University of Puerto Rico Mayagüez Campus: *Proteus* Technical Report

Yomar Ruiz, Karinne Vélez, Esteban L. López, Cristhian Tacoronte, Ricardo Álvarez, José M. Quiñones, Hilton Pérez, Juan Lastra, Estefanía Torres, Carlos Figueroa, Christian Santiago, Jose Biescas, Christian Alapont, Leonel Osoria, Héctor Bencosme, Michael González, Carlos Garcia, Jonathan López, Eduardo Castillo, Greg Santiago, Madeleine Malave, Laura Villegas, Gabriela Pérez, Kevin Rivera, Lein Bajandas, Juan Rivera, Rose Sánchez, Genesis Rodriguez, Jaime Correa, Kevin J. Montalvo, Kevin N. Ramírez, Pedro Santiago, Carlos López, Fernando Colón, Carlos Ocasio, Froilan Colón, Alec Espina, Yousef Salaman

1. *Abstract*

As a third time competitor, Rumarino started a fresh agenda for the new generation of the team, which was the mentorship program since the project founders graduated. This year, Proteus, the University of Puerto Rico's Autonomous Underwater Vehicle (AUV), became a more sustainable model than years before by innovating and challenging the members with new ideas. It has upgrades in various areas such as: the mechanical design, embedded systems, software, vision, and control system fields. In the mechanical design, Proteus has 3D printed designs using PLA, ABS and four iron rods for stability. With these changes, a new mechanical model required an advanced controller designed by the control systems team. All software related teams kept improving the development The embedded system created a kit to corroborate the functionality of the thrusters and to help the team track the AUVs movement and precision, the software team designed and developed a graphical user interface (GUI) that improves testing and debugging; and vision found a different strategy to track objects underwater.

2. *Competition Strategy*

This year's competition strategy was to tackle the missions with a more robust and effective AUV. This would allow the team to face this year's challenges by modifying its previous build and make it more effective and expandable if needed. Working on improving their AUV also allowed the team to show the new members the previous system and the errors that were being modified and corrected.

In each Robosub competition Rumarino has participated in the team learned abundantly from these experiences and started to implement new ideas from errors encountered or from networking with other teams. Every working team was inspired to implement and develop their fresh ideas as best they could. This was done with certain key quality metrics that the team wanted to achieve; these include usability, maintainability, reliability, modularity, and expandability. Another goal was to make sure that the team would be able to implement the design and development in time for testing before the competition.

The Autonomous Architecture Division focused on developing a GUI (Graphical User Interface) which would provide feedback of certain sensors, such as cameras, motors intensity, and the inertial measurement unit (IMU). Moreover, another task tackled was the implementation of a simulator utilizing Gazebo. The benefit for this on the team would be allowing them to test the mission codes before testing them on the AUV, as well as to test the mechanical structure and how it affects the buoyancy. Lastly, the team modified the architecture of the code to be more modular.

The Vision Systems Team first started working on looking back on what has been done so far and how it could improve on it. Considering that most missions relating computer vision algorithms has been changing every year, the team doesn't have much to work on in terms of reusing code. However, this allows for a great opportunity to innovate and start algorithms with a different point of view from what was done before. This year, the team decided to take it a step further by integrating the computer vision algorithms into the control system. This presented a challenge for the team because greater accuracy and frame rate was required, which tend to be very difficult on probabilistic algorithms with great mathematical complexity. The team took on this challenge by learning how to program efficiently and the many alternatives to computer vision algorithms that existed and which are not necessarily included on programming libraries. Another improvement made in order to accommodate for the new requirements is to run tracking algorithms instead of detection algorithms in every single frame. These algorithms tend to have fewer costs in resources and faster response time [1]. Another thing taken into consideration was parallel programming. Given that the current on board computer has a GPU integrated into the system, it would allow the team to implement algorithms with real time capabilities. With this in mind and taking into consideration the hardware limitations, the team set out to develop algorithms capable of complying with every requirement given to them.

