Team Owltonomous
Journal Paper for the 2022 RobotX Competition

Hayley Rienzo, Adriana McKinney, Eloise Cariou-Allard, Clayton Workman

***Abstract*—This document presents the design overview and challenges within the development of the 2022 WAM-V USV16 (16-foot Wave Adaptive Modular Vessel) as it is adapted to achieve the RobotX task requirements. The team is comprised of Florida Atlantic University (FAU) students. The vessels sensor suite employs a previously used RobotSub autonomous underwater vehicle (AUV). The vehicle has been modified to meet these RobotX specifications, RGB-D vision system robust to lighting variations, underwater USBL (Ultra Short Base Line) acoustic localization system, GPS aided MEMS-based Inertial Measurement Unit, and Velodyne VLP16 3D LIDAR. As a research platform, the FAU vehicle has served to further the development of adaptive control and robot vision systems. For example, the controller is robust to various environmental disturbances, including wind force, current, lighting variations and rain, while the vision system implements a novel data fusion agglomerative hierarchical clustering (AHC) to produce unique object detection. A hierarchical structure and finite state machine allow for the development of modular routines which can be rapidly implemented and are utilized to conduct mission-level control. Path planning, mapping, obstacle avoidance and navigation, and three degree of freedom state estimation embody the focus of Team Owltonomous WAMV USV16 platform.**

I. Introduction

This document serves as the means to convey our team's approach to the RobotX 2022 Competition Tasks. As a constructive Pacific-Rim Partnership between Universities from five countries, the fourth biennial RobotX 2022 Challenge presents autonomous system development for the maritime domain. The challenge is organized by the Association for Unmanned Systems International (AUVSI) and the Office of Naval Research (ONR) and seeks to foster innovation and develop competitive engineers with a thirst for robotics in the maritime domain.

The scenarios presented for each competition task simulate problems encountered in real-world applications, while dynamic task association enforces modularity and the need for a robust state machine within the software development scope. Success with this year's competition requires the RobotX vessel to demonstrate proficiency in the following:

1. Vehicle propulsion strength and speed
2. Combined Acoustic and Vision navigation agility
3. Identify and communicate light sequences from a target object
4. Autonomous identification and navigation of a docking scenario
5. Visually survey and deliver a package to a target of known dimension and location
6. All performance is independent of human involvement

II. Paper Contents

A. Design Strategy

The interactive nature of the challenges for this year's competition marks a clear step in the direction of true automation where the vessel is capable of performing different actions depending on how the environment is perceived by the sensors. In order to accomplish this, Team Owltonomous set out to make a mechanical system that robust, an electrical system that provides large amounts of system data, and a Guidance, Navigation, and Control (GNC) system that is as modular as possible. This design strategy also meshes well with the commitment of the team to research projects centered around the WAM-V outside of RobotX.

1) Vision Based Tasks

For the mainstay of the AUVSI competitions the teams vision system is a coupled nodding LiDAR and camera. The vessel will use the large field of view of the lidar to performing preliminary detection routines in the polar coordinate frame. If the object lies within the camera field of view, color detection is used as its position. This provides a hardware low-pass filter for object detection since the lidar will only return a value for objects of interest and not the surface of the water.

2) Acoustic Pinger-Based Transit

Coupling the vision, acoustic, and control system with a path planner and a prior knowledge is one of the challenges Team Owltonomous was most interested in because the fusion of multi-environment data is a real-world challenge for USV's. Since the gate buoys are laid out in a persistent manner, the team is able to box the data received from the acoustic system and vision system into 1 of 6 possible choices for the possibilities of entry and exit gates.

B. Vehicle Design

1) Requirement Analysis

The team developed a number of system requirements by analyzing the rules and task descriptions as they were updated in addition to the experience from previous competitions. The requirements include:

- An acoustic navigation system
- An RGB-D acquisition system
- A multi-computer network
- > 4 Hours of run time

- Tilting thruster
- Azimuthing thrusters
- Data transmission in a saturated network

2) Functional Decomposition

To provide a clear blue sky environment during design, the team laid out a functional decomposition with abstraction held paramount. The idea was to break down each challenge into the discrete steps that the vehicle would need to take in order to accomplish each task. By defining exactly what the vessel was required to do enabled the designers to manipulate the vehicle software to be utilized in various research topics.

3) Trade Studies

Once research and initial designs were completed, the team completed a series of trade studies based on the Kepner-Tregoe method. The major factors for weighting results were cost, reliability, and ease of implementation. Values were primarily assigned through conversation and group consensus.

