TERRA Technical Design Report - RoboSub 2022

Abner Gabriel de Souza, Amanda Pereira, Aryane de Paula Cezario, Bruno Ricardo Seixas, Cristian Piehowiak, Eduardo de Bittencourt Ribeiro, Gabriel Mariano Gonçalves Santos, Gabriela Custódio Martins, Giovanna Ambrósio de Souza, Igor dos Santos Soares Pinto, Juliana Pereira Meireles, Kalyne Marques Moreira Dias, Luara Ferreira Lino, Lucas Eduardo Piana, Luiz Guilherme de Oliveira Appel, Mariana Dias Martins, Patricia Mistura Klippel, Rafael D'Amaro Chiara and Thais Cristina Gaino.

Faculty Advisors: Prof. Andrea Piga Carboni, Prof. Lucas Weihmann, Prof. Roberto Simoni

Abstract – **For the 2022 edition of RoboSub, Terra Competition Team focused on improving the old systems and developing new systems in order to accomplish a greater number of tasks when compared to the previous year's project. In order to achieve these objectives, the mechanical sector developed the project of a torpedo launcher and carried out experimental tests to evaluate its performance. In turn, the electronics sector has advanced in computer simulations and optimization of computer vision code, ensuring greater autonomy to our AUV.**

I. Competition Strategy

Last year, the project developed by the team was only able to pass through the starting gate of the run. For this year, we decided to improve and develop new systems in order to allow our AUV to be able to perform a greater number of tasks. In this way, when starting the run, our AUV will be ready to perform the alignment with the gate required in the task "With Moxy(CoinFlip)". Then, our AUV will be able to recognize the gate structure and pass through it in the "Choose Your Side (gate)" test. Continuing our run, our vehicle will be capable of recognizing the markers and head towards the "Make the Grade (Buoys)" task. Once there, it will be able to choose and touch in one of the buoys positioned in the pool. As we still haven't the proper acoustic sensors, our AUV will head to the "Survive the Shootout—Torpedoes" task by visual recognition and when it gets there it will fire torpedoes with the aim of hitting the targets.

This year we decided to name our vehicle the Fênix, as a way of symbolizing the restart and the ability to overcome the adversities lived throughout this project.

II. Design Creativity

A. Mechanics sector

For the 2022 edition of RoboSub, the team chose to kick-start the development and carry out a battery of concept tests to implement a torpedo launching system. The propulsion principle was selected from the analysis of the TDRs made available by the teams participating in RoboSub in previous editions.

A1. Modeling & External Geometry

Saghafi, Lavimi (2019) [1] present nose and tail equations for the radius distribution of axisymmetric geometries, optimized for reducing viscous drag, stagnation pressure ahead of the nose and recirculation aft of the tail.

The equations for the nose and tail geometries are parameterized by the lengths of these sections, the maximum diameter of the body and, for the tail, a characteristic arc angle.

The reference keeps the total length constant. Here, in order to evaluate the influence of the torpedo length on the directional stability in free launch, a systematic variation of the length of the cylindrical section will be applied, parameterized by multiples of the maximum diameter. Three geometries were modeled, with cylindrical midbody lengths of 1, 2 and 5 times the maximum diameter (see figure 1).

The points that define the external geometry and the cavity for buoyancy adjustment were generated in Octave, converted into a three-dimensional model via SolidWorks modeling.

A2. Materials

The prototypes were obtained using conventional additive manufacturing methods (3D

printing) in PETG, a thermoplastic denser than the test tank fluid (fresh water). Given the exposure of the structure to an underwater environment, it is desired that hygroscopicity is minimized.
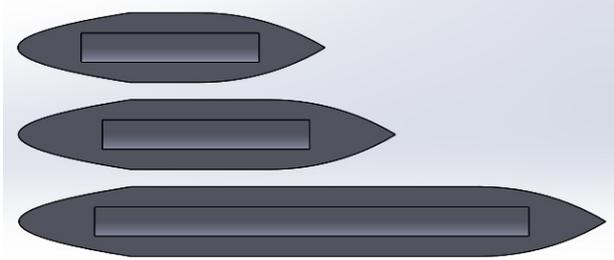

Figure 1: Geometries of torpedoes modeled .

