Controlling a Remotely Operated Vehicle (ROV)

Brennan P. Miller-Klugman (BSCO), Hank M. Pham (BSCO), Ethan S. Rowe (BSEN)

Advisor: Eric Gallimore, Douglas E. Dow, Pilin Junsangsri

ENGR 5000 Senior Design II

School of Engineering

Wentworth Institute of Technology

Boston MA

August 17th, 2022

Introduction

The ocean contains a multitude of features of value that can be explored using underwater vehicles. Many of these features, including coral reefs and shipwrecks, are delicate and should not be disturbed. Various classes of vehicles exist that offer different capabilities regarding underwater exploration, such as autonomous underwater vehicles (AUVs), remotely-operated vehicles (ROVs) which require a human operator, and hybrid ROVs (H-ROVs) that combine features of both ROVs and AUVs. These different classes of vehicles vary in capabilities and have advantages or disadvantages for tasks. For example, ROVs can be used for a myriad of tasks including inspection of harbor structures [1].

For AUVs, autonomous underwater navigation is difficult since GPS and other radio frequencies (RF) do not travel well in water [2]. Being able to determine a location underwater both consistently and accurately is a difficult task. Some methods for localization do exist like using markers to estimate a relative position [5]. However, while this approach might work well in an environment with known associations between landmarks and positional data, this approach struggles in unexplored seascapes where there are no existing landmarks to draw positional data from. Due to these issues with localization some tasks that would be difficult with AUVs are better suited for ROVs, such as inspections, maintenance, and repair [6].

A potential way to recognize objects around the underwater vehicles is using simultaneous localization and mapping (SLAM). SLAM offers features like more robust path planning and obstacle detection. However, due to conditions including floating particles and constantly changing lighting conditions, visual SLAM underwater is difficult. By adding image pre-processing, some of these issues regarding the quality of underwater imaging can be avoided [3].

SLAM also has other disadvantages like the fact that increasing the map size of SLAM requires large amounts of calculations and thus computational power [5]. Given that AUVs operate on batteries adding more computational power means draining the batteries quicker and reducing total operating time.

Given these difficulties in underwater visual SLAM, sensor fusion can be used to help improve the image of the environment surrounding a vehicle. Additional sensors such as sonars can be used in conjunction with cameras to help create a more complete image. Object detection using sonar is generally noisy and cluttered. However, by utilizing a probabilistic framework noise can be significantly reduced while filtering can help emphasize regions of interest [4].

ROVs have the advantage of being operated by a human who can process the sensor and camera data to make the quick adjustments needed in high-risk environments. Operating a vehicle manually is difficult and at times operators might have to manage multiple camera and sensor feeds. Some ROVs simplify some of the repetitive tasks needed to be completed by operators like automatically keeping the vehicle in place relative to the operating site of the ROV to help operating complexities [5].

The H-ROV presented in [7] combines the advantages of both AUVs and ROVs by offering features like autonomous navigation to a predefined destination in the event of communication loss between the operator and the vehicle. In a high-risk environment where the tether connecting the vehicle to the surface is at risk, this solution allows the finite control of an ROV while retaining autonomous operations in case of emergencies.

By using sensors, such as sonar, depth and temperature sensors, a camera, and an Inertial Measurement Unit (IMU) in conjunction with a human operator many of the benefits of AUVs can be retained while also enabling the finite control only possible through a human-controlled vehicle. An IMU makes it possible to implement autonomous features such as keeping the vehicle in position relative to the operating site using feedback loops, while a sonar can provide the ability to recognize and preplan for oncoming obstacles.

The purpose of this project was to augment the usability and control of an existing ROV platform by adding new sensors and feedback loops. The development of such a system would reduce the risk of damaging marine ecosystems or artifacts during ROV exploration.

Methodology

Hardware

The proposed system utilizes the NUGV ROV chassis, shown in Figure 1, which was designed and assembled by previous members of Wentworth's IEEE chapter. The vehicle is 20in by 16.5in by 14.5in.

