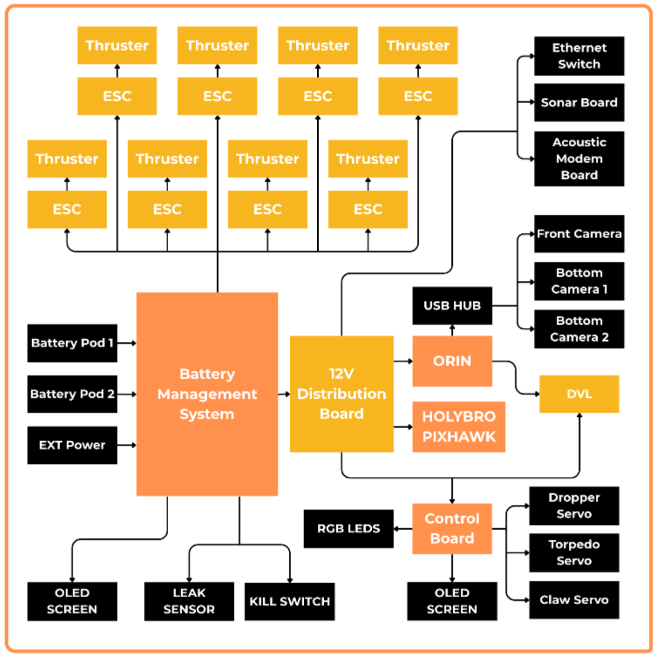

Figure 5. Kenai Power Architecture Diagram

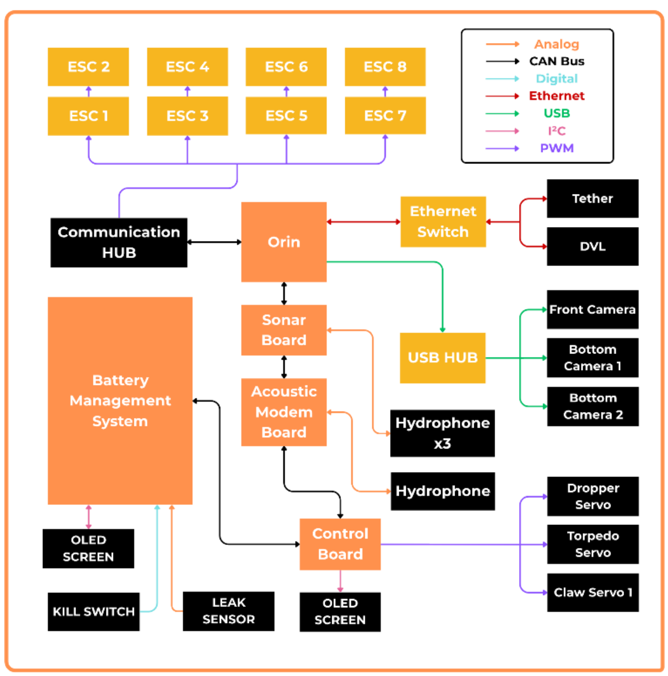

Figure 6. Kenai Communications Architecture Diagram

Appendix C: Software Architecture

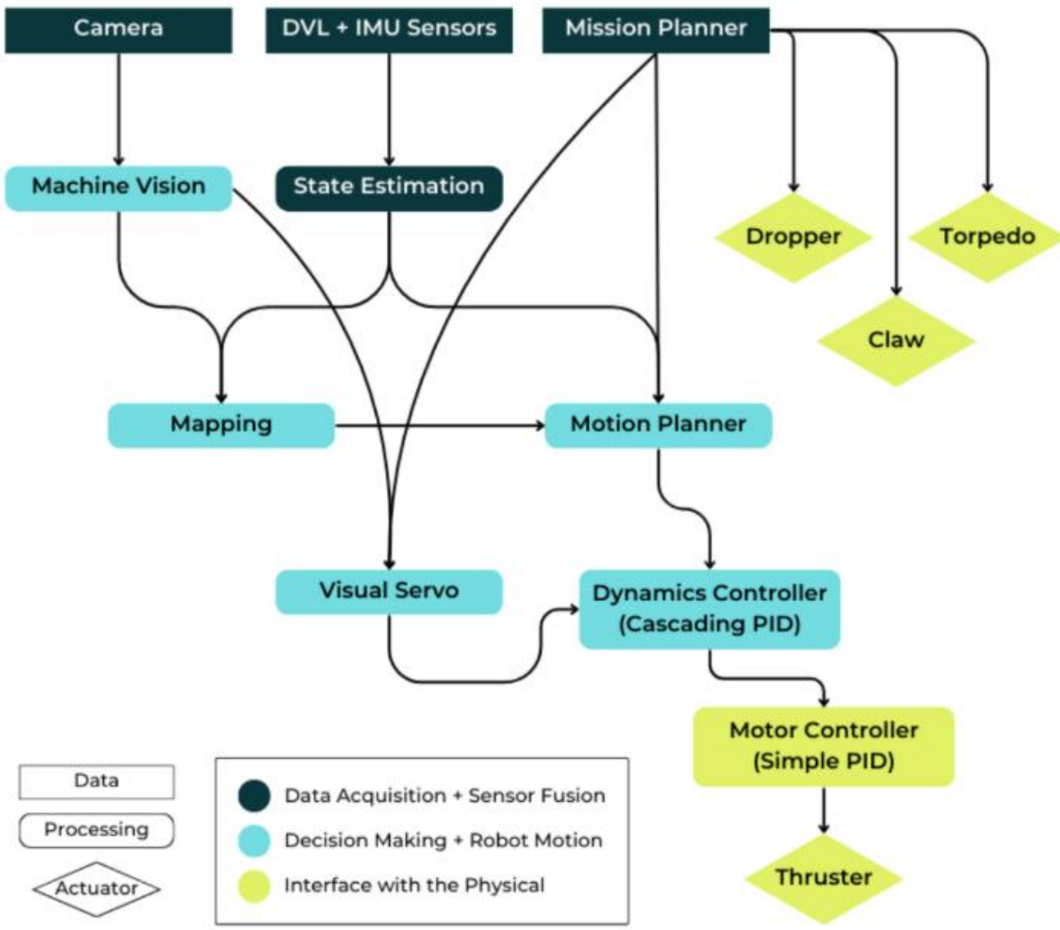

Figure 7. Kenai Software Architecture Diagram

Appendix D: Pool Testing Procedures

- *Prior to Pool Test (Day of, 4 hours in advance)*
    - Pull latest main branch onto onboard NVIDIA Jetson Orin
    - Build docker container using build script and start container
    - Build ROS workspace in repository and source workspace
    - Check CANBus connection for all 8 thrusters, internal environment sensor, leak sensor, and batteries
    - Connect to robot camera feeds and verify correct camera orientation and that each camera is connected to correct USB port
    - Pack pool test boxes for each team according to subteam packing lists
- *Before Leaving (1 hour in advance)*
    - Build and seal robot; pull vacuum on robot to prepare for pressure test
    - Prepare battery pods for pool test
- *At The Pool*
    - Connect internet switches to power. Connect tether into ethernet switches and connect to robot over SSH
    - Use internet forwarding script to forward internet from member's laptop to robot
    - Connect member's laptops to robot ROS connection to allow for viewing robot data and visualizations
    - Run pressure test
    - Build competition obstacles, attach weights, and place in pool to replicate RoboSub course
    - Following a successful pressure test, move the robot to the pool. Use XBox Controller script to test successful movement along all axes.
    - Begin planned pool test procedures

Appendix E: Pool Test Time Allocation

Completed Pool Test Time (As of June 30th)

