Tartan Autonomous Underwater Vehicle Design and Implementation of TAUV-22: Kingfisher

Theo Chemel, John Duncan, Sarah Fisher, Rishabh Jain, Rylan Morgan, Katia Nikiforova, Micah Reich, Sam Schaub, and Tom Scherlis

Carnegie Mellon University

Abstract - TartanAUV (TAUV) is a Carnegie Mellon University affiliated Robosub team, driven by undergraduate students from diverse backgrounds and disciplines. For our third year of competition, we achieved our goal of developing a large and extendable AUV: Kingfisher. Equipped with a Doppler Velocity Log, imaging sonar, and multiple stereo cameras, Kingfisher is now capable of prolonged underwater navigation with an astonishing degree of accuracy. Our new onboard computer enables computationally intensive vision and trajectory planning algorithms that were previously out of reach, and fine-tuned software stacks have opened the possibility for the completion of a broader array of underwater tasks. With this strong foundation, our team will be able to expand Kingfisher's capabilities for years to come.

1 Competition Strategy

Although TartanAUV is a third-year competition team, we consist primarily of first-year students working toward the driving goal of advancing Kingfisher, our inherited submarine, to its fullest potential. This robust AUV, used in only one prior year of competition, has continued to demonstrate its intended versatility and sustainability. We have taken the learnings from our previous two years of competition and adapted our strategy accordingly, combining it with the improvements we have made to our vehicle to fully take advantage of Kingfisher's competition potential.

We realized after our first year of competition that it is important to focus on mastering a few tasks instead of halfheartedly attempting all of them. Therefore, we adapted our mechanical, electrical, and software interfaces with specific intent.

Our already modular and versatile electrical and mechanical systems have undergone modifications to improve performance and enhance their compatibility with the competition tasks. New to our submarine this year is the incorporation of a marker dropper mechanism and a mechanical arm. These new additions challenged our mechanical team while giving Kingfisher the ability to complete a broader set of competition tasks.

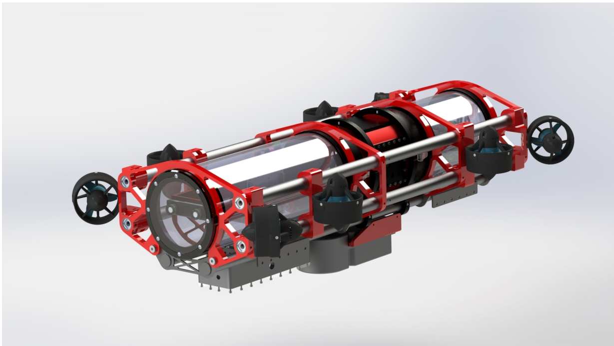

Figure 1: Kingfisher

Furthermore, the new software system upholds autonomous flexibility, capable of adapting Kingfisher to its environment - a massive improvement over our previous hard-coded, task-specific stack. This reimplementation of our software system has also been more accessible to our growing team's new members, who offered major contributions and new problem-solving approaches.

We spent considerable effort this year on the transfer of knowledge from our more experienced members to our future experts as we expanded our team after the pandemic. However, we still maintained our plan to complete the first several tasks with a high degree of precision and expanding Kingfisher's flexibility. We primarily focused on software and maneuverability-based tasks, with a further goal of incorporating the mechanical arm manipulator that has already been designed and engineered. We were able to master the coin flip, gate, buoy, and bin tasks by rigorously testing these scenarios in both simulation and water tests. We also focused our experimentation on our upgraded sensor suite, which now includes an imaging sonar, stereo cameras, and DVL.

