Autonomous Navigation for Unmanned Underwater Vehicles: Real-Time Experiments Using Computer Vision

Adrian Manzanilla, Sergio Reyes, Miguel Garcia, Diego Mercado-Ravell, Rogelio Lozano

Autonomous Navigation for Unmanned Underwater Vehicles: Real-Time Experiments Using Computer Vision

Adrian Manzanilla, Sergio Reyes, Miguel Garcia, Diego Mercado, and Rogelio Lozano

***Abstract*—This letter studies the problem of autonomous naviga-tion for unmanned underwater vehicles, using computer vision for localization. Parallel tracking and mapping is employed to localize the vehicle with respect to a visual map, using a single camera, whereas an extended Kalman filter (EKF) is used to fuse the vi-sual information with data from an inertial measurement unit, in order to recover the scale of the map and improve the pose estima-tion. A proportional integral derivative controller controller with compensation of the restoring forces is proposed to accomplish trajectory tracking, where a pressure sensor and a magnetometer provide feedback for depth control and yaw, respectively, while the remaining states are provided by the EKF. Real-time experiments are presented to validate the navigation strategy, using a commer-cial remotely operated vehicle (ROV), the BlueROV2, which was adapted to perform as an autonomous underwater vehicle with the help of the robot operative system.**

***Index Terms*—Marine robotics, autonomous vehicle navigation, AUVs, trajectory tracking, visual-based navigation.**

I. INTRODUCTION

In the last decades, the scientific and tecnological developments in underwaters robotics have played a very important role in many civilian and military applications, such as the attention of disasters, environmental inspection and industrial tasks. Within the Unmanned Underwater Vehicles (UUV) there are several classifications, among which stand out the Remotely Operated Vehicles (ROVs) and Autonomous Underwater Vehicles (AUVs), which can be used for coast patrolling, surveillance, bridge inspection, marine cartography, 3D reconstruction, mapping and tracking [1], etc.

In order to increase the level of autonomy in navigation, localization arises as one of the main challenges. The use of the mathematical model to improve the measurement in some navigation techniques is a common approach, such is the case of dead reckoning, which also makes use of inertial measurements to integrate accelerations and velocities to estimate the position, although they present unbounded error accumulation (drift), and can not be thrust in long term missions. Alternatively, the Doppler Velocity Logs (DVLs) are used to estimate the velocity of the vehicle using acoustic signals directed to the bottom of the ocean and observing the Doppler effect. Nevertheless, integration is required to recover position, resulting in error drift.

Due to these issues, most of the localization methods are based on acoustic sensors, which are mostly heavy, large, expensive and with some other deficiencies such as: a small bandwidth, high latency in water, reduced battery life as they increase the power used to transmit the ultra sound signals, variable sound speed due to temperature and salinity. Some systems track pings from underwater units, measuring the Time-Of-Arrival (TOAs). Different solutions enter this classification depending on the number of transducers and receivers, and the locations of them, for example, using a receive-only Ultra-Short Baseline (USBL) array, to determine the location of the objects and then transmit it to the vehicle using acoustic modem technology [2]. Other related alternatives are the Long-Baseline (LBL) systems, USBL systems and the related Super-Short-Baseline (SSBL). Global Positioning Systems (GPS) Intelligent Buoys (GIB) work as an inverted LBL device where the transducers are replaced by floating buoys, self-positioned by GPS. The tracked position is calculated in real-time at the surface from the TOAs of the acoustic signals sent by the underwater vehicle, and acquired by the buoys [3]–[5].

Vision-based localization is a powerful alternative for autonomous navigation. In some works, the use of visual Simultaneous Localization and Mapping (vSLAM) [6] has been proposed, which can be implemented through the use of conventional cameras, providing a significant amount of information at low cost, such as color, edges, size, texture, shape, etc. This data allows a vehicle to be located in a certain way, using a sequence of captured images that estimates the movement of the vehicle in a mostly static scene. The implementation of SLAM in real underwater environments can be considered an open problem of great interest in robotics [7].

This work was supported by the National Council of Science and Technology CONACyT, Mexico. *(Corresponding author: Diego Alberto Mercado Ravell.)*

A. Manzanilla, S. Reyes, and M. Garcia are with the French-Mexican Laboratory on Computer Science and Control LAFMIA-UMI 3175, CINVESTAV-IPN, Mexico City 07360, Mexico (e-mail: amanzanilla@ cinvestav.mx; sergio.reyes.sanchez1@gmail.com; miguelgarciar@cinvestav.mx).

D. Mercado is with the Research Center in Mathematics CIMAT-Zacatecas, Mexico (e-mail: diego.mercado@cimat.mx).

R. Lozano is with theSorbonne Universitès, Université de Technologie de CompiègneCNRS, Heudiasyc UMR 7253, 60203 Compiègne, France, and also with the , French-Mexican Laboratory on Computer Science and Control LAFMIA-UMI 3175, CINVESTAV-IPN, Mexico City 07360, Mexico.