Date	Testing Done
Sept-Dec 2024	No formal pool tests. Frequent visits to the water resource lab to pressure test Arctos and check if it holds pressure during Arctos restoration.
January 23 (1hr)	Tub time test with Arctos to verify all CANBus connections and cameras before the first pool test of the year.
January 25 (3hr)	First pool test for Arctos. Verifying basic software xbox control, PID, and basic functionality of motion planning, mission planning and vision systems, testing for regressions.
February 23 (3hr)	Begin preparation for community showcase by verifying Arctos autonomy on the gate, buoy, and bin task. Recalibrating Doppler Velocity Logger to reduce yaw drift.
March 6 (1hr)	Tub time test with Koda to confirm all connections and thrusters prior to its first pool test.
March 8 (3hr)	First pool test for Koda and tuning PID for Koda. Verifying the full sequence of gate, buoy, and bin task together on arctos.
March 15 (3hr)	Continued PID tuning for Koda; Starting to re-verify robosub 2024 torpedo task on Arctos, running into vision model issues.
April 5 (3hr)	Testing basic autonomous movement for Koda with a new motion planner. Collecting more data for vision models with Arctos as existing models were struggling.
May 2 (3hr)	Testing the RoboSub 2024 torpedo and surfacing tasks with Arctos; skipped test for Koda. Diagnosing and debugging an issue where path traversal would get stuck.
May 10 (3hr)	Diagnosed the path traversal issue to be speed controller related, and implemented a fix on Arctos; testing the bin and surfacing tasks from RoboSub 2024 for showcase. Testing a simple gate task with Koda.
May 17 (3hr)	Practice runs for our community showcase event on May 18th; running full tests of RoboSub 2024 course with Arctos. Running a simple gate task with Koda.
May 18 (6hr)	Community showcase event. Running full runs of the RoboSub 2024 gate, buoy, bins, torpedo, and surface tasks, verifying that functionality from last year is fully stable and ready to be adapted for RoboSub 2025.
May 31 (3hr)	Pool party to celebrate a successful showcase!

June 7 (3hr)	Testing vision model framerates and detection ability on Koda, and framerate improvements from converting vision model to TensorRT and quantizing to FP16. Arctos retirement.
June 20 (2hr)	Tub time in the water resources lab to verify all CANBus connections, thrusters, and cameras on Kenai before pool test.
June 21 (3hr)	First pool test with Kenai. Re-tuning PID system for Kenai's thruster layout. Testing oriented bounding box vision model alignment with Kenai. Collecting vision data on all RoboSub 2025 tasks using Koda.

Future Pool Test Plans (As of June 30th)

Date	**Testing Planned**
July 5 (3hr)	RoboSub 2025 Torpedo, Bin, and Gate tasks verified using a newly trained vision model on Kenai. Testing visual servoing on Koda for the gate task.
July 12 (3hr)	Robosub 2025 Torpedo and Claw testing for Kenai. Testing Slalom task for Koda. Intersub communication testing.
July 26 (3hr)	Further testing and refinement of claw and slalom tasks for Kenai and Koda respectively. Intersub communication testing.
August 3 (3hr)	Running full practice runs of the RoboSub 2025 course, as well as being a flex spot to catch up on any critical items.

Appendix F: Pressure Testing Procedure

Setup and data collection:

1. Pull a 10 kPa vacuum (equivalent to 1 m dive depth).
2. Start CANBus parsing ROS node on onboard NVIDIA Jetson Orin computer
3. Start pressure and temperature data recording script on onboard computer
4. Wait for 15 minutes.
5. Copy log file with data to local computer for analysis

Analysis:

1. Initial pressure (P) and temperature (T) values, and the internal hull volume (V) are used to calculate the amount of air (n) in the hull using the ideal gas law.

$$PV=nRT$$

2. This starting amount of air is used with temperature measurements to calculate an expected internal pressure assuming no leak for the rest of the test.
3. Measured and expected pressure values are plotted and the difference recorded.

