The Design of Team VORTEX's 2021 AUV

Amr Essam (Team Captain), Islam Gamal, Mahmoud Assem, Lina Nassar
Ahmed Ayman, Omar Ahmed, Zeyad Ahmed, Mohamed Magdy, Islam Wael, Mohamed Salem, Abdelaziz Assem, Mariam Mahmoud, Fatma Mohamed, Gad Mohamed, Mohannad Kandil, Abdalla Ibrahim, Eyad Ahmed, Anas Khamees, Rania Sami, Shororuk Talaat, Nawaar Nasrat, Heba Hesham, Mahmoud Nawar, Mohanned Shahen, Ahmed Tarek, Abdullah Hosny, Ahmed Bahaa, Hossam Ibrahem, Rafael Zaher, Ahmed Solyman
Vortex Robotics, Alexandria, Egypt

Abstract— Team VORTEX is the first manufacturer of ROV in the Middle East. With the help of our experience in the underwater industry, our team of the most experienced and talented undergraduates and graduates designed our first AUV, named SWIFT, for the 2021 RoboSub competition. During the development phase, our main concern was to achieve a flexible, modular, maintainable design, as well as ensuring safety and reliability, as we intended that it would be an industrial project. In designing SWIFT, we used main components like ZED2 camera, DVL, Sonar, and Hydrophones. We also learned NMPC, EKF and VISION algorithms like GANS and Denoising auto-encoders; our software is implemented as ROS2 packages. Due to COVID-19 pandemic, our team relied on unit testing; the mechanical team tested the enclosures that withstand a pressure up to 7 bars, the electrical team made a Destructive Test to test the endurance and behavior of a 6mm width copper trace, and the software team developed our simulator.

I. COMPETITION STRATEGY

Since 2016, we have been participating annually in the MATE ROV competition and achieved high rankings, including the first position in 2018. Now, we have decided to take a further step and accept a higher challenge, by building our first AUV "SWIFT" and participating in the 2021 RoboSub Competition, so Vortex Co has gathered a team from the most experienced and talented undergraduates and graduates from around Egypt to build a strong structured flow of work.

We started to analyze the competition rules and past years programs to determine the requirements for building SWIFT, considering that passing through the start gate is a required task and the other tasks are optional, we design a strong visual perception module for SWIFT to identify the start gate, different objects like buoys and to determine whether the tasks correspond to the G-man or Bootlegger, as well as localization system and motion control system for SWIFT to navigate to the required task with confidence. To ensure gaining maximal points, we have added hydrophones to navigate to "Survive the Shootout" and "Cash or Smash" missions, designed a torpedo for "survive the shootout" mission, and designed droppers for "Collecting (Bins)" mission.

Then we started to plan how to accomplish these tasks effectively focusing on the modularity and flexibility of the design to facilitate the maintenance process and easily fix the affected parts without requiring a change in the entire design, also considering future enhancements.

The software team applied the V-Model, for the Software Development Life Cycle, analyzing the AUV system, determining the requirements, and designing the software

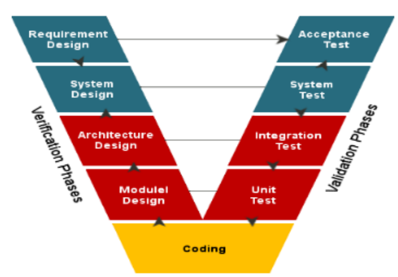

Figure 1 V model

architecture, to draw the path followed then started to develop the software as ROS2 packages.

The visual perception module is mainly divided into four main subsystems; each solves major problems that counters SWIFT's underwater vision to accomplish its tasks: Image Enhancement, Object Detection, Target Analysis, and Bias Filtering.

Considering the underwater noise on images, More than one approach was used (Image Processing - GANS – Auto Encoder) for the images to be clear for the object detection model to identify the start gate and other objects like buoys.

For navigation and motion control modules, we developed a sensor fusion system to fuse data from multiple sensors (DVL [1], Pressure sensor, and IMU) in addition to using the positional tracking from the ZED2 camera to minimize the impact of individual sensor failure in determining the current state of SWIFT. While non-linear model predictive controller (NMPC) and a reference model for trajectory tracking were developed to keep track between the current state and the next desired state.

Reducing the SWIFT's size down to (75x65x30cm) whilst keeping optimal functionality was also a challenging part for the mechanical team as achieving maximal bonus points is a goal of ours.

As a part of testing, the software team started prop building the start gate and other figures from the competition for collecting data and testing model measures, develop a simulator for SWIFT to test its software algorithms, while the electrical and mechanical teams relied on unit testing and solid work simulation.

Figure 2 Prop building the start gate

II. Vehicle Design

A. Mechanical

1) Internal structure and frame: Previously, "as ROV-makers", we used to use Polyamide nylon 6 cylindrical tube with two flanges made by Polyamide nylon 6 as a sealed enclosure, but now, as the modularity was our strategy of work while building SWIFT, we had to use many enclosures to make our system as much modular as we could. For reducing the wasted areas in the cylinders and reducing the volume and the buoyancy of the vehicle, we decided to change the cylindrical enclosure by rectangular ones that are made of aluminum.