The Embedded Systems Team focused on the efficiency of troubleshooting and thorough research on how to complete more complex missions. They created an infrared rpm meter, that would benefit the team during testing runs. The main reason for this significant work was due to the inconsistency of the motor output, allowing the team to receive real-time thrust feedback compared to the desired thrust. On the research, the main focuses included data-wise sensors such as the implementation of a push button and creating an LED Status Display. Last year, RUMarino faced some coding problems which resulted in the AUV running missions at an incorrect time. Because of this, the team undertook the task of implementing a Status Display to verify whether the AUV was running the correct mission code when needed, in addition to perceiving any other failures. This display would suggest what components were running and what sensors had been activated. Another of the problems Rumarino faced was the uncertainty of whether an object was touched or not. The team first tackled the obstacle utilizing the cameras, yet due to the inaccuracy, a different method was used. Push buttons were the most optimal answer due to their low cost and simple implementation and due to the LED Display, Proteus could be able to determine if an object was touched successfully.

The Mechanical Structures Division and Electrical Systems Division had a new drastic perspective because of last year's competition. They wanted to design and implement a new mechanical structure from scratch. The problems to be solved were the rigidity of the structure, buoyancy, stability and the organization of electrical components. In order to accomplish the goal, a redesign was made taking in consideration the benefits of the previous structures (6.5 inches hull with a 4 inches hull underneath), the time frame available, and the time that control division requires to adjust their codes an realizes tests. The design implements a full 3D printed symmetrical chassis to provide a lightweight structure that also provides an adequate positive buoyancy. The materials chosen for the 3D printing process were PLA (Polylactic Acid) and ABS (Acrylonitrile

Butadiene Styrene), both offers different benefits for different parts around and inside the AUV. The joining method for the assembly uses only M4 and M8 screws with nuts that goes inserted in the 3D printed parts compared to the previous model that screws ranges from M3 to M8 and are screwed using T-slots and fastened directly into the 3D printed parts. The design also features channels on the side to organize the thruster wires and numbers for each part that helps for replacement and assembly process.

In the same manner, the electrical system division performs a rewiring and labeling process to facilitate replacement of all the different components inside the AUV. Also in collaboration with the mechanical division design a fixed base to organize the components in a manner that enables the easy access and does not affect drastically the center of buoyancy of the structure.

Assembling and disassembling the Submarine has always been a tedious task. Many different components tightly fitted and wired in to a negatively pressurized cabin results in many wires that need to be correctly connected. The electrical division focused on developing PCBs for the future submarine to minimize the cables run in the AUV. This year the electrical team soldered, labeled and replaced connectors for it to be impossible to incorrectly connect cables together, The electrical team also created an assembly manual. This facilitates the assembly process of our submarine to the point that any individual from the team, regardless of educational background is able to completely assemble the AUV.

This helps the electrical team to adapt the new changes form the Mechanical division. These structures would be designed in such a manner that they would be able to keep up with the rapidly developing nature of the team; thus, creating a new method design that will help in the future troubleshooting for the team.

The Operations Management Division focused on improving documentation in testing, troubleshooting, and mechanical and electrical assembly. Which is important for the team growth, since this will help the team in creating knowledge transfer. Due to the lack of workspace due to Hurricane Maria, the team was granted a sponsorship from Harris to design and create a laboratory for RUMarino. In collaboration with the Operations team, the Business Management Division focused on getting as many sponsors as possible, getting the team involved in the outreach initiative and had collaborated in creating the video promo for the team.

The team also noticed since many previous members had graduated this year there was a knowledge gap on the control systems of their AUV. Due to this problem, a new division was added: Control System Division. The Control Systems Team focused on upgrading and improving the existing controls of their AUV. They troubleshoot and modified their control system from a Proportional controller into a PI (Proportional + Integral) control system, which helped the AUV be more precise for the Align controller and with the help from Autonomous Architecture team; they developed new codes for the control systems of the AUV.

The team had to face immense struggles due to knowledge transfer, recruits, and pool access. Since this year most of the team leaders and members graduated, the team was in an immense recruit stage, where they were facing problems with the knowledge gap and mentoring the new members and new leaders. Otherwise, the team’s last struggle would be the access of the pool since this year the team had difficulties in testing and acquiring the pool facilities. However, the team managed to keep progressing and maintain their goal to compete this year.

3. *Design Creativity*

The design creativity took after their previous competitions, they decided to design an easier way to acquire their materials for their AUV, to have a fast replacement plan and designed a dynamic structure for this year's competition.