A3. Internal Structure and Ballast

In order to keep the external geometry proposed in the literature unchanged, the adjustment of buoyancy and convergence of the longitudinal coordinates of the centers of mass and buoyancy to avoid free-floating rotations were achieved through the positioning of a cavity inside the geometry (see flowchart in figure 2). The characteristic dimensions of the cavity are dependent on the density of the selected material. Before the torpedoes were printed, the post-print density was verified via the Archimedes method, as described by Spierings, Schneider, Eggenberger (2011) [2] . From this measurement the final geometries were generated.

A3. Launching System

The initial proposal was to evaluate the effectiveness of torpedoes via compressed air ejection. The purpose of the tests was to collect information relevant to the ongoing development of the launcher-projectile pair. Test protocols were generated by the team that allowed the assessment of the torpedo's range and directional stability, together with the exposure of technical problems not foreseen in previous stages of the project.

The pneumatic system used is composed of a reservoir for the working fluid (atmospheric air), a valve and a guide tube. Pressurization and firing were performed manually. This was the simplest and most economically efficient way found to circumvent the problems involving electronic

valve control, allowing the reproducibility of the output speed through the control of the internal pressure of the reservoir, a magnitude easily measurable during testing.
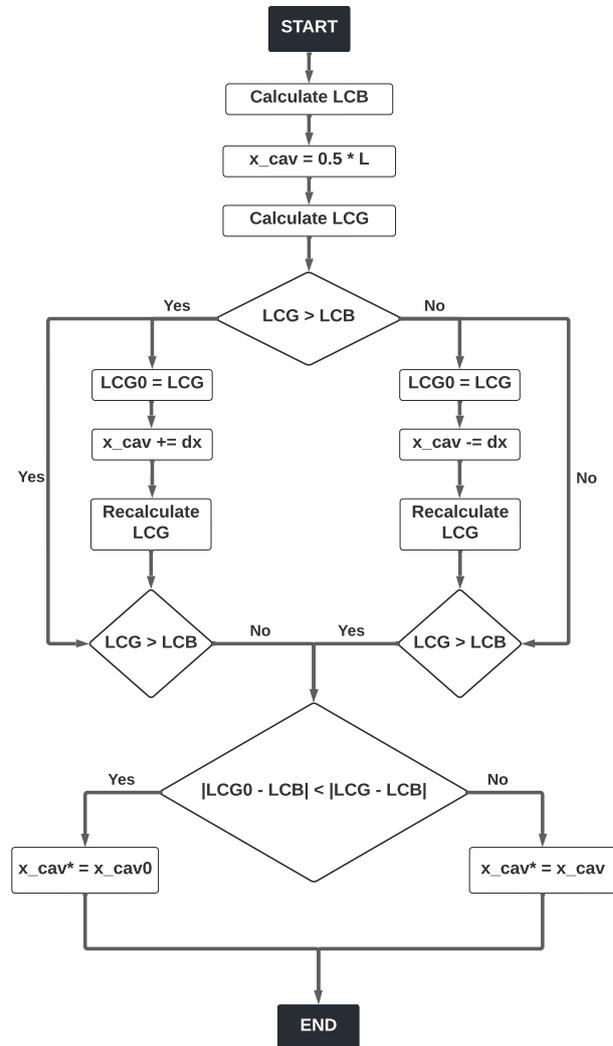

Figure 2: Flowchart regarding the positioning of a cavity.

A4. Battery

To enable the execution of all the tasks stipulated by the team in this year's competition, it was decided to use two batteries, the Turnigy and "Battery A". Turnigy is a battery purchased with LI-PO (lithium polymer) technology with a minimum capacity of 10000mAh and a maximum voltage of 14.8V in the 4S1P configuration. Battery A" is an assembled battery that has an average voltage of 12V (with a maximum voltage of 16.8V) and a capacity of 12500mAh in a 4S5P configuration.

