AUV pipeline following by artificial vision

Eduardo Campos Mercado, Frédéric Comby, Vincent Creuze, Jorge Torres Munoz, Elaba Antonio, Olivier Strauss, Jesus Monroy Anieva

AUV pipeline following by artificial vision

E. Campos,[2] F. Comby,[3] V. Creuze,[3] J. Torres,[1] E. Antonio,[1] O. Strauss, [3] J. Monroy,[1]

[1] Automatic Control Department, CINVESTAV, D.F., México
[2] Universidad Politécnica de Pachuca, Hidalgo, México
[3] LIRMM, CNRS-Université Montpellier 2, Montpellier, France
E-mail: ecampos@upp.edu.mx, vincent.creuze@lirmm.fr

***Abstract*—This paper presents the modeling, sensing and control of an autonomous underwater vehicle (AUV) developed in order to perform pipeline following. The pipeline detection and the relative angular position of the vehicle with respect to the pipeline are obtained by an artificial vision algorithm. The Proportional-Derivative (PD) paradigm with gravity compensation is used for the control of the vehicle. An analysis of performances is presented on Real-time experiments.**

***Keywords*—Underwater vehicle, PD Controller, Control, Underwater vision, Real-time experiments.**

I. INTRODUCTION

The development of Remotely Operated Vehicles (ROV) and Autonomous Underwater Vehicles (AUV) has known a great growth due to their wide application. Nowadays these vehicles are used to perform marine exploration tasks, hydrographic surveys, monitoring, among others (for example [1], [6], [7]). A classical example is the inspection of marine structures, which is currently performed by ROV type submarines equipped with a camera, allowing the pilot to control the movements of the vehicle with respect to the object of interest. However, this task is not easy and requires trained operators to handle the vehicle. A better option for this kind of tasks is the use of AUV submarines. In this case, a sensing and control strategy is implemented, so as to autonomously control the movements of the vehicle. Given the nonlinear dynamics and the difficulty to identify the hydrodynamic parameters involved in vehicle dynamics, these topics still remain of interest for research (see [11], [8], [16]).

The mini submarine used in this work, called Lirmia 2, offers the capability of being used either as a ROV or as an AUV, depending on the performed task. In the present case, we will use it as an AUV to track a pipeline using an artificial vision algorithm. One of the most widely used technique for the identification of ducts is based on the application of the Hough's transformed, with which straight lines can be identified, corresponding to the edges of the duct. The drawback of this method is that it is not possible to identify the lines corresponding to the edges of the pipe when this latter is formed by very long curved lines (see [2], [10]). For this reason, we have implemented another artificial vision algorithm robust to the different phenomena induced by the underwater image acquisition process as well as by the disposition of duct.

The control strategy implemented in our vehicle is based on a PD control technique with gravity compensation, taking into account the restoration forces and moments generated by buoyancy and weight of the vehicle. This paper is organized as follows: In section II, we briefly describe the dynamic model of the mini-submarine Lirmia 2. The artificial vision algorithm is presented in section III. In section IV, we describe the control strategy. The real-time experiments are presented in section V. Finally, concluding remarks and future works are given in section VI.

II. DYNAMICAL MODEL

The Lirmia 2 submarine is depicted in Figure 1, with its body fixed frame (O_b, x_b, y_b, z_b). The center (O_b) of this latter frame corresponds to the center of gravity of the vehicle, and its axes are aligned with the main axes of symmetry of the vehicle. The motion in the horizontal plane is referred as $surge$ (along x_b axis) and $sway$ (along y_b axis), while $heave$ represents the vertical motion (along z_b axis). Roll, pitch, and yaw, denoted (ϕ, θ, ψ), are the Euler angles describing the orientation of the vehicle's body fixed frame with respect to the earth-fixed frame (O_I, x_I, y_I, z_I), while (x, y, z) denote the coordinates the center of the body-fixed frame in the earth fixed frame. The propulsion system consists in six thrusters, as depicted in Figure 2, which generate the rotational and translational motion. Concerning the rotational motion of this vehicle, yaw control is performed through differential speed control of the thrusters 3, 4, 5 and 6. Pitch control is obtained similarly using thrusters 1 and 2, whereas the roll motion is unactuated. On the other hand, the translational motion of the z axis is regulated by decreasing or increasing the combined speed of thrusters 1 and 2; similarly, the translational motions along the x_b and y_b axes are obtained by using thrusters 3, 4, 5, 6 and by controlling the yaw angle.