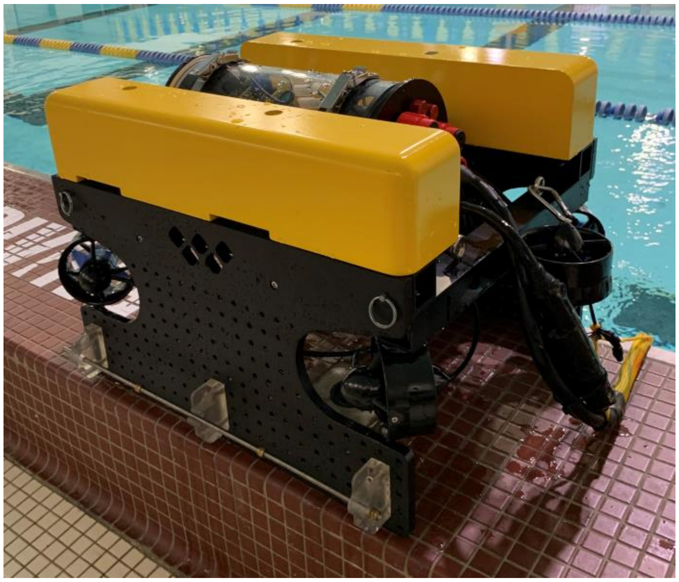

Figure 1. NUGV ROV Platform

Internal to the NUGV, a Raspberry Pi (RPi) is used to communicate with an Adafruit PCA9685, a 16-channel 12-bit PWM/Servo Driver, using I^2C to create pulse width modulation (PWM) signals. These PWM signals are then communicated to a Blue Robotics Basic Electronic Speed Control (ESC) which directly interfaces with the Blue Robotics T100 thrusters used to propel the vehicle. Additionally, an Adafruit Precision NXP IMU is used to collect accelerometer, magnetometer, and gyroscope data. A general diagram of this architecture can be seen in Figure 2.

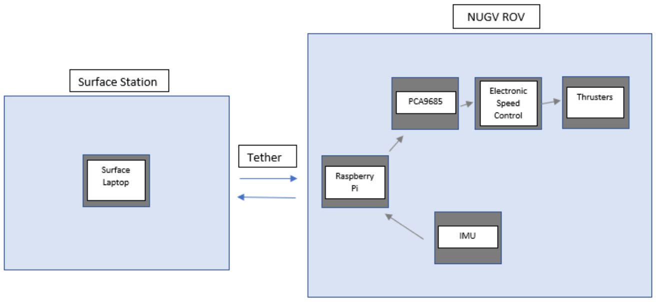

Figure 2. NUGV Core Vehicle Overview

External to the NUGV, the surface control equipment for the vehicle, including the surface laptop and DC power supply, is housed in a pelican™ brand case as seen in Figure 2.

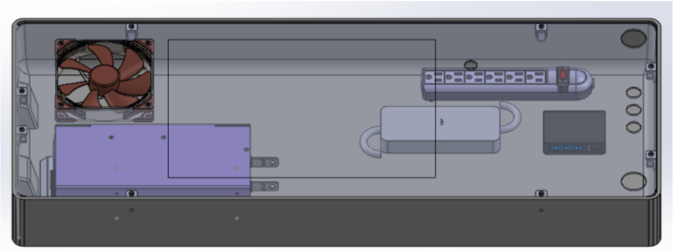

Figure 3. Surface Control Station

Additionally, the surface control station has room for monitors to be mounted which will display camera feeds and other pertinent information.

The NUGV utilizes a custom tether to facilitate a connection between the vehicle and the surface. This tether utilizes waterproof connectors that transfer power and data to the vehicle. Data transfer to and from the RPi is handled through CAT-5 ethernet, which is run alongside DC power wires.

Software Overview

The existing codebase utilizing the Robot Operating System (ROS) Melodic was previously developed to control the NUGV. This architecture was outdated, and many of the utilized packages were deprecated requiring the codebase to be refactored. To achieve full system functionality, the entire software package was updated to the more recent ROS Noetic running on Ubuntu 20.04.