2 Design Creativity

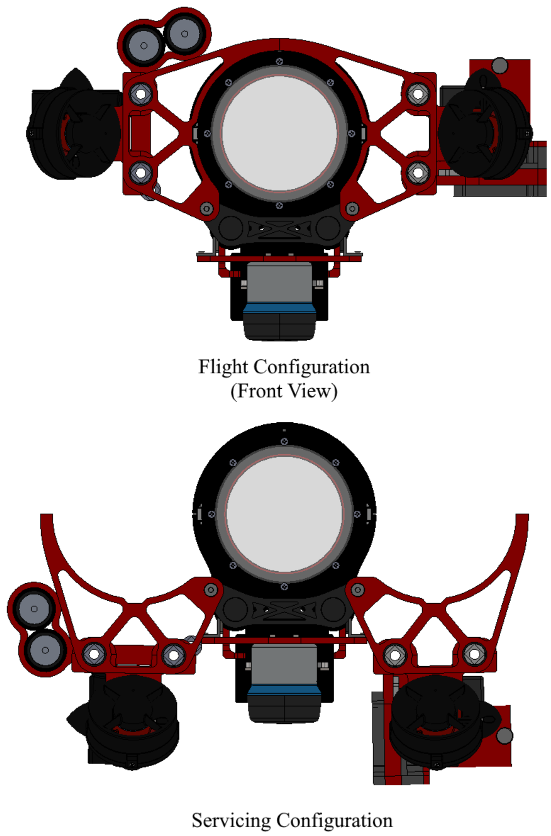

Figure 2: Structural System CAD Model

2.1 Mechanical Design Drivers

This year, we continued to develop Kingfisher's mechanical platform that began to take shape during the 2020-2021 season. Our primary mechanical objectives sought to implement manipulators to accomplish tasks requiring more than simple navigation while maintaining modular accessibility to our recently upgraded electronics and sensor systems. In the main chamber of the vehicle, we identified accessibility to internal electronics, modularity of vehicle internals, and thermal management of our internal electronics as three major areas requiring improvement. External to the vehicle, we also developed two manipulator systems - a marker dropper and servo-powered gripper - to grasp and release objects with precision.

2.2 Structural System

The core of our AUV is a pressure vessel, consisting of two transparent acrylic tubes joined by a central aluminum "midcap." The tubes are joined to the midcap with temporary double o-ring seals, so they can be easily removed in a field environment for on-the-fly electronics servicing. The midcap is a large (7" diameter), 5-axis CNC machined part, which constituted a significant manufacturing challenge for our team. The midcap contains two o-ring tube seals on each end and four face seals around the circumference of the part, which seal four removable panels containing cable pass-throughs. The enclosure is separated into two halves, one for battery and power electronics and one for compute and perception electronics.

Supporting the entire system sits a folding chassis design, where a collapsible aluminum "ribcage" allows us to have a compact profile during operation and shipping while also giving open access with minimal obstructions during electronics servicing. The main acrylic electronics enclosure is reinforced underneath with a steel weldment, termed the "spine," to reduce vibration inside the enclosure where sensitive electronics sit.

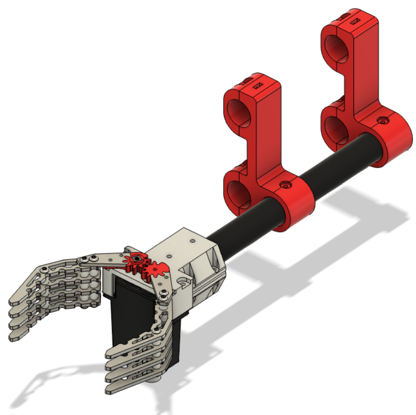

Figure 3: Robotic Gripper (with mounts)

2.3 Manipulators

Beyond the internal components of the AUV sit two manipulator mechanisms, a marker dropper and a robotic gripper. While the marker dropper is specially designed for a specific task, the robotic gripper is designed as a general-purpose, reusable actuator that can pick up and release objects that are 0.75 to 5 inches wide, including a bin cover and PVC bottles. The gripper is controlled with a Blue Trail underwater servo with power transmission by spur gears mounted on the superstructure of the AUV in the same way as our thrusters: using custom adjustable clamping mounts. The marker dropper is designed to release a torpedo-shaped marker at the bin. To drop the marker, a servo rotates 90 degrees while the marker is held in place, breaking the magnetic attraction between the marker and the servo and causing the marker to be released downwards; the marker is also designed to travel a path with minimum drift as it reaches the floor.