4) Vessel Platform

Since the inaugural RobotX Challenge in 2014, the WAMV USV16 Platform was provided by ONR and Marine Advanced Research (MAR) to all competition teams as the primary development platform to support the competition tasks. The USV16 is a catamaran style vessel which that uses differential thrust as the propulsion system to increase the degrees of freedom. Each of the catamaran pontoons are equipped with a suspension system allowing it to keep the payload tray level in all sea states. This stability and dampened motion from wave interaction serves to reduce noise observed by any instrumentation on the vehicle. The vehicle's suspension system and simple configuration makes the WAM-V platform ideal for sensor integration and maneuvering around a range of obstacles. The current vehicle can be seen below in Figure 1.

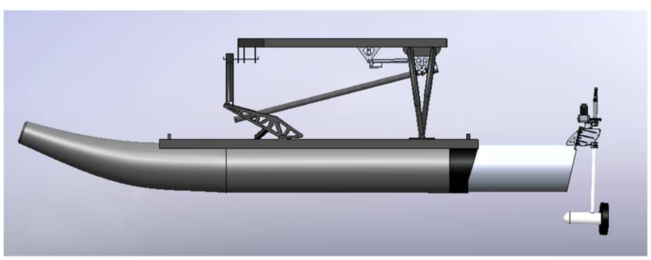

Figure 1: Side View of the Team WORX WAMV USV16 Platform

5) Actuated Thruster Mounting System

The thrusters chosen are re-purposed trawling motors with anodized and custom-made mounts. The two 36V Minn Kota thrusters have a capability of up to ±90 degrees. The thruster's azimuth range was increased from ±45 degrees by integrating servos into the new design. The custom made mounts give the ability to raise and lower the thrusters at an angle, which reduces the potential for damaging the thrusters during the deployment and retrieval process.

The actuated system employs two GearWurx high-torque servos paired with external gears which are mounted directly to the thruster shaft.

6) Acoustics Boom

The need of a low-cost solution for the acoustics subsystem was fulfilled by utilizing components already present in the lab space: transducers, carbon fiber pole and a linear actuator. The system is mounted below the payload tray to not interfere with the LARS system towards the vehicle stern. Figure 2 shows the carbon fiber casing and linear actuator mounted on the vehicle. Figure 3 provides a close up of custom-made transducers utilized in the acoustics system.

When retracted, the system allows the WAMV to clear a buoy. When deployed, the transducers are located two feet below the water surface. All components for this subsystem were machined by hand by team members in the FAU machine shop. The triangular plate was fabricated using the water jet. The spacers which mount to the carbon fiber boom are 3D printed from ABS and employ heat inserts for structural integrity.

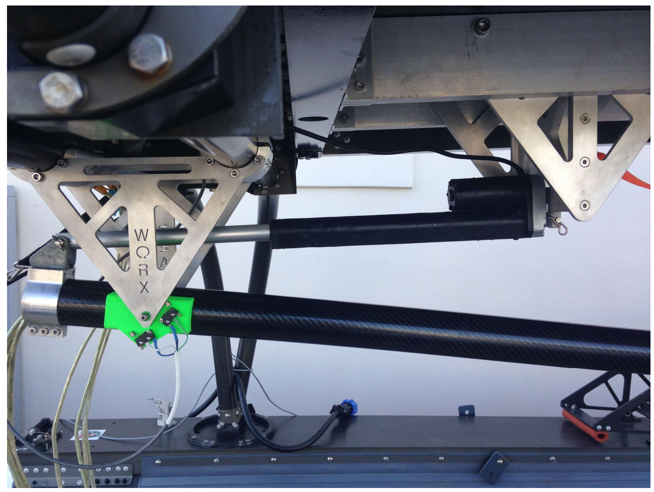

Figure 2: Acoustic Boom and actuator configuration shown mounted beneath the WAM-V payload tray