The pre-existing software package was examined node by node, and faulty code was either repaired or outright replaced. Some nodes, like the Adafruit PCA9685 and joystick control nodes, suffered from outdated packages and needed to be replaced while other nodes such as the IMU sensor library had to be written from scratch. Additionally, these nodes were updated to be compatible with ROS Noetic, and all nodes critical to vehicle operation have received refactors that reflect an object-oriented programming style. Utilizing pre-existing ROS packages, a Madgwick filter was used to ensure the accuracy of IMU data. Additional changes were made to future proof the code including updating messages to match with the ROS standards such as the REP 103 standard for sensor measurement and coordinates.

Given the importance of responsive vehicle operation, a considerable amount of time was put into overhauling the joystick control node. The node was modified to take in a configuration file as a runtime argument which specifies which type of controller should be used as an interface for the ROV. More controllers can be added via a JavaScript Object Notation (JSON) configuration file and have allowed the testing of multiple forms of controllers including a Logitech joystick as well as Xpad game controllers. The bindings created in the JSON file are customizable, and thus can be modified to fit the preferences of the vehicle operator.

External forces present in underwater operating environments such as currents make it difficult to ensure vehicle stability. In missions where vehicle stability is crucial, it is paramount to counteract the effects of these external forces. A proportional–integral–derivative (PID) controller was integrated to remedy this challenge.

The PID controller takes as inputs the setpoint value, commands received via user input to an Xbox remote, and the vehicle’s current acceleration and orientation values received from the IMU. A block diagram of a PID controller is shown below.

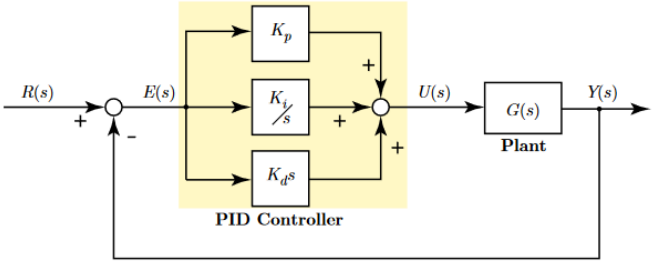

Figure 4. PID Controller Flow Chart [8]

By changing the constants Kp, Kr, and Kd, the output of the PID can be tuned to yield the best results.

Sensors

The addition of a camera, sonar, temperature, and depth sensor will allow more data to be provided to the operator. The aggregation of additional data will serve to help provide the operator with an enhanced view of the operating environment.

Adding a digital camera will provide visibility underwater, as well as the future ability to add computer vision algorithms to assist in autonomous navigation. To remedy the possible issues of operating the vehicle in a low-light environment, a PWM-controlled Blue Robotics light is utilized. Given that the camera needs to communicate via CAT-5 ethernet from the vehicle’s Raspberry Pi to the surface station router, and then finally to the surface station laptop, latency is a concern that needs to be addressed. By compressing the video feed, load can be taken away from the network at the cost of CPU usage on the Raspberry Pi.

Integrating a Bar30 depth and pressure sensor, created by Blue Robotics, onto the ROV will allow depth and water temperature to be monitored. This information can be displayed to the operator to ensure that the vehicle does not submerge beyond a desired depth.

Through retrofitting a Blue Robotics Ping360 sonar, shown in Figure 3, on the NUGV platform a system can be developed to detect obstacles and subsequently assist a human operator in navigation.

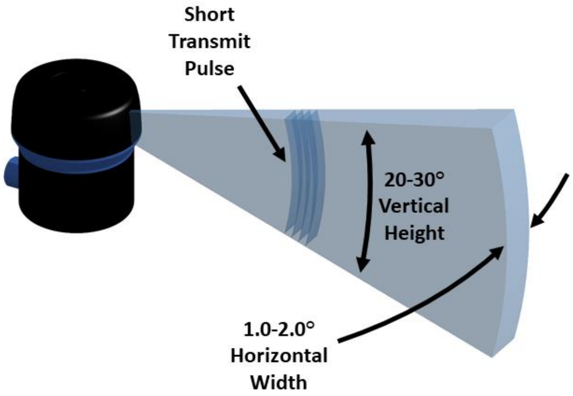

Figure 5. Blue Robotics Ping360 Sonar [9]

The Ping360 is a mechanical scanning sonar that can scan a radius of 50 meters at a depth rating of 300 meters [9]. The sonar sends out sound pulses that are reflected when hitting obstacles.