Within the literature of vSLAM applied to underwater vehicles, we can find in [8] the use of a direct mapping method that estimates in real time a dense reconstruction in 3D of an underwater scene and the position of the vehicle, using a monocular camera. In [9], a technique for loop closures grouping the key points is presented, with the purpose of validating it in an AUV, in marine environments with low textures. Based on insights from [10], they use Speeded-Up Robust Features (SURF) for feature detection, showing that the algorithm is capable of localizing an AUV in different environments. However, this is not a purely camera driven SLAM solution since they incorporate a DVL and sonar into their framework. The authors in [11] propose an underwater SLAM called DolphinSlam, which claims to be able to improve the perception at low resolution, ambiguity of perception and long-term tasks for a 2D plane, using a neural network.

The application of these techniques depends mainly on the computational cost, light and image conditions, i.e., illumination, shadows, blurriness, turbidity, scale, rotation, reflectance, among some other environmental factors. These are negative effects that affect the features detection and alter the estimation of the pixel shift between image frames, leading to errors in the estimation [12].

A qualitative study of visual navigation algorithms, using SLAM in the underwater scenario, is presented in [13], where several SLAM algorithms are tested off-line in data sets captured by ground, aerial and marine vehicles. The authors encounter that the best results for underwater applications are achieved by PTAM (Parallel Tracking and Mapping), ORB-SLAM, COLMAP, g2o and Ceres algorithms. The document further studies the ORB-SLAM algorithm and implements it on the BlueROV2.

In this work, a conventional low-cost monocular camera is used for underwater location, based on the pose estimation obtained from the PTAM algorithm [14], fused with data provided by the Inertial Measurements Unit (IMU) using an Extended Kalman Filter (EKF). This method, originally conceived for augmented reality, is used to estimate the scaleless pose of the camera in an unknown scene. The algorithm can be divided into two separate tasks, one in charge of tracking and the other in mapping, processed in parallel threads in a computer. This algorithm can be employed even in a phone camera [15]. PTAM has been successfully implemented for localization of aerial vehicles, fusing it with inertial measurements to recover the scale and improve the estimation [16]. To our knowledge, this technique has not been applied to underwater vehicles before. Estimation from the EKF is complemented with a pressure sensor and a magnetometer to measure depth and yaw respectively. Once a suitable feedback of the pose of the vehicle is available, a Proportional Integral Derivative controller (PID) with compensation of the restoring forces is used to accomplish trajectory tracking in fully autonomous operation. Real-time experiments are provided to validate the proposed strategy, in a customized BlueROV2 vehicle (see Fig. 1), adapted to perform autonomously. The proposed scheme has potential application in inspection missions for flat structures such as hulls, damns and bridges, as well as coast patrolling, among others.

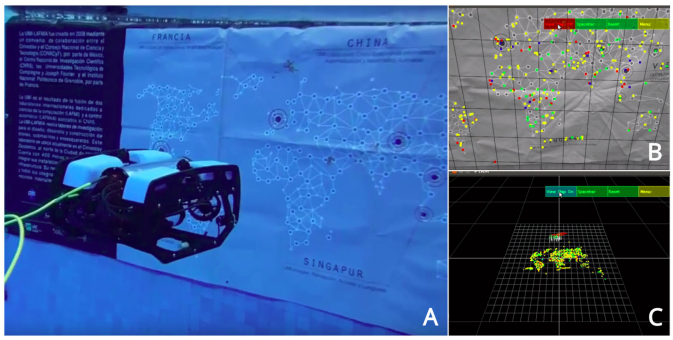

Fig. 1. A) AUV in an autonomous mission using vision based localization. B) A frontal camera in the vehicle is used to obtain the feature points for localization. C) Map generated by the PTAM algorithm for localization.

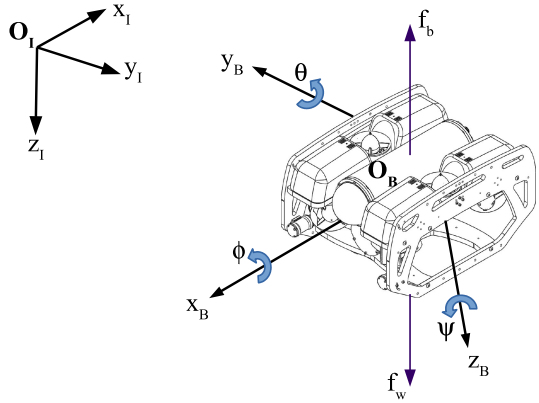

Fig. 2. Coordinate frames. A body fixed frame O_B moves with respect to an inertial frame fixed to the earth O_I.