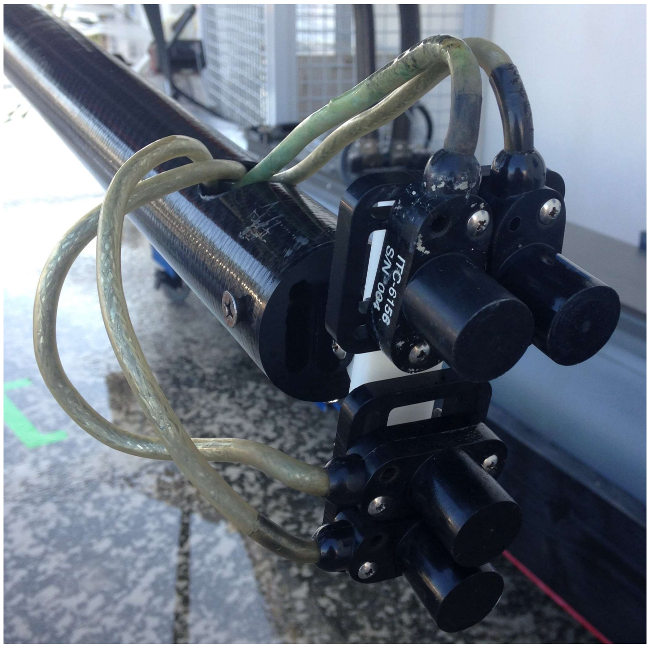

Figure 3: Physical Hydrophone Configuration for Acoustic Data Acquisition System

7) *Acoustic Data Acquisition System*

The design for the acoustic positioning system was based on the system from the RobotX 2014 Challenge, which used an Ultra-short baseline algorithm. The design uses a STM32F4 processor to perform both the Data acquisition and Digital Signal Processing (DSP). The components that have been kept from the 2014 challenge were the hydrophones. The hydrophones consist of two pairs of pie-electric elements with each tuned to a certain frequency which will be known before the mission run. Communication between the acoustic system and the high-level computer utilizes a TTL serial connection since the STM32F4 used does not have an Ethernet MAC.

The current design went through two hardware iterations. The first design required the signal to constantly be sampled in order to search for the beginning of the signal. This proved to be a major limitation because the sampling could be interrupted at any point by any of the service routines such as the communication which would likely result in missing the beginning of the signal. Revision two of the design incorporated a hardware trigger circuit so that a threshold could be set and if a signal was detected a high priority interrupt service routine would be called which would trigger the ADC sampling which would then write directly into the DMA without the processor needing to be involved. This means that the signal acquisition can no longer be interrupted by another service routine. Another improvement from the first iteration is the addition of a hardware hold off which allows the processor to disable the input signal until the system is ready for a new waveform. Doing this in hardware solved some complications with the interrupt service routines used for the hold off in the first revision. The last improvement from revision one was a change in the front-end which fixed some voltage offsets in the amplifiers caused by the amplifiers having a slightly different analog reference; these have now been combined so that the reference is identical on all the amplifier stages.

The algorithm used for the USBL is to perform a Fourier transform on the input data using the DSP library. Once the data is captured and the FFT is performed, the maximum amplitude frequency is compared to the desired frequency and if it matches then it proceeds to the next stage. Since the bearing is calculated from the phase angle between the signals on both hydrophones in each pair, the angle can be calculated by converting the complex FFT to phase and magnitude and subtracting the phase information between each pair of hydrophones. Once this phase angle is gathered, the resultant bearing can be estimated using the equation: bearing = arcsin(phase/pi).

This does, however, leave one issue in that the angle can be ambiguous since the source could be located on either side of the hydrophones and still create the same phase difference. In the future this can be eliminated by placing one pair perpendicular to the other so that two pairs of hydrophones can give each other information on where the source is located. This concept would require remote operation [3].

8) *Electrical*

The inspiration for the electronics box was based off previous layouts used for competition. There have been some changes regarding on board computers, but all main components necessary to complete specific missions remain. The main control box was rewired and given a new computer to replace the high-level Jetson Xavier that was previously used in other years. Mission critical components in the electronics box include the Jetson Xavier, the Jetson Orin, an STM32 microcontroller, power distribution circuits, telemetry systems, and the motor control interface. The main control box wiring can be seen in Figure 5. All major electrical components can be seen connected to the control box since this electrical circuit links all systems together. Emphasis was placed on making sure each component individually and concurrently without overloading the power distribution systems. Four emergency stop buttons were implemented into the control box. If any of the five emergency stop switches were to be pushed. The wiring of the emergency stop buttons connected to each of the thruster systems can be found in Figure 5 the system would not run implementing the requirement states in RobotX 2022 handbook[6].