It is still important to retain a human operator as not all objects are detected by the sonar. As stated in [9], some objects such as rock, concrete, and metal will be “very reflective and have strong echoes” whereas other materials including mud, silt, sand, and plants “will have weaker echoes as they either have a similar density to water or absorb acoustic energy”. Additionally, the sonar has a minimum range of 1.5 meters, which means that obstacles that are too close to the vehicle will not be detected. A human operator will allow objects that are incorrectly detected or not detected at all to be considered.

Testing and Results

Camera Latency Testing

Low camera latency is critical in providing the vehicle operator with an up-to-date view of the operating environment. If the latency camera is too high, then the operator is reacting to past events which could easily lead to a mission-ending mistake like crashing into an obstacle.

To accurately measure camera latency, a ROS node was constructed to measure the difference in the epoch time and received time of ROS compressed image messages. The time taken to compress the USB camera image was not included in this testing, as it is highly variable based on the algorithm chosen.

Table 1 includes the average measured latency along with the average size of the ROS message. Averages were collected at 500, 1000, 1500, and 2000 images received. The larger the total number of images received, the more accurate the average latency and image size.

Total Number of Images Received	Average Latency (Seconds)	Average Image Size (Bytes)
500	0.0398	56038.99
1000	0.0407	56922.08
1500	0.0415	56697.42
2000	0.0428	56546.08

Table 1. Camera Latency

An average latency of 0.04s for a 57546 Byte image demonstrates that the time it takes for a compressed image to transverse the tether is negligible.

PID Testing

Underwater remotely operated vehicles (ROVs) are subject to environmental conditions that can make operation difficult. When operating in environments where vehicle stability is important, such as maintenance and repair-related tasks, it can be mission critical to assure that the vehicle is as steady as possible. Additionally, since GPS signals do not propagate underwater, ensuring that the vehicle does not veer off course is crucial in tracking heading reliably.

To create the PID controller a preexisting PID library (https://github.com/ivmech/ivPID) was utilized to complete the necessary calculations. Additionally, a ROS tool called dynamic reconfigure (http://wiki.ros.org/dynamic_reconfigure) was used to enable dynamic adjustment of the Kp, Kr, and Kd values in real-time.

To test the PID controller, a simulation was created using artificial IMU and joystick values. Figure 6 depicts the output of the PID controller in red and the desired input in blue. In the simulation, a desired linear x-axis acceleration value of 0.5 is sent to the PID controller, and an IMU simulator slowly increases the current x-axis acceleration until it also reaches 0.5.