The biggest upgrade this year was set to be the mechanical structures. They created a new challenge to developing and creating a mechanical structure using 3D printed materials. Since their previous competition, most of the structure was unstable due to weight imbalance and some previous test with 3D printed extrusions shows a good behavior and alternative to the previous aluminum structure. They used PLA (Polylactic Acid) and ABS (Acrylonitrile Butadiene Styrene) for the parts depending on the forces that each part needs to handle. The process requires test parts with different infill percentage, perimeter shells, and layers. In addition, this process requires to test each printer tolerance to set the clearance parameters for the parts and join all the parts perfectly. Also, this design was a big upgrade, to developing the capability of having 6 thrusters or 8 thrusters layout available at any moment. This is due since the team was developing a new control system that will help the AUV be more dynamic and mobile as possible. All of the 3D printed design also has engraved numbers to facilitate the assembly of the AUV and fast replacement plan if the structure is damaged it could help locate the piece by number and print it out.

The use of rapid prototyping helped to keep the structural design simple and creative while being able to make quick changes when they were necessary. Using the same core design of last year’s competition, the AUV was developed as compact as possible, while keeping the design lightweight and at a manufacturing cost under $100 including the filament and all screws required. The 3D printed components are now the main structural element, due to the benefit with regards to buoyancy and simple replacement. The team opted to utilize metal rods and plates in the bottom which allow Proteus to maintain its stability. These new structures and components also add to the aesthetics of the AUV, projecting a new style and colored uniform look, thus Proteus 3.5 is born (**Figure** 1)

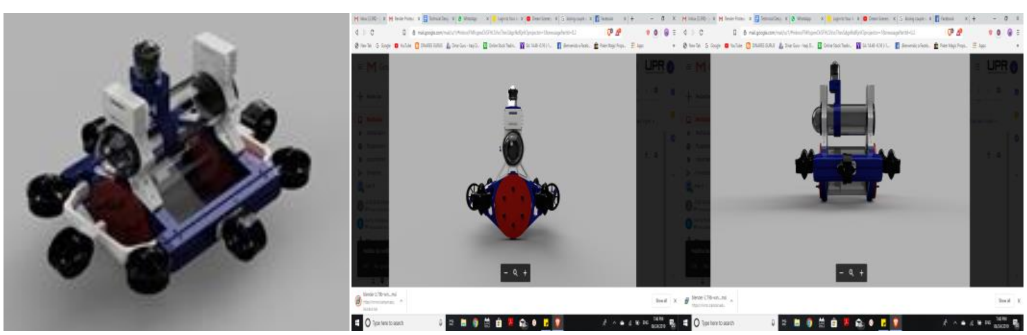

Figure 1*: Proteus 3.5 upside down.*

The high-level code was designed considering the Autonomous Architecture team diverse levels of skill. The team opted to code in Python. This decision arises since Python offers itself as a versatile programming language, providing a simpler learning curve, in comparison to other high-level languages. However, it is still powerful enough to encapsulate the overall necessities needed from the project. Its relative simplicity allows for new members to learn and contribute to the software development, whereas the more senior programmers and members may benefit from its versatility and be able to produce a more complex code. This allowed the team to add in object-oriented code and designs.

Last year, most of the code was done “in-house”. This meant the team members were developing by themselves software that was already available, which slowed progress down. Therefore, the Autonomous Architecture team, Vision team, Embedded team, and Control Systems team saw the benefits of implementing the popular ROS framework and decided to adopt it. With the adoption of ROS, a new software architecture was made to take advantage of all the functionality that ROS provides [2]. Given that ROS has a relatively high learning curve it took the team some time to get into its mindset, as a result, the development of the new architecture took more time than expected. The new architecture was more autonomous in its core since the architecture starts to be able to account for abnormal function situations, such as not finding an obstacle or getting lost, something that last year’s architecture was unable to do. A layer of abstraction was also added to this new architecture which allowed the Mission Code and the Vision Code to work seamlessly with ROS without the team members of those sub-teams having to learn about the intricacies of the communication between the two. ROS also permitted to run the controllers, which were originally run on microcontrollers on the main computer. This meant that the microcontrollers would serve as hardware interface boards.

The Autonomous Architecture team decided to develop a GUI (Graphic User Interface) which will provide the team with an overview of their AUV. The team had to adapt to the changing environment of other teams since changes in their design affect how the missions can play out. This means that they had to be watchful for these changes. So they wanted to keep up to date with them and help each team to facilitate their troubleshooting during the competition. The GUI developed provides a camera view from both frontward and downward position. It displays each of the motors intensity and the Inertial Measurement Unit (IMU).