B. Electronics sector

Our biggest challenge in the last competition was establishing a connection with Gazebo. So this year we decided to migrate to Unity. This idea arose out of the simulation made by Inspiration Team [3], available in the app, which would allow the team to focus less on the simulation development. With this, our main focus would be to optimize the vision code and design a mission plan.

With Unity, we were able to do a higher quality simulation, as well as being able to simulate the amount of light in the water, turbidity, and more, improving AUV capability. However, despite the qualities of the application, we found it immensely difficult to use it, given the lack of experience.

Considering that the electronics sector is divided between control and vision, each subsector was responsible for different objectives, as follows:

B1. Control

The tasks performed by the AUV's control subsector were: creating a mission plan, so that image recognition would work more fluidly, by making it possible to carry out the tasks given.

The first task the team tried to do was "With Moxy (CoinFlip)", where the AUV is placed in the water in a random position, so that the sub could align itself to the gate and pass through. After the success of this first chore, the goal was to pass the "Choose Your Side (gate)" test, where the AUV should choose a character and go through the corresponding portal, which also proved to be a success. For the last test run, "Make the Grade (Buoys)", the sub should touch one of the two buoys positioned in the pool, getting a higher score if it chose the same side as in "Choose Your Side".

In the simulations, the AUV was able to complete the entire mission plan established, therefore accomplish the main control intent.

B2. Computer Vision

To accomplish the tasks of the competition, the vision members implemented software programs to identify the portal, the path makers, the buoys, and the characters. The idea of using OpenCV [4] was intended to reduce the use of neural networks for the identification of certain competition objects, considering that we have at our disposal a board with limited processing power for the task of neural networks , a Raspberry Pi 3B+.

To improve the quality and contrast of the images obtained from the simulator, some techniques were used, such as contrast limited adaptive histogram equalization (CLAHE) [5] and Gamma correction. These filters helped to remove the effect of greening and turbidity of the water. In Experimental Results, images with and without filters for enhancements can be seen.

In the gate detection, image segmentation tools were used from the HSV mode, in addition to other filtering tools, as well as contour extraction tools. After extracting the contours and approximating them by rectangles, the central coordinate of this object is mapped at each instant of time. The relative sizes of each of the legs are also verified to check the angulation of the AUV regarding the gate. In Experimental Results, a flowchart illustrating the process of controlling the AUV during the step of passing through the gate can be seen.

For pool bottom markers, the contours of this object are extracted, and the angulation of this object relative to the horizontal is checked. That way, the AUV can orient itself to the next task.

As for the detections of the characters fixed on the portal and the buoys, a deep learning solution was used. This solution was chosen due to the greater complexity of these tasks. Thus, using traditional vision tools for these cases would bring many difficulties and the solution obtained would not be very robust.

Images of the simulated environment of Transdec in Unity were used as a basis for the team's dataset, since we were unable to perform vision tests with the AUV in person. After collecting the dataset, the image annotation process was also carried out.

Subsequently, studies began on possible models of convolutional neural networks that could be used, taking into account the hardware resources available to the team. Three models were studied and tested, namely: Yolo-v3 Tiny,

SSD MobileNet and Faster R-CNN (from the Detecto library). After the tests were performed, it was noticed that the most accurate model was the one from the Detecto library. However, the neural network size was too large, and our board would not be able to load the model into its memory to perform the inferences.

Therefore, it was decided to use another model, less precise but lighter and with faster inferences. Both Yolo-v3 Tiny and SSD-Mobilenet had these characteristics, however it was noticed that the model trained with the SSD architecture was performing more accurately and with faster detections than the Yolo architecture.

With that in mind, SSD Mobilenet architecture neural network [6] was chosen for the competition, used from the TensorFlow object detection API [7], in order to obtain a reasonable precision with greater speed in the detections, considering the existing hardware for the team.

