Towards an immersive and natural gesture controlled interface for intervention underwater robots

Juan C. García*, Bruno Patrão†, Javier Pérez*, João Seabra†, Paulo Menezes†, Jorge Dias‡ and Pedro J. Sanz*
*Department of Computer Science and Engineering,
Universitat Jaume I, Castellón de la Plana, Spain
Email: {garciaju, japerez, sanzp}@uji.es
†Institute of Systems and Robotics,
University of Coimbra, Coimbra, Portugal
Email: {bpatrao, jseabra, paulo}@isr.uc.pt
‡Khalifa University of Science and Technology and Research,
Abu Dhabi, UAE
Email: jorge@deec.uc.pt

***Abstract*—During underwater intervention missions, one of the critical issues, from the human-machine interaction perspective, concerns the operator's stress, which should be able to control the intervention system, mainly due to the high complexity of information displayed through specifically designed Graphical User Interface, and the usually available master-slave control architecture. So, this paper introduces a new approach with the aim to minimize the aforementioned drawbacks and, in the meantime, increasing the user's immersive feeling. Preliminary results are discussed highlighting the pros and cons of this novel procedure.**

I. INTRODUCTION

Currently, there is a large number of applications that can be carried out in underwater robot intervention. Some examples can be found in the oil and gas industry (i.e. operating submerged infrastructures), search and recovery missions (i.e. grasping a crashed airplane blackbox), deep water archeology or scientistic missions. Usually, the typical robot used in these missions is Remotely Operated Vehicle (ROV), which needs high economical and logistic requirements. In addition, these robots uses a master/slave architecture and relays all the responsibility in the pilot, who suffers cognitive fatigue and stress [1]. So, Human-Robot Interaction (HRI) plays a central role in underwater robotics intervention, where it is necessary to merge multiple capabilities, from perception to control the robot's actions. Besides this commercial systems, the evolution of this kind of robot, the Autonomous Underwater Vehicle (AUV), has started. Although this new technology reduces the user responsibility, adding some autonomous behaviors, the human is still the control loop (e.g. supervising the mission, specifying the mission). Nevertheless, due to the limited intervention capabilities (e.g. mapping, survey), the evolution of this robot is the Autonomous Underwater Vehicle for Intervention (I-AUV). The robotic arm endowed to the robot opens up a new intervention mission type with physical interaction (e.g. operate a valve, grasp an object).

Nevertheless, both kinds of intervention systems (ROV and AUV) still have problems related to the control of the robot and the way the user interacts with the system. The pilot uses different joysticks with several buttons to control the robot, which represents another drawback in sense of HRI. In both cases, the Graphical User Interface (GUI) to control the robot is focused to an expert user, who should be an expert in both the robot and mission to perform, due to the complexity of the different screens. Thus, the pilot should pay attention to different screens and control panels to get all the information about the robot, sensors and cameras. In the case of ROVs, there are more problems, such us the tethering system (this is, the cable which connects the pilot and the robot), communication delays and the limitations of technology that can be used. On the other hand, the user interfaces for AUV robots are also complex, but the main problem is the lack of feedback to the user, because there is no physical connection between the robot and the pilot.

The most relevant works about underwater GUI are related to the research projects DVECS (Distributed Virtual Environment Collaborative Simulator) [2], SAUVIM [3][4], TRIDENT [5][6], TRITON [7] and PANDORA [8]. In general, the DVECS and SAUVIM GUI were developed for an expert user. These interfaces were complex, depends on the robot hardware and do not present any type of interconnection with immersion or haptic devices. On the other hand, in TRIDENT and PANDORA projects, the HRI was developed taking into account that it could be used by a non-expert user [9] and does not rely on a specific robot platform.

Mostly all the AUV simulators developed in research projects [10], were platform dependent and access to the source code and collaborative development possibilities are normally very limited. In the case of I-AUV, is still more difficult to find support for underwater manipulators. Of course, there are also commercial simulators such as VortexSim [11], ROVsim [12], ROVolution [13] or DeepWorks [14]. However, these are normally addressed to ROV pilot training, they are closed, and thus, difficult to integrate with custom control architectures.

