Altitude control of an underwater vehicle based on computer vision

Pedro M. Rodrigues* Nuno A. Cruz (Supervisor)† Andry M. Pinto (Co-supervisor)‡

Abstract—It is common the use of the sonar technology in order acquire and posteriorly control the distance of an underwater vehicle towards an obstacle. Although this solution simplifies the problem and is effective in most cases, it might carry some disadvantages in certain underwater vehicles or conditions. In this work it is presented a system capable of controlling the altitude of an underwater vehicle using computer vision. The sensor capable of computing the distance is composed of a CCD camera and 2 green pointer lasers. Regarding the control of the vehicle, the solution used was based on the switching of two controllers, a velocity controller (based on a PI controller), and a position controller (based on a PD controller). The vehicle chosen to test the developed system was a profiler, which main task is the vertical navigation. The mathematical model was obtained and used in order to validate the controllers designed using the Simulink toolbox from Matlab. It was used a Kalman filter in order to have a better estimation of the state variables (altitude, depth, and velocity). The tests relative to the sensor developed responsible for the acquisition of the altitude showed an average relative error equal to 1 % in the range from 0 to 2.5 m. The UWsim underwater simulation environment was used in order to validate the integration of the system and its performance.

Index Terms—Underwater vehicle; Altitude control; Computer vision; Profiler; Kalman filter; PID Controller; UWsim.

I. INTRODUCTION

The desire of improving and developing new technologies targeting the ocean's supervision and exploration is continuously increasing. A common method of exploration of the oceans is the underwater exploration missions. Since these tasks might involve hostile environments far too hazardous for human intervention, it is typical to resort to systems based on underwater vehicles. During an underwater mission using underwater vehicles, the distance between the vehicle and the sea floor or other obstacles must be warily controlled to ensure its safety and the reliability of the mission. Commonly, to deal with this task, a system based on sonar technology is used. Although this solution simplifies the problem and is effective in most cases, it might carry some disadvantages in some underwater vehicles or conditions. Particularly, the sensors based on acoustic waves, like the sonar, might present difficulties on the interpretation of the signals received when the vehicle is extremely close to the obstacle, requiring a minimum distance to retrieve valuable and reliable information. Furthermore, the inclusion of the sonar sensor demands an increase on the energetic cost of the system that in the case of vehicles powered by an external source through a cable like the

*up201306231@fe.up.pt
†nacruz@fe.up.pt
‡andry.pinto@fe.up.pt

ROVs is not a problem, but in AUVs, it might be valuable to avoid it since these vehicles are powered by batteries. Lastly, sometimes the space occupation of the sonar sensor represents a problem in some vehicles with meticulous limits relative to space usage, a common problem found in AUVs.

To overcome the problems mentioned, in this work it is presented the development of a system capable of controlling the altitude of an underwater vehicle using computer vision. To provide this functionality, it is required that the system gathers information about the altitude of the vehicle in real-time based on capturing and the processing of images containing known marks imposed by laser devices.

The vehicle used in order to validate and test the system created was a profiler developed in [1], illustrated in figure 1. The vehicle has approximately 1.35 m of length and a mass of 11.3 kg and its main purposes are the navigation in the vertical axis in order to acquire information related to the water properties and the inspection of the sea floor. The application of the system to be developed is opportune in this vehicle since it has energy limitations due to being an AUV, reduced diameter making it not possible to incorporate a sonar device, and it navigates mainly in the vertical direction, which makes the control of its altitude extremely important to control. In this work the concept of altitude is interpreted as the distance of the vehicle towards the closest obstacle in its movement direction.
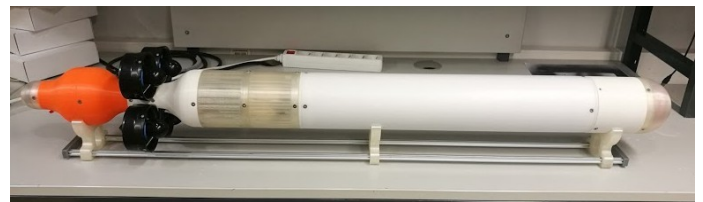

Fig. 1. Image of the profiler

II. STATE OF THE ART

The distance of the vehicle towards the sea floor is one of the most important information needed for its control. In order to gather this information, several solutions have been presented through the years.

One option to acquire the distance of the vehicle to the sea floor is the use of methods based on computer vision. In [2] it is presented a method based on geometrical triangulation using a laser beam pointer and a Charge Coupled Device (CCD) camera to capture it. The distance of the vehicle to the objects is calculated through simple laser triangulation.

A similar method using two laser points and a CCD camera was used in [3]. In this case, two green laser pointing devices were placed parallel to one another beside the camera. Distances are also calculated by using laser triangulation and processing of the images acquired in order to locate the laser points.

In [4] an algorithm able to calculate distance information about the vehicle surroundings is presented using a CCD camera and two laser line generators mounted on top of each other and parallel to the camera's viewing axis instead of two laser pointers. The use of two laser line generators allows the determination of the distance to an object at multiple points of the projected line in the image, enhancing the perception and the performance of the vehicle's ability to navigate.