The outline of the document is as follows: in Section II the general dynamic model of the vehicle is presented. In Section III the implementation of the vision-based localization and the data fusion are described, while in Section IV the control strategy is provided. In Section V the characteristics and modifications of the prototype are discussed. In Section VI, the experimental results of trajectory tracking for fully autonomous operation using visual-based navigation are presented. Finally, a discussion about this work and future developments is detailed in Section VII.

II. Dynamic Model

The dynamic model for an UUV must consider a body reference frame O_B and an inertial one O_I. This model considers the hydrodynamic effects, generalized inertial forces, gravity, buoyancy and force given by the thrusters, as shown in Figure 2. The dynamics of the vehicle can be represented based on the matrix equations proposed by Fossen [17], which are based on the general notation for submarine vehicles populated by [18], and can be written as follows:

$$\begin{aligned} M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) &= \tau + \omega \\ \dot{\eta} &= J(\eta)\nu \end{aligned} \tag{1}$$

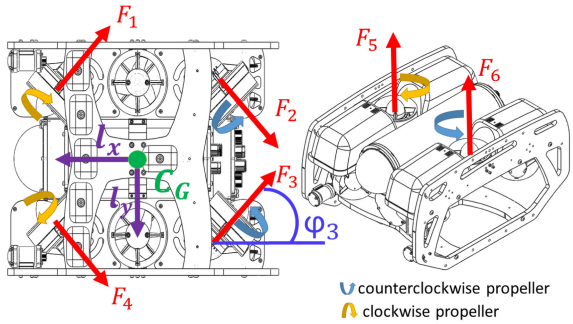

Fig. 3. The forces acting on the submarine are due to 6 thrusters. Thrusters 1 to 4 are used for the rotation movement in yaw and the advance-recoil, while thrusters 5 and 6 are used to raise and descend. Roll and pitch are stable by design. Thrusters 1, 4 and 5 use clockwise propellers, while 2, 3, and 6 are counter-clockwise, cancelling the reactive moments.

where M represents the inertial matrix and is formed by M_{RB} representing the mass and the inertial matrix of the rigid body and M_A denoting the matrix due to the kinetic energy of the fluid (added mass and added inertia), $C(\nu)$ describes the terms of Coriolis and is the sum of the Coriolis matrix of aggregate mass $C_A(\nu)$ and the Coriolis matrix of the rigid body $C_{RB}(\nu)$, $D(\nu)$ describes the hydrodynamic damping matrix and is composed by the body friction $D_S(\nu)$ and the potential damping $D_P(\nu)$, wave drift damping $D_W(\nu)$ and damping due to vortex shedding $D_M(\nu)$, also $g(\eta)$ represents the vector of hydrostatic forces, $\tau = [X, Y, Z, K, M, N]^T$ represents the vector of forces and moments in the body frame and ω defines the disturbance vector, $J(\eta)$ is the kinematic transformation between the body and the inertial frame, $\eta = [x, y, z, \phi, \theta, \psi]^T$ represents the vehicle's position and orientation in the inertial frame and $\nu = [u, v, w, p, q, r]^T$ denotes the linear and angular velocities in the body frame.

A. The BlueROV2 Underwater Vehicle

For the particular case of the BlueROV2, it is composed of six thrusters from which 1 to 4 are used for the rotational movement in yaw, as well as the forward-reverse and lateral movements, while to raise and descend, the thrusters 5 and 6 are used. The configuration of the thrusters allows the vehicle to move freely within these four degrees of freedom, as illustrated in Figure 3. By design the submarine robot has symmetry in two of its planes and is stable mechanically in the roll and pitch angles, thanks to a careful selection of the center of buoyancy, i.e., $\phi, \theta \approx 0$, which implies that the submarine has the ability to return to steady state after a disturbance (movement generated by water flow), therefore $K, M, p, q, \dot{p}, \dot{q} \approx 0$. From now on, we consider only the reduced state vector $\bar{\eta} = [x, y, z, \psi]^T$ and the reduced force vector $\bar{\tau} = [X, Y, Z, N]^T$.

From equation (1), using the short notation $\sin(a) = s(a)$ and $\cos(a) = c(a)$, following the methodology as in [19] and considering only the reduced system, the translational and rotational dynamics particularly for this robot can be expressed as follows

$$X = (m - X_{\dot{u}})\dot{u} + X_{\dot{v}}\dot{v} - (my_g - X_{\dot{r}})\dot{r} + (Y_{\dot{v}}v + mw)r - (X_u)u + f_B s(\theta) \quad (2)$$

$$Y = (m + Y_{\dot{v}})\dot{v} + (mx_g - Y_{\dot{q}})\dot{r} - (X_{\dot{u}}u - (mu))r - (Y_v)u - f_B c(\theta)s(\phi) \quad (3)$$

$$Z = (m - Z_{\dot{w}}) - (Z_w)w - f_B c(\theta)c(\phi) \quad (4)$$

$$N = (-my_g - X_{\dot{r}})\dot{u} + (mx_g - Y_{\dot{r}})\dot{v} + (I_{zz} - N_{\dot{r}})\dot{r} - (Y_{\dot{v}}v - mv)u + (X_{\dot{u}}u + mu)v - (N_r)r \quad (5)$$