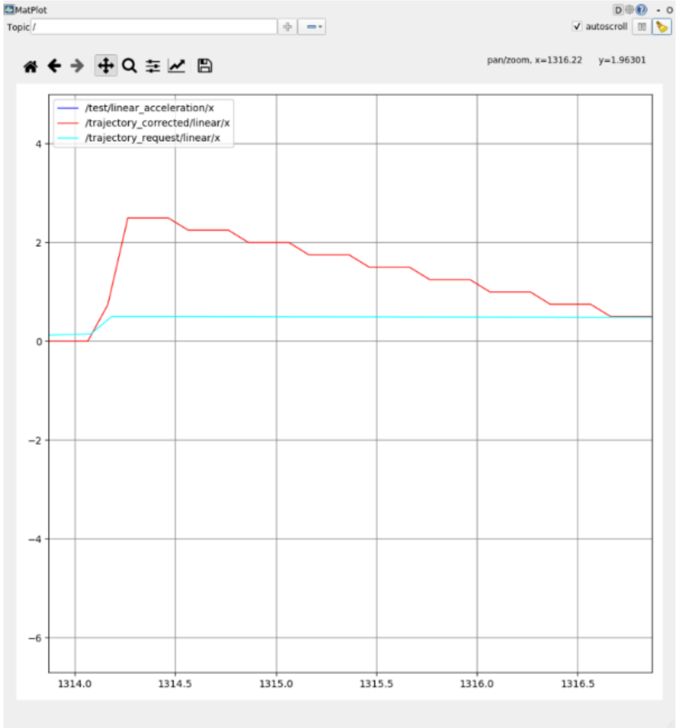

Figure 6. Simulated PID Output

After verifying the outputs of the PID controller were satisfactory, the code was uploaded to the vehicle and taken for testing in a pool. Tests in the pool consisted of pushing the vehicle along the linear x-axis and observing the output of the PID. As seen in Figure 7, the PID output, shown in red, demonstrates an overdamped oscillation that first overshoots the desired value, shown in blue, and then oscillates until it returns to 0.

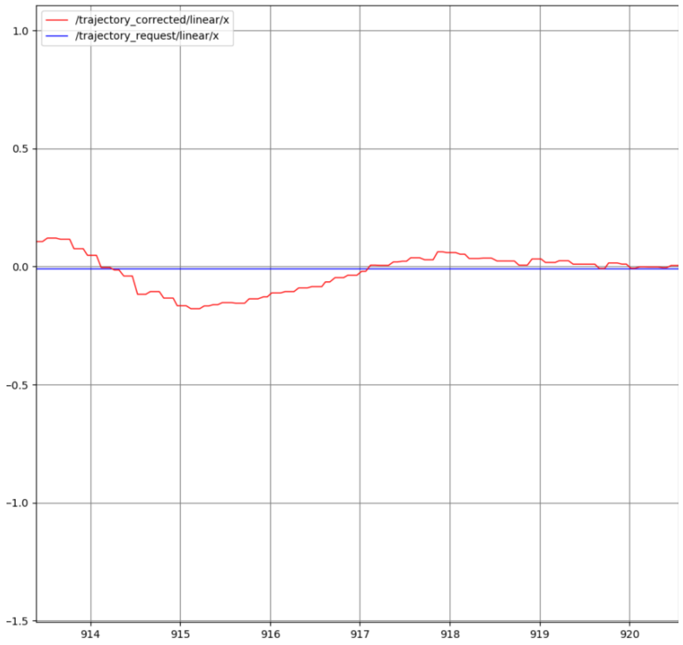

Figure 7. Experimental PID Output

The PID controller was successfully able to help keep the vehicle in position despite external forces, such as a human push.

Sonar Testing

The Ping360 plays a critical role in providing the driver with an accurate image of the operating environment surrounding the vehicle. Unlike a camera, the sonar can collect data despite obstacles related to visibility like muddy water and floating particulates. A range of tests was conducted with various sonar settings to verify the sonar range and its ability to detect obstacles in the environment.

As shown in Figure 8, the sonar was able to detect the human subject who was placed 5 meters from the sensor. The outline of the pool is also visible from the sonar image. However, there is a lot of noise being picked up.

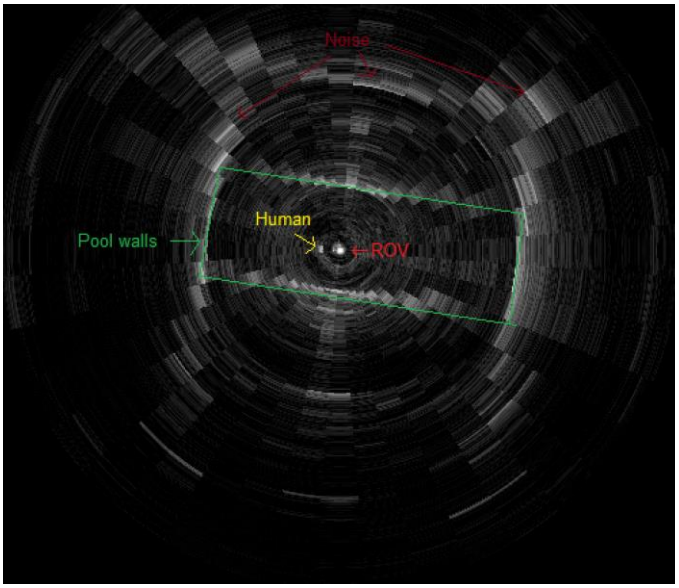

Figure 8. Sonar Test with Human Subject at 5 Meter Range

Discussion and Future Work

Given the rapidly changing environment that ROVs operate in, providing data to the vehicle pilot is crucial in ensuring safe vehicle operation. By augmenting an ROV platform with additional sensors including a sonar and a camera and by adding control loops, the experience of piloting an ROV can be greatly improved.