Regarding the control of the altitude of the vehicle, the most common control approach is the classic Proportional-Integral-Derivative controller (PID), which is a feedback controller. In [5] a simple PID controller is presented to control the altitude of the vehicle while it moves horizontally. Several other cases of success have been reported relative to the control of AUVs based on the PID approach, demonstrating its effectiveness in this ambit [3], [6], [7].

Besides the control algorithm, the application of filters is a valuable way of removing noise and preparing the information to be given to the control phase, increasing the reliability of the system.

Estimations filter are commonly used to enhance the accuracy in the measurement and to filter out high deviations [8], [9]. The Kalman filter is a general error tracking and estimation based filter. At each time interval, the Kalman filter updates its prediction based on the characteristics of the process, then using the measurements obtained through its sensors, the final value is obtained by adjusting the prediction value with the measurement. The more accurate the measurement the more impact it will have in the correction phase. Furthermore, when the vision system could not acquire the information needed to know the state of the system, it can use the prediction given by the prediction phase or even integrates more information coming through other sensors [10].

III. OVERVIEW OF THE SYSTEM

To achieve a system capable of controlling the altitude of an underwater vehicle using sensing based on computer vision, it was developed a module capable of measuring the distance (Sensor module) and a module able to filter the data gathered by the sensor module and capable of using these data to control the distance of the vehicle to an obstacle (Filtering and Control module). Figure 2 illustrates a block diagram that demonstrates an high level view of the context where these modules are inserted and how they interact with each other.

The diagram shows that the Filtering and Control module and Sensor module are inserted in a distance control loop sequence of a vehicle based on the input of a reference that is compared to the value obtained by the Sensor module after filtering.

The division on two modules allows the implementation of the two main inherent features of the system (sensing
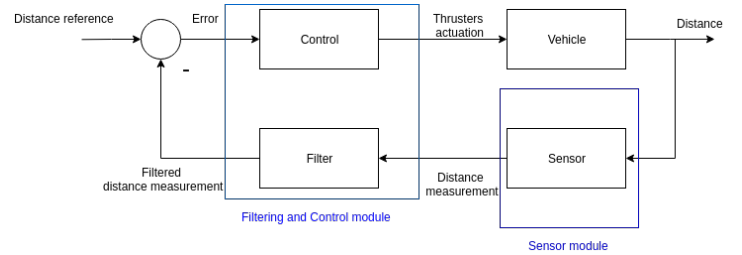

Fig. 2. Block diagram of the overall system integrating the modules developed in this work

and control) independently and therefore enables the introduction or the improvement of features on any module without imposing significant changes on the other module. This is especially important since it was intended that the sensor module would be developed in a way that allows its use in other underwater applications other than the control of the distance of an underwater vehicle.

IV. SENSOR MODULE

A. Hardware

The sensor device was implemented based on two laser pointer devices placed parallel to one another beside a CCD camera. The choice of using laser pointer devices over laser line devices is mainly supported by the inherent lower computer demanding and complexity of the image processing phase required. The camera used was the already integrated camera on the profiler, which is a Mako G-125 camera based on the Vimba SDK from Allied Vision. It is well compacted, allows the powering over Ethernet, and in full resolution is able to provide 30.3 frames per second. Although it is able to provide the mentioned frame rate, it was used an underwater Ethernet cable in order to gather the information limiting the maximum frame rate to approximately 9 frames per second.

Relative to the laser pointer devices, it was used two green (532nm) laser modules from OdicForce Lasers. It was performed an adjustment in order to by-pass the switch pressure button in order to work in continuous operation whenever power is applied.

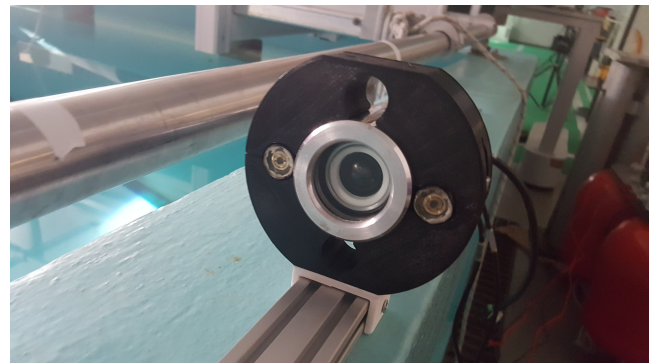The overall structure of the sensor is presented in the figure 3.

Fig. 3. Structure containing the camera and the laser devices
.

B. Software

The software developed was structured in a C++ class capable of performing 4 main features: Acquisition of the frame; Calculation of the distances; Data handling; Communication.

The first feature is responsible for the configuration, communication with the camera device, and the acquisition of new frames.

The Calculation of the distances feature is divided into 3 phases: detection of the laser dots on the image; calculation of the centroids of the laser dots; triangulation method in order to compute the distance. First, the image is processed using image processing methods in order to obtain a left and right segmented image containing only the left and right detected laser respectively. Then, the centroid of each laser dot is obtained and sent to the last phase where based on laser triangulation, a distance towards the obstacle is obtained for each dot in use.

The Data handling is responsible for two main operations: calculation of the quality of the detection of the laser dots and application of certain operations based on the software configuration, for example, application of a circular average on the measurements.

