Design of Semi-Autonomous Underwater Vehicle Platform using Robot Operating System (ROS) for Marine Robotics Research

line 1: 1st Given Name Surname
line 2: *dept. name of organization (of Affiliation)*
line 3: *name of organization (of Affiliation)*
line 4: City, Country
line 5: email address or ORCID

line 1: 2nd Given Name Surname
line 2: *dept. name of organization (of Affiliation)*
line 3: *name of organization (of Affiliation)*
line 4: City, Country
line 5: email address or ORCID

line 1: 3rd Given Name Surname
line 2: *dept. name of organization (of Affiliation)*
line 3: *name of organization (of Affiliation)*
line 4: City, Country
line 5: email address or ORCID

line 1: 4th Given Name Surname
line 2: *dept. name of organization (of Affiliation)*
line 3: *name of organization (of Affiliation)*
line 4: City, Country
line 5: email address or ORCID

line 1: 5th Given Name Surname
line 2: *dept. name of organization (of Affiliation)*
line 3: *name of organization (of Affiliation)*
line 4: City, Country
line 5: email address or ORCID

line 1: 6th Given Name Surname
line 2: *dept. name of organization (of Affiliation)*
line 3: *name of organization (of Affiliation)*
line 4: City, Country
line 5: email address or ORCID

***Abstract*—In the marine robotics research, individuals face large development overhead. This paper discusses the development of a semi-autonomous underwater vehicle (Semi-AUV) that combines the features of autonomous under-water vehicles (AUV) and remotely operated vehicles (ROV). We contribute open-source hardware and software plans and documentation. The platform can be configured with multiple sensors and cameras that use visual-simultaneous localization and mapping (V-SLAM) technology. Additionally, the platform is easy to modify to fit the use case of several fields, including but not limited to underwater research, robotics research, machine learning-based controllers, and ecological and environmental assessment. The vehicle runs the open-source Robot Operating System (ROS) framework enabling easy software development including software-in-loop modelling with Gazebo.**

Keywords—Underwater Robotics & Vehicles, AUV, ROV, V-SLAM, Robot Operating System, Software-in-Loop Modelling.

I. Introduction

Underwater robotic vehicles play a major role in environmental, commercial, military, and emergency operations. Underwater robotic vehicles can be divided into two groups: manually operated, and autonomous, both with their different applications [1]. In recent development, Robot Operating System (ROS) [2] expanded rapidly through community support to include software libraries, simulation platforms, and compatible hardware components to facilitate robotics development. However, the wide-reaching base of available ROS and similar software, in addition to the difficulties of developing underwater-compliant robotic systems, means that the barrier of entry into marine robotics needs reduced. This paper aims to propose a method that combine the advantages of the manual and autonomous vehicles with a low overhead, while applying the open-source ROS framework. This hybrid vehicle should be able to produce maps of maritime life underwater and analyze their ecological status using use visual-simultaneous localization and mapping (V-SLAM) technology [3], and other object detection technologies. The project also includes designing an integrated underwater vehicle simulation platform through which we can test command and control systems in various marine environments such as high currents and polluted environments to avoid the problems that the submarine may encounter in the working environment, and this feature is needed by everyone who works in the field of autonomous vehicles. On the commercial level, the Semi-AUV "Fig. 1" will be able to conduct survey patrols of the depths of water. The vehicle qualifies it to carry out industrial operations using the automatic or autonomous command system.

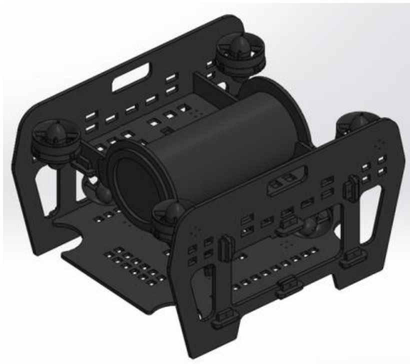

Fig. 1. Design of Semi-AUV formed on SOLIDWORKS CAD.