The remainder of this paper is organized as follows: sections 2 and 3 explain our proposal and the architecture used; sections 4 and 5 details how to get an immersion feeling and improve the user interaction. Finally, section 6 describes a preliminary experiments using our proposal and section 7 offers some conclusions.

978-1-4799-8736-8/15/$31.00 ©2015 IEEE

II. THE NEW APPROACH

Bearing in mind the aforementioned problems related to underwater robots, this paper presents a new approach focus on enhancing the way the pilot interacts with the system and increasing the GUI usability. The improvements we propose, can be divided in three different parts:

- Defining an abstraction layer. This layer will manage the input devices (joysticks, 3D mouse, gamepad). Each available device will be activated depending on the task to be carried out, including sensor data fusion capabilities. Moreover robot safety measures to prevent robot damages will be also included and, suitable information will be filtered to the user in real time.
- Getting an immersive feeling. The integration of the Inertial Measurement Unit (IMU) data of the Head-Mounted Display and the head tracking system data, allow the user to control the camera point-of-view (POV) just moving his/her head.A first advantage of working through such an immersive device is that avoid distractions, improving, in the meantime, the necessary concentration for enhance the human-machine interaction. It is worth mentioning, that in the field training underwater domain, it is crucial minimize the solar reflections effects, so working with this immersion capabilities this problem is overcome.
- Improving the user feedback. Instead of paying attention to multiple screens, our proposal reduces the information to be displayed to the user by filtering only the most relevant. This information will be displayed visually effective and intuitively in the developed virtual windshield.

III. ARCHITECTURE

In 2011, the Universitat Jaume I (Castellón, Spain) started to develop a new open source software tool for visualization and simulation of underwater robotic missions, the UWSim [15]. This simulator was part of the TRIDENT and TRITON projects and was used in different projects funded by European Commission (MORPH [16], PANDORA). The UWSim relies on OpenSceneGraph (OSG) [17] and osgOcean [18] libraries and uses the middleware Robot Operating System (ROS) [19], making easier the integration in any architecture.

In both projects (TRIDENT and TRITON), the architecture was based on ROS, which facilitates the communication between both the real robot and the simulator, using an asynchronous streaming of data over "topics". The real robot and UWSim have exactly the same topics, which allows the user send the same command to both systems. In addition, the UWSim could be used for visualizing the robot mission, just subscribing it to the robot topics.

So, in order to improve the UWSim and adapt it to our proposal, a User Interface Abstraction Layer (UIAL) has been developed. This abstraction layer can be integrated easily in most of the architectures, although nowadays is mainly focused to be used with ROS. Due to the constrains in order to use the real robot, we have developed this abstraction layer using only the UWSim. As we mention before, both the robot and the UWSim uses the same topics, so the UIAL considers the robot and the UWSim as the same object. In fact, if both are in the same network (e.g. if there is a cable connection between them), they can be controlled using the same commands. Fig. 1 shows the UIAL architecture.

To sum up, the main UIAL features are:

- Receive data from the different controllers (keyboard, hand trackers, joysticks).
- Receive data from sensors (virtual or real).
- Adapt and filter the information to be shown to the user.
- Reconfigure the use of both the input and output devices according to the mission or the task.
- Apply safety measures to prevent damages to the robot. E.g. if the robot is close to the seafloor, will discard all the commands to send the robot deeper.

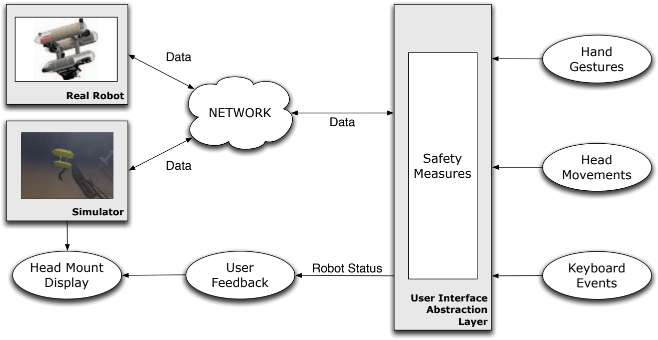

Fig. 1. UIAL manages the input devices and filters the most relevant information.