Also, the Autonomous Architecture team was in development on creating the first simulator for the team. By using ROS, the framework offered the simulation engine called Gazebo that could help transfer the mission codes of the team and test them before launching the AUV in the water. However, it was not able to design the simulation completely this year, by some challenges that occurred during their semester. But it still in development for their next year.

One of the main challenges faced when designing the AUV’s vision system was the lack of depth perception because of the use of a single camera as the front and bottom facing inputs [3]. The team worked around this by implementing a hybrid yaw and depth controller. These controllers would receive two different inputs, one being from the corresponding sensor and the other from the detected object in the vision algorithm. However, this came with some difficulties. This was because detection of the object had to be precise and robust so that it could give a continuous input to the corresponding controller. To do this, a vision pipeline was created. This pipeline would receive the image from the camera and would output the pixel distance from the center of the camera to the center of mass of the object detected. To make the pipeline more robust and with a higher frame rate, a tracking algorithm was implemented along with a detection algorithm. Multiple tracking algorithms were taken into consideration such as the MIL, KCF, TLD, and MOOSE. A decision matrix was created with the requirements needed in the tracking algorithm. The algorithm needed to be fast and precise, with a higher emphasis on fast. It also needed to be accommodated with the library versions on the software architecture. KCF was selected as the tracking algorithm. Even though KCF is not as fast as MOOSE, it is more accurate and that's what tilted the balance towards KCF [4]. It was determined that once an object was detected the pipeline would run this tracking algorithm so that it would have higher accuracy and smaller processing time. However, during testing, the tracking algorithm would start to have lower accuracy during long intervals of time being used. A correction to the tracker was made every N frames by running the detection algorithm and giving it the new coordinates of the detected object. With this new implementation, we were able to achieve a higher throughput to the controllers and a more precise movement and alignment with the detected objects.

4. Experimental Results

Due to many setbacks, the period allocated for testing had to be delayed. It was decided that if they were to compete, they had to be able to upgrade and modify their AUV with their new members. Since their semester is over, more time is being dedicated to quickly implement and test the AUV's system.

As of the writing of this paper, the tested codes were the embedded systems, controls systems and Autonomous Architecture. Since, the Embedded Systems Team is analyzing the conversion rate of the T100-Thrusters by Blue Robotics from revolutions per minute (rpm) to thrust. Due to this semester's time constraints, the team has considered purchasing a tachometer due to its simplicity. Finally, it was decided for the employment of the push button and the LED Display to be postponed for future prototypes.

As well as the new update of the control systems of the AUV, to verify that is responding and working properly.Thought the semester the team was troubleshooting the system to make more precise and verifying that the new update would keep up to date with the new structure. Lastly the Autonomous Architecture using the GUI, to test and verify that the program runs properly before the teams head in the competition.

The updated mechanical structure demonstrated superior stability in the tests. Computational fluid models helps defining drag forces over the AUV and improve design aesthetics. The new chassis provides more confidence in the joints between parts and help the control division to refine their controllers system, an important improvement over the periodically loosen thruster in previous model. The chassis could not be tested with the 8-thruster layout due to the time required, but it will be done in the near future.

5. *Acknowledgements*

RUMarino would like to thank all of its team members and the faculty members of the Engineering Department of the University of Puerto Rico for their unwavering support. There is a special acknowledgment to all the sponsors who have helped guide us and challenge us to become better; RUMarino would not be here today without their help. From small to large, the team appreciates all the contributions and donations that have led to the team's participation in the upcoming RoboSub 2019 competition. The team would also like to thank all the students who researched in collaboration with the Industrial Affiliate Program (IAP) and excelled within the development of the AUV. The passion and knowledge passed on by these students have left a legacy within RUMarino.

6. *References*

1. Software.intel.com. (2019). A Closer Look at Object Detection, Recognition and Tracking. [online] Available at: https://software.intel.com/en-us/articles/a-closer-look-at-object-detection-recognition-and-tracking [Accessed 7 Jul. 2019].
2. Martín Gómez, David & Marín, Pablo & Hussein, Ahmed & de la Escalera, Arturo & Armingol, J.M.. (2016). ROS-based Architecture for Autonomous Intelligent Campus Automobile (iCab). 257-272.
3. Bertozzi, M., Broggi, A. and Fascioli, A. (2000). Vision-based intelligent vehicles: State of the art and perspectives. Robotics and Autonomous Systems, 32(1), pp.1-16.
4. "OpenCV Object Tracking," 04-Feb-2019. [Online]. Available: https://www.pyimagesearch.com/2018/07/30/opencv-object-tracking/.