A. Related work

Creating an underwater robotics research platform involves significant overhead, works exist that aim to reverse engineer existing vehicles to support open-source software [4] [5] [6]. Additionally, some systems achieve autonomy but rely on expensive commercial platforms [7]. Open-source platforms are scarce and out-of-date or have incomplete instructions [8] [9]. Other open-source platforms are targeting low-cost without considering additional devices payload for underwater robotics research [10] [11] [12]. Manual-control

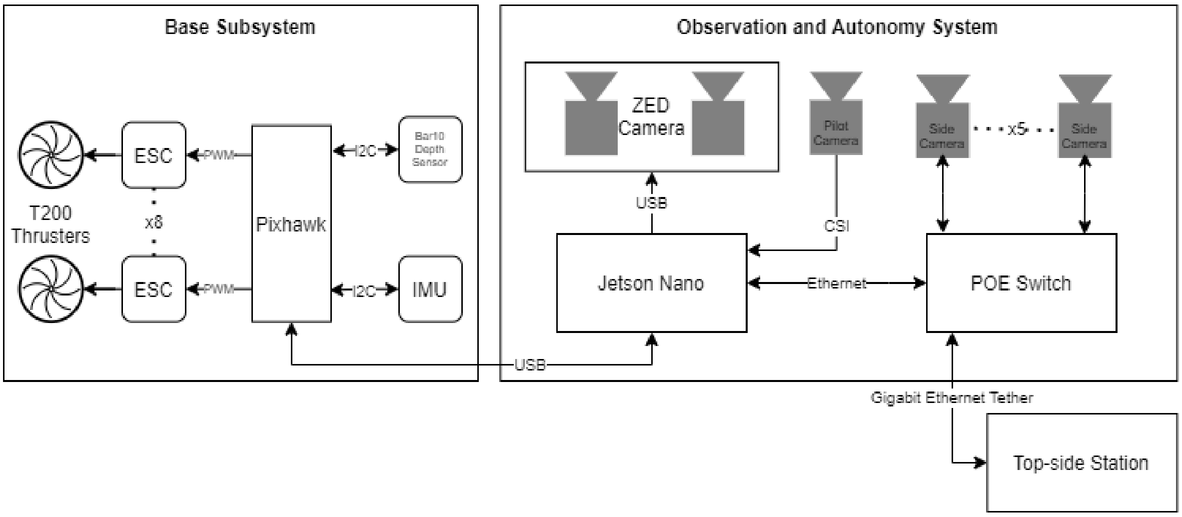

Fig. 2. Diagram of the data flow inside the Semi-AUV. The base manually operable system (left), and the autonomy hardware and camera system (right). All connected via Gigabit Ethernet to the top-side station.

only platforms offer additional examples of the mechanical and electrical design of underwater robotics [13].

B. Contributions

We contribute full design plans for an easily manufactured marine robotics research platform, excluding thrusters. The design uses readily available materials and manufacturing processes (wood routers, laser-cutters, metal work). Also included are the electrical system diagrams, and software tools used. In particular we contribute the following:

- Software setup, ROS software libraries, documentation of hardware-software considerations.
- Documentation of helpful websites.
- Electrical system components, power circuit schematics, inter-system communication and tether.
- Mechanical system with body plans and CAD files.

The open-source documentation and files are available at:

https://github.com/Walid-Rovo/Semi-AUV

II. System Design

While designing the Semi-AUV the main aim was to reach an underwater robotics research platform. The design is separated into three disciplines: mechanical, electrical, software. The mechanical design facilitated interlocking acetal sheets, and acrylic tubing for insulation. The electrical system consists of the power circuit, and the data circuit shown in "Fig. 2 ". Power is supplied via the tether with optional batteries on-board for verifying vehicle dynamics and software independent of the tether. The software is based on a collection of open-source libraries under the ROS framework with access to software-in-loop simulations in Gazebo.

A. Hardware

The system is designed from the ground-up for ease of manufacturing and modularity for research. The system considers local alternatives to importing systems from companies like BlueROV to reduce cost and decrease financial barrier of entry. Various make-decisions were made such as with manufacturing the main electronics insulated enclosure from acrylic tubing and metal flanges made at a metal workshop.