3 Electronics

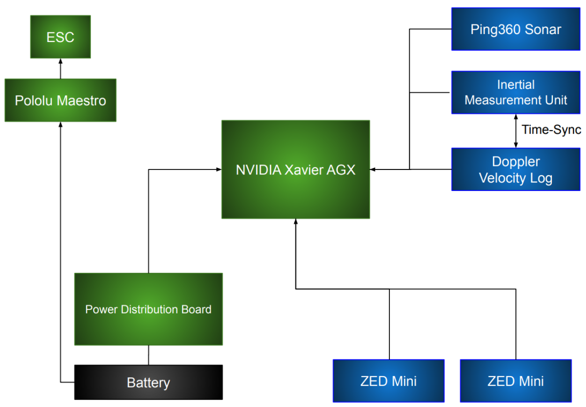

Figure 4: Electronics System Overview

3.1 Compute

In order to enable the development of more computationally intensive algorithms, especially for vision and navigation tasks, we upgraded Kingfisher's onboard compute capabilities. The NVIDIA Jetson AGX Xavier single board computer (SBC) was replaced with the latest generation NVIDIA Jetson AGX Orin SBC, which offers more CPU and GPU cores. Our software stack takes full advantage of multithreading, so the larger number of CPU cores significantly boosts performance despite the slower single-core speeds of the Orin SBC. Our vision stack also makes heavy use of CUDA GPU acceleration, so the Orin's improved GPU enables significant speedups in that domain as well. Despite this surplus of compute power, the new Orin SBC has the same form factor and connectors as the Xavier SBC, allowing for quick swaps if maintenance is necessary.

3.2 Power

Kingfisher's power system is still managed by our custom power distribution board (PDB), shown in Figure 4. While testing, we discovered stability issues with the PDB triggered by rapidly changing thruster inputs; thankfully, these issues were mitigated with only a few component changes and did not require a redesign of the PDB. In addition to the existing magnetic kill switch, which safely cuts powers to the thrusters, we added a dummy plug that can be removed to restart the entire power system, including the Orin SBC. In the event of communications issues, this can be helpful to power-cycle the SBC without opening the pressure vessel.

3.3 Sensing

This year saw the final integration of Kingfisher's new suite of sensors, the first of which is the Xsens MTi-200-VRU Inertial Meaurement Unit (IMU). We originally chose the Xsens IMU because of its resistance to unmodelled magnetic field distortion, which was a significant issue for our previous vehicle, Albatross. Coupled with the Xsens IMU is a Teledyne Pathfinder Doppler Velocity Log (DVL), which provides direct measurements of linear velocity that can be fused with orientation from the IMU to track the vehicles state with exceptional accuracy. Synchronization between the IMU, DVL, and SBC is particularly important to maintain accuracy, which we solved with a two-part system. First, the IMU and DVL are synchronized in hardware using the Xsens IMUs sync-out port, which allows DVL measurements to be timestamped relative to the IMU's internal clock. Then, a Kalman Filter inspired by the Cuckoo Time Translator [1] is used to continually estimate the bias and offset between the IMU and SBC clocks. Timestamps from the IMU can then be translated to the SBCs clock that runs the guidance system.

We also integrated the Blue Robotics Ping360 Sonar, which enables Kingfisher to detect objects at significant distances, even with poor visibility. Since the scan time of the sonar is so slow (on the order of 30s per revolution) it is necessary to accumulate sonar returns while the sub is moving through the water, building up a map as it explores the environment. The high degree of positional accuracy provided by the IMU and DVL certainly helped, but limiting the effects of noise on the sonar maps still required a significant understanding of the physical limitations of the device. One significant obstacle was the large beam angle of the Ping360 Sonar, which created uncertainty in the position of detected objects that needed to be corrected for.

Finally, Kingfisher was outfitted with two ZED-M Mini Stereo Camera Modules, one oriented forwards and one oriented downwards. By calculating the disparity between the left and right cameras on each module, a measure of depth can be obtained, which aids in the detection and localization of objects. Since the bandwidth of these cameras is so high, an external PCI USB hub was needed to connect them to the SBC.