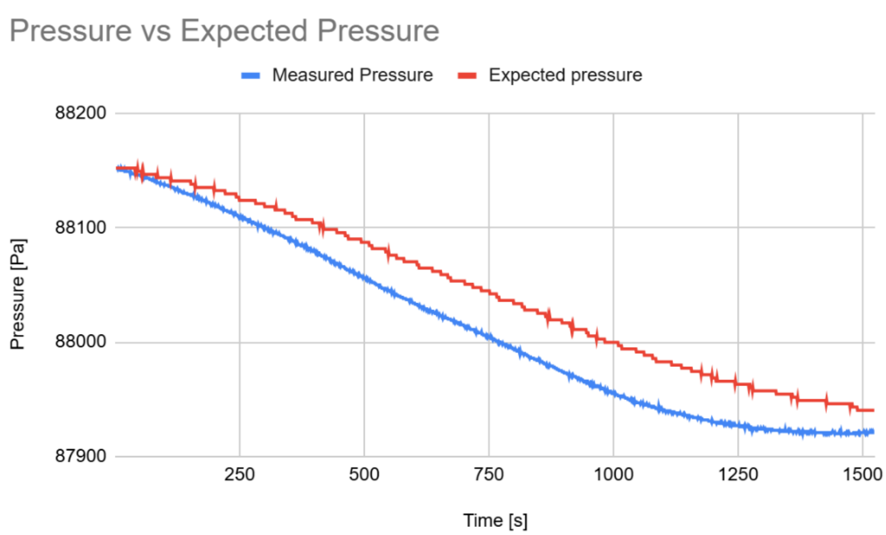

Figure 8. Example of successful pressure test plotted data.

Figure 8 above shows a successful pressure test, as the measured pressure remains below the expected pressure.

Appendix B: New Hull Testing Plan

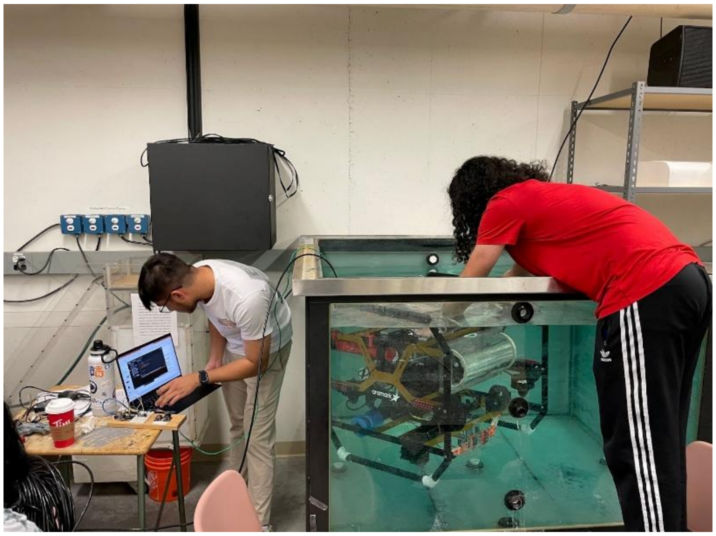

Figure 9. Water resources lab setup for testing hull integrity (pictured is previous robot, Arctos, being tested).

Scope:

- Confirm successful seal of machined hull at flanges and penetrators.

Resource and Tools:

- BlueRobotics Bar30 High-Resolution Pressure/ Depth Sensor
- BlueRobotics Celsius Fast-Response Temperature Sensor
- Logic level converter
- Arduino

Environment:

- University of Alberta Water Resources Lab tank (shown in figure _ above)

Steps:

1. Grease and insert o-rings in flanges, penetrators, and subcons.
2. Tape paper towels at ends of hull and around penetrators.
3. Assemble hull with endcaps and locking rings.
4. Follow pressure testing procedure in air to ensure seal.
5. If successful, pull a 60 kPa vacuum (equivalent to our maximum 6 m dive depth).
6. Fully submerge the hull in water.
7. Start recording pressure and temperature data at 30 s intervals for a duration of 3 hours (average length of our pool tests).
8. Remove from water and towel dry.
9. Disassemble hull, inspecting interior paper towels for water and plot measured pressure and calculated pressure values to check that measured does not exceed calculated.

a. If water is present, the paper towel should indicate where the water entered.

10. If successful, repeat overnight and re-inspect in the morning.

Results:

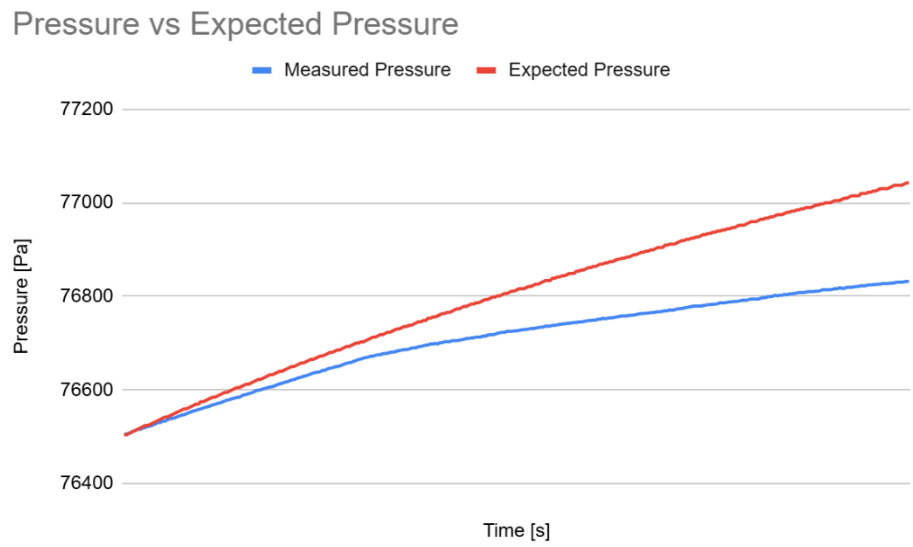

Figure 10. Results from overnight test.

As seen in figure _ above, the pressure remained below the expected pressure throughout the overnight trial. Additionally, after both the 3 hr trial and overnight trial, no water was observed in the hull. This is considered a success and the hull is viable for regular operation.

Appendix H: Control and Monitoring Verification and Validation Plan

Verification

Objective:
Verify each electrical subsystem of Control Board Rev B prior to integration.
Tools Used:

- Teensy 4.1 MCU
- Oscilloscope
- Variable electronic load (up to 3 A)
- Serial console and logging
- Multimeter

Firmware Used:

- C++ firmware acceptance test suite
- Custom logic for PWM, CAN, servo, I²C, and environmental sensor verification

Subsystems Tested:

- **CAN Bus:** Transmit frames at 10 Hz; confirm loopback and RX/TX integrity.
- **PWM Outputs:** Sweep from 30–1000 Hz across all output channels; verify with oscilloscope.
- **INA219 Current Sensors:** Use adjustable load to step current from 200 mA to 2.7 A; confirm sensor linearity and timing.
- **MS5611 & HIH61xx Sensors:** Capture and verify pressure, humidity, and temperature data; compare to reference.

Results:

- All six INA219 sensors read current within ±5% of expected value.
- All PWM channels operated continuously without jitter or dropout.
- CAN bus maintained stable communication under varying message loads.
- Sensor data was consistent with ambient conditions.

Validation

Objective:
Confirm control board functionality in full mission context using Kenai's hardware stack.
Method:

- Connect all mission-critical actuators and sensors via final wiring harness
- Integrate control board into Kenai's CAN and power system
- Run simplified task plans to evaluate performance under task-specific loads

Tasks Evaluated:

- **Claw Task:** Monitor INA219 sensor feedback; verify no stall detection logic triggers retry behavior
- **Dropper Task:** Verify predictable marker release via servo control
- **Torpedo Task:** Validate control timing and pulse width accuracy for firing
- **Intersub Communication:** Confirm RGB LED signaling timing underwater
- **CAN Communication:** Confirm reliable data transfer between Jetson and Teensy; no missed commands

Results:

- All tasks completed successfully during testing
- Stall detection logic operated as intended

- No hardware faults, connector issues, or signal instability observed
- Full end-to-end performance confirmed