The enclosures were manufactured by a CNC router machine and they were designed and tested to withstand a pressure of up to 70 meters of water.

All the enclosures are connected with plug and play connectors and fixed on both sides of an aluminum sheet, the enclosures on one side are fixed back-to-back to the enclosures on the other side to make it easier to access each enclosure directly.

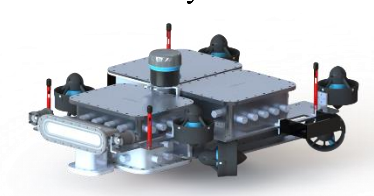

Figure 3 Internal structure

The whole internal structure is covered by a fiber streamlined cover that reduces the drag force on SWIFT, to be as low as possible, and the cover was also manufactured by a CNC router machine.

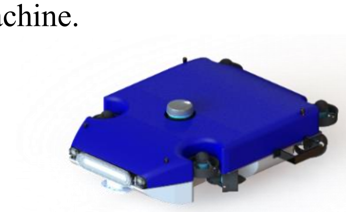

Figure 4 Streamlined cover

2) *Droppers*: The dropper mechanism is designed as a half-circular box with two dropping holes in the bottom, two loading holes at the top, and guiding arms in the interior.

Markers inserted into the loading holes will be held by the guiding Arms. These Arms are connected to a servo motor that will push the markers to the dropping holes. Each rotation direction of the servo motor will drop only one marker.

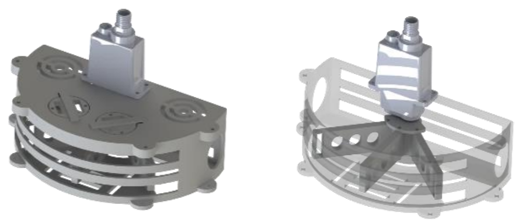

Figure 5 Dropper mechanism

3) *Torpedoes:* The torpedo launcher consists of a scotch yoke mechanism with double arms connected to a servo motor. These arms fit in the torpedoes' back holes.
The 3D-printed torpedoes are designed and tested to reduce the drag force to the minimum. In the start position these torpedoes will be loaded by compressing the springs behind them, staying held by the Scots Yorke arms, then when we turn the servo motor in the clockwise direction, the scotch yoke arm will slide out of the torpedo releasing the spring and the torpedo. For the anticlockwise direction, the other arm will release the other torpedo.

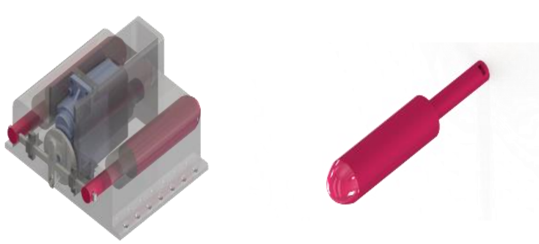

Figure 6 Torpedo launcher mechanism; Torpedo

B. Electrical

As for our Electrical system, we thought of flexibility and ease for debugging and troubleshooting, So, we divided our system into smaller sub-systems:

- Battery E-pod.
- BMS (Battery Management System) E-pod which includes an STM32 microcontroller.
- Propulsion E-pod which includes a Pixhawk Flight Controller.
- Main Computer E-pod (Nvidia AGX).
- Localization E-pod that includes a micro-controller specified for acoustic localization.
- Payloads E-pod.

Our Electrical team provided a Wiring Table stating the connections between each sub-system and their internal connections.

From			Specs				To		
Enclosure	Connector No.	Type	Length (cm)	Pin	Colour	Usage	Type	Connector No.	Enclosure
Propulsion	T1	Bulkhead	35	1	Green	CS for thruster 1	Simple Penetrator	M12	BMS
				2	Blue	CS for thruster 2			
				3	White	CS for thruster 3			
				4	Brown	CS for thruster 4			
				5	Black	CS for thruster 5			
				6	Red	CS for thruster 6			
	T2	Bulkhead	35	1	Green	CS for thruster 7	Simple Penetrator	M13	BMS
				2	Blue	CS for thruster 8			
				3	White	CAN 1			
				4	Brown	CAN 1			
				5	Black	GND			
				6	Red	-----------			
	T3	Bulkhead	25	1	Green	Pixhwak USB Terminal 1	Bulkhead Cobalt Series Cable, Double-ended	C5	Nvidia
				2	Blue	Pixhwak USB Terminal 2			
				3	White	Pixhwak USB Terminal 3			
				4	Brown	Pixhwak USB Terminal 4			
				5	Black	Leakage prob (Terminal 1)			
				6	Red	Leakage prob (Terminal 2)			
	T4	Bulkhead	N/A	1	Green	Yellow Signal	Termination Kit	LED Bluerobotics	External (LED)
				2	Blue	Red Power			
				3	Black	Black Ground			
				4	Brown	-----------			
	T5	Bulkhead	55	1	Green	Negative Terminal 1	Simple Penetrator	M14	BMS
				2	Blue	Negative Terminal 2			
				3	Black	Negative Terminal 3			
				4	Brown	-----------			
	T6	Bulkhead	55	1	Green	Positive Terminal 1	Simple Penetrator	M15	BMS
				2	Blue	Positive Terminal 2			
				3	Black	-----------			
				4	Brown	Positive Terminal 4			
	T7	Bulkhead	N/A	1	Red	ESC Blue phase	Pre-installed Connector		Thruster 1
				2	Black	ESC Green phase			
				3	Yellow	ESC white phase			
	T8	Bulkhead	N/A	1	Red	ESC Blue phase	Pre-installed Connector		Thruster 2
				2	Black	ESC Green phase			
				3	Yellow	ESC white phase			
	T9	Bulkhead	N/A	1	Red	ESC Blue phase	Pre-installed Connector		Thruster 3
				2	Black	ESC Green phase			
				3	Yellow	ESC white phase			
	T10	Bulkhead	N/A	1	Red	ESC Blue phase	Pre-installed Connector		Thruster 4
				2	Black	ESC Green phase			
				3	Yellow	ESC white phase			
	T11	Bulkhead	N/A	1	Red	ESC Blue phase	Pre-installed Connector		Thruster 5
				2	Black	ESC Green phase			
				3	Yellow	ESC white phase			