All the information acquired is then sent to the user through UDP communication in the last phase and the software awaits a new frame to process (Communication).

The information that the system is capable of acquiring includes for example, the distance calculated using each laser dot, an quality factor representative of the quality of the detection of the laser dots, and the result of the average between the 2 measurement. The choice of the information acquired is configured by an XML file.

Figure 4 illustrates the procedures that occur in a generic execution of the software from the acquisition of the frame to the dispatch of the data to the user.

C. Characterization and Tests

In order to characterize and test the sensor module, it is first necessary to know the limitation of the technology used in the distance calculation. Through the triangulation method it is known that the accuracy of the measurements is inversely proportional to the distance of the obstacle, and therefore the performance of the acquisition of the distance is limited by this inherent error. This is caused by the fact that the determinant factor of the distance calculation is the choice of the pixel where the laser dot is detected. Therefore, it is then possible to assert that the range of values of the distance obtained is "discrete" and has an error equal to the distance necessary to go through that makes the detection of the laser to be in the forthcoming pixel.

The procedure of calculation and analysis of this error is detailed in [11]. In I it is illustrated the value of the inherent error caused by the use of the triangulation principle for 4 different distances.

It was also studied the divergence of the camera relative to the target and the divergence of the laser devices in order to obtain the linear regression equations that allow the calibration
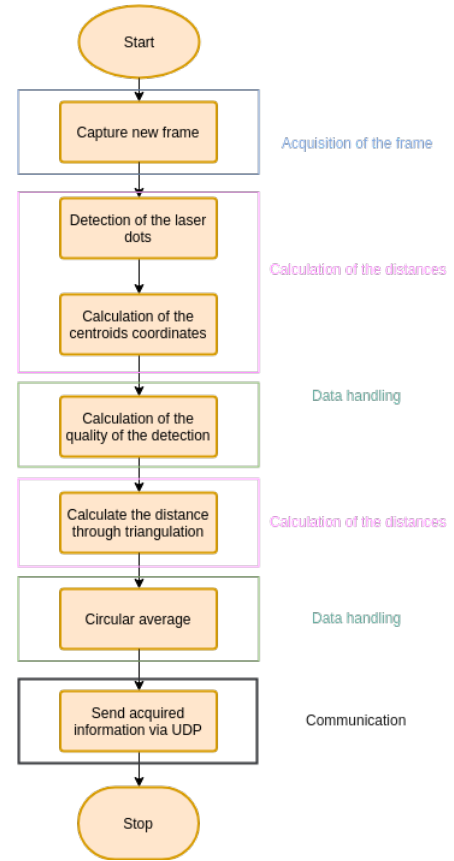

Fig. 4. Example of a generic high-level execution of the software related to the Sensor module

TABLE I
INHERENT ERROR CAUSED BY LASER TRIANGULATION

distance	error	error(%)
0.258m	0.1cm	0.39
1.077m	1.8cm	1.69
1.794m	5.2cm	2.90
3.076m	15.4cm	5.00

of the sensor. The structure set up in order to test the Sensor module is illustrated in the image 5.

The equation 1 represents the divergence of the camera relative to the target.

$$Deviation = 0.02033 \times D + 0.001956 \quad (1)$$

Relative to the divergence of the lasers, they were placed 4cm to the center view of the camera and the equations 2 and 3 shows the divergence with the increase of the distance.

$$D_{\text{left-center}} = 0.01404 \times D + 0.03990 \quad (2)$$

$$D_{\text{right-center}} = -0.01316 \times D + 0.03872 \quad (3)$$

After a procedure of calibration using the linear equation in order to counteract the error induced by the divergence of the components, a series of tests were performed from the range of 10 to 250 cm, moving the target 20cm between each test in order to validate the performance of the Sensor module. The
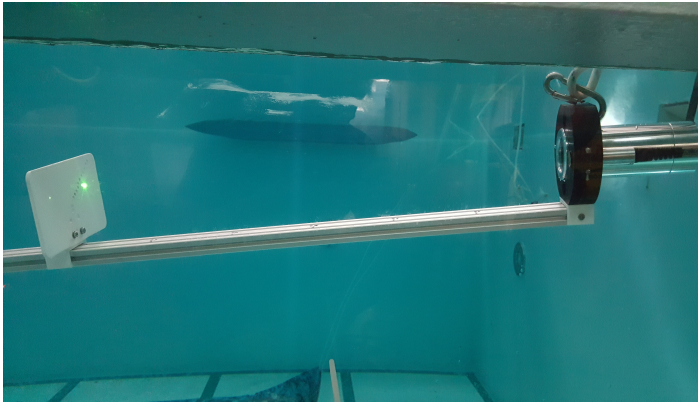
Fig. 5. The full structure underwater, containing the target on the left

.

Fig. 7. Measurements obtained by the right laser dot

.

results obtained for the distance calculated using the left laser dot (D_l) and right laser dot (D_r) are presented in the table II.