Appendix A: Component Specifications

Component	Vendor	Model/Type	Specs	Cost (if new)
Frame	Hatchbox	ABS Filament	Diameter: 1.75 mm 1 Kg Spool	$20
	Hatchbox	PLA Filament	Diameter: 1.75 mm 1 Kg Spool	$20
Waterproof Housing	BlueRobotics	Watertight Enclosure: Acrylic Tube (4" & 2")	Max Depth: 330 ft ID: 4in & 2in OD: 4.5in & 2.25in Length: 13.15in	4" Series: $183 2" Series: $107
	CrustCrawler Robotics	WaterProof Vessel	Depth Rating: 150 ft ID: 6.5 in OD: 7 in Length: 12.75 in	
Waterproof Connectors	BlueRobotics	Cable Penetrators for 6mm and 8mm Cables	Bolt Threading: M10	

Thrusters	BlueRobotics	T100 Thruster	Max Thrust – Forward: 5.2lbf Operating Voltage: 12V Max Power: 130W Diameter: 3.8 in	$119.00
Motor Control	BlueRobotics	Basic ESC	Voltage: 7-26 V Current: 30 Amps Signal: Pulse-width (PWM) Max Reverse: 1100 µs Stopped: 1500 µs Max Forward: 1900 µs Deadband: 1475-1525 µs	$25.00
High Level Control				
Battery	Venom Power	LiPo 3 Cell Batteries	Capacity: 5000 mAh Voltage: 11.1 V	
CPU	A57	Cortex	j	
Internal Comm Network				
External Comm Interface				
Programming Language 1	Python 2/3			
Programming Language 2	C++/Arduino			
Inertial Measurement Unit (IMU)	VectorNav	VN-100	3-axis accelerometer, 3-axis gyros, 3-axis magnetometers, and a 32-bit processor.	
Cameras	BlueRobotics	Low-Light HD USB Camera	Field of View (Horizontal): 80° Field of View (Vertical): 64°	$89.00

Table 1: Slicing parameters and density of parts

Number and STL par name	Infill		Printing Time	Actual Printing Time		Approximated Material Used (g)			Filament Color	Volume (mm3)	Density (kg/m3)
	Percentage %	Type	Slicer (Hours)	Hours	Minutes	Slicer	# of prints	Total			
1ChassisV2	35	Grid	18.9	49	13	425.38	2	850.76	Blue	609073.37	698.41
2ChassisOppositeV2	35	Grid	18.9	49	5	425.19	2	850.38	Blue	611197.04	695.67
3LateralBracketv2	30	Grid	0.9	2	12	17.02	2	34.04	White	17805.59	955.88
4DepthBracketV2	35	Grid	3.3	8	29	70.72	2	141.44	Blue	88643.44	797.80
5DepthBRacketOppositeV2	35	Grid	3.3	8	27	70.41	2	140.82	Blue	88639.38	794.34
6RTmountV2	30	Grid	4.7	12	10	103.56	2	207.12	White	146357.38	707.58
7LTmountv2	30	Grid	5.1	13	10	109.37	2	218.74	White	150824.01	725.15
8secondcabsupportv2	35	Grid	5.0	13	2	99.58	2	199.16	White	92595.92	1075.43
9secondcabweightsv2	30	Grid	5.8	14	59	113.6	2	227.2	White	117467.00	967.08
10Weightscoverv2	30	Grid	0.7	1	49	14.18	2	28.36	White	11535.83	1229.21
11Cammountv2	30	Grid	7.8	10	8	154.35	1	154.35	White	150236.48	1027.38
12CamBracketv2	30	Grid	1.3	1	43	26.98	1	26.98	White	25581.52	1054.67
				184	33	Total		3079.35		2109956.95	

Table 2: Materials for structures

Materials	**Quantity**	**Cost ($)**
1Kg Filament spool	3	19.99
M4 x 15 mm screw (55 pack)	1	8.75
M4 Hex Nut (100 pack)	1	6.90
M8 x 35 mm (20 pack)	1	11.49
M8 Hex Nut (20 pack)	1	6.79
	Total Cost	93.90