Figure 7 Demo for wiring table

We relied on CAN Bus as our communication protocol with 4 nodes, as in Fig. 8 each representing a subsystem: Nvidia AGX, BMS, localization, payloads. The exception is the Pixhawk controller in the propulsion E-pod as it is connected to the Nvidia through serial communication. Using CAN Bus allowed us to minimize internal wiring while maintaining a decent data transmission rate.

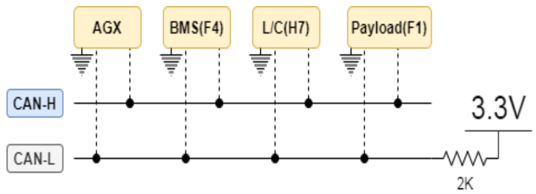

Figure 8 Can Bus

Design Measures like safety, endurance, plug & play, and size costed multiple brainstorming meetings and effort to create the most compact design, Therefore, we designed and fabricated our customized PCBs using onboard fuses and current sensors for protection, 120-micron thick copper traces to withstands high currents and On-board ESCs to reduce wiring. A sample of our PCBs is shown in Fig. 9.

This design concept was applied to all of our PCBs. To solve the problem of high temperatures in MOSFETs, we attached small fans to heat sinks in each MOSFET to provide cooling and continuous air circulation.

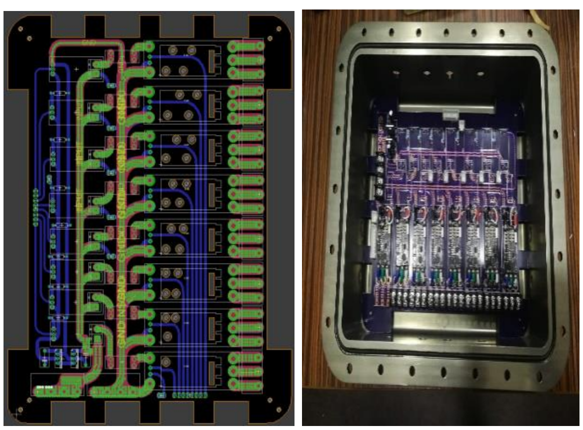

Figure 9 Propulsion PCB design; Propulsion PCB

Moreover, one of our major concerns was to make it easy for us to test and debug, so we made SWIFT support powering by an external power source when attached (Tethering), which allowed us to have both Tether-powered mode and Battery-powered mode. These 2 modes are switched via high power Relay that switches between the external power source and the battery. This idea gave us more flexibility while testing SWIFT without having to wait for the battery to fully charge, as we wanted to simply test without having to manually connect each enclosure or to remove any of the components to test externally. This idea also helped us increase our battery life by decreasing the load on it.

C. *Software*

We focused on keeping our system modular. To do so we designed a software architecture diagram as seen in Fig. 10, and then started developing our software as ROS2 packages to maximize flexibility and maintain high reliability and safety.

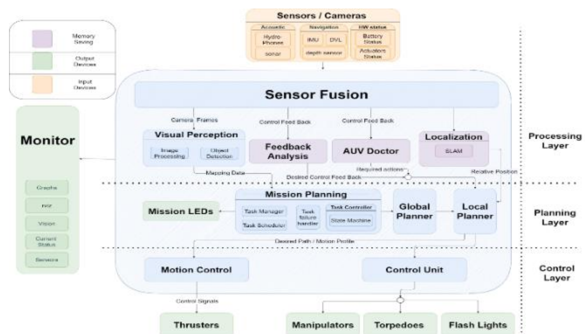

Figure 10 Software architecture

1) *Computer Vision*

The computer vision algorithms enable the AUV to identify objects while working on its underwater missions. The process starts with a video stream from the cameras, which are fed into the visual perception module in the form of raw frames.