TABLE II
RESULTS OBTAINED FOR SENSOR PERFORMANCE

D	D_l	D_r	$Error_l$	$Error_r$
0.10 m	0.1004 m	0.1014 m	0.4 mm	1.4 mm
0.30 m	0.3011 m	0.3017 m	1.1 mm	1.7 mm
0.50 m	0.4942 m	0.5027 m	5.8 mm	2.7 mm
0.70 m	0.7022 m	0.6889 m	2.2 mm	1.1 cm
0.90 m	0.9106 m	0.8769 m	1.06 cm	2.31 cm
1.10 m	1.0832 m	1.0737 m	1.68 cm	2.63 cm
1.30 m	1.2795 m	1.2785 m	2.05 cm	2.15 cm
1.50 m	1.4908 m	1.4712 m	0.92 cm	2.88 cm
1.70 m	1.6958 m	1.7196 m	0.42 cm	1.96 cm
1.9 m	1.8983 m	1.8742 m	0.17 cm	2.58 cm
2.1 m	2.1740 m	2.1606 m	7.4 cm	6.06 cm
2.3 m	2.3579 m	2.2253 m	5.79 cm	7.47 cm
2.5 m	2.5635 m	2.4225 m	6.35 cm	7.75 cm

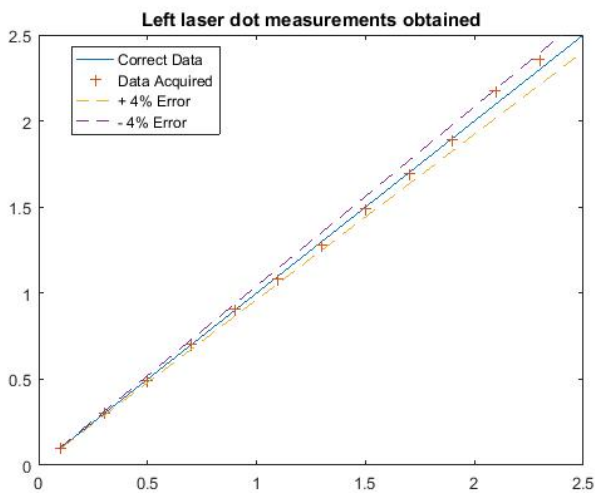In images 6 and 7 a better visualization of the error obtained in these tests is presented.
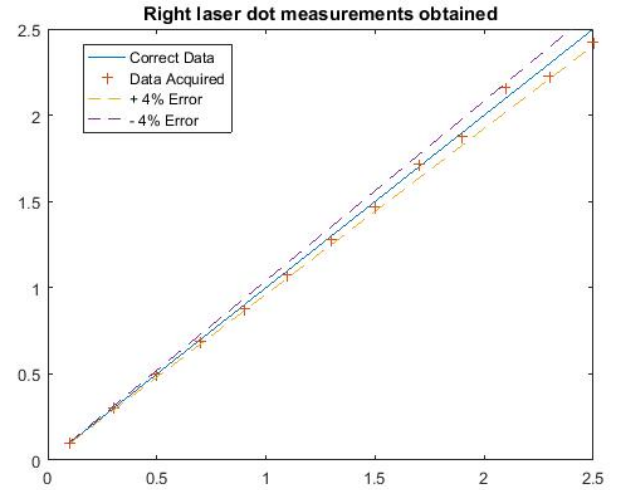

Fig. 6. Measurements obtained by the left laser dot

.

Through the observation of the graphics presented, it is possible to assert that the measurements obtained presented an error inferior to 4 % for the full range of measurements.

Since 2 measurements were acquired for the same altitude, using a simple average between the 2 measurements for the same altitude resulted in a significant improvement of the measurement obtained. The average relative error of the left and right measurements was 1.26 % and 1.88 %, respectively, and after the application of the average operation, the average relative error was equal to 1%.

V. FILTERING AND CONTROL MODULE

Since the Filtering and Control module is meant to deal with the filtering of the received data and the control tasks, it was divided into two phases: Filtering phase and Control phase.

A. Filtering phase

The Filtering phase is responsible for 3 tasks: communicate with the Sensor module in order to retrieve the information gathered; apply a Kalman filter to the data received; transmit the estimates to the vehicle through a shared memory block. This phase serves as the link between the vehicle software control and the Sensor module software.

The Kalman filter implemented has 3 state variables, predicting and updating the altitude, depth, and velocity. These states are fed by the Sensor module and by the pressure sensor of the profiler.

Since this work's main objective is the control of the altitude of the profiler when it moves vertically, the filter was designed in a way that allows the update of the altitude of the vehicle even if the Sensor module could not retrieve any measurements. This is possible since the variation of the depth is the inverse of the variation of the altitude when the profiler moves vertically and therefore in cases where the sensor does not calculate any reliable measurements, the variation of the depth is used in order to have an approximate update of the altitude value.

B. Control phase

The solution adopted regarding the control is based on the switching of two controllers, a velocity controller, and a position controller. The idea is that the vehicle moves with a specified velocity imposed using the velocity reference while it is still considered far away from the reference and then a position controller is used to complete the movement and reach the position reference. This allows more control over the movement of the vehicle through the mission especially in the velocity of the movement and avoids the application of high value steps into a position controller that would result in a high power consumption. Additionally, this division allows the design of the velocity and position controllers based on the demands that each phase of the movement requires, approximation and reach of the reference, respectively.