3.4 Communications

We continued to succeed with our existing topside communications setup from previous years, including the Blue Robotics Fathom Tether and Subconn connectors. The wet-matable Subconn connector enabled us to quickly connect and disconnect from the vehicle when testing autonomous runs, and the connected WiFi router allowed multiple team members to interact with the sub simultaneously.

In terms of internal communications, we expanded our use of RS-232 for interfacing between the IMU, DVL, and SBC, and relied on USB for communication between the cameras, sonar and SBC. We also added a simple I2C bus to connect the SBC with the battery monitor chip on the PDB, allowing for Kingfisher's control software to continually monitor battery voltage. Since the actual force provided by our thrusters varies based on the battery voltage, this information is fed into the control system to enable corrections for drooping battery voltage.

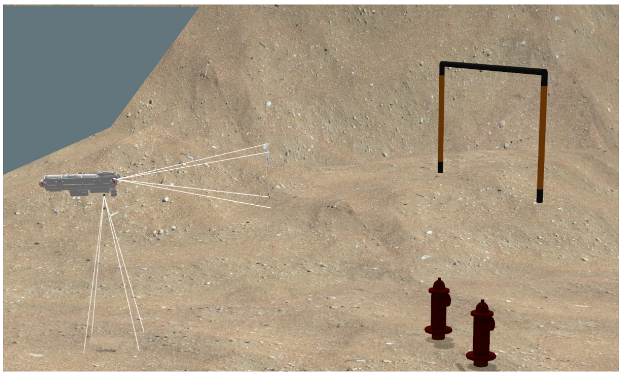

Figure 5: Simulated Environment

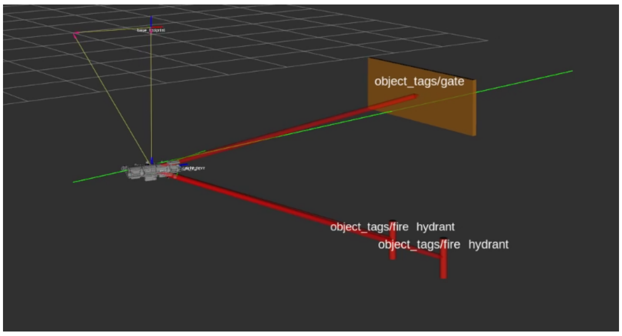

Figure 6: Internal Perception System

4 Software Systems

While expanding our software stack for this year's competition, our software team focused on creating robust, reusable solutions to key autonomy problems that could be used at a high level to solve different tasks. We continued our use of ROS on our SBC because of its ease of use and excellent support for multithreading. Our software stack was split and worked on as four interconnected categories: Perception, State Estimation, GNC (Guidance, Navigation, and Control), and Mission Planning.

4.1 Perception

Kingfisher was designed with a versatile object awareness system, capable of detecting and tracking any number of objects of any size and category specified by the submarine's object recognition system. The tracking aspect of the system is handled by Kingfisher's Detector Bucket, a software structure managing the calculated global positions for each distinct object of interest.

The Bucket System relies on object detections from Kingfisher's base perception framework, which was refined and retrained this year for better competition performance. The submarine's general object detection method was updated to be based on the Darknet YOLOv4 Neural Network [2], which was trained on underwater recordings of competition object replicas from Kingfisher's cameras. This system was paired with improved classical Computer Vision detectors for simpler perception tasks such as gate identification, which used traditional line-fitting to reduce overhead and Neural Network complexity.

The Bucket System process chain was also completed this year, updated to be invoked by any detection from Kingfisher's joint perception system. Using the submarine camera's depth perception map, the detection's corresponding YOLOv4 or Computer Vision bounding box dimensions, and the submarine's global orientation – sourced from Kingfisher's state estimation system, a position for the object relative to the submarine was calculated to begin the tracking process. By pairing this information with Kingfisher's global position, an estimate for the global position of the detected object was found.