IV. GETTING AN IMMERSION FEELING

UWSim offers multiple camera point-of-view (POV) controls. One of the most used is the trackball, which means that the user needs to control it by using the mouse. Although this represents an advance control for expert users, it is hard to control and get a properly experience for non-expert users. The I-AUV used in TRIDENT and TRITON project allowed a different camera location and configuration, depending on the mission to be perform, but usually, the main camera used is located in the front-middle position of the robot.

One of the improvements with our proposal, bearing in mind the user experience, is the way the user control the on-board camera. In order to achieve this, some 3D techniques has been applied to the UWSim, using a Head-up Mounted Display (HMD) to control the camera's POV. The developed system combines the information from the Inertial Measurement Unit (IMU) of Oculus Rift DK-1 [20] and the information obtained from a camera/marker system (Fig. 2). The IMU, composed by accelerometer and gyro provides the orientation, while the camera/marker system provides the position.

As we can see in Fig. 2, we fusion the data obtained by each subsystem, IMU and camera/marker system, enabling a full mapping of the user's head movements to the virtual world

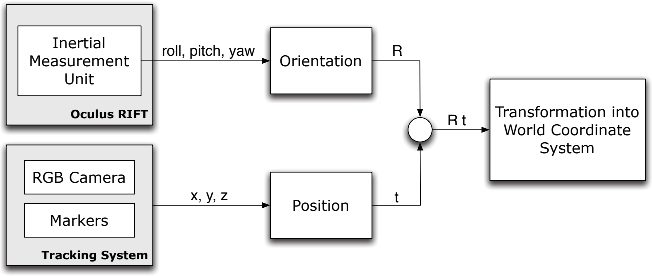

Fig. 2. This diagram shows how the information from the HMD and the head tracking system is merged, to transform it into world coordinate system.

(UWSim). Thus, it provides a better experience and immersion level. From the combination of camera and marker we extract the 3D position [21] and because the marker is physically aligned to the IMU we can apply its orientation on that point. At this stage, we have the transformation (translation and rotation) of the user's head ready to position the virtual camera on UWSim, and consequently, change the camera's POV and translation inside the virtual cockpit.

V. IMPROVING THE USER FEEDBACK

As we mentioned before, one of the main drawbacks in the already GUI addressed to ROV/AUV systems, is the GUI complexity. As we can see in Fig. 3, which represents a robot mission trial related to the TRITON project, the user has to pay attention to different screens, due to UWSim uses a Linux command line to show all the information to the user. In this setup (left-right, top-down), screen 1 is used for monitoring the arm, screen 2 is used for the visual tracking system, screen 3 is used for monitoring the vehicle status and screen 4 is used for monitoring the valve intervention algorithm.

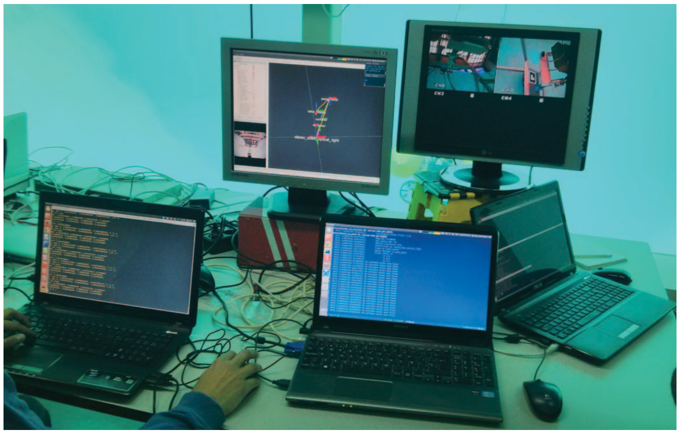

Fig. 3. During a trial related with TRITON project, the user should pay attention to different screens.