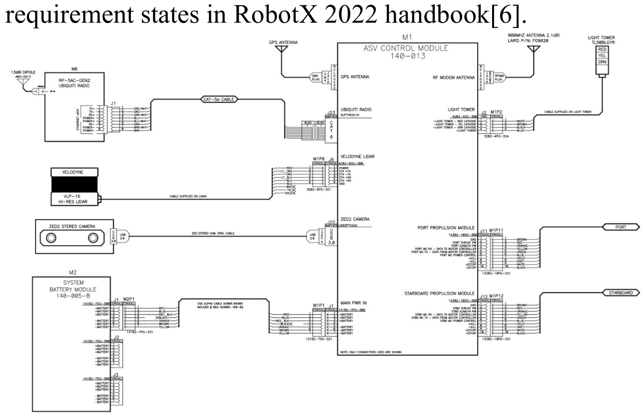

Figure 4: ASV Control Module Wiring Diagram

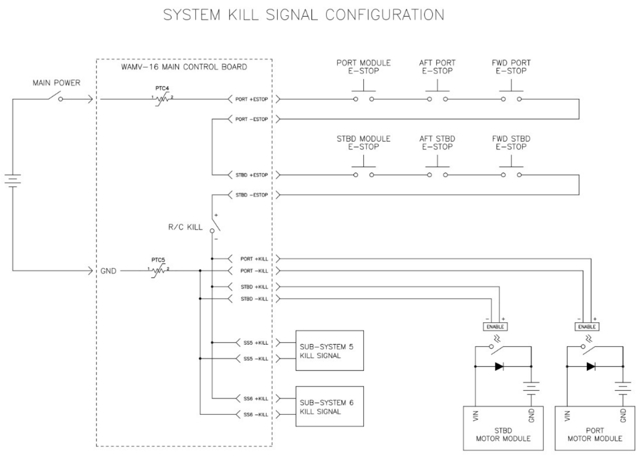

Figure 5: Emergency Kill Configuration

a) Jetson Xavier and Jetson Orin

The Jetson Xavier and Jetson Orin compute boards were chosen to act as the CPUs for the entire vehicle system. The design ensured that these boards were able to communicate with the other systems while operating missions in any type of condition. In order to sync this communication with accurate timing, a Real Time Clock (RTC) was chosen to be battery operated so that during communication failure, the Jetsons would still have the correct time.

b) STM32: The STM32 Microcontroller acts as the Bottom Level Manager (BLM), where the low-level real-time devices are managed. This board has a primary function of measuring capacity in the battery supplies and indicating their state to the system. This was important to ensure the power distribution system runs properly.

c) Motor Control: The motor control (PWM) interface was designed to provide hardware control over the motors from shore with a standard RC remote. A multiplexer control line originates from a PIC microcontroller with a singular job of detecting which way a switch is set on the RC remote, exceeding the level of control a user could have got from software solutions.

d) Light Indicator Station:

For the RobotX 2022 competition, Team Owltonomous has added a light indicator. This light indicator is wired to the main control box and shows a yellow light when the vehicle is being manually operated. The green light indicates when the vehicle is in autonomous mode. When one of the emergency kill switches is activated the red light will show indicating that the propulsion system is disabled.

9) Software

The software for the WAM-V is split between low-level and high-level systems.

a) Low-Level: The low-level subsystems constitute the set of nodes to be executed in order to obtain basic functionality of the USV. The primary components of this suite are positional sensors and a low-level control system.

Navigation and inertial sensor drivers allow data acquisition from the XSENS IMU/GPS and OS-5000 digital marine compass. The XSENS unit provides filtered data regarding the current state of the USV in terms of its position, orientation, corresponding speeds, and accelerations. The pose information is available in both the North-East-Down (NED) and East-North-Up (ENU) frames inertial and body-fixed frames as appropriate. Both conventions are readily available and permanently defined to allow for a smooth transition between the Virtual Marine RobotX Challenge (VMRC) simulation environment (built in ENU) and on water operation (built in NED). The transformations shown in equation 1 are applied via ROS's TF pipeline.

A rich set of previously developed station keeping controllers remain available for the current distribution. An Enclosure Based Steering [11] approach to mitigating cross track error via a coupled heading and velocity controller has been implemented as per last competition. The complete solution shown in Fossen's Handbook for Marine Controls is left out for brevity, the equations which require solving are show in (1)-(3). Results of these two are shown in sections results.

$$\chi_d(t) = atan2(y_{los} - y(t), x_{los} - x(t)) \qquad (1)$$

$$[x_{los} - x(t)]^2 + [y_{los} - y(t)]^2 = R^2 \qquad (2)$$

$$tan(\alpha_k) = \frac{y_k + 1 - y_k}{x_{k+1} - x_k} = \frac{y_{los} - y_k}{x_{los} - x_k} \qquad (3)$$

b) Secondary low-level systems provide information about the vehicles operating environment. The optics suite includes the Velodyne VLP16 3D LiDAR driver, a StereoLabs ZED 2, and a projection of the LiDAR voxels onto the webcam's field of view. Laser-based obstacle recognition takes advantage of the fact that the LiDAR only returns information from obstacles floating on the water surface, but not from the water surface itself. However, for certain tasks color information is required. To leverage the sparsity and robustness of the LiDAR, the entire point cloud is first segmented by use of a Nearest-Neighbors unsupervised algorithm, called Agglomerative Hierarchical Clustering (AHC) [13]. The benefit of AHC over supervised approaches, such as K-means or KNearest Neighbors, is that the number of detected objects is not and a priori requirement. While the computational cost of AHC can be larger than K-means or KNN, $O(n)^3 * O(n)^2$, this performance hit is not truly felt on the water, as the available information is far less sparse when compared with on land operation.