Each detection identification and its corresponding position estimate is reported to the Bucket System, triggering a Detector Daemon spawn to process the data and update the submarine's world map, represented as a list of identified distinct objects and their estimated positions. Independently of new detections, each Detector Daemon filters and analyzes its assigned data, which is then asynchronously merged with the other available data. This abstraction accounts for asynchronous sensor information and minimizes the latency between detections and updates to the global map.

When spawned with new detections, each Detector Daemon uses the Mahalanobis Distance and the Kuhn–Munkres (Hungarian) algorithm [4] to attempt to match detections to previously identified objects in Kingfisher's world map or else establish the detection as a newly identified object. Due to the inherent uncertainty of perception sensors and possible state estimation drift that may cause deviations in the calculated positions of different detections of the same object, using Mahalanobis Distance is vital to accommodate for positioning variability. Additionally, prior known information about objects of interest is incorporated into this detection matching algorithm using a configurable object manifest, which allows the system to consider the dimensions, prior location estimates, and the quantity of objects of any particular classification during detection-entry matching.

After a detection is classified as a new object or one referring to an existing world map object, it is respectively either directly added as a Bucket System entry or used to update the existing entry's position estimate using a constant position 3D Kalman Filter Tracker. With this consistent recalculation of global map object positions using every new detection as an entry position estimate, Kingfisher's 3D perception system is used to build out a thorough and efficient object awareness model that allows for accurate navigation (Figure 6).

4.2 State Estimation

Maintaining an accurate estimate of Kingfisher's position, orientation, and velocity is critical to our ability to accurately navigate between tasks. Underwater state estimation is made particularly difficult by the lack of GPS and visual features; however, we found great success by fusing information from our IMU, DVL, and depth sensor. An Extended Kalman Filter (EKF) continually updates an estimate of the vehicle's position, velocity, acceleration, orientation and angular velocity based on new sensor measurements and its understanding of the kinematic equations governing the motion of the vehicle.

One notable technical challenge was presented by the time delay of our DVL sensor. In order to prevent phase margin issues in the control system, the state estimation system must run as close to real-time as possible. However, Kingfisher's Teledyne Pathfinder DVL takes roughly 400ms to process velocity data before it is made available to the state estimation system. In order to alleviate this issue, the system maintains a circular buffer of the last 400ms of states, allowing it to roll the EKF back in time to account for new velocity measurements as they arrive from the DVL.

4.3 Guidance, Navigation and Control

Kingfisher's GNC system is divided between two primary controllers. The stability of the vehicle about the pitch and roll axes is controlled by an attitude PID controller, which supports changing the target roll and pitch for complex maneuvers. The motion of the vehicle in the x, y, z, and yaw axes is controlled by a more advanced linear Model Predictive Controller (MPC). At the highest level, the MPC controller solves for optimal control inputs to follow a given trajectory using the OSQP quadratic programming toolbox. These control inputs are accelerations along the x, y, z, and yaw axes, which are translated into a force and torque wrench by a differentially flat dynamics model.[3] This model accounts for buoyancy, lift, and drag to enable precise control of the vehicle. The resulting force and torque wrench is then acted upon by the Thruster Allocation Matrix (TAM) to determine the individual thruster forces. Finally, these thruster forces are fed through the inverse thruster dynamics model to determine the thruster speeds that should be commanded.

High level guidance is provided using a Motion Utilies service, which provides easy-to-use trajectory primitives for mission applications. The Mission Planning system can easily construct trajectory primitives, and the Motion Utilities service continually provides a 1 second horizon reference trajectory to the MPC controller. These trajectory primitives have been expanded to include finite jerk linearly interpolated trajectories, minimum snap trajectories computed by a custom 5th order polynomial spline optimizer based on OSQP, and simple ray trajectories for following a straight line indefinitely. (Figure 7)