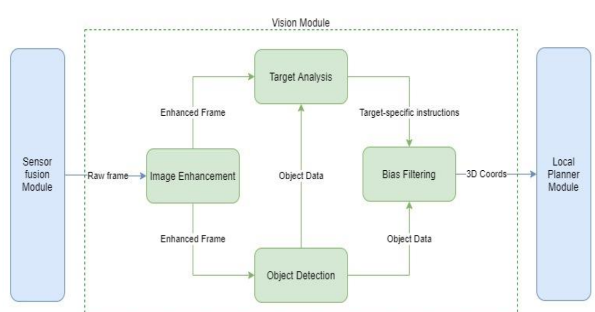

Figure 11 Vision module

a) Image enhancement: The image enhancement node within the visual perception module utilizes a combination of computer vision algorithms to improve the quality of the raw images, such as flattening the image histogram and cumulative curve, converting the RGB-normalized image to HSV, and applying histogram stretching to the S and V channels before converting back to RGB, as shown in Fig. 12. [2] [3]

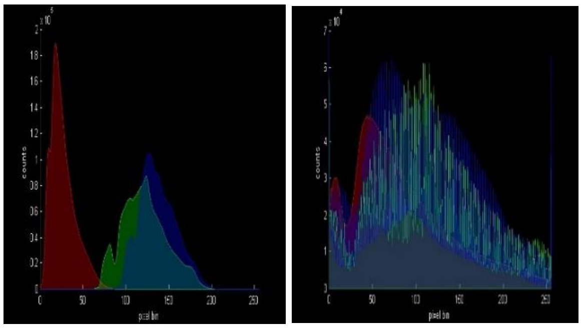

Figure 12 Image enhancement with image channels manipulation

Another functional component to improve the image enhancement was implementing a GAN model [4], which is well-trained on hundreds of different underwater image noises and problems and how to fix and enhance the images, as shown in Fig. 13 and Fig.14. Then comes the last function that enhances the images, the denoising autoencoder [5] [6]. Through tweaking the image properties, the D-autoencoder model will remove any noises in the raw image.

Figure 13 Image enhancement with GAN model

Figure 14 Image enhancement with GAN model [4]

With the power of the 3-image enhancement algorithms combined, we can ensure very well that the incoming raw image will be passed to the object detection in its most optimal form, significantly increasing the chance of the object being identified.

b) Object detection: The object detection node uses the enhanced image to output the detected objects IDs, their confidence thresholds, 2D coordinates, and depth, through the Yolov5l model [7] as trained by the team. Examples of detections are shown in Fig. 15 and Fig. 16.

Figure 15 Gate detected under bad circumstances

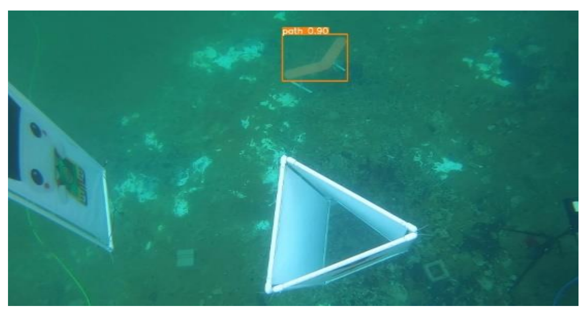

Figure 16 Path detected with 0.90 confidence

The well-trained object detection model alone was not enough for all cases, but the addition of the image enhancement algorithms made possible for successfully detecting the objects under most circumstances. In Fig. 17, the object detection only succeeded at detecting the gate when the image enhancement algorithms were applied.

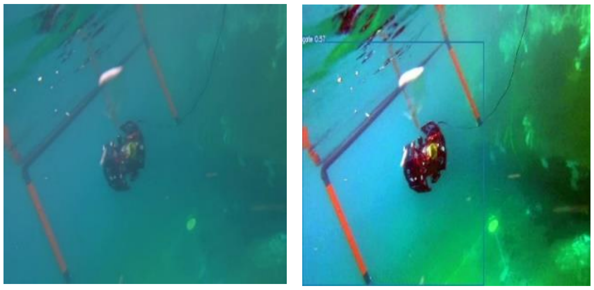

Figure 17 Object detection results before and after applying the image enhancement algorithms

c) Target analysis: When the object is successfully detected, the target analysis node takes the enhanced frame and the object data and starts analyzing the region of interest, producing target-specific instructions. For example, the target analysis node can identify the object's orientation to SWIFT, which will help SWIFT align itself to the object before approaching it. As shown in Fig. 18, approaching the gate without the target analysis process would result in SWIFT colliding with the gate pipes.

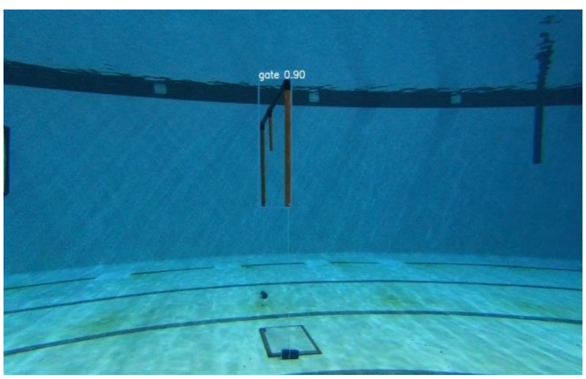

Figure 18 Bad orientation of the AUV to the gate that will be solved using the target analysis node

The target analysis node identifies the gate's orientation to the object by considering the lengths of the side pipes and the angle of the horizontal pipe, as well as comparing the distances of between the middle and left pipe and the middle and right pipe. This process is done using image processing and k-means clustering. After identifying the orientation, the Local Planner module receives 3D coordinates instructed by the target analysis node to align itself ahead of the object.

d) Biased coordinates filtering: Before any coordinates are passed to the local planner module, whether of object's coordinates or the desired point for aligning the AUV to the object, they are first passed to the Bias Filtering node, which filters any biased readings/coordinates, by comparing the current coordinates with the previous coordinates, and checking whether there is a 60% similarity, as if so, the coordinates are confidently passed to the Local Planner module.