Due to we propose the use of a HMD, we have developed a virtual windshield to improve the way the user gets this information. This virtual windshield has animated elements placed on a virtual sphere that surrounds the user's location, simulating the feeling of being in a real cockpit. The combination of the head tracking and the Oculus Rift DK-1, creates a sense of immersion and telepresence that is desirable when conducting a mission with a teleoperated vehicle. So, the virtual windshield is a practical way of presenting the required information to the user and it is made to be intuitive, simple and informative without being intrusive, confusing and overwhelming.

Fig. 4 shows the evolution, from the standard setup using UWSim (one screen for the simulator plus one screen for the output data such as navigation information or sensor data) to our proposal, where most of the information is shown graphically and it is filtered depending on its relevance (e.g. a blinking warning signal appears when the robot is close to the seafloor or the surface, a battery indicator appears when the battery charge is down a predefine value).

The possibility of a dynamic interface is not the only advantage of our approach. The feeling of being inside a real vehicle with a dashboard, warning lights and indicators also creates a familiar and immersive experience. With the aforementioned developed head tracking system, the user can even lean on any of the interface components like he/she would do in a real situation. All these little details that sometimes go unnoticed by the user create a sense of presence that is impossible to match just by looking at a screen (or several screens).

VI. EVALUATION EXPERIMENTS & RESULTS

In order to compare and evaluate our proposal with the traditional setup, we have created a new scenario in the UWSim. In this mission, the user has to control the robot navigation passing through five rings, that are not collinear and have different orientations. We consider three different setups:

- Setup 1 (traditional): the user needs to pay attention to two different screens and control the camera POV control using the mouse. The robot will be controlled using a gamepad.
- Setup 2 (immersive traditional): the user will use the HMD to get the immersive experience and to control the camera POV. The robot will be controlled using a gamepad
- Setup 3 (immersive with LeapMotion): the user will the HMD to get the immersive experience and to control the camera POV. The robot will be controlled using a LeapMotion. For expert users or users with experience playing video games, the gamepad is the best way to control the robot. Nevertheless, the LeapMotion could be representative of a more natural and intuitive control method.

The participants are between 20-50 years-old with a mean age of 26, 10 males and 3 females, with each trial lasting 3 minutes. All participants reported normal or corrected to normal vision. Participation was voluntary and experiments are in accordance with ethical principles of research.

The participants received a short description about the mission with a non-detailed map and the pertinent information, before they started the experiment. The non-detailed map represents the position of the simulated robot and the waypoints in a top view representation. Then, we divide the participants in different groups, so they will start the experiment running a different setup since the beginning. Due to the scene is the

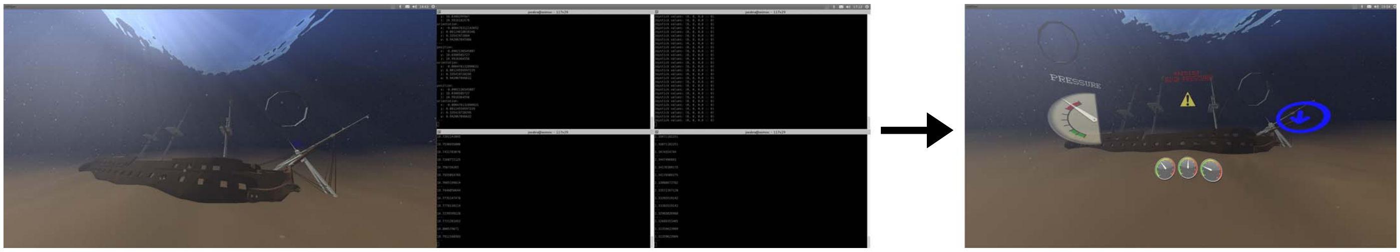

Fig. 4. Left: the user needs two screens to get all the information: the simulated view and the info (up-down, left-right: navigation data, commands sent the robot, pressure sensor (the user will know the robot depth) and the range sensor (the user will know the distance to the seafloor). Right: the UWSim using our proposal, displaying the same information in a graphical way.

same in all the setups, this will reduce the learning effect. It is obvious, that the users may improve their results after each test, because they will know where are the waypoints. Finally, after each setup, the user will answer a brief questionnaire to get suggestions and opinions. In order to compare the results, we are going to measure the time needed to gain confidence with the robot controller, the time needed to arrive at each waypoint and the overall time needed to perform the whole mission.