This performs a pre-classification routine based on clustering points into single identities associated to the corresponding obstacles of the challenge. This information is also used to train a convolutional neural network for a definitive classification inspired by models such as [8].

Mapping functionality is provided by implementing the move_base package from the ROS Navigation Stack, which computes local and global cost maps around instances of obstacles captured by the LiDAR in order to avoid collisions while minimizing the traversed distance. Global cost maps can be computed over static maps of the environment generated by methods such as SLAM. Local cost maps, on the other hand, build a map that travels locally around the

USV, considering only the most updated data from the perception system.

c) High-Level: Path-planning is also implemented using the move_base package from ROS Navigation Stack. It is given a start and goal pose for the vehicle to achieve and generates the corresponding global and local trajectories. Global trajectories are computed using the global cost map as a series of discrete points that defines it. Local trajectories unfold from using local maps to compute linear and angular velocity commands for vehicle motion, preventing collisions while minimizing deviations from the original global trajectory.

From a procedural perspective, the high-level mission planner, defined in the system as the_planner, is a sequence of switch cases, each of which flags and executes a particular task from the RobotX challenge if associated conditions are met. Every task is defined as a derived class that inherits from a general base class. This provides a uniform and modular interface for the user and allows all missions to share common communications and persistent data.

d) Localization, Mapping and Motion Planning: At the base of the High-Level Mission Planner resides fundamental navigation requirements for the USV, namely localization and mapping, which provides the system with data regarding vehicle pose as well as knowledge of the surrounding obstacles. This information is a crucial aspect for the autonomy of the USV, since it supplies all the required data in order to compute optimal trajectories across safe navigation area. Potential obstacles could be any RobotX challenge task, natural objects, (e.g., shoreline, mangroves, rocks, etc.) or other man-made obstacles (such as boats or docks in the vicinity).

Over the design stage, our approach was first tested using the Virtual Maritime RobotX Challenge (VMRC) environment. This simulation provides all the physical models for the WAM-V USV (geometry, mass, and inertia) and environmental conditions using the Gazebo physics engine. Gazebo plugins were also provided to simulate the GPS and IMU sensors, as well as the Velodyne VLP16 3D LiDAR, which is the actual 3D Lidar used in the USV configuration for the competition. The implementation of this virtual environment allowed us to quickly assess the localization and mapping approach explained above, to determine its viability for its further implementation in the actual USV. Fig. 6 illustrates this approach, in which the mapping functionality is tested based on the virtual model from Gazebo and shown at the same instance of time.

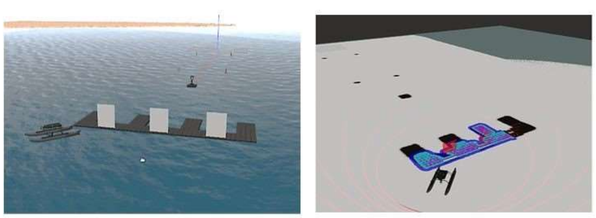

Figure 6: Left: Gazebo representation of RobotX challenge using VMRC. Right: LiDAR-based mapping, visualized using RViz

Localization was approached by fusing data from the GPS and IMU sensors into an extended Kalman filter (EKF), implemented using the robot localization package from ROS. This also incorporates a general non-holonomic motion model of the vehicle in order to estimate a true belief of its current state. The state of the vehicle is represented as a 15-dimensional vector:

$$\{X, Y, Z, \phi, \theta, \psi, \dot{X}, \dot{Y}, \dot{Z}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \ddot{X}, \ddot{Y}, \ddot{Z}, \ddot{\phi}, \ddot{\theta}, \ddot{\psi}\}$$

where the first six components correspond to the linear position and angular motion (roll, pitch, yaw), respectively. The rest appertain to their first and second derivatives (only first derivative for the angular motion). A slightly different approach was considered for the actual implementation of the localization system, which did not require using the robot localization package. Instead, the filtered output from a XSENS device was used. At the core, the output of the XSENS unit is also based on fusing GPS and IMU data using an EKF; however, the unit entails a highly calibrated system which provides very accurate localization data.

Mapping of the environment is achieved by implementing the ROS Navigation Stack, specifically the move_base package. The_package provides the functionality of constructing local and global cost maps around the obstacles, as well as to choose among various types of local and global motion planners. Cost maps consider the geometries of both obstacles and the vehicle to compute clear and efficient trajectories. Global cost maps may use prior knowledge of the world, provided from previous maps generated by any kind of technique (SLAM, map rasterization [9] and map-to-image coordinates transformations). Correspondingly, global motion planners use data from the global cost map to compute a global trajectory from the current position of the vehicle to a certain goal location [5], as shown in Fig. 5. Local cost maps (shown as the colorful portion of the large obstacle in Fig. 5), are built around and travel with the vehicle, using only the most updated sensor data available. This in turn allows the vehicle to avoid obstacles as their positions update in time.