2) Localization: We decided to use an open-source software package to develop our sensor fusion system "the robot localization ROS package" [8], as it supports fusion of an arbitrary number of sensors and Per-sensor input customization, with adaptations made in the configuration files to extend its use for SWIFT in the 3D space, so we could discover different implementations of Kalman filters EKF, ESKF and UK.

Combining velocity data from DVL and Z position from the pressure sensor, we decided to use the EKF filter as it provides faster processing than UKF with a slight reduction in accuracy.

For future improvements, we considered developing a SLAM algorithm based on obstacle avoidance sonar.

3) Motion Control: many researches, discussions, and brainstorming, we decided to use the Non-linear model predictive control NMPC in SWIFT. Based on shen [9] the most adequate control technique for the trajectory tracking control problem is the Model predictive controller due to its ability to handle both state constraints and actuators constraints, which are ubiquitous for the different types of control techniques like the PID, Nonlinear PID, and LQR.

To design an NMPC-based tracking controller, we considered both dynamics and kinematics of SWIFT, but this comes to the computational complexity of each MPC iteration. Based on shen [9] to overcome this problem we had to distribute the optimization problem of the MPC to maneuver and depth sub-problems, this solution is very effective as long as we drive at low-speed due to the coupling between SWIFT's states.

SWIFT's autopilot consists of the following components as seen in Fig. 19:

- Waypoints database: Known in advance or generated on the fly by the objects detection system.
- Trajectory generator: Converts the discrete set of waypoints to a time-varying continuous function with its higher order derivatives; these derivatives are proportional to the vehicle velocity and acceleration.
- Reference model: Uses the generated path parameters and its derivatives along with SWIFT's state and generates a feasible reference state for the MPC to track.
- MPC: receding horizon optimizations, which has the space to solve both the path planning and the trajectory tracking problems.

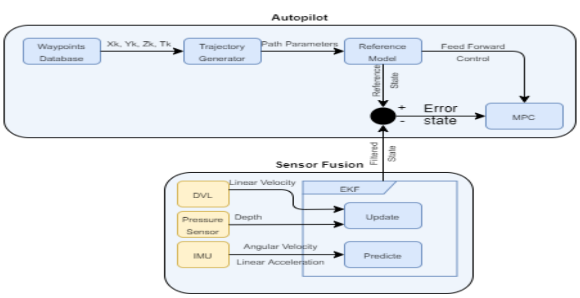

Figure 19 Motion control and Sensor fusion architecture

4) *Hydrophones:* Because of its importance we decided to give great attention to the localization task our system based on making best use of the sound characteristics so that we made use of the sound Speed our calculations based on the time it takes the sound waves to travel from point A to point B so we placed the hydrophones as distant as possible so we can get the highest accuracy and most reliable results, Using 4 Hydrophones those located at the corners of the AUV and with sampling rate of 200kHz we continuously scan the surroundings and divide the collected data into frames then using Fourier Transform[10] we could find the exact frame that contains the desired frequency then using Cross-Correlation function[11] we can estimate the time delay of arrival between any pair of Hydrophones or which is known for short as TDOA

The gathered data is then processed accordingly to determine the Angle of Arrival (AoA).[12]

Combining the data from two microphone pairs and by using the process of triangulation, we eventually compute the exact distance and direction of the Sound Source.

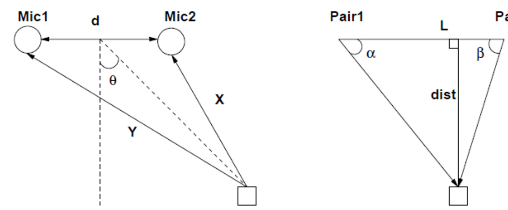

Figure 20 Angel estimation; Distance estimation

III. Experimental Results

A. Enclosures testing

Theoretically, the sealed enclosures were designed to work under a pressure up to 100 meters of water (10 bars). The Solidworks stress analysis results show that the enclosures are safe and no deflection occurs.

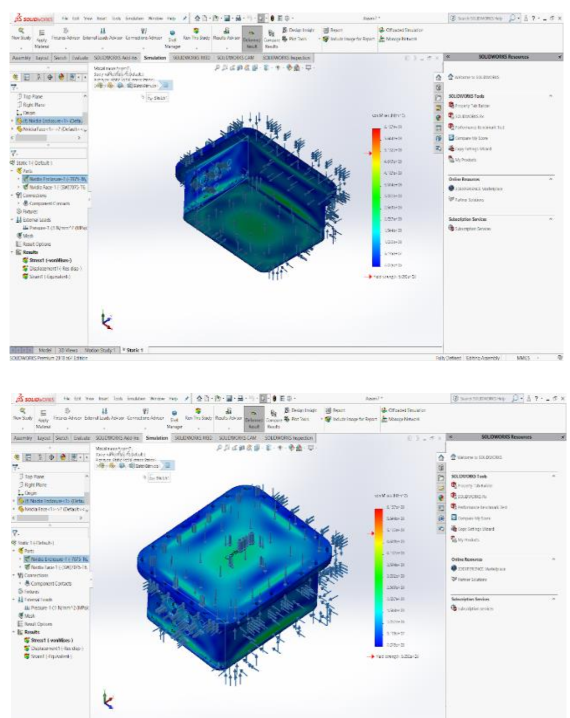

Figure 21 Control enclosure stress analysis

Experimentally, we tested the enclosures under a pressure of 7 bars.