where X_u, Y_v, Z_w, N_r are the linear damping coefficients; $X_{\dot{k}}, Y_{\dot{k}}, Z_{\dot{k}}, K_{\dot{k}}, M_{\dot{k}}, N_{\dot{k}}$ with $\dot{k} : \dot{u}, \dot{v}, \dot{w}, \dot{p}, \dot{q}, \dot{r}$ represent the hydrodynamic added mass coefficients; m defines the mass of the vehicle. In addition, $r_g = [x_g, y_g, z_g]^T$ form the distance vector from the origin of the body fixed frame O_B to the center of gravity of the robot C_G; $f_B = W - B$ stands for the restoring forces vector resulting from the difference between the vehicle's weight W and the buoyancy force B. Finally, I_{zz} is the inertia moment in z.

Taking into account the physical properties of the vehicle and that only slow motions below $2m/s$ are considered, moreover, the origin of the body fixed frame O_B coincides with the center of gravity C_G, and is colinear with the main axes of the inertial reference frame O_I, the reduced vector of forces and moments $\bar{\tau}$ that acts in the vehicle can be expressed as:

$$\bar{\tau} = \begin{bmatrix} -F_1 c(\varphi_1) - F_2 c(\varphi_2) - F_3 c(\varphi_3) - F_4 c(\varphi_4) \\ F_1 s(\varphi_1) - F_2 s(\varphi_2) + F_3 s(\varphi_3) - F_4 s(\varphi_4) \\ F_5 + F_6 \\ \xi \end{bmatrix} \quad (6)$$

with $\xi = l_x(F_1 c(\varphi_1) - F_2 c(\varphi_2) - F_3 c(\varphi_3) + F_4 c(\varphi_4)) + l_y (F_1 s(\varphi_1) + F_2 s(\varphi_2) - F_3 s(\varphi_3) - F_4 s(\varphi_4))$. $F_{i:1,\dots 6}$ are the forces generated by each thruster; $\varphi_{i:1,\dots 4}$ are the angles between the x_B axis and the force $F_{i:1,\dots 4}$ applied in the AUV; l_x and l_y are the distances between the horizontal and vertical thrusters with the center of gravity C_G (see Figure 3). Note that all propellers are in the same plane with the C_G.

III. Vision-Based Localization

In order to accomplish fully autonomous navigation underwater, real-time localization arises as one of the main challenges. Given that acoustic sensors, which are the preferred solution, are normally heavy, large and expensive, computer vision presents an interesting alternative.

PTAM is an algorithm developed to estimate the pose of a camera in an unknown environment and was originally conceived for augmented reality (AR), the main advantages of this algorithm are that it does not require special markers, previously known maps or structured scenarios, and it only requires a single camera. The operation basically consists of separating the tracking and mapping into two separate tasks, processed in parallel, generating a 3D map of the observed characteristic points

[14]. The generated map estimates a dominant visual plane from the feature points while refining the map, having a fast, accurate and robust tracking.

The monocular vision algorithm provides a scaleless estimation of the pose of the camera, without drift over time, at a slow rate. With the purpose of recovering the scale of the position and to improve the estimation, the PTAM algorithm is fused with inertial measurements from the IMU (accelerometers and gyroscopes) using an EKF. This allows as well to increment the estimation rate, taking advantage of the faster IMU measurements. In order to do so, the sensor fusion libraries from [20], [21] are used. This libraries where developed for aerial vehicles and are also available as a ROS package. The PTAM estimation is used as the update sensor, while the accelerations and angular speeds are used directly in the prediction step, running at a faster rate. Another interesting advantage of this approach is that it auto-calibrates itself since it does not only attempt to estimate the pose and velocity, it also estimates the missing scale and the intrinsic rotations between the different coordinate frames, ensuring long-term operations without error drift (only spacial drift may be present from the visual algorithm).

IV. Control Strategy

Provided that a suitable state feedback is available from the IMU, pressure sensor and vision algorithm with the EKF, the control objective is to track a time varying trajectory for monitoring and inspection applications. It is common to use sensor measurements referred to the Euler angles (for orientation) and the inertial frame (for the position). Then, the reduced dynamic model of an AUV is expressed in the frame fixed to the ground using the kinematic transformations proposed by Fossen [17], and is represented by the following equation:

$$M_\eta(\bar{\eta})\ddot{\bar{\eta}} + C_\eta(\nu,\bar{\eta})\dot{\bar{\eta}} + D_\eta(\nu,\bar{\eta})\dot{\bar{\eta}} + g_\eta(\bar{\eta}) = \bar{\tau}_\eta \qquad (7)$$