B. Sensors and computing

The sensors and computing hardware is consisted of incorporates ZED 2K stereographic camera, 3 IP Cameras that provide clear vision for sides, bottom and front of the vehicle. In addition to an anti-vibration IMU sensor that locates the orientation of the vehicle, as well as Blue Robotics BAR03 pressure sensor which provides precise identification for the depth of the vehicle.

The system is connected to the Nvidia Jetson Nano which has been selected for its AI processing capabilities.

III. Autonomy & Control Software

The control system of the Semi-AUV is ROS based, where the system consists of a set of "Nodes". Each node represents a single process that can then communicate with other nodes. Nodes are a convenient method to develop software for each subsystem in the vehicle.

Different ROS nodes communicate together and exchange topics to perform the desired mode of operation. In the Semi-AUV case, depth and image data received from the ZED camera, and other cameras installed on vehicle, in addition to IMU sensors, set the intelligent action of the vehicle. The control computer commands the real-time Pixhawk controller that provides signals to thrusters to perform the maneuvering decision. This system allows for expandability and adaptability as most microcontrollers can be used for the base system through libraries such as "rosserial" for Arduino-supported devices.

A. V-SLAM Technology

The SLAM can be defined as follows: given the robot's controls U, the observations of the world Z determine the map of the environment M, and the robot's pose X.

And in the probabilistic world as every element exhibits some errors, it can be expressed as in "(1)".

$$p(x_{0:T}, m \mid z_{1:T}, u_{1:T}) \tag{1}$$

It can also be represented as a flow diagram as in "**Error! Reference source not found.**".

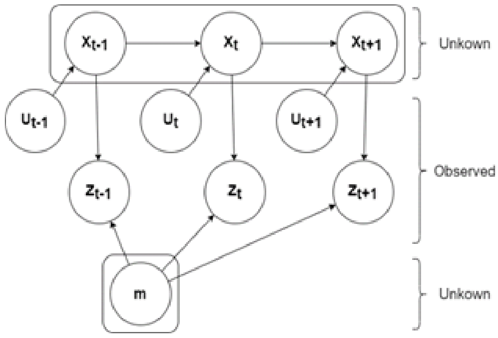

Fig. 3 The flow diagram of typical SLAM system. It's constitutes are system state (X), odometry (U), observations (Z), and environment map (m).

Upon the general idea of SLAM technology, ZED 2K Stereoscopic Camera convert the visual data into a point cloud. The "rtab_map" ROS library is used to perform pathing and add further SLAM capabilities.

B. PID Hovering

The aim of this function is to keep the vehicle vertically and horizontally stable as much as possible by using the Blue Robotics pressure sensor to calculate the depth & orientation of the vehicle, which gives the pilot an advantage when doing several in the ROV mode - tasks on the same depth by changing the desired depth on the graphical user interface (GUI) input box. Using "(2)" pressure can be easily converted to depth where P denotes pressure, ρ denotes density, g denotes gravity's acceleration, and h denotes depth.

$$P = \rho * g * h \tag{2}$$

For the orientation of the vehicle, the IMU provides clear data for the ROS system to identify the Semi-AUV status underwater, this is maintained within the IMU ROS node using the PID control method that provides stable and smooth control upon the desired sampling rate. Flowchart "Fig. 4" declares the mode of operation of the IMU ROS node.