In order to get the most out of the ranging and mapping capabilities from the Velodyne VLP16 3D LiDAR, a singular configuration of the move base_node was implemented. Point cloud data was acquired from the sensor driver in order to be used for marking instances of new obstacles and performing an artificial conversion of the point cloud data into laser scan information using the pointcloud_to_laserscan ROS package. The computation of navigation trajectories not only rely on the position of the obstacles, but also on the knowledge of free and unknown-state cells in the map. Three potential states of the map cells can be inspected from Fig. 7, with cells depicted in black, white, and grey, for occupied, free and unknown states, respectively.

Global and local trajectories are defined parametrically different. While global trajectories (from global planners) are

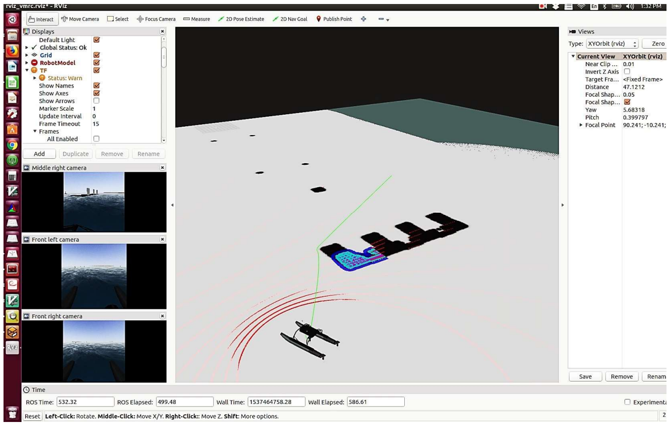

Figure 7: Global Path-Planning

defined as a sequence of points comprising the desired (global) path, the local motion planner computes linear and angular velocity commands as inputs for the actuators of the vehicle, from which the actual local trajectory of the vehicle unfolds [4]. The desired trajectory generated by the local planner is generated according to a selected holonomic or non-holonomic motion model of the vehicle, as defined in the move base package, which in general are associated with motion models of ground robots.

However, the dynamics and the medium associated with the operation of a USV greatly differ, so some adaptations are necessary. Provided that the low-level controllers implemented in the USV effectively control the vehicle state (defined in terms of position, speed, and orientation), one possible adaptation is to recover the local trajectory as a sequence of points in a horizontal plane. This can be done by using the velocity commands computed by the local planner as input parameters for a non-holonomic kinematic motion model, from which it predicts the most suitable local trajectory. It also accounts for the original global trajectory and goal location, all of which are weighted inside an optimization function.

Simultaneous Localization and Mapping (SLAM) was also explored for some time. The hector_slam package from ROS was implemented because it did not require odometry data [10], since this can be a problem in marine vehicles. It relies entirely on the information provided by the LiDAR at a high update rate. This approach was tested on ground and marine operations, successfully providing information regarding the pose of the vehicle while simultaneously mapping the surroundings. The success was dependent upon the number of features in the environment, preferably of a structured nature, such as buildings, walls, or large ships. However, given the scarcity of these features in regular conditions in the marine environment, this approach was swiftly discarded. A move towards the more traditional way of solving the localization problem was chosen: mapping the environment based on the localization information provided by the fusion of GPS and IMU data [6].

10) Vision-Based Perception System

In addition to typical camera information such as color, morphology, and the angular location of objects within the camera's Field of View (FOV), a LIDAR/Video vision system provides depth information. Depth is extremely important for ASVs, as it enables the use of advanced vision algorithms like simultaneous localization and mapping (SLAM) and object-based color analysis [7].

The level of autonomy required for the 2022 RobotX Challenge Tasks is achieved by using the LIDAR/Video vision system mounted at the front of the payload tray near the control box shown in Figure 8. In this configuration, the LIDAR scanner (Velodyne VLP16 3D LiDAR) scans on a plane and is used to obtain distance information while the video camera (StereoLabs ZED 2) is used to obtain color and morphological information about the objects in the vision system's FOV.