where the subindex η represents coordinates in the inertial frame. In this section, the implementation of a PID controller is approached for two cases, depth and yaw stabilization, and for trajectory tracking in x and y position. The PID controller with compensation of the restoring forces is implemented in the underwater robot for regulation, and is given by:

$$\bar{\tau}_\eta = g_\eta(\bar{\eta}) - J^T(\bar{\eta})\left[K_p e + K_d \dot{e} + K_i \int e\right] \qquad (8)$$

where $e = \bar{\eta} - \bar{\eta}_d$ defines the error and the feedback gains K_p, K_d and K_i are diagonal positive definite matrices such that $K_{j:\{p,d,i\}} = diag(k_{jx}, k_{jy}, k_{jz}, k_{j\psi})$. The proposed controller renders the system asymptotically stable, as can be probed by Lyapunov's second method of stability and LaSalle's invariance principle (see [17], chapter 12).

V. Experimental Platform

The robot used for this work is the BlueROV2, by Blue Robotics, presented in Figure 4. It is a small-size ROV of approximately $45.71 \times 33.81 \times 22.1$cm [22]. The electronic components are installed in a hermetic acrylic tube, for the battery there is another acrylic tube which has a direct connection to the electronic tube. The prototype has the capacity of sinking up to 100 m depth according to the manufacturer. It is equipped with six Blue Robotics T200 thrusters configured as shown in Figure 3. The main physical parameters of the prototype are presented in Table I, where SI base units are adopted.

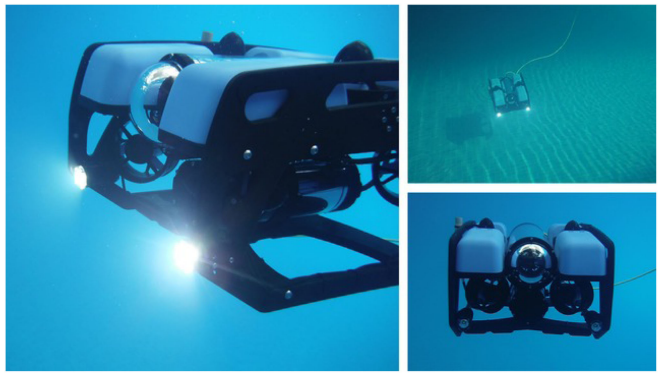

Fig. 4. Underwater vehicle BlueROV2 during a mission. The BlueROV2 was modified in this work to accomplish fully autonomous operation as an AUV instead of remotely operation as a ROV.

TABLE I
Physical Parameters for the Vehicle

param.	m	W	B	$l_x;\ l_y$	$K_{\dot{p}},\ M_{\dot{q}},\ N_{\dot{r}}$
value	11	107.91	109.87	0.156; 0.11	-0.12
param.	I_{zz}	$X_{\dot{u}}$	$Y_{\dot{v}}$	$Z_{\dot{w}}$	r_g
value	0.16	-5.5	-12.7	-14.57	$[0,0,0]^T$

The BlueROV2 uses the open source software ArduSub installed on a PixHawk autopilot, furthermore, it runs ROS in a Raspberry Pi on-board computer, while the MAVLink protocol is used to communicate the on-board computer and the autopilot, to obtain the sensors information. Via ROS the vehicle is connected to a ground station using a Fathom-X tether interface trough 100 m of wire providing communication with a bandwidth of 80 Mb/s. Video from a frontal camera, with a resolution of 640×480, is transmitted to the ground station at a frame rate of 15fps. The vision algorithm, along with the EKF and the control strategy are computed in the ground station and return the Pulse-Width Modulation (PWM) value to the vehicle's thrusters.

Figure 5 displays the main ROS nodes running in the ground station, where the BlueROV2 node provides the sensor measurements from the vehicle, including the video stream which is fed to the PTAM node. The sensor fusion node reads the pose estimation from PTAM and merges it with inertial measurements from the embedded IMU. The control node was developed for this work and uses the pose estimation from the EKF, along with the pressure and magnetometer measurement to accomplish trajectory tracking using a PID with compensation of the restoring forces, and delivers the PWM signals to the ROV's thrusters. Real-time visualization of the states is possible thanks to the rqt_plot node, while online parameter tuning is available through a Graphical User Interface (GUI).