1) Mathematical model of the vehicle: The first step related to the Control phase, was the calculation of the mathematical model of the vehicle. In [1] the model of the vehicle for its 6 degrees of freedom was obtained following the procedures presented in [12]. The results obtained in [1] related to the vehicle's model and its experimental parameters were used in this work in order to retrieve the equation that expresses its vertical movement since that its control is the focus of this dissertation.

The next step was the linearization and calculation of the state space of the model obtained. The last step was the calculation of the transfer function that represents the relation between the input given to the thrusters of the vehicle and the variation of its position in the vertical direction. An detailed description of the procedure to obtain the transfer function is present in the full document refer [11].

The transfer function obtained for the relation between the input given to the thrusters of the vehicle and the variation of z can be given by:

$$G(s) = \frac{0.0860}{s(s + 0.1029)} \qquad (4)$$

Based on this transfer function it is easy to conclude that a transfer function that relates the input given to the thrusters of the vehicle and the variation of velocity on the vertical direction can be given by:

$$G(s) = \frac{0.0860}{s + 0.1029} \qquad (5)$$

2) Position controller: The position controller was implemented based on the conventional PD (Proportional – Derivative) approach. The reason behind this choice is given by the analysis of the transfer function obtained referent to the position of the vehicle. Since it is a second-degree transfer function with a pole in the origin, it is possible to use a PD controller in order to introduce a zero that cancels the plant's pole located in the left plane of the root-locus. Therefore, the gain K is the only parameter that can be changed in order to change the performance of this controller. The increase of this parameter would mean an decrease on the time necessary to reach the steady state, however this also implies an increase on the energetic cost. In this work, after analyzing the main

objectives, it was decided that an response time of 10s would be the imposed factor as well as ensuring no overshoot, resulting in a value of the parameter K equal to 5.

Relative to the switching point, the idea behind the calculation of the switching point is that the actuation that the position controller requires on the switching point is approximately equal to the actuation required by the velocity controller on that exact moment, making the transition smooth for the thrusters. This is achieved by considering that when the switch occurs, the velocity of the vehicle is constant and equal to the velocity reference. This allows the knowledge of the actuation that is being applied to the thrusters since the actuation is only trying to counteract the damping and flotation forces. Therefore it is possible to determine what is the value of the step that can impose on the position controller that value of actuation, obtaining the switching point.

Since the position control is meant to control the altitude of the vehicle it is essential that when the switching occurs, the Sensor module can calculate reliable information related to the altitude. Therefore, the switching point must occur when the altitude of the vehicle is inferior to the maximum distance that the sensor module can work. Analyzing the data collected for the sensor developed it was decided that in this control application, the range where the sensor was going to be used was from 0.10 to 2.5 m since after that value the absolute error was considered not appropriate for this application.

3) Velocity controller: The velocity controller was implemented based on the conventional PI (Proportional-Integral) approach. In the same way as the position controller, this choice is given by the analysis of the transfer function obtained referent to the velocity of the vehicle. Since it is a first-degree transfer function, it is possible to use a PI controller in order to introduce a zero that cancels the plant's pole and add a pole placed at the origin. This allows the tuning of the controller through the gain parameter of the controller.

In contrast to the position controller, The value of the gain parameter in the velocity controller is flexible towards the mission's demands and the vehicle's characteristics.

Considering that the velocity controller task is to approximate the vehicle towards the obstacle while it is still far away, it is considered that in most situation it will be the controller responsible for most of the navigation of the vehicle and therefore the main concern for its design is the energetic cost. The energetic cost depends on the power enforced by the controller and the time necessary to reach the switch point to the position controller and therefore this information is used in order to design the controller. Furthermore, the choice of the parameter also takes into account the maximum current enforced by the controller in order to guarantee that the thrusters can provide the necessary force.

The procedure created in order to tune the controller starts with the mathematical calculation of the value of the gain of the velocity controller that induces the maximum current that the thrusters are able to provide. Since the final system is a first degree system, its response to a step will be an exponential that has its maximum derivative at the start. Using this value and the information present in the data sheet of the thrusters of the vehicle, it is possible to tune the velocity controller

to initially impose a determined maximum current. From this point and through simulation of the closed loop system it is possible to tune the gain of the controller in order to adapt the controller towards the missions' demands. In this work, a Simulink diagram, including both controllers, was created in order to visualize all the information needed to adapt the controller to the characteristics of the vehicle and mission. The full explanation of the tuning and the simulink diagram is present in [11]

VI. INTEGRATION AND EVALUATION OF THE MODULES

After the test of each module separately, it is necessary to validate the behavior of the system when the 2 modules developed are integrated and therefore it is used the Sensor module to gather the information that feeds the control. Previously to the integration of the modules in the profiler, it is mandatory the performance of tests using a simulation environment for underwater vehicles. Through this simulation it is possible to simulate a vast number of environments and conditions to predict the behavior of the system in a real environment.