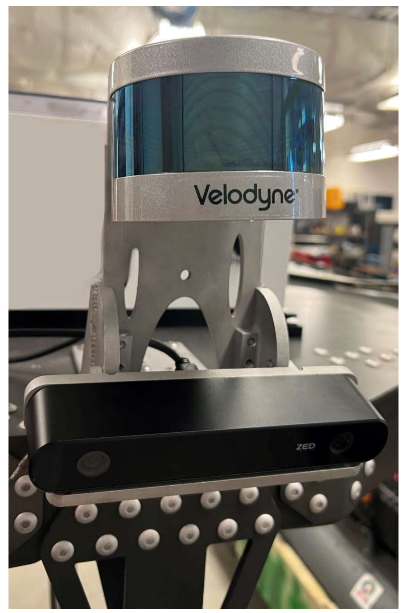

Figure 8: Vision System Package utilizing the ZED camera and Velodyne LiDAR

The vehicle's stand-alone vision system is designed with the capability to control the gimbal, measure data, and perform image-processing. The primary computational resource for this task is a laptop capable of running Ubuntu 20.04, a Linux operating system. The laptop is programmed using ROS and OpenCV. Both the Velodyne LIDAR device and the ZED 2 can connect directly to the laptop.

Objects in the LIDAR image can become distorted as the vehicle turns. An example of this can be seen in Figure 9b, where the target buoy appears tilted. This occurs because points in the depth image are not collected at the same instant as when the LIDAR scans the environment. To overcome this issue, one of the innovations implemented in the presented LIDAR system is the use of Lissajous-like scan patterns to allow a trade-off between speed and resolution without constraining the FOV [1].

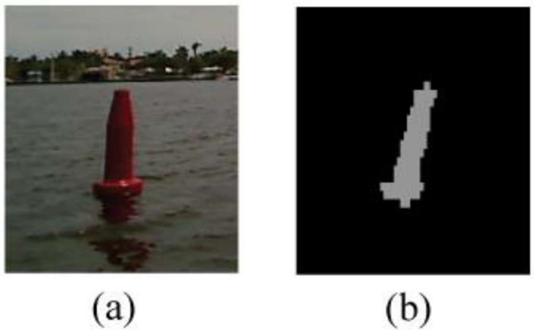

Figure 9: Example (a) camera and (b) depth images showing issues associated with motion

The LIDAR and ZED produce two outputs that must be fused: 1) Depth information, with corresponding gimbal and LIDAR angles, enabling the production of a depth image (Figure 10a), 2) An RGB image (Figure 10b). Figure 10c shows detail of the approximate RGB image region from (a).

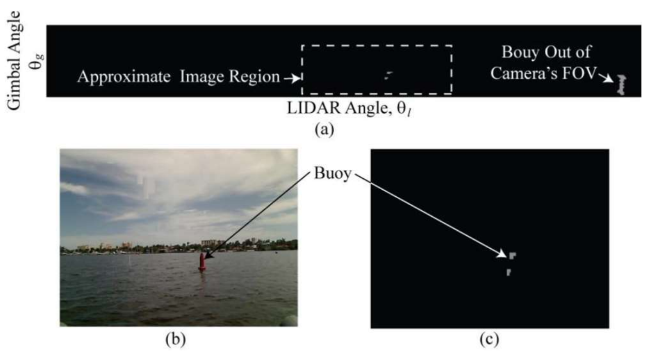

Figure 10: (a) Depth image obtained from the LiDAR system. (b) RGB image obtained from the camera. (c) Detail image of the approximate image region in (a)

For buoy identification, the LIDAR/ZED fusion algorithm uses the depth image to identify objects of interest (not just buoys, but anything in the FOV of the LIDAR). It is advantageous to carry out the object identification in the depth image because floating objects are automatically isolated, both from the water (as the LIDAR is unable to return the distance to the water surface) and from the background (since LIDAR range is limited). As seen in Figure 8a, this typically results in a small subset of Points of Interest (POI) in the depth image [3].

To fuse the LIDAR and ZED images, the location of the POI needs to be transformed into RGB camera pixel coordinates. This is done by converting the LIDAR points to Cartesian XYZ coordinates, translating the origin to that of the ZED camera, and finally calculating the pixel values through use of an intrinsic model. This typically results in a sparse mapping of depth points to the pixel frame, as the camera has significantly higher resolution.

To better correlate the object being detected in the depth image with its corresponding object in the RGB image, the depth information is set for a desired range and the remaining points are joined to make a continuous area using morphological operations (Figure 11c). Mathematical details can be found in [2,14].

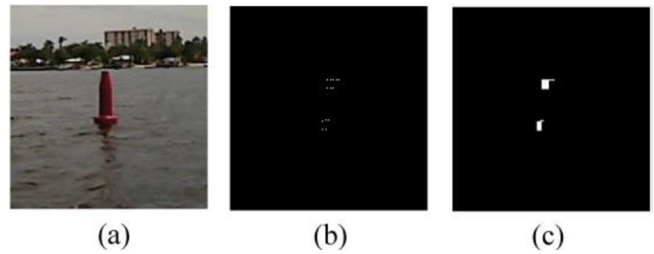

Figure 11: Example images showing (a) the original RGB image, (b) LIDAR data plotted in the camera coordinates, and (c) joined LIDAR data, creating a region of interest in the camera image

Heavily coupled to the vision system is the vehicle's Simultaneous Localization and Mapping (SLAM) capability. SLAM works to combine sensor measurements and control inputs to form an estimate about the robot's pose as well as features within a robotic environment. For the purpose of the boat, these features will be considered as buoys in the water. The particular variant of SLAM used in this implementation was FastSLAM [12].