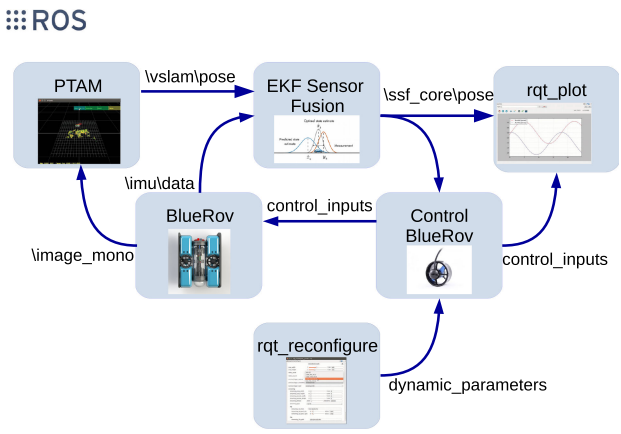

Fig. 5. Main ROS nodes used in the implementation. The BlueROV2 node provides information from the embedded sensors in the vehicle. The PTAM node takes the video stream from the BlueROV2 and estimates the scaleless pose of the vehicle w.r.t. the visual map. The sensor fusion node merges together the PTAM estimation along with inertial measurements from the IMU to improve the pose estimation. The control node implements a PID controller to accomplish autonomous trajectory tracking. Finally, the rqt_reconfigure and rqt_plot nodes are used respectively for on-line parameter tuning and real-time visualization.

TABLE II
CONTROL GAINS

param.	k_{px}	k_{dx}	k_{ix}	k_{py}	k_{dy}	k_{iy}
value	130 $\frac{Kg}{s^2}$	20 $\frac{Kg}{s}$	4 $\frac{Kg}{s^3}$	130 $\frac{Kg}{s^2}$	15 $\frac{Kg}{s}$	4 $\frac{Kg}{s^3}$
param.	k_{pz}	k_{dz}	k_{iz}	$k_{p\psi}$	$k_{d\psi}$	$k_{i\psi}$
value	400 $\frac{Kg}{s^2}$	150 $\frac{Kg}{s}$	3 $\frac{Kg}{s^3}$	4 $\frac{Nm}{rad}$	30 $\frac{Nms}{rad}$	0.5 $\frac{Nm}{s\ rad}$

VI. REAL-TIME EXPERIMENTS

In this section, the validity of the vision algorithm for underwater applications is demonstrated. The main objective is to autonomously track a trajectory, using vision-based localization and applying the control law given by (8).

The test was performed in an olympic pool whose dimensions are $(25 \times 50 \times 1.6m)$. It consisted in stabilizing yaw and depth at a constant reference $\psi_d = 110°$, $z_d = -35$cm, respectively, and performing trajectory tracking in (x, y). For the trajectory tracking of the longitudinal movements the desired signal is given by: $x_d = 0.7 * \cos(2\pi t/25)$ and $y_d = 0.5 * \sin(2\pi t/25)$. Several perturbations were applied during the test to further test the robustness of the proposed algorithms, at seconds 310, 400 and 475. The controller parameters were heuristically adjusted and are presented in Table II, using SI base units. The covariance matrix for the EKF update step is provided by the PTAM algorithm, while for the prediction step the noise variances for the accelerometers σ_a^2; gyroscopes σ_ω^2 and their bias σ_{ba}^2 respectively $\sigma_{b\omega}^2$ are taken into account, i.e., $\sigma_a^2 = 0.25\frac{m^2}{s^3}$, $\sigma_{ba}^2 = 0.025\frac{m^2}{s^5}$, $\sigma_\omega^2 = 0.013\frac{rad^2}{s}$ and $\sigma_{b\omega}^2 = 0.0013\frac{rad^2}{s^3}$ respectively. For more details about the EKF implementation see [20], [21].

In Fig. 6, we can observe the x, y and z position of the vehicle during the path tracking, as measured by the PTAM (pink dotted) and estimated by the EKF (dashed blue) against the desired trajectory (solid red line). We can appreciate that the proposed strategy is good enough to accomplish this task, with bounded error, in spite of unexpected disturbances at time 310s, 400s and 475s. The orientation of the AUV is depicted in Fig. 7. There we can appreciate the mechanical stability property of the vehicle in roll ϕ and pitch θ angles, achieved through a suitable selection of thr center of buoyancy. An aerial view of the position estimation and the behavior of the controller before the perturbations can be seen in Fig. 8. We can note the action of the EKF (dashed blue) in order to recover the missing scale in the PTAM algorithm (dotted pink), and improve the estimation. Finally, the control inputs are shown in Figure 9.

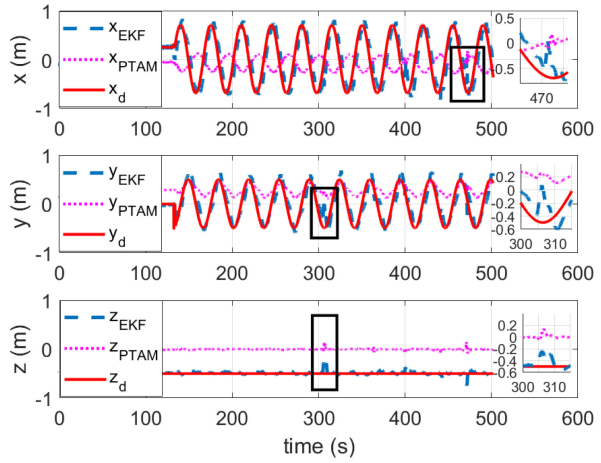

Fig. 6. Autonomous trajectory tracking using vision-based localization: position. Desired (solid red line); PTAM measurement (pink dotted line) and EKF estimation (dashed blue line) position; x coordinate response on top, y coordinate on the center and z coordinate on bottom. Important perturbations were applied at time 310s, 400s and 475s.