It was used the UWsim simulation environment [13] that allows the simulation of the movement of an underwater vehicle in an underwater virtual scenario. The objectives of the simulations were the following:

- Validation of the overall architecture developed that integrates both modules in the profiler's system;
- Validation of the controllers in discrete time;
- Integration of the sensor module and the controller module, and validation of the control of the profiler using the measurements from the sensor module;
- Evaluation of the Kalman filter, especially in situations where the Sensor module could not acquire a reliable measurement and the pressure sensor must be used;
- Visualization of the movement of the vehicle in an underwater virtual scenario in order to better evaluate the controllers.

In order to simulate the system using the UWsim, the example Girona500 given in the tutorials of the UWsim [13] was used in order to start from a generic example. Several changes were made in the files presented in this example to implement the profiler characteristics, the controllers designed, and to create an underwater virtual scenario that allows the objectives mentioned.

The main 4 files that were created or adapted are the following:

- dynamicsAUP - Python file based on the Girona500 example that implements the dynamics of the vehicle;
- dynamicsConf - File containing the dynamic model parameters;
- aupCamera - XML file that determines the underwater virtual scenario.
- ThrusterControl - Python file that is responsible for the application of the speed and position controller;

The main difference made in the dynamicsAUP file in comparison to the Girona500 example is the determination of the flotation force that was adapted in order to consider the drag coefficient and the flotation force of the profiler.

The dynamicsConf format is the same as the one presented in the Girona500 example, where the values were changed according to the parameters obtained for the model of the profiler.

The aupCamera file was created based on the conventional underwater virtual scenarios presented on the tutorials of the UWsim [13], and it contains a virtual camera in order to gather images simulating the acquisition of the Sensor module, a light projector to simulate the laser pointer device, a pressure sensor, and a sonar device. The overall scenario simulated is a typical ocean-like environment with a box object near the bottom of the ocean in order to guarantee that the surface is flat and more controllable.

Furthermore it was developed a ROS workspace referent to the Sensor module in order to make possible the interaction between the UWsim and the Sensor module software. It contains the following 3 main components:

- A main file responsible for the interaction between the Sensor module and the UWsim;
- The sensor software developed;
- The kalmanfilter class developed.

The overall architecture was developed in order to make this simulation as close as possible to the real operation of these modules after their integration in the profiler.

The figure 8 illustrates the architecture and the typical sequence that runs in the simulation.

A. Tests performed

The objective of the simulation was the validation of the overall architecture. Furthermore, this initial test also allows the validation of the discrete controllers in this underwater environment, the usage of the sensor module, access to the shared memory block, and the evaluation of the actuation of the Kalman filter in situations where the sensor module could not retrieve information and therefore it is necessary to resort to another sensor.

For this test the parameters and settings used are the following:

- Starting point at approximately $4\ m$ of altitude;
- Position controller with an input reference of $0.3\ m$ and the following parameters, $K_p = 0.5145$ and $K_d = 5$;
- Velocity controller with an input reference of $0.1\ m/s$ and parameters $K_p = 10$, $K_i = 1.0290$;
- Switching at 1.1687 (Switching error point) + 0.3 (position reference) m, calculated as stated in section V-B;
- Consideration that the measurements of the Sensor module when the vehicle is at an altitude higher than $2.5\ m$ are not reliable;
- Usage of the z position given by the dynamic instead of the pressure sensor simulated by the UWsim in order to have more control over the error present in those
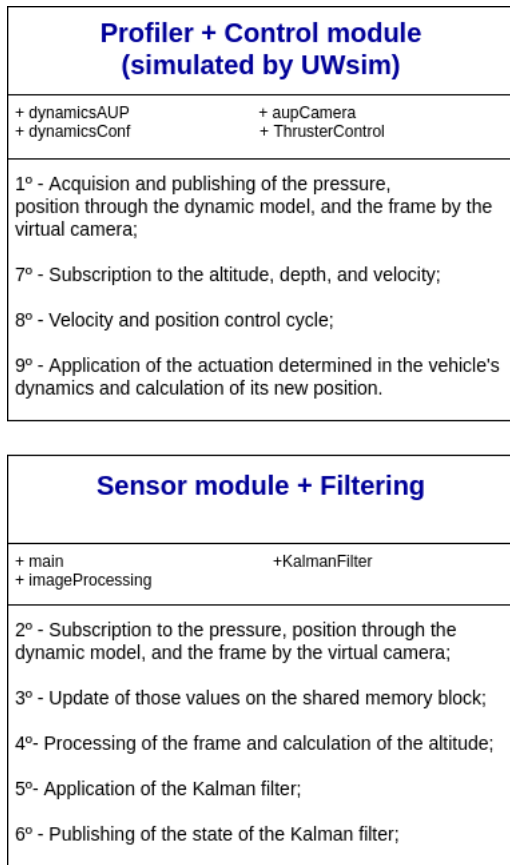
Fig. 8. Overall architecture and sequence of actions that take place in the simulation

measurements, guaranteeing that for this test the sensor used is considered a "perfect" sensor.