<u>Procedures:</u>

- Safety first, wearing gloves, eye safety glasses, and of course the face masks while testing.

Figure 22 Safety precautions

- Assemble the enclosures by fastening the face screws, placing Orings, and fastening the connectors.
- Plugging the air hoses in the connectors which are connected to the air compressor.
- Turning the compressor on and raising the pressure up to 7 bars.

Figure 23 Enclosure testing; Compressor pressure regulator

B. CFD Simulation

Using ANSYS, we made this study at a velocity of 1 m/s as maximum velocity. We made sure that the minimum resulting thrust force equals the maximum drag force on the AUV Seeking stability. We made some changes in the cover design to keep the center of drag (center of pressure) and the center of gravity as aligned as possible.

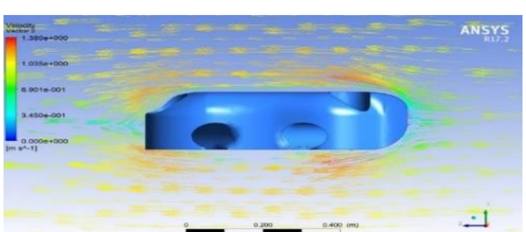

Figure 24 Flow analysis using ANSYS

C. Testing copper Trace Endurance

During development, we tested our PCBs to know the maximum current limit, this destructive test was done on 15VDC and using Load blocks of various values that we added in parallel to draw the required current for testing, we discovered that:

At 44.5A, a 2.2 mm width single layer trace was burned as seen in Fig.25, at 50A, there was a very high temperature, at 55A, there was a color change on a 4.4 mm width and 30 Microns thickness trace as seen in Fig. 26, and at 65A, the soldering came off at the highest temperature in the circuit as seen in Fig. 27.

As a result, we enlarged the traces in our propulsion PCB into 6 mm width and 120 Microns.

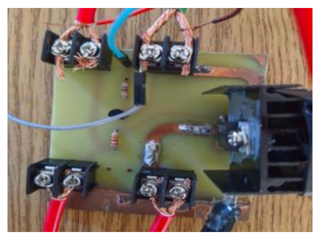

Figure 25 soldering testing pcb

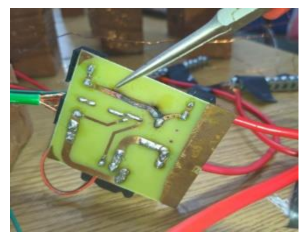

Figure 26 Color changed at 55A

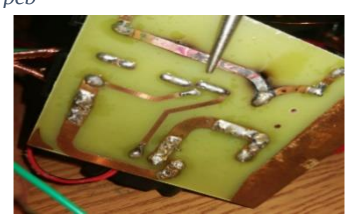

Figure 27 the soldering came off at 65A

D. Motion control testing

The response of our NMPC controller for tracking a linear x-path and a sinusoidal y-path is shown in the Fig. 28

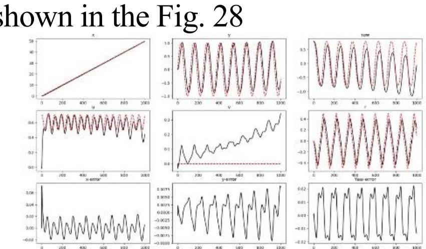

Figure 28 [reference in dashed-red, actual state in black]

ACKNOWLEDGEMENTS

Vortex would like to express gratitude to everyone who has supported us along SWIFT's journey. Without their assistance, Vortex would not have been able to compete in Robosub 2021. We would like to thank Seasplit for their monetary contributions and our logistics sponsor Arab Academy for Science, Technology & Maritime Transport. We received engineering support from Water Linked [13], Aquarian [14], Blue Robotics, and Blue Trail. We are grateful to our technical advisor Abdelrahman Elmaradny. We appreciate RoboSub organizers for putting the competition together to challenge us, and special thanks to everyone who made it easier for us along the road either in software licenses, simulation, or hardware components. Lastly, we are indebted to our family and friends for their continuous support.

REFERENCES

[1] Water Linked, "Waterlinked WL-11007 DVL A50," Aug. 3, 2020, https://waterlinked.com/wp-content/uploads/2020/08/W-MK-20001-3_DVL_A50_Datasheet.pdf (accessed Nov. 6, 2020).

[2] Nickloomis, (2015, September 11). Automatic Color Correction: Underwater Photos. Adventures in Loom-Science. https://loomsci.wordpress.com/2013/06/20/automatic-color-correction-underwater-photos/. accessed Dec. 1, 2020).