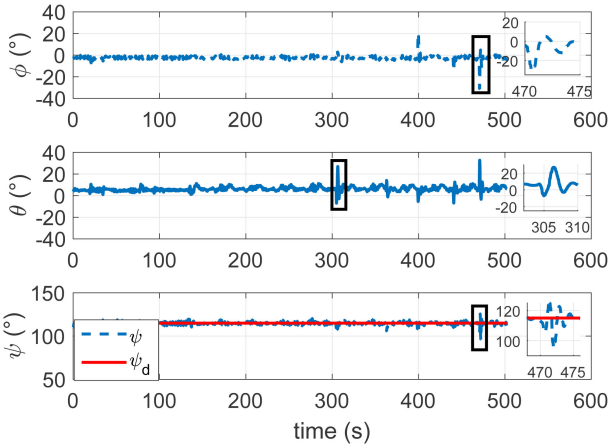

Fig. 7. Autonomous trajectory tracking using vision-based localization: orientation. Desired (solid red line) vs IMU measurements (dashed blue line). The vehicle is mechanically stabilized by design in roll ϕ and pitch θ angles, thanks to a careful selection of the center of buoyancy.

The behavior of the AUV during this experiment can be appreciated in the following video: https://www.youtube.com/watch?v=v0Uj-IuztIs. From there, we can observe the system performance and robustness while facing unexpected disturbances such as important rotations and displacements, or camera occlusions, causing the vehicle to deviate from the trajectory. As expected, the use of an EKF with data from the IMU helps to recover the map.

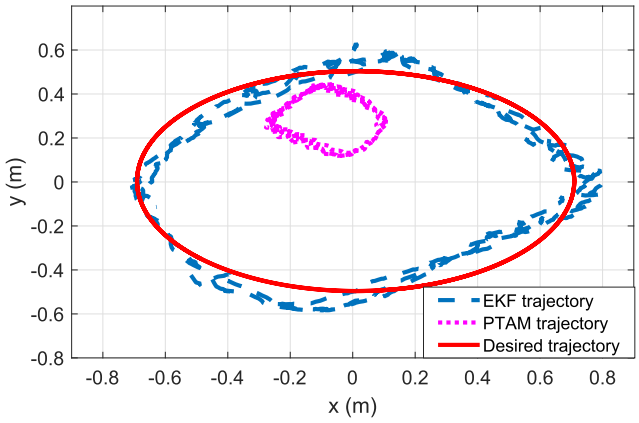

Fig. 8. Autonomous trajectory tracking using vision-based localization: aerial view before perturbations. Desired trajectory (solid red line); PTAM measurement (dotted pink line) and EKF estimation (dashed blue line).

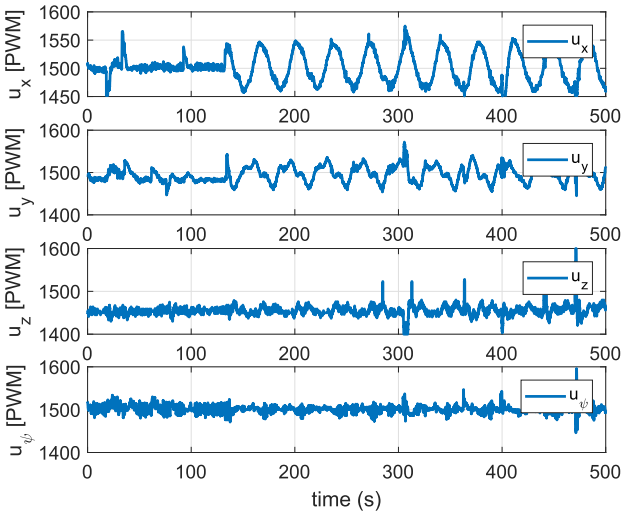

Fig. 9. Autonomous trajectory tracking using vision-based localization: control inputs. The motors operate in a PWM range from 1100 to 1500 in one direction and from 1500 to 1900 in the other.

VII. Conclusions and Future Work

In this work, fully autonomous trajectory tracking was successfully achieved using PTAM and IMU data merged via an EKF. The PID controller with compensation of restoring forces showed a good performance, even when unexpected disturbances were present. The implementation of different control laws is contemplated as future work to improve the system's behavior, for example, adaptive control may be used to compensate unknown dynamics, or predictive control to take care of potential communication delays in a tetherless vehicle.

The use of PTAM in an aquatic environment was validated in real time experiments, exposing potential use for underwater applications in real world scenarios such as bridge, dams and hull inspection; coast patrolling; monitoring in shallow waters; etc. Field experiments are envisioned to demonstrate the limits of the algorithms under different real conditions.