Fig. 5. The participant is running an experiment (setup 2). The HMD will use the splitted imaged display in the screen to generate the 3D image.

After analyzing the time needed by the users, the vast majority of the users improved their overall time when using the immersive system. This is due to the camera's POV is intuitively connected to the participant's head movements, so it is not needed to manually switch between controlling the robot and camera's POV. All the participants reported that the head movement camera control is "natural and intuitive" and required short adaptation times. According to the comparison about the gamepad and the LeapMotion, the people with experience in video games prefer the gamepad because they are get used. On the other hand, people without experience in video games prefer the LeapMotion because "the movements to control the robot are easy to remember and they do not need time to learn". Nevertheless, all the participants agree that the use of LeapMotion and the Oculus Rift is the most difficult, because the HMD blocks the view between the user's hand and the LeapMotion, losing the reference among them. After the authors noticed about this problem, the setup was modified adding a fan close to the LeapMotion, so the user would be able to feel the airflow and localize the device.

Regarding the questionnaire, the results can be summarized as follows:

- 92% of the participants preferred using Oculus Rift DK-1, instead of the traditional system with the double screen setup.
- 77% of users considered Oculus Rift DK-1 very important or essential for improving the mission performance.
- 85% of users considered the virtual interface (windshield) very important or essential.

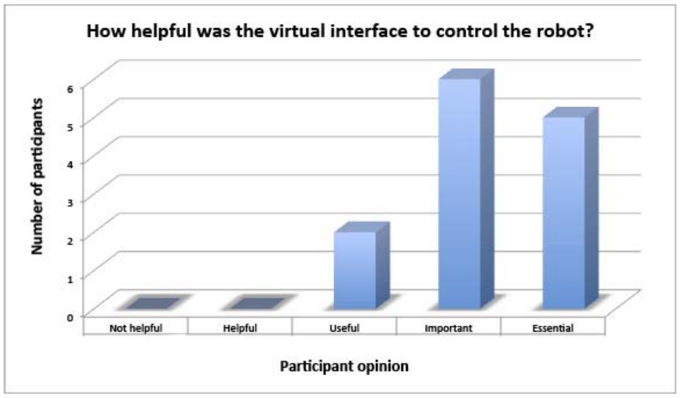

Fig. 6. The virtual interface (windshield) helps the user to controll the robot accurately.

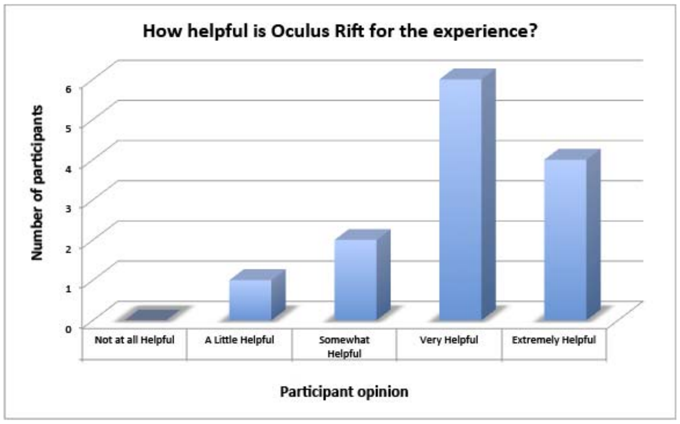

Fig. 7. Oculus Rift DK-1 improves the user experience.

VII. CONCLUSION

To sum up, this paper presents a new GUI approach, focused in improving the HRI in the underwater intervention domain. The three main improvements of our proposal are:

- Defining an abstraction layer.
- Getting an immersive feeling.
- Improving the user feedback.