Although testing results showed that the propagation time of compressed images from the camera was negligible, a minor delay was still visible when piloting the vehicle. Implementing a faster compression algorithm could mitigate this effect and ensure that the vehicle operator is receiving up-to-date information.

The PID controller was successfully able to keep multiple axes steady at the same time and did improve the overall experience of piloting the vehicle. Tuning the PID loops more accurately would serve to further this effect.

The sonar provided a long-range image of the environment surrounding the vehicle. In conditions with poor lighting or floating particulates where utilizing the camera is not a viable solution the sonar can provide a valid method of navigation. Given that the sonar has a minimum range of 1.5 meters, a rather large blind spot surrounds the vehicle which makes it unrealistic to utilize the sonar in size-constrained environments like shipwrecks. In the future, using the sonar could be used in conjunction with the camera to implement autonomous navigation. Additionally, it could be beneficial to add a grid containing distance markers onto the sonar live feed.

Conclusions

Given the limitations of AUVs in terms of battery life and communications, some tasks are not suitable to be completed by this class of vehicles. ROVs offer a promising solution as they have the advantage of a human operator who can make quick on-the-spot decisions. By supplementing the capabilities of a human operator with automation, a system can be created wherein the operator has enough incoming information to make informed and effective decisions. By augmenting the usability and control of an ROV platform by adding sensors including a camera and a sonar, and by integrating feedback loops, an operator can gain a better perception of the environment surrounding the vehicle. This system aims to prevent damage to the vehicle as well as the surrounding environment making it more viable for underwater exploration.

References

[1] Jinwoo Choi *et al*, "Development of a ROV for visual inspection of harbor structures," in 2017, Available: https://ieeexplore.ieee.org/document/7890285. DOI: 10.1109/UT.2017.7890285.

[2] W. Zhao *et al*, "Review of SLAM techniques for autonomous underwater vehicles," in Sep 20, 2019, Available: http://dl.acm.org/citation.cfm?id=3366262. DOI: 10.1145/3366194.3366262.

[3] S. Rahman, A. Q. Li and I. Rekleitis, "SVIn2: An underwater SLAM system using sonar, visual, inertial, and depth sensor," in Nov 2019, Available: https://ieeexplore.ieee.org/document/8967703. DOI: 10.1109/IROS40897.2019.8967703.

[4] V. Ganesan, M. Chitre and E. Brekke, "Robust underwater obstacle detection and collision avoidance," *Auton Robot*, vol. 40, *(7)*, pp. 1165-1185, 2015. Available: https://link.springer.com/article/10.1007/s10514-015-9532-2. DOI: 10.1007/s10514-015-9532-2.

[5] P. Jasiobedzki *et al*, "Underwater 3D mapping and pose estimation for ROV operations," in Sep 2008, Available: https://ieeexplore.ieee.org/document/5152076. DOI: 10.1109/OCEANS.2008.5152076.

[6] A. Palomer, P. Ridao and D. Ribas, "Inspection of an underwater structure using point-cloud SLAM with an AUV and a laser scanner," *Journal of Field Robotics*, vol. 36, *(8)*, pp. 1333-1344, 2019. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21907. DOI: 10.1002/rob.21907.

[7] G. Meinecke, V. Ratmeyer and J. Renken, "HYBRID-ROV - development of a new underwater vehic brotherle for high-risk areas," in Sep 2011, Available: https://ieeexplore.ieee.org/document/6106913. DOI: 10.23919/OCEANS.2011.6106913.

[8] R. A. Paz, “The Design of the PID Controller,”

[9] Blue Robotics, "Understanding and Using Scanning Sonars," Available: https://bluerobotics.com/learn/understanding-and-using-scanning-sonars/.