The main contribution of this work is then, the implementation of the PTAM algorithm to perform autonomous navigation of an underwater vehicle, and the validation on real-time experiments. We are currently working on the implementation of the algorithms fully embedded in the vehicle, with this we intend to avoid the use of a long tether, increasing the level of autonomy and the coverage of the vehicle.

Acknowledgment

The author would like to thank the *Dirección de Desarrollo y Fomento Deportivo del IPN, Mexico*, for facilitating the use of the swimming pool.

References

[1] J. Melo and A. Matos, "Survey on advances on terrain based navigation for autonomous underwater vehicles," *Ocean Eng.*, vol. 139, pp. 250–264, 2017.

[2] B. Megdal and H. Scholze, "Underwater acoustic navigation systems and methods," U.S. Patent 8654610, Feb. 18, 2014.

[3] J. Snyder, "Doppler velocity log (DVL) navigation for observation-class rovs," in *Proc. IEEE OCEANS*, 2010, pp. 1–9.

[4] J. Melo and A. C. Matos, "Tracking multiple autonomous underwater vehicles," *Auton. Robots*, vol. 43, pp. 1–20, 2018.

[5] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE J. Ocean. Eng.*, vol. 39, no. 1, pp. 131–149, Jan. 2014.

[6] T. Taketomi, H. Uchiyama, and S. Ikeda, "Visual slam algorithms: A survey from 2010 to 2016," *IPSJ Trans. Comput. Vis. Appl.*, vol. 9, no. 1, pp. 1–11, 2017.

[7] F. Guth, L. Silveira, S. Botelho, P. Drews, and P. Ballester, "Underwater slam: Challenges, state of the art, algorithms and a new biologically-inspired approach," in *Proc. IEEE Biomed. Robot. Biomechatronics*, 2014, pp. 981–986.

[8] A. Concha, P. Drews, Jr, M. Campos, and J. Civera, "Real-time localization and dense mapping in underwater environments from a monocular sequence," in *Proc. IEEE OCEANS*, Genova, Italy, 2015, pp. 1–5.

[9] P. L. Negre, F. Bonin-Font, and G. Oliver, "Cluster-based loop closing detection for underwater slam in feature-poor regions," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 2589–2595.

[10] R. Garcia and N. Gracias, "Detection of interest points in turbid underwater images," in *Proc. IEEE OCEANS*, 2011, pp. 1–9.

[11] L. Silveira *et al.*, "An open-source bio-inspired solution to underwater slam," *IFAC-PapersOnLine*, vol. 48, no. 2, pp. 212–217, 2015.

[12] M. O. Aqel, M. H. Marhaban, M. I. Saripan, and N. B. Ismail, "Review of visual odometry: Types, approaches, challenges, and applications," *SpringerPlus*, vol. 5, no. 1, 2016, Art. no. 1897.

[13] C. Kahlefendt, "Implementation and evaluation of monocular SLAM for an underwater robot," Master's thesisRobot. Autom. Lab, Univ. Western Aust., Perth, WA, Australia, Jan. 2018.

[14] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in *Proc. 6th IEEE ACM Int. Symp. Mixed Augmented Reality*, 2007, pp. 225–234.

[15] G. Klein and D. Murray, "Parallel tracking and mapping on a camera phone," in *Proc. 8th IEEE Int. Symp. Mixed Augmented Reality*, 2009, pp. 83–86.

[16] G. R. Rodríguez-Canosa, S. Thomas, J. Del Cerro, A. Barrientos, and B. MacDonald, "A real-time method to detect and track moving objects (DATMO) from unmanned aerial vehicles (UAVS) using a single camera," *Remote Sens.*, vol. 4, no. 4, pp. 1090–1111, 2012.

[17] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Hoboken, NJ, USA: Wiley, 2011.

[18] The Society of Naval Architects and Marine Engineers, "Nomenclature for treating the motion of a submerged body through a fluid," Technical and Research Bulletin No. 1–5, Apr. 1950, pp. 1–15.

[19] A. Manzanilla, M. Garcia, R. Lozano, and S. Salazar, "Design and control of an autonomous underwater vehicle (AUV-UMI)," in *Marine Robotics and Applications*. New York, NY, USA: Springer, 2018, pp. 87–100.

[20] S. Weiss and R. Siegwart, "Real-time metric state estimation for modular vision-inertial systems," *IEEE Int. Conf. Robot. Autom.*, 2011, pp. 4531–4537.

[21] S. Weiss, M. W. Achtelik, M. Chli, and R. Siegwart, "Versatile distributed pose estimation and sensor self-calibration for autonomous MAVs," *IEEE Int. Conf. Robot. Autom.*, 2012, pp. 31–38.

[22] Blue Robotics, "Bluerov2," 2017. [Online]. Available: http://docs.bluerobotics.com/brov2/. Accessed on: Sep. 7, 2018.