- The frequency of frames given by the simulated camera is 9 fps in order to better simulate the camera used by the profiler.
- The period of the control cycle is 100ms;

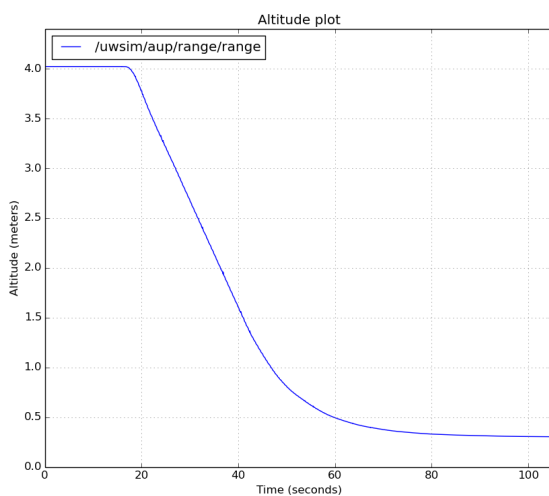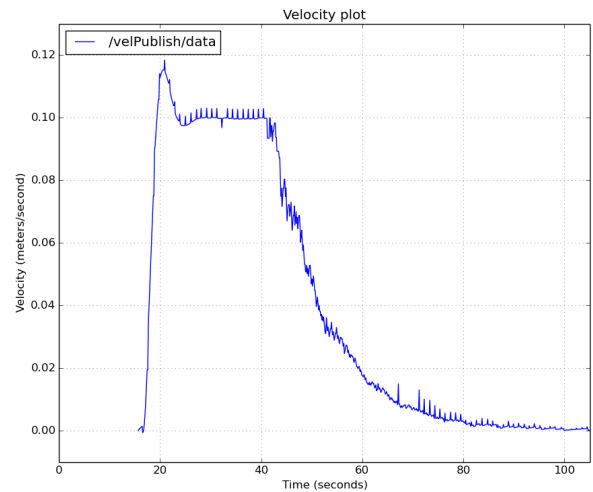The results obtained are presented in the images 9 and 10.

Fig. 9. Altitude plot

Fig. 10. Velocity plot

This test proved that the architecture worked as designed, and the Kalman filter is able to use the depth value whenever the sensor module can not gather viable measurements.

The next test is focused on the evaluation of the performance of the Kalman filter. In order to do so, the following conditions were used:

- Same starting point, controllers' parameters, switching point as the last test;
- Same consideration of the sensor module, where the measurements when the vehicle at an altitude higher than 2.5 m are considered not reliable;
- Usage of the z position given by the dynamic containing a Gaussian error induced;
- Introduction of forced errors on the sensor module measurements.

The forced errors introduced are the following:

- Error introduced when the vehicle reaches 1.5 m of altitude where the sensor module produces a high deviation measurement of 2 m during 5 frames (approximately 0.5 seconds)
- Error introduced when the vehicle reaches 0.5 m of altitude where the sensor module during 5 frames is not able to produce any valuable information; (approximately 0.5 seconds)

The errors introduced are meant to simulate situations where for example a particle or object circulates in front of the camera which can cause an incorrect detection, or cause the calculation of an incorrect measurement.

The figures 11 to 13 illustrate the results obtained.

The figure 11 shows that the overall intended behavior was achieved using the conditions mentioned. These conditions approximate this simulation to the behavior of the profiler navigating in a real environment where the use of sensors with an inherent error is unavoidable.

In the figures 12 and 13 it is possible to notice the errors induced. The error that causes the sensor to retrieve a high deviation measurement for 5 cycles is present at approximately 63 s and the error causing the measurements to not be reliable
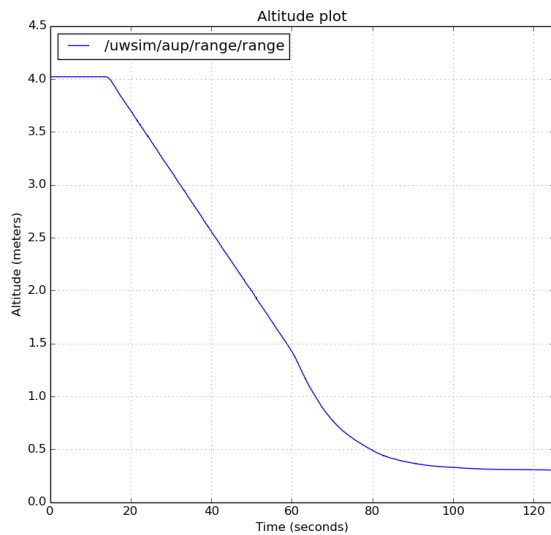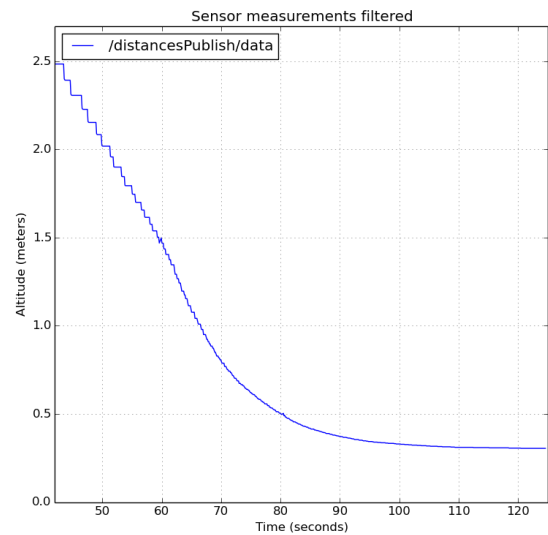
Fig. 11. Altitude plot