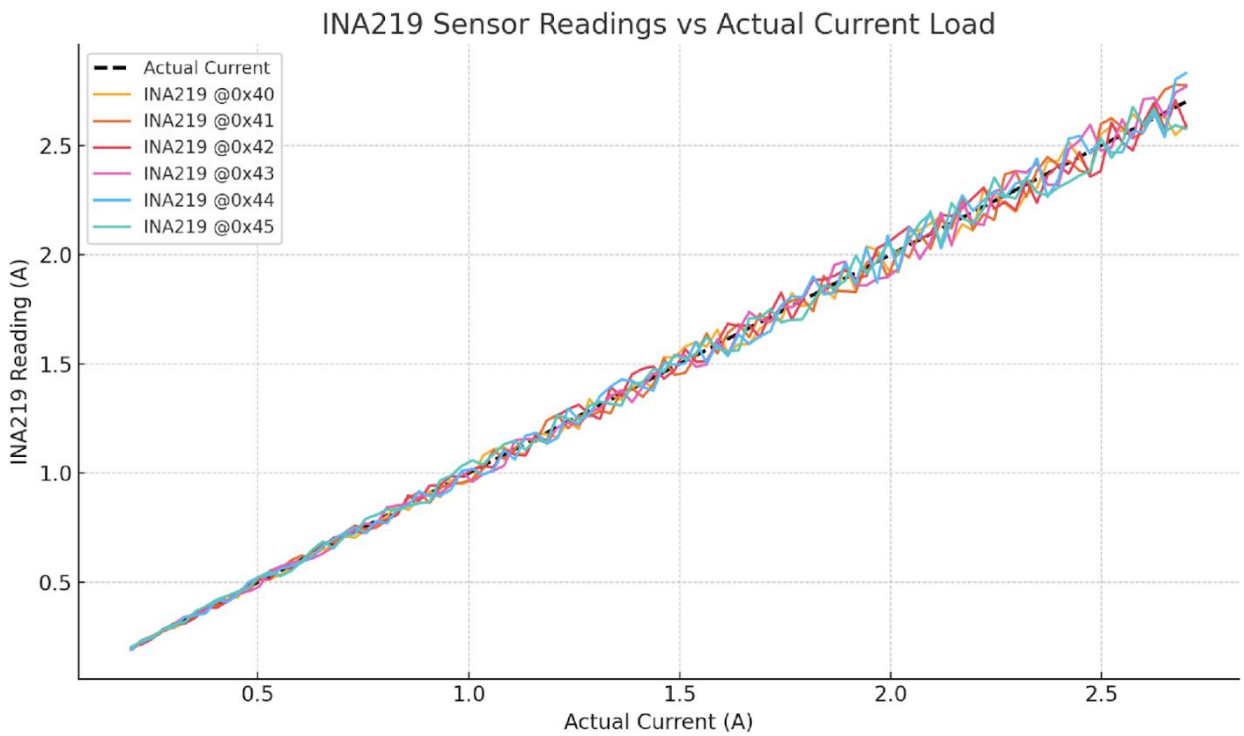

Figure 11: INA219 Sensor Readings vs Actual Current Load
This graph compares actual applied current (0.2 A to 2.7 A) with sensor readouts from six INA219 channels. Each line represents a unique INA sensor address. Readings remain within ±5% error margin across the full test range, confirming the sensors' linearity and suitability for stall current detection and feedback logic. The black dashed line represents the ideal reference line for perfect measurement.