The dynamics of the vehicle expressed in the body-fixed frame can be written in a vectorial setting according to Fossen [14]:

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{g}(\boldsymbol{\eta}) = \boldsymbol{\tau} + \boldsymbol{w}_e \tag{1}$$

$$\dot{\boldsymbol{\eta}} = J(\boldsymbol{\eta})\boldsymbol{\nu} \tag{2}$$

where $\boldsymbol{M} \in \mathbb{R}^{6\times6}$ is the inertia matrix, $\boldsymbol{C}(\boldsymbol{\nu}) \in \mathbb{R}^{6\times6}$ is the coriolis-centripetal matrix, $\boldsymbol{D}(\boldsymbol{\nu}) \in \mathbb{R}^{6\times6}$ represents the hydrodynamic damping matrix, $\boldsymbol{g}(\boldsymbol{\eta}) \in \mathbb{R}^{6\times1}$ describes the vector of gravitational/buoyancy forces and moments, $\boldsymbol{\tau} = (\boldsymbol{\tau}_1, \boldsymbol{\tau}_2)^T = ((\tau_X, \tau_Y, \tau_Z), (\tau_K, \tau_M, \tau_N))^T \in \mathbb{R}^{6\times1}$ is the vector of control inputs; $\boldsymbol{w}_e \in \mathbb{R}^{6\times1}$ is the vector of

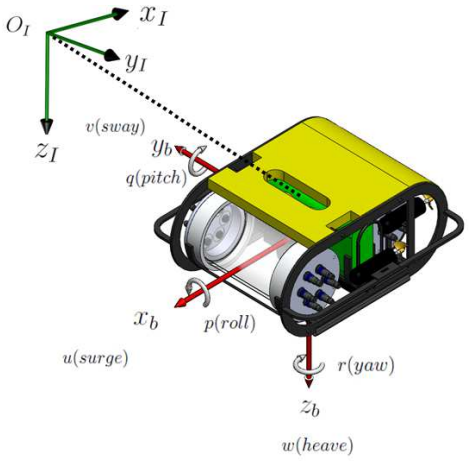


Figure 1: The Lirmia 2 vehicle, its body fixed frame (O_b, x_b, y_b, z_b), and the earth-fixed frame (O_I, x_I, y_I, z_I).

disturbances; $\boldsymbol{\nu} = (\boldsymbol{\nu}_1, \boldsymbol{\nu}_2)^T = ((u, v, w), (p, q, r))^T \in \mathbb{R}^{6\times1}$ denotes the linear and angular velocity vector in the body-fixed frame; $\boldsymbol{\eta} = (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2)^T = ((x, y, z), (\phi, \theta, \psi))^T \in \mathbb{R}^{6\times1}$ is the position and attitude vector decomposed in the earth-fixed frame, and $\boldsymbol{J}(\boldsymbol{\eta}) \in \mathbb{R}^{6\times6}$ is the transformation matrix between body and earth-fixed frames (more details, see [5],[9]).

A. Gravity/Buoyancy forces and torques

According to Archimedes' principle, the buoyancy force f_B applies on the center of buoyancy and acts in the opposite direction of vehicle weight f_W. This leads to:

$$f_B = -\begin{bmatrix} 0 \\ 0 \\ \rho g \nabla \end{bmatrix} \qquad f_W = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{3}$$

where ρ represents the fluid density, g the gravitational acceleration, ∇ the displaced fluid volume and m the mass of the vehicle. Now, the weight and buoyancy forces can be transformed to the body fixed coordinates system by:

$$F_B = J_1(\eta_2)^{-1} f_B \qquad F_W = J_1(\eta_2)^{-1} f_W \tag{4}$$

Considering that $W = mg$ and $B = \rho g \nabla$ and using the zyx-convention for navigation and control application (see, [15]), then the matrix transformation is $J_1(\eta_2) = R_{z,\psi} R_{y,\theta} R_{x,\phi}$. Therefore, the forces with respect to the body fixed frame are written as:

$$F_B = \begin{bmatrix} Bsin(\theta) \\ -Bcos(\theta)sin(\phi) \\ -Bcos(\theta)cos(\phi) \end{bmatrix} \quad F_W = \begin{bmatrix} -Wsin(\theta) \\ Wcos(\theta)sin(\phi) \\ Wcos(\theta)cos(\phi) \end{bmatrix} \tag{5}$$