Finally, some preliminary tests have been carried out for validation. Almost all the participants agree with the idea of the virtual windshield and the use of the HMD for an immersive experience, getting only the most relevant information. Nevertheless, the use of a hand tracking device such as LeapMotion, was not successful, due to the lack of feedback and because the user loses the reference between his/her hand and the device. The use of the fan airflow reduces the problem, but does not remove completely.

ACKNOWLEDGMENT

This research was partly supported by Spanish Ministry of Research and Innovation DPI2011-27977-C03 (TRITON Project).

REFERENCES

[1] T.B. Sheridan, *Telerobotics, Automation and Human Supervisory Control*. Massachusetts, USA: MIT Press, 1992.

[2] S. K. Choi, O. T. Easterday, *An Underwater Vehicle Monitoring System and Its Sensors*, in Experimental Robotics VII, LNCIS 271, pp. 551-560, Springer-Verlag Berlin Heidelberg, 2001.

[3] (2015, 2) SAUVIM project webpage [Online]. Available: http://www.eng.hawaii.edu/%7Easl/Sauvim/

[4] G. Marani, S.K. Choi., J. Yuh, *Underwater Autonomous Manipulation for Intervention Missions AUVs*. in Ocean Engineering. Special Issue: AUV, Volume 36, Issue 1, January 2009, Pages 15-23.

[5] (2014, 9) Interactive & Robotic Systems Lab webpage. [Online]. http://www.irs.uji.es

[6] P.J. Sanz, P. Ridao, G. Oliver, G. Casalino, Y. Petillot, C. Silvestre, C. Melchiorri, A.Turetta, *TRIDENT: An european project targeted to increase the autonomy levels for underwater intervention missions* in OCEANS'13 MTS/IEEE conference. Proceedings. San Diego, CA (2013)

[7] A. Peñalver, J. Pérez, J.J. Fernández, J. Sales, P.J. Sanz, J.C. García, D. Fornas, R. Marín, *Autonomous Intervention on an Underwater Panel mockup by using Visually-guided Manipulation Techniques*. in IFAC World Congress, Ciudad del Cabo, Africa, 2014.

[8] (2014, 9) Pandora EU-research project website. [Online]. https://persistentautonomy.com/

[9] J.C. García, J.J. Fernández, P.J. Sanz, R. Marín, *Increasing Autonomy within Underwater Intervention Scenarios: The User Interface Approach*, in: 2010 IEEE International Systems Conference. 2010 IEEE International Systems Conference. San Diego, CA, USA; 2010:71-75.

[10] O. Matsebe and C. M. Kumile. *A review of virtual simulators for autonomous underwater vehicles (AUVs)*, in NGCUV 2008, Lakeside Hotel, Killaloe, Ireland, 2008.

[11] (2015, 2) CMLabs. Vortexsim. [Online]. Available: http://www.vxsim.com/en/simulators

[12] (2015, 2) Marine Simulation. Rovsim. [Online]. Available: http://marinesimulation.com.

[13] (2015, 2) GRL. Rovolution. [Online]. Available: http://www.generalrobotics.co.uk.

[14] (2015, 2) Fugro General Robotics Ltd. Deepworks. http://www.fugrogrl.com/software/.

[15] M. Prats, J. Pérez, J.J. Fernández, P.J. Sanz, *An open source tool for simulation and supervision of underwater intervention missions*, in 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2577-2582, 7-12 Oct. 2012

[16] (2014, 9) Morph EU-research project website. [Online]. http://www.morph-project.eu/

[17] (2015, 2) OpenSceneGraph [Online]. Available: http://www.openscenegraph.org.

[18] (2015, 2) osgOcean [Online]. Available: http://code.google.com/p/osgocean.

[19] (2014, 9) Robot Operative System webpage. [Online]. Available: www.ros.org

[20] (2014, 9) Oculus Rift website. [Online]. http://www.oculus.com/

[21] B. Patrão, *Biblioteca para desenvolvimento de aplicações de Realidade Aumentada em marcadores binarios [Augmented Reality Library using binary markers]*, (University of Coimbra Master Thesis), DEEC - FCT, University of Coimbra, 2011