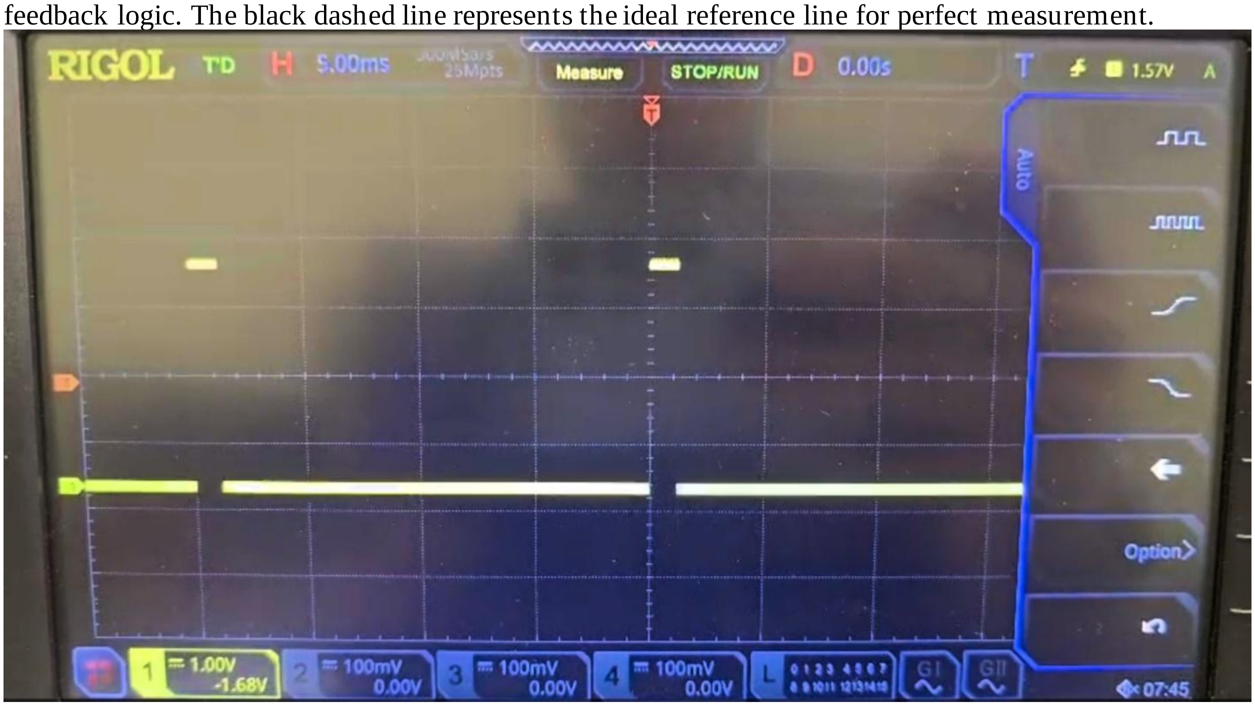

Figure 12: PWM Signal Output Captured on Oscilloscope
Oscilloscope capture of one PWM output channel from the control board during a frequency sweep test. This snapshot confirms clean digital transitions and a consistent duty cycle under a test load, verifying signal integrity at ~5 ms intervals. This type of validation is critical for servo control accuracy and signal stability over long wiring harnesses.

```
Sent → ID=0x10 LEN=8 DATA=[ 0xB8 0x00 0x00 0x00 0x00 0x00 0x00 0x00 ]

--- Sensor Readings ---
Pressure    = 927.92 hPa
Temperature = 28.94 °C
Humidity    = 28.20 %RH
HIH Temp    = 28.85 °C

--- Current Readings (INA219) ---
INA219 @0x40: BusV=7.424 V, Shunt=-0.040 mV, Curr=-8.0 mA
INA219 @0x41: BusV=7.420 V, Shunt=0.010 mV, Curr=2.0 mA
INA219 @0x42: BusV=7.424 V, Shunt=0.000 mV, Curr=-2.0 mA
INA219 @0x43: BusV=7.400 V, Shunt=0.020 mV, Curr=2.0 mA
INA219 @0x44: BusV=7.388 V, Shunt=5.610 mV, Curr=1122.0 mA
INA219 @0x45: BusV=7.392 V, Shunt=-0.030 mV, Curr=-6.0 mA
```

Figure 13: Control Board Serial Output During Verification
This image shows raw serial output from the control board verification test. The log includes successful CAN message transmission, sensor readings from the MS5611 and HIH61xx (pressure, temperature, humidity), and INA219 current sensor readings from all six channels. Channel @0x44 registers a current draw of 1122 mA, confirming live load test functionality, while other channels show idle or near-zero current.

Appendix I: Vision and Mapping Testing Procedure

Phase 1: Code Validation and Frame Rate Testing

Scope:

- Ensure model inference speed is sufficient for 15 FPS frame rate for bounding boxes, the frame rate we run model inference at.

Resource and Tools:

- Main onboard computer, NVIDIA Jetson AGX Orin
- DWE underwater camera used for both Kenai and Koda
- Sample robosub object as a test object for models

Environment:

- Custom Docker container on Jetson AGX Orin
- Testing done outside of a pool test with onboard computer and camera only

Steps

1. Copy a trained YOLO bounding box, oriented bounding box, and segmentation vision model to Jetson Orin
2. Connect camera to Jetson Orin through USB.
3. Run camera node to begin publishing camera images
4. Place test object in front of the camera
5. Run vision node with arguments for bounding box model; ensure no crash
6. Use ROS command line utilities (rostopic hz) to check output rate of vision detection topic.
7. Use YOLO verbose mode to record inference time for an image with the model
8. Use vision model visualization to check correct prediction on the test object
9. Repeat steps 5-8 with OBB model and segmentation model