[3] Schettini, R., & Corchs, S. (2010, April 21). Underwater Image Processing: State of the Art of Restoration and Image Enhancement Methods. EURASIP Journal on Advances in Signal Processing.https://aspeurasipjournals.springeropen.com/articles/10.1155/2010/746052#Sec5. accessed Dec. 12, 2020).

[4] Xahidbuffon. (2020). xahidbuffon/FUnIE-GAN. GitHub. https://github.com/xahidbuffon/FUnIE-GAN (accessed Jun. 10, 2021

[5] Deep Learning. Introduction to autoencoders · Deep Learning. (2020, March 10). https://atcold.github.io/pytorch-Deep-Learning/en/week07/07-3/. accessed Jun. 10, 2021).

[6] Atcold. (2020 ,March 18). Atcold/pytorch-Deep-Learning. GitHub. https://github.com/Atcold/pytorch-Deep-Learning/blob/master/10-autoencoder.ipynb. accessed Jun. 6, 2021).

[7] Ultralytics. (2020). ultralytics/yolov5. GitHub. https://github.com/ultralytics/yolov5. accessed Feb. 6, 2021).

[8] Moore, T. (2020). *Robot Localization*. ros.org. http://wiki.ros.org/robot_localization.

[9] Shen, C., Shi, Y., & Buckham, B. (2016). Nonlinear model predictive control for trajectory tracking of an AUV: A distributed implementation. 2016 IEEE 55th Conference on Decision and Control (CDC), 5998-6003.

[10] ST. (2018). *Digital signal processing for STM32 microcontrollers using CMSIS*. ST. https://www.st.com/resource/en/application_note. (accessed Apr. 20, 2021)

[11] Bourke, P. (1996). Cross Correlation. http://paulbourke.net/miscellaneous/correlate/.(accessed Apr. 20, 2021)

[12] Umbarkar, Anurag & Subramanian, Varun & Doboli, Alex. (2010). Improved sound-based localization through a network of reconfigurable mixed-signal nodes. 1 - 6. 10.1109/ROSE.2010.5675318.

[13] Water Linked, https://waterlinked.com (accessed Oct. 15, 2020).

[14] Aquarian Audio & Scientific, https://www.aquarianaudio.com (accessed Oct. 15, 2020).

[15] Kristoffer Rakstad Solberg, "Manta v1: A Deliberative Agent Software Architecture for Autonomous Underwater Vehicles" Master's thesis in Engineering and ICT, Jan. 2020

Appendix A: Component Specifications

Component	Vendor	Model/Type	Specs	Cost	Status
Buoyancy Control	N/A	N/A	N/A	N/A	N/A
Frame	Locally manufactured	N/A	Material: Fiber	$128	installed
Waterproof Housing	Locally manufactured	N/A	Material: Aluminum Depth rating: 7 bars (70 meters)	$575 (For 9x Enclosure)	installed
Waterproof Connectors	Blue Trail engineering	Cobalt Series Bulkhead Connector	Material: 316 stainless steel Depth rating: 600 meters	$1680 (For 40x Bulkhead Connector)	installed
Thrusters	Blue Robotics	Blue Robotics T200 Thruster	https://www.bluetrailengineering.com/product-page/blue-robotics-t200-thruster-with-cobalt-connector	$1800 (For 8x T200 Thrusters)	installed
Motor Control	Blue Robotics	Basic ESC	https://bluerobotics.com/store/thrusters/speed-controllers/besc30-r3/	$216 (For 8x Basic ESCs)	installed
High Level Control	Pixhawk	Pixhawk 4	https://docs.px4.io/master/en/flight_controller/pixhawk4.html	$250	installed
Actuators	Blue Trail Engineering	ser - 110X	depth rating 100 meter Operating voltage 4.8 : 7.4 V Torque: 18 : 29 kg.cm Travel angel 74 degree	380$	installed
Propellers	Blue Robotics	T200 Propellers	https://www.bluetrailengineering.com/product-page/blue-robotics-t200-thruster-with-cobalt-connector	Included with thrusters	installed
Usb hub	TP-Link	TP-Link USB 3.0 to Ethernet Adapter	Number of Ports 3 Item Dimensions LxWxH 3.78 x 1.22 x 0.91 inches	$40 (For 2x USB)	installed
Switch	Blue Robotics	Switch	https://bluerobotics.com/store/comm-control-power/switch/switch-10-5a-r1/	$60 (For 4x switches)	installed