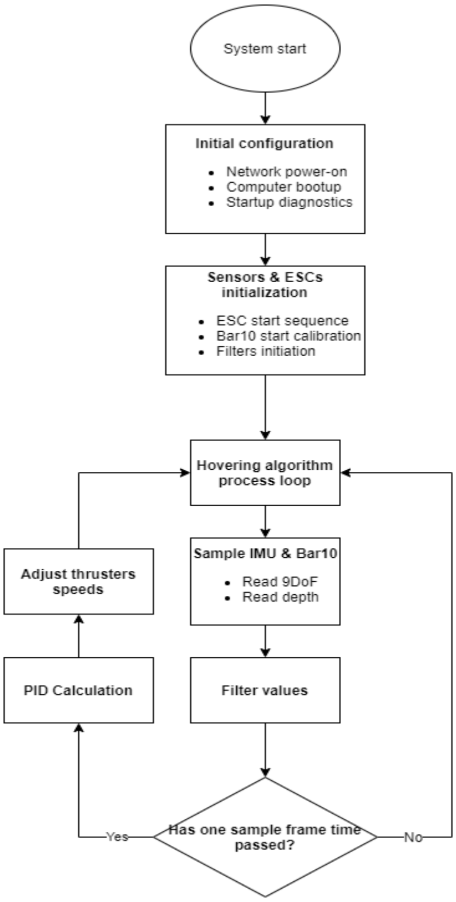

Fig. 4. PID hovering Flowchart

C. Software In-loop simulation

A simulation model can be very useful in the process of designing control systems. The control system can be tested on a simulation model, which is both less expensive and simpler than testing the control system in the actual process. Since the disturbances and environment can be easily changed in a simulation model, the system's stability can be easily verified. The main challenge in such a system is to make the simulation model as realistic as possible. The final force operating out of the water on the vehicle is the product of many factors. and the most critical ones should undoubtedly be considered when constructing a simulation model [14].

The goal is to develop the foundations of a ROV/AUV simulation platform. Two primary elements, a simulator, and a control system, will consist of the platform. Dynamic ROV simulations will be provided by the simulator, including versions of the various sensors.

The control system will be the ROV control program which will contain: an estimator, a path planner, a guidance system, and a controller. The aim is that, when the software is used on the real vehicle, the simulation platform will provide a framework for evaluating control software without significant modifications necessary [15].

Without manually testing the vehicle in service, the platform can be used for testing applications for vehicle operations and missions. It will be tested on a with simulated sensor performance during the production of the control system. The program will be checked with the individual ROV in the loop when the production is completed. Based on thruster inputs and environmental parameters, the simulation

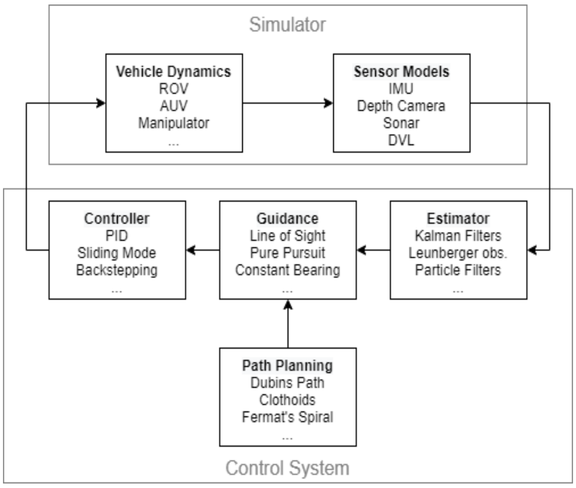

Fig. 5. Block diagram showing components of control system inside the software in-loop simulator

platform must be able to replicate ROV motions. This requires a practical calculation of the hydrodynamic forces acting on the ROV, as well as a realistic estimate of the thrust forces for each propeller for a given number of revolutions per minute (RPM).

In the control system portion of the simulation framework, the necessary components for the simulation of underwater robotics systems must be used, including sensor models with all possible add-ons “Fig. 5” such as sonar and Doppler velocity logs (DVL), an estimator, a route planner, a guidance system, and a controller. The framework is structured to be module-oriented, with each module performing a specific function. Modules interact with one another, and one module may be replaced with another of the same kind if necessary. If a user wants to test a new controller algorithm, they can do so by replacing the controller module on the simulation board with another controller module.

IV. Conclusion

We have showcased a marine robotics research platform with provided open-source resources and documentation on hardware and software. The platform is modular and easy to manufacture. The platform should help lower barrier of entry into underwater robotics. As a research application we explored V-SLAM using ZED module, and Software-in-Loop modelling using Gazebo with ROS. The platform was deployed as a Semi-AUV with two modes of operations. Further development and documentation of our methods and of sensor data analysis and comparisons will follow.