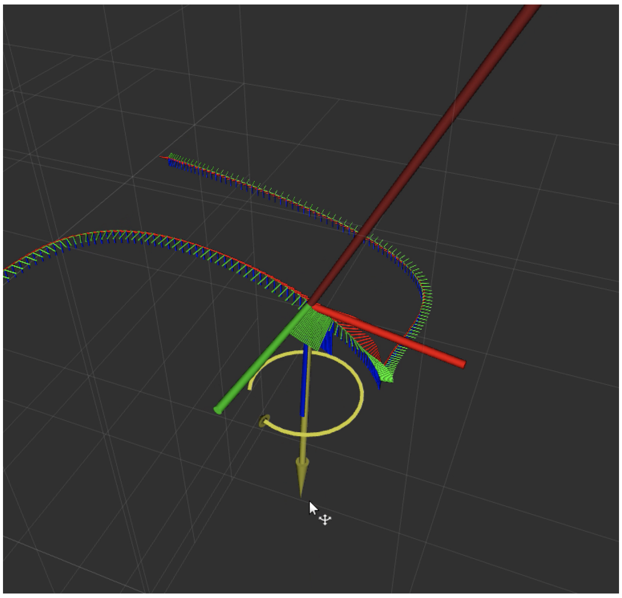

Figure 7: GNC System

4.4 Mission Planning

The Mission Planning software system is the high-level orchestrator of the vehicle, interacting with all other systems in order to complete tasks. Each task has an associated Mission Planner that interacts with the Perception system to identify and localize objects of interest and uses the Mission Utilities service to perform the required movements. Each Mission Planner maintains control of the vehicle until its specified task has been completed and the vehicle has reached a state where it can begin attempting the next task. By dividing the Mission Planning system in this way, it becomes easy to isolate specific tasks while developing and testing the system.

4.5 Simulator

The ability to test software systems offline is critical for fast iterations and robust software solutions. To enable this, we continued development of our simulator framework, which builds on a foundation of the UUV Simulator [5] project. By creating custom plugins that emulate the behavior of our sensors and designing digital replicas of the competition environment, we are able to continue development and validation of our software stack even when water testing is unavailable.

4.6 Frameworks

Finally, we have continued the development of a set of frameworks aimed at improving software reliability within our codebase. Most importantly to system safety, we designed a fault tracking service that allows different software components to set and unset fault statuses in order to centralize information about system health. By including information about fault severity, the Mission Planning system can respond intelligently to different events, attempting to recover in the case of minor problems or aborting in response to severe errors.

5 Experimental Results

Thankfully, the lifting of COVID-19 restrictions enabled us to spend more time physically testing our submarine than was possible in recent years. We began our series of water tests by thoroughly checking Kingfisher's control systems in the high bay of Carnegie Mellon's Field Robotics Institute. We manually checked the functionality of these controls, comparing state estimation results to the observed motion of the vehicle. Once the controls were functional, we moved to Carnegie Mellon's campus pool to continue testing. In this larger space, we were able to setup our home-made buoys, gates, and other props, and swim alongside the vehicle in an environment similar to the competition venue. The larger pool also allowed for a testing environment with fewer echoes, which proved difficult for the DVL and sonar to handle. Furthermore, this competition-like environment allowed us to check the functionality of Kingfisher's perception capabilities with appropriate underwater footage. Beyond water testing, we also continued our simulation testing efforts in order to validate software changes between water tests, especially to our perception system.