Based on the performed tests in the Unity simulation environment, it was possible to verify that the chosen model presented good detection results. It was noticed that the model worked very satisfactorily for medium and large objects in relation to the image size, but for small objects, the model faces some difficulties, even though it manages to perform the detections. However, by performing the detections of these smaller objects (gate characters) only in regions closer to the portal, the instability of the detections was overcome. Experimental Results shows image objects being detected in the simulator images inside the Transdec environment.

III. Experimental Results

A. Torpedo Field Tests

After assembly and pressurization of the system, launches were carried out to establish a correlation between initial pressure, ejection velocity and range for the available geometries. Test launches at pressures of 20, 40, 60 and 80 psi were conducted for each of the existing prototypes.

The effective range for all geometries was small ($\approx 0.3m$) and not sensitive to the varia-

tion of the cylinder's internal pressure. The short range defeated the purpose of the initially proposed evaluation of the trajectory deviation.

During the tests, the following problems were observed:

1. The drag imposed on the torpedo quickly dissipates the kinetic energy given to it during launch.

2. Difficulty in maintaining system watertightness. Part of the working fluid flows between the torpedo and the inner walls of the guide tube. This problem persisted even after adjustments of the external diameter of the torpedoes.

3. The recoil caused by the ejection of water that accumulates in the portion of the guide tube ahead of the nose may be deleterious to the navigation and positioning of the AUV after launch. It is desirable that mechanical work is applied to the torpedo in a smoother manner.

The tests showed the ineffectiveness of the proposed system in carrying out the task. The team interprets that the implementation of solutions to improve the current system is inefficient. The launching concept is to be reviewed. New guidelines discussed by the team after testing point to the creation of a self-propelled torpedo, as well as the inclusion of stabilizers along the outer geometry.

B. Computer Visual Simulation

The electronic simulations were performed in Unity and the results referring to the discussions in the previous section will be presented below.

Regarding the techniques used to improve the qualities of the images obtained by Unity, it is possible to observe in figure 3 the original image of the gate in the simulator and in figure 4 the same image corrected with the filters used to remove the greening effect and turbidity of the water.
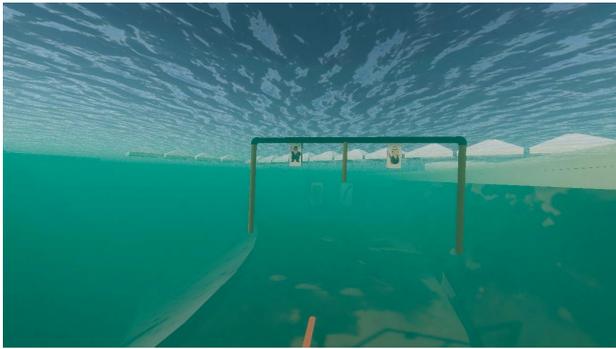
Figure 3: Original image of the gate in the simulator.
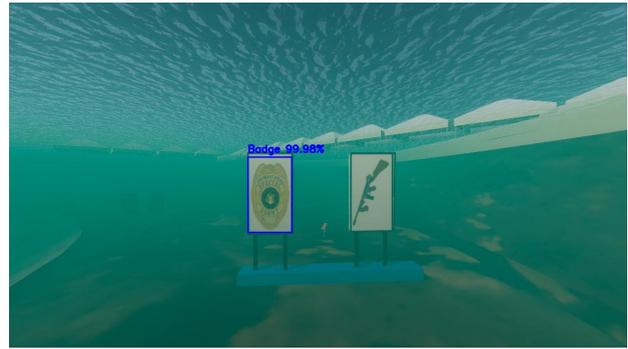

Figure 5: Object detections of the badge.