Risk Management:

- Negligible risk due to separate git branch

Results: Detailed outcomes of test cases

- Output from test without model optimization is as follows, with all models being based on the "small" variant from YOLOv11:
    - YOLO Bounding Box: 27.701 fps
    - YOLO Segmentation: 25.143 fps
    - YOLO Oriented Bounding Box: 17.791 fps
- Result is a success. Slower speed on oriented bounding box model indicates need for conversion to TensorRT model for this

Phase 2: Mapping Integration Simulation Test

Scope:

- Ensure orientation from the OBB model is correctly parsed by the mapping node into the object orientation.

Resource and Tools:

- ARVP Orca server

Environment:

- Custom Docker container on Jetson AGX Orin
- Custom Gazebo simulator for Kenai
- Simulated vision detections running

Steps

1. Launch Gazebo simulator with any object placement configuration file
2. Start mock vision plugin of Gazebo simulator to start publishing oriented bounding boxes
3. Launch mapping node
4. Use ROS topic utilities to view output position from mapping of an object of choice.
5. Compare output position and orientation from mapping to the true position and orientation from the simulator's configuration file; validate whether the mapping estimate converges to near to the true position.
6. Repeat steps 3-5 with the robot at a different position relative to the object

Risk Management:

- Negligible risk due to separate git branch

Results:

- Success; in simulation mean squared error between true robosub 2024 torpedo banner and estimated torpedo banner pose <2cm. Orientation error less than 0.1 radians.

Phase 3: End-to-End Testing of Alignment Ability

Scope:

- Test ability for Kenai to detect and align to match the orientation of a table object using new oriented bounding box models

Resource and Tools:

- Kenai
- Main onboard computer, NVIDIA Jetson AGX Orin
- DWE bottom facing underwater camera
- RoboSub 2025 Table
- RoboSub 2025 collection bins, bottles, and ladles

Environment:

- Custom Docker container on Jetson AGX Orin
- Table placed on pool floor with one bottle and one ladle randomly assorted
- Kenai beginning directly above table on the pool surface

Steps

1. Begin with Kenai aligned above table, and one bottle and ladle on table
2. Start vision node with YOLOv11 oriented bounding box model
3. Start mapping node
4. Launch alignment mission
5. Observe the robot motion and final alignment; note down whether it matches object alignment
6. Repeat steps 1-5 several times with different locations of the bottle and ladle

Risk Management:

- Minimal risk of robot losing control and hitting wall/floor; however, deemed unlikely due to success of previous two testing phases, testing well away from walls, and swimmers prepared to kill robot if necessary.

Results:

- In-water alignment testing over 10 trials showed success on all 10
- Further full testing on RoboSub 2025 course tasks in progress at time of report writing.

Appendix J: Detailed Acknowledgements

We would like to thank our principal advisor, Dr. Samir Ead, for providing design feedback, management guidance, and logistical support.

We also recognize additional mentors: Nathan Chong and Logan Krezan for supporting systems engineering, documentation reviews, and project management; Nazmul Arefin, David Lenfesty, and Alain Letourneau for their contributions to electrical systems development and testing; Stevie Desmarais for mechanical development support.

Special thanks to industry contributors Dr. Zhaohui Wang for underwater acoustics consultation, Paul Bakhmut for fundraising and policy support, and Frankie Mantella for PCB design expertise.

We are grateful to the University of Alberta's Experiential Learning Department, Water Resources Lab, Engineering Safety and Shipping, and Engineering Student Society for financial, testing, and logistical support.

Additional thanks to Marl Technologies, Anderson Anodizing, Altium, GitLab, and Elegoo for their in-kind contributions. We appreciate our partner sponsors — Altium, UofA Engineering, and Aramark — as well as our support sponsors: Marl Technologies, Copperstone Technologies, ESS, Elegoo, Rail Shop Services, GitLab, and Calvary Fence.

Finally, we thank again our 27-year-strong alumni community for their continued advice, mentorship, and engagement at events!