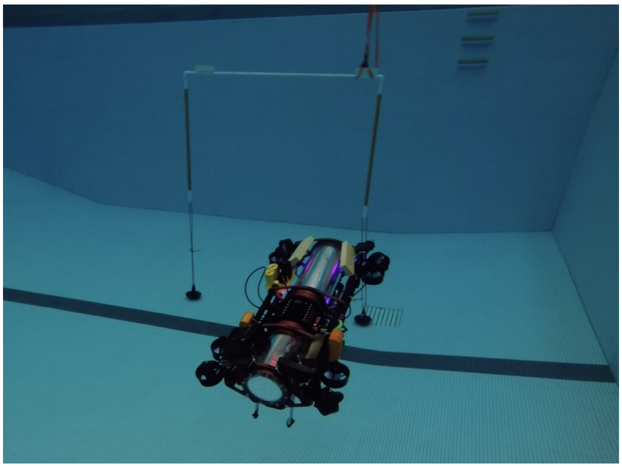

Figure 8: Kingfisher's Prequalification Run

6 Acknowledgements

TartanAUV would like to thank our parent organization, the CMU Robotics Club, as well as faculty member Michael Kaess, for their incredible mentorship and support. We would like to thank Chuck Whittaker and the Field Robotics Center for training and access to the Robotics Institute water testing facilities. We would also like to thank to Alicia Gorman and the Carnegie Mellon Athletics department for access to the Cohon University Center pool for additional testing times in the competition pool environment. We would like to thank Jim Bain (CMU ECE), Margaret Noel (CMU ECE), and Catherine Copetas (CMU SCS) for their crucial help in negotiating sponsorship deals. We would like to thank our gold sponsors: Leidos, Carnegie Mellon Student Senate, and Altium designer, our silver sponsors: Fischer connectors, Blue Trail Engineering, IMC Training, Lockheed Martin, NAPA Altoona, Stihl, and the Robotics Institute, and our bronze sponsors: NVIDIA, Robot Perception Lab, Advanced Circuits, Connect Tech inc, Molex, Blue Origin, and IMC. We look forward to everyone's continued support in our effort to bring RoboSub to the Carnegie Mellon University community.

References

[1] Cuckoo Time Translator, ETH Zurich. https://github.com/ethz-asl/cuckoo_time_translator.

[2] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *CoRR*, abs/2004.10934, 2020.

[3] G. Rigatos, G. Raffo, P. Siano. AUV Control and Navigation with Differential Flatness Theory and Derivative-Free Nonlinear Kalman Filtering.

[4] H.W. Kuhn. The Hungarian method for the assignment problem. 1955. https://web.eecs.umich.edu/~pettie/matching/Kuhn-hungarian-assignment.pdf.

[5] Musa Morena Marcusso Manhães, Sebastian A. Scherer, Martin Voss, Luiz Ricardo Douat, and Thomas Rauschenbach. UUV simulator: A gazebo-based package for underwater intervention and multi-robot simulation. In *OCEANS 2016 MTS/IEEE Monterey*. IEEE, Sep 2016. https://doi.org/10.1109%2Foceans.2016.7761080.

A Components List

Components	**Vendor**	**Model/Type**	**Cost (if new)**
Buoyancy Control	Blue Robotics	Buoyancy Foam	Already Owned
Frame	Various	Aluminum	Already Owned
Waterproof Housing	(Generic)	6in Acrylic Tube	Already Owned
Waterproof Housing	Custom	Aluminum Midcap	Sponsored
Waterproof Connectors	Blue Trail Engineering	Cobalt Series	Sponsored
Waterproof Connectors	Subconn	DBH8F/DOM8M	Already Owned
Thrusters	Blue Robotics	8x T200	Already Owned
Motor Controls	Blue Robotics	Generic ESC	Already Owned
High Level Controls	Custom		
Actuators	Blue Trail Engineering	Waterproof Servo	Sponsored
Propellers	See Actuators		
Battery	Various	4s LiPo 16Ah	Already Owned
Converter	Not Used		
Regulator	Texas Instruments	Various	Already Owned
CPU	Nvidia	Jetson AGX Orin	2000
Internal Comm Network	RS232/USB		
External Comm Network	Ethernet		
Programming Language 1	Python		
Programming Language 2	C++		
Programming Language 3	MATLAB		
Compass	See IMU		
IMU	Xsens	MTi-200-VRU	Sponsored
DVL	Teledyne	Pathfinder	Sponsored (Leidos)
Camera(s)	Stereo Labs	2x ZED mini	
Hydrophones	Aquarian	AS-1	2200
Manipulators	Custom		
Algorithms: vision	Darknet YOLOv4 Template Matching		
Algorithms: acoustics	FFT		
Algorithms: localization/mapping	EKF Pose Graphs		
Algorithms: autonomy	Model Predictive Control (MPC)		
Open source software	OSQP OpenCV ROS Gazebo UUV-Simulator		
Team size (number of people)	9		
HW/SW expertise ratio	1:2		
Testing time: simulation	> 100 hours		
Testing time: in-water	30 hours		