Without an internal map of its surroundings, the vehicle response can only react to sensor measurements (e.g., video, LIDAR) at a current time. To provide a higher level of functionality to the system, the vehicle must be able to leverage past sensor readings with current sensor data and form some estimate of important features in its environment. With an internal map of the environment, the vehicle can perform tasks such as path planning with obstacle avoidance, recursive color estimation for objects, and will have more accurate vehicle pose estimates.

C. Experimental Results

FAU's SeaTech campus allows for multiple days of testing since it is located on a marina. Pushing towards water-testing has been a driving force for the team based on the experience of systems working beautifully when in the confines of a lab then mysterious errors occurring as soon as it is placed on the water. The processing for launching the WAM-V is bringing the vessel to a davit less than 100 yards from the lab is it stored in.

Team Owltonomous has been testing the vehicle on the water since the last competition in Hawaii. Much of this time was spent fine tuning control algorithms and verifying the finite state machine, with significant portions also comprised of USBL testing and high-level mission logic.

D. Conclusion

In preparation for the 4th biennial RobotX challenge posed many different challenges. Owltonomous is a ragtag group of individuals that put their effort into making the competition possible for FAU. There were changes with the electrical system to make the electronics more efficient. Our team learned how to use ROS and OpenCV in order to fully understand the vehicle. Overall, the team feels confident that RobotX 2022 is possible and will the biggest learning experience for each of us.

E. Acknowledgements

Team Owltonomous is increasingly grateful for the technical expertise and support from the Florida Atlantic and associated advisors: Ed Henderson, John Frankenfield, Don Harper, Armando Sinisterra, Travis Moscicki, and Edoardo Sarda. Each one provided invaluable guidance in both electrical, mechanical, and software design. As always, thank you.

F. References

[1] Anderson J. W. and Clayton G., “Lissajous-like scan pattern for a gimbaled LIDAR.” 2014 Conference on Advanced Intelligent Mechatronics (AIM).

[2] Anderson Lebbad, Nicholas DiLeo, Chidananda Matada Shivananda, Frank Ferrese and C. Nataraj, "Probabilistic Vision Based Guidance & SLAM for Autonomous Surface Vehicles", Naval Engineers Journal, December 2014.

[3] Bianca Mesa, Travis Moscicki, et. al, “Team worx: Journal paper for the 2016 robotx competition,” Boca Raton, FL, 2016.

[4] D. Fox, W. Burgard, and S. Thrun, “The dynamic window approach to collision avoidance,” *IEEE Robotics Automation Magazine*, vol. 4, no. 1, pp. 23–33, March 1997.

[5] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige, “The office marathon: Robust navigation in an indoor office environment,” in *2010 IEEE International Conference on Robotics and Automation*, May 2010, pp. 300–307.

[6] Eric Nieves, Armando Sinsiterra, et. Al, “Team Owltonomous: RobotX Journal Paper 2018,” Boca Raton, FL, 2018.

[7] Haung S., Wang Z., and Dissanayake G., "Simultaneous Localization and Mapping: Exactly Sparse Information Filters," World Scientific Publishing Co., Danvers, MA, 1994.

[8] M. Engelcke, D. Rao, D. Z. Wang, C. H. Tong, and I. Posner, “Vote3deep: Fast object detection in 3d point clouds using efficient convolutional neural networks,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 1355–1361.

[9] N. Otsu, “A threshold selection method from gray-level histograms,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, pp. 305 – 324, 1979.

[10] S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf, “A flexible and scalable slam system with full 3d motion estimation,” in *2011 IEEE International Symposium on Safety, Security, and Rescue Robotics*, Nov 2011, pp. 155–160.

[11] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley Sons, Ltd, 2011.

[12] Thrun S., Burgard W., and Fox D., Probabilistic Robotics, Cambridge, The MIT Press, 2006

[13] W. H. E. Day and H. Edelsbrunner, “Efficient algorithms for agglomerative hierarchical clustering methods,” *Journal of Classification*, vol. 1, no. 1, pp. 7–24, Dec 1984. [Online]. Available: https://doi.org/10.1007/BF01890115

[14] Yoshida T., Irie K., Koyanagi E., and Tomono M., "3d laser scanner with gazing ability,” 2011 IEEE International Conference on Robotics and Automation (ICRA).