Thus, the restoring forces acting on the vehicle are f_g=F_B+F_W, this is

$$f_g = \begin{bmatrix} (B-W)sin(\theta) \\ (W-B)cos(\theta)sin(\phi) \\ (W-B)cos(\theta)cos(\phi) \end{bmatrix} \tag{6}$$

On the other hand, the restoring moments are described by the following equation

$$m_g = r_w \times F_W + r_b \times F_B \tag{7}$$

where $r_w = [x_w, y_w, z_w]^T$ and $r_b = [x_b, y_b, z_b]^T$ represent respectively the positions of the center of gravity of the vehicle (CG) and of the center of buoyancy (CB). Based on the design of the vehicle and in order to reduce computations, the origin of the coordinate system fixed to the body is placed in the center of gravity, so that $r_w = [0, 0, 0]^T$; while the center of buoyancy is $r_b = [0, 0, -z_b]^T$. For practical reasons, the buoyancy force is greater than the weight, that is $W - B = -fb$, but $-f_b$ should be smaller than the force produced by the thrusters. Then from equation (6) and (7), we have:

$$\boldsymbol{g}(\boldsymbol{\eta}) = \begin{bmatrix} f_g \\ m_g \end{bmatrix} = \begin{bmatrix} f_b sin(\theta) \\ -f_b cos(\theta)sin(\phi) \\ -f_b cos(\theta)cos(\phi) \\ -z_b B cos(\theta)sin(\phi) \\ -z_b B sin(\theta) \\ 0 \end{bmatrix} \tag{8}$$

B. Forces and torques generated by the thrusters

Figure 2 shows the forces generated by the thrusters acting on the mini submarine, these are described relative to the body-fixed coordinate system, as

$$\hat{\boldsymbol{f}}_{\mathbf{1}} = \begin{bmatrix} 0 \\ 0 \\ f_1 \end{bmatrix} ; \hat{\boldsymbol{f}}_{\mathbf{2}} = \begin{bmatrix} 0 \\ 0 \\ f_2 \end{bmatrix} ; \hat{\boldsymbol{f}}_{\mathbf{3}} = \begin{bmatrix} f_3 \\ 0 \\ 0 \end{bmatrix}$$

$$\hat{\boldsymbol{f}}_{\mathbf{4}} = \begin{bmatrix} f_4 \\ 0 \\ 0 \end{bmatrix} ; \hat{\boldsymbol{f}}_{\mathbf{5}} = \begin{bmatrix} f_5 \\ 0 \\ 0 \end{bmatrix} ; \hat{\boldsymbol{f}}_{\mathbf{6}} = \begin{bmatrix} f_6 \\ 0 \\ 0 \end{bmatrix}$$

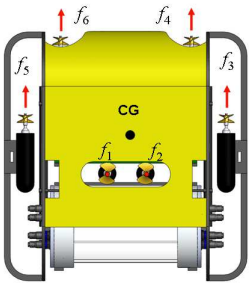


Figure 2: Forces f_i $\forall i = [1...6]$ generated by the six thrusters of the vehicle. Position of the center of gravity, CG.

summarizing and using the notation of [13], we have that

$$\boldsymbol{\tau}_{\mathbf{1}} = \begin{bmatrix} \tau_X \\ \tau_Y \\ \tau_Z \end{bmatrix} = \begin{bmatrix} f_3 + f_4 + f_5 + f_6 \\ 0 \\ f_1 + f_2 \end{bmatrix} \tag{9}$$

and the body-fixed torques generated by the above forces, are defined as

$$\boldsymbol{\tau}_2 = \sum_{i=1}^{6} l_i \times \hat{\boldsymbol{f}}_{\boldsymbol{i}} \tag{10}$$

where $l_i = (l_{ix}, l_{iy}, l_{iz})$ is the position vector of the force $\hat{\boldsymbol{f}}_{\boldsymbol{i}}$ $\forall\ i = 1, .., 6$, with respect to the body-fixed reference frame. Then the torques generated by the thrusters are described as