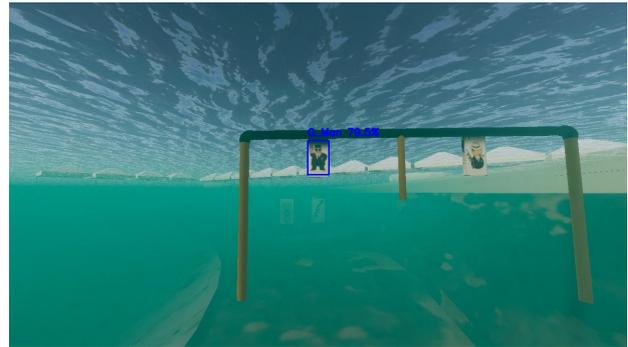Figure 4: Corrected image of the gate in the simulator.

Figure 6: Object detections of the character.

As for the detections of the characters fixed on the portal and the buoys, a deep learning solution was used. In general, the model displayed a good accuracy for medium and large objects in relation to the image size, as can be seen in figure 5. For small objects, the model presented some difficulties, but still managed to perform the required detections as can be seen in figure 6.

As previously discussed, in the detection of the gate and the markers, the contours are first extracted and approximated by rectangles. Then, for the gate, the central coordinate of the object is mapped and the relative sizes of each of the legs are verified to analyze the angulation of the AUV in relation to the gate, allowing the AUV to orient itself correctly. The figure 7 shows in detail how the passage of the AUV through the portal works.
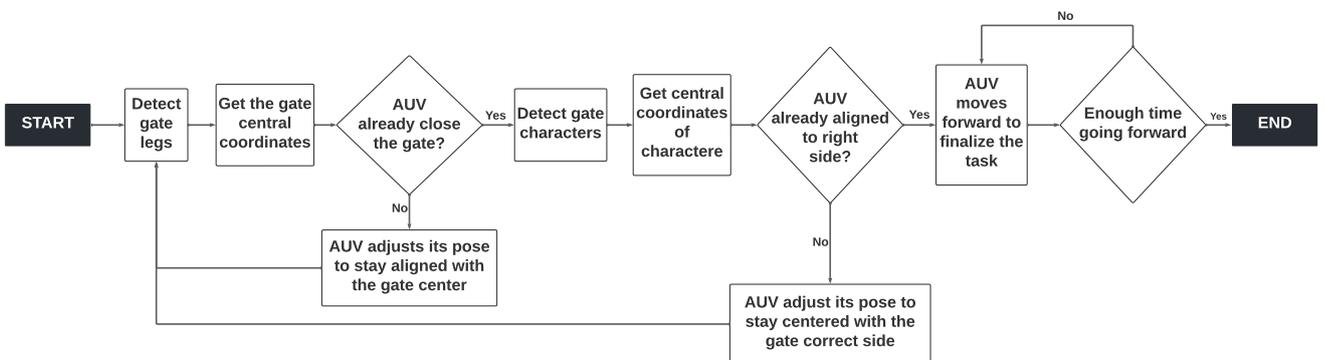

Figure 7: Control software flowchart regarding the "Pass the Gate" task.

ACKNOWLEDGEMENTS

REFERENCES

[1] Saghafi, M; Lavimi, R. **Optimal design of nose and tail of an autonomous underwater vehicle hull to reduce drag force using numerical simulation.** Journal of Engineering for the Maritime Environment, Iran, 2019.

[2] Spierings, A.; Schneider, M.; Eggenberger, R. **Comparison of density measurement techniques for additive manufactured metallic parts.** apid Prototyping Journal, vol.17 n.5, Emerald, 2010.

[3] Transdec Environment simulation in Unity made by Inspiration Team. [Online] Available in: `https://github.com/InspirationRobotics/RoboSub-Simulation`

[4] OpenCV. (2022) [Online] Available: `https://opencv.org/`

[5] Verna, Aryan. **CLAHE Histogram Equalization – OpenCV.** (2021) [Online] Available: `https://www.geeksforgeeks.org/clahe-histogram-eqalization-opencv/`

[6] Forson, Eddie. **Understanding SSD Multibox.** (2017) [Online] Available: `https://tinyurl.com/3sp4mm7p`

[7] Patel, Khush. **Custom object detection using Tensorflow from scratch.** (2019) [Online] Available: `https://tinyurl.com/3zm8uhwt`