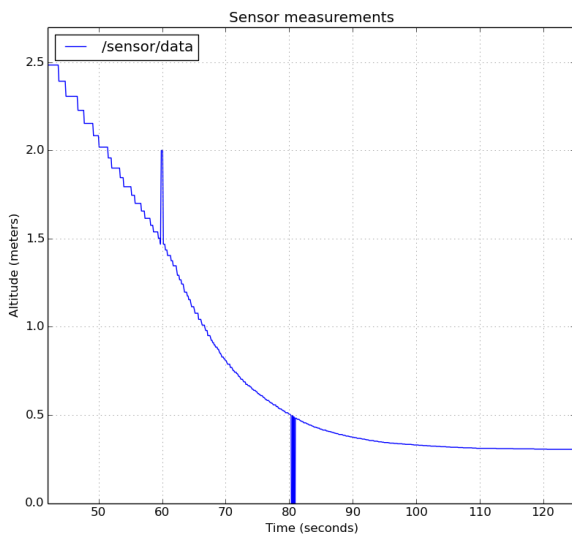Fig. 13. Filtered measurements from the sensor module after 2.5 m

Fig. 12. Measurements gathered by the sensor module including the induced errors after 2.5 m

for 5 cycles occurs at approximately $81s$. It is possible to assert that the Kalman filter eliminated the errors induced by comparing the 2 figures.

VII. Conclusion

This paper reported the development of a system capable of controlling the altitude of an underwater vehicle using computer vision.

Overall, the use of image processing was successfully in order to gather the distance to a target, showing an error of less than 4 % in the range from 0 to 2.5 m, and an average relative error after the application of an average operation between the 2 measurements equal to 1%. The triangulation principle revealed a considerable error inherent to its technology, limiting the range of altitude where it is beneficial to use this sensor module. It was also verified

that the error on the measurements is highly dependent on the calibration, especially the calibration of the angle of the camera and the laser devices.

The use of two phases of control (velocity and position) and the use of a switching point revealed itself very effective in order to be able to tune the controllers' parameters independently according to their objective, allowing a better control on the overall movement. It was given special attention to allowing flexibility on the controllers' parameters depending on the mission's demands. The methods of tuning the controllers' parameters were successfully in determining a good relationship between the mission and vehicle's demands and the smoothness of the switching between the controllers.

Based on the results given for each module and for the tests made with the underwater simulated environmental using UWsim, it is possible to conclude that the main objective of this work was achieved.

References

[1] J. M. Monteiro and N. A. Cruz, "Development of an autonomous underwater profiler for coastal areas," in *OCEANS 2017 8211; Anchorage*, pp. 1–8.

[2] C. Hsin-Hung and L. Chia-Ju, "A simple underwater video system for laser tracking," in *OCEANS 2000 MTS/IEEE Conference and Exhibition. Conference Proceedings (Cat. No.00CH37158)*, vol. 3, pp. 1543–1548 vol.3.

[3] H. Kondo and T. Ura, "Navigation of an auv for investigation of underwater structures," *Control Engineering Practice*, vol. 12, no. 12 SPEC. ISS., pp. 1551–1559, 2004.

[4] C. Cain and A. Leonessa, "Laser based rangefinder for underwater applications," in *2012 American Control Conference (ACC)*, pp. 6190–6195.

[5] F. Takemura, R. Taba, K. Hirayama, S. Tansuriyavong, K. Kawabata, S. Sagara, and K. Ogasawara, "Development of an altitude-keeping system for underwater robots using laser beams," *Artificial Life and Robotics*, vol. 22, no. 4, pp. 405–411, 2017.

[6] P. H. Buscariollo, *Sistema de posicionamento dinâmico baseado em visão computacional e laser*. Thesis, 2008.

[7] H. Kondo, T. Maki, T. Ura, Y. Nose, T. Sakamaki, and M. Inaishi, "Structure tracing with a ranging system using a sheet laser beam," in *Proceedings of the 2004 International Symposium on Underwater Technology (IEEE Cat. No.04EX869)*, pp. 83–88.

[8] G. C. Karras and K. J. Kyriakopoulos, "Localization of an underwater vehicle using an imu and a laser-based vision system," in *2007 Mediterranean Conference on Control & Automation*, pp. 1–6.

[9] N. Hansen, M. C. Nielsen, D. J. Christensen, and M. Blanke, "Short-range sensor for underwater robot navigation using line-lasers and vision," *IFAC-PapersOnLine*, vol. 28, no. 16, pp. 113–120, 2015.

[10] W. Christopher, *An Investigation into Computer Vision Techniques for Underwater Object Recognition*. Thesis, 2006.

[11] P. M. Rodrigues, N. A. Cruz, and A. M. Pinto, *Altitude control of an underwater vehicle based oncomputer vision*. Dissertation, 2018.

[12] T. Fossen, *A Guidance and Control of Ocean Vehicles*. 08 1994.

[13] J. F. J. S. P. Prats, M.; Perez, "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.