Acknowledgment

The authors wish to thank the Academy of Scientific Research and Technology, Arab Academy for Science and Technology, and Underwater Robotics Research Center for their unlimited financial and motivational support and guidance.

References

[1] A. Tahir and J. Iqbal, "Underwater robotic vehicles: Latest development trends and potential challenges," *Science International*, vol. 26, pp. 1111-1117, 2014.

[2] M. Quigley, K. Conley, B. Gerkey, T. F. J. Faust and J. Leibs, "Ros: an open-source robot operating system," in *ICRA workshop on open source software, vol. 3, no. 3.2.*, Kobe,Japan, 2009.

[3] Manzanilla.A, Reyes.S, M. Garcia, Mercado.D and L. R., "Autonomous navigation for unmanned underwater vehicles: Real-time," *IEEE Robotics and Automation letters*, vol. 4, p. 1351–1356, 2019.

[4] Terrill, E. Gallimore, R. Stokey and E., "Robot Operating System (ROS) on the REMUS AUV using RECON," *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, pp. 1-6, 2018.

[5] K. DeMarco, M. E. West and T. R. Collins, "An implementation of ros on the yellowfin autonomous underwater vehicle (auv)," *OCEANS IEEE*, pp. 1-7, 2011.

[6] N. Lawrance, R. DeBortoli, D. Jones, S. McCammon, L. Milliken, A. Nicolai, T. Somers and G. Hollinger, "Shared autonomy for lowcost underwater vehicles," *Field Robotics*, vol. 36, no. 3, pp. 495-516, 2019.

[7] Wilby, Antonella and E. Lo, "Low-Cost, Open-Source Hovering Autonomous Underwater Vehicle (HAUV) for Marine Robotics based on BlueROV2," in *OES Autonomous Underwater Vehicles Symposium (AUV)(50043)*, San Diego, California, 2020.

[8] M. Dunbabin, J. Roberts, K. Usher, G. Winstanley and P. Corke, "A Hybrid AUV Design for Shallow Water Reef Navigation," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 2105-2110, 2005.

[9] Prats, Mario, Romagós, D. Palomeras, N. Sánchez, J. Carlos, Nannen, V. Wirth, S. Fernandez, J. Beltrán, J. Campos, R. Rida, P. S. P. Oliver, G. Carreras, M. Gracias and N. M. Pra, "Reconfigurable AUV for intervention missions: A case study on underwater object recovery," *Intelligent Service Robotics*, vol. 5, pp. 19-31, 2012.

[10] R. Codd-Downey, M. Jenkin and K. Allison, "Milton: An open hardware underwater autonomous vehicle," 2017.

[11] Joordens and Matthew, "Design of a low cost underwater robotic research platform," *2008 IEEE International Conference on System of Systems Engineering, SoSE 2008*, pp. 1-6, 2008.

[12] Hostler, A. a. Benson, B. a. Warner and Joshua, "BudgetROV: An ultra low cost robotics platform for education and research," *OCEANS*, pp. 1-4, 2017.

[13] B. M. M. Anwar, M. A. Ajim and S. Alam, "Remotely operated underwater vehicle with surveillance system," *2015 International Conference on Advances in Electrical Engineering (ICAEE)*, pp. 255-258, 2015.

[14] M. M. M. Manhães, S. A. Scherer, M. Voss, L. Douat and T. Rauschenbach, "UUV Simulator: A Gazebo-based Package for Underwater Intervention and Multi-Robot Simulation," OCEANS’16 MTS/IEEE Monterey, Monterey, 2016.

[15] S. M. Mo, "Development of a simulation platform for ROV systems," NTNU, Trondheim, 2015.

[16] F. Eren, "Pose Detection and control of unmanned underwater vehicles -utilizing an optical detector array," University of New Hampshire, Durham, 2015.

[17] A. K. Osman, A. M. Abdelghany, A. A. Salah, P. Y. Yasser, S. A. Adel, S. M. M. ElBaroudy, Y. J. Mohamed, Y. O. A. Elmansy and Z. M. Hassan, "Building ROS-based Software for an Autonomous Underwater Vehicle (AUV)," Ain Shams University (Unpublished), Cairo, Egypt, 2019.