APPENDIX A

TABLE I
Component Specifications

Component	Vendor	Model/Type	Technical Specifications	Cost	Status
Buoyancy Control	Blue Robotics	BlueROV2 Machined Buoyancy Foam	**Quantity:** 4 **Material:** LAST-A-FOAM® R-3318	$ 53.00	Installed
		Ballast Weight	**Quantity:** 5 **Material:** Stainless Steel	$ 9.00	Installed
Frame	Blue Robotics	BlueROV2 Frame	**Material:** High-density polyethylene	$ 339.00	Installed
		Payload Skid	**Material:** High-density polyethylene	$ 279.00	Purchased
Waterproof Housing	Blue Robotics	Watertight Enclosure	**Diameter eletronic en-clousure:** 4 in	$ 224.00	Installed
			Diameter battery en-clousure: 3 in **Quantity:** 2	$ 123.00	One installed and one purchased
Waterproof Connectors	Blue Robotics	M10 Cable Penetrator	**Quantity:** 8 **Diameter Cable:** 6 mm	$ 4.00	Installed
			Quantity: 2 **Diameter Cable:** 8 mm	$ 5.00	Installed
Thrusters	Blue Robotics	T200 Thruster	**Quantity:** 6 **Operating Voltage:** 7-20 V **Thrust Maximum (20 V):** 5.25 kgf	$ 179.00	Installed
Electronic Speed Controller	Blue Robotics	Basic ESC	**Quantity:** 6 **Maximum Current:** 30 amps **Firmware:** BLHeli_S **Operating Voltage:** 7-26 V	$ 27.00	Installed
Camera	Blue Robotics	Low-Light HD USB Camera	**Maximum Resolution:** 1080 p **Field of View :** 80° (Horizontal) and 64° (Vertical) **Compression format:** H.264/MJPEG	$ 99.00	Installed
Pressure Sensor	Blue Robotics	MS5837-30BA	**Measurement:** 30 bar (300m / 1000 feet deep)	$ 72.00	Installed

Component	Vendor	Model/Type	Technical Specifications	Cost	Status
Light	Blue Robotics	Lumen Subsea Light	**Quantity:** 4 **Operating Voltage:** 10-48 V	$ 115.00	Installed
Microcontroller	Arduino store	ATmega328 (Arduino Nano 3.x)	**Microcontroller:** ATmega328 **Flash memory:** 32 KB, of which 2 KB is used by the bootloader	$20.70	Installed
CPU	Submarino	Raspberry Pi 3B+	**Processor:** BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC at 1.4 GHz **RAM memory:** 1 GB LPDDR2 SDRAM **Internal memory:** Micro SD 16 GB **Power input:** 5 V / 2.5 A DC	$ 115.00	Installed
Flight controller	Banggood	pixhawk px4 pix 2.4.6	**Processor:** 32 bits STM32F427 Cortex M4 core with FPU - 168 MHz 256 KB RAM 2 MB Flash **Security coprocessor:** STM32F103 **ST Micro Gyroscope:** L3GD20 **ST Micro Accelerometer/Magnetometer:** LSM303D **Barometer:** MEAS MS5611	$ 128.00	Installed
Stepper motor	Blue Robotics	HS-5055MG	**Velocity:** 0.2 s for 4.8 V 0.17 s for 6 V **Output torque:** 1.3 kg-cm for 4.8 V 1.6 kg-cm for 6 V	$ 39.00	Installed

TABLE II
Software Information

Control Unity	Language(s)	Tools/Libraries
Algorithms: computational vision	Python	OpenCV
Algorithms: simulation	C++	Unity
Algorithms: state machine	Python	ROS
Algorithms: engine control	C++	ROS

TABLE III
Additional Information

Team Size (number of people)	18
Expertise ratio (electronic vs. mechanic)	9:4 (9 electronic, 4 mechanic, 2 marketing, 2 administration, 1 management)
Testing time: simulation	100 hours
Testing time: in-water	20 hours