Battery	Lumenier	Lumenier 22000mAh 4s 20c Lipo Battery	https://www.getfpv.com/lumenier-22000mah-4s-20c-lipo-battery.html	$160	installed
Battery Charger	Blue Robotics	Lithium Battery Charger	https://bluerobotics.com/store/comm-control-power/powersupplies-batteries/lithium-battery-charger/	$150	purchased
Converter	N/A	N/A	N/A	N/A	N/A
Regulator	ACEIRMC	BUCK (12V)	https://www.amazon.com/Aceirmc-Converter-Adjustable-Regulator-Protection/dp/B0823MM1DV/ref=sr_1_18?dchild=1&keywords=buck+converter&qid=1602871023&sr=8-18	$96 (For 8x regulators)	installed
Regulator	ACEIRMC	BUCK (5V)	https://www.amazon.com/Adjustable-Converter-1-25-36v-Efficiency-Regulator/dp/B079N9BFZC/ref=psdc_10967761_t3_B0823MM1DV	$13	installed
CPU	NVIDIA	Jetson AGX Xavier Developer Kit	https://developer.nvidia.com/embedded/jetson-agx-xavier-developer-kit	$703	installed
Internal Comm Network	N/A	CAN and UART	N/A	Priceless	installed
External Comm Interface	N/A	Ethernet	N/A	Priceless	installed
Lumen Subsea Light	Blue Robotics	Lumen Subsea Light (Pre-Connected Sets)	https://bluerobotics.com/store/thrusters/lights/lumen-sets-r2-rp/	$800 (For 2x lumen)	installed
Compass	Pixhawk	Pixhawk 4	Magnetometer: IST8310	Included with Pixhawk	installed
Inertial Measurement Unit (IMU)	Pixhawk	Pixhawk 4	Accel/Gyro: ICM-20689, Accel/Gyro: BMI055	Included with Pixhawk	installed

Doppler Velocity Log (DVL)	water linked	DVL A50	https://waterlinked.com/product/dvl-a50/	$6290	installed
Sonar	Blue Robotics	Ping360 Scanning Imaging Sonar	https://bluerobotics.com/store/sensors-sonars-cameras/sonar/ping360-sonar-r1-rp/	$1975	installed
Additional hardware to support the Hydrophones	Aquarian hydrophones	Aquarian PA4 preamp	https://www.aquarianaudio.com/pa4.html?variation_id=95	$59.99	installed
Additional hardware to support the Hydrophones	STM	STM32-H747I-DISCO	https://www.st.com/en/evaluation-tools/stm32h747i-disco.html#overview&secondary=st-featured-products	$31.93	installed
Hydrophones	Aquarian hydrophones	AS-1 HYDROPHONE	https://www.aquarianaudio.com/as-1-hydrophone.html	$1580 (For 4x hydrophones)	installed
Pressure & Temperature sensor	Blue Robotics	Bar30 High-Resolution 300m Depth/ Pressure Sensor	https://bluerobotics.com/store/sensors-sonars-cameras/sensors/bar30-sensor-r1/	$72	installed
Leak Sensor	Blue Robotics	SOS Leak Sensor	https://bluerobotics.com/store/sensors-sonars-cameras/leak-sensor/sos-leak-sensor/	$120 (For 4x SOS Leak Sensors)	installed
Vision	Stereo Labs	Zed2 Camera	https://store.stereolabs.com/products/zed-2?_ga=2.214446167.1152906347.1601928193-782092441.1599849388	$500	installed
Camera	Blue Robotics	Low-Light HD USB Camera	https://bluerobotics.com/store/sensors-sonars-cameras/cameras/cam-usb-low-light-r1/	$300 (For 3x Low-Light HD USB Cameras)	installed
Manipulator	locally manufactured	N/A	laser cutting acrylic + 3D printed parts	$31.92	installed
Algorithms: vision	OpenCV	Custom	Histogram stretching for image channels by a specific value (1% and	Free	installed

			99% of the cumulative curve cutting), Biased detections filtering, Finding object orientation to AUV, Kmeans clustering		
Algorithms: vision	Open-Source	GANs, Denoising auto-encoders, Yolov5L	N/A	Free	installed
Algorithms: acoustics	In-house	Custom	Fast Fourier Transform (FFT) Generalized Cross-Correlation (GCC)	Free	installed
Algorithms: localization and mapping	Open-Source	Robot Localization ROS package	Extended Kalman Filter	Free	installed
Algorithms: autonomy	In-house	Custom	Local and Global planner (state machine)	Free	installed
Open-source software	Open-Source (n/a)	Open Computer Vision, Robot Operating System, Python, C++, Linux	Computer Vision, Inter-process communication, programming, computer operating system	Free	installed
Team Size (number of people)	30	undergraduates and graduates	undergraduates, fresh graduates	Priceless	N/A
HW/SW expertise ratio	12/19	Hardware sub teams: mechanical, electrical Software sub teams: motion control, sensor fusion, computer vision	N/A	Priceless	N/A
Testing time: simulation	50	Software sub teams: Motion control, Sensor fusion	N/A	N/A	N/A

Testing time: unit testing	145	Enclosures, PCBs	N/A	N/A	N/A
Inter-vehicle communication	N/A	N/A	N/A	N/A	N/A
Programming Language 1	C++	N/A	N/A	Free	installed
Programming Language 2	Python	Python 3	N/A	Free	installed

Appendix B: Outreach Activities

Vortex competes in MATE ROV every year, with new teams of young enthusiasts that we mentor, passing them our underwater robotics experience to help them reach their goals. Vortex competes in Seaperch regional competition at the Arab Academy for Science, Technology, and Maritime Transport in Egypt, in preparation for the Seaperch 2021 competition, and so Vortex won the first place.

Figure 29 Seaperch Team

Vortex participated in the ROV day in Arab Academy for Science, Technology & Maritime Transport to demonstrate what is ROV to City International School - CIS Students.

Figure 30 CIS Students