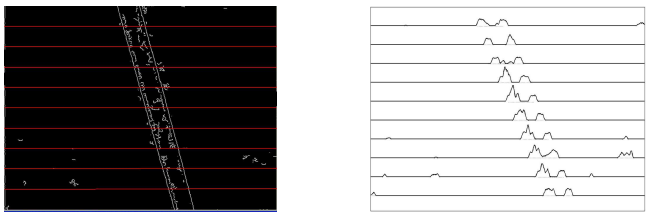
$$\boldsymbol{\tau_2} = \begin{bmatrix} \tau_K \\ \tau_M \\ \tau_N \end{bmatrix} = \begin{bmatrix} 0 \\ l_{1x}(f_1 + f_2) \\ l_{3y}(f_3 - f_5) + l_{4y}(f_4 - f_6) \end{bmatrix} \tag{11}$$

Now considering that two thrusters are connected to the same driver, we assumed that $f_1{=}f_2$, $f_3{=}f_4$ and $f_5{=}f_6$, then

$$\boldsymbol{\tau} = \begin{bmatrix} 2f_3 + 2f_5 \\ 0 \\ 2f_1 \\ 0 \\ 2l_{1x}f_1 \\ (f_3 - f_5)(l_{3y} + l_{4y}) \end{bmatrix} \tag{12}$$

III. ARTIFICIAL VISION ALGORITHM

Assuming that the camera is fixed to the vehicle, the aim of our vision algorithm is to estimate the relative angular position (ψ_R) of the camera with respect to the pipeline. This algorithm has to be robust to ill knowledge of the pipeline model and has to be of low complexity to be integrated in the onboard calculator. Our approach is mainly based on the algorithm proposed by Zingaretti [12]. This method analyses the cumulative profiles to select candidate contour points in the edge maps. In this paper, we introduce two changes to the algorithm, namely the introduction of a Gaussian filter in order to reduce the noise produced primarily by changes in lighting and irregularities on the pipeline and an optical flow estimation to determine the direction and magnitude of movement in the image plane. These changes improve the robustness of the algorithm towards sudden movement of the vehicle.



a) b)

Figure 3: The Image produced by Gaussian filter and Canny's method is shown in Figure (a) and the horizontal profile of each region are shown in Figure (b).

All the steps of the vision algorithm are illustrated in Figure 4. First Gaussian filter is applied in order to reduce the noise produced primarily by changes in lighting and irregularities on the pipeline. Then, a Canny edge detector is used to produce a binary image by assigning $\forall\ h \in H_i,\ w \in W_i$: $I(h, w) = 1$ (white color) to the pixels that belong to an edge, and $I(w, h) = 0$ (black color) for the others, where $I(h, w)$ is the value of the pixel located at position (h, w), H_i the image height, and W_i the image width. This is illustrated in Figure 3 (a).

The next step is to divide the height of the image in B_i regions, so the image is split into B_i blocks with a height of H_i/B_i and a length of W_i, as depicted in Figure 3 (a). Subsequently that the image has been divided, the horizontal profile of each region is computed and stored in a vector of W_i elements, as shown in Figure 3 (b). Usually these profiles have values larger at the edges of the duct. In some cases, mainly due to noise in the image or phenomena like algae or rocks there will be peaks that do not correspond to the duct edges. Dividing images into B_i blocks allows to isolate outlier edges in the next steps.

Three methods are alternatively used to perform the profile analysis depending on the state of previous analysis. Their goal is to provide the coordinates of a pair of points associated to the duct edge. These three methods are called: $Initialisation$, $Narrow$ and $Broad\ search$. The $Initialisation$ method (INI) is used in the detection of the pipe when the position of the pipe is not available on previous image. The $Narrow - search$ (NS) is the faster and most robust procedure of the three, as it exploits the information coming from previous image processing, this method is used when the position of the pipe was found in the previous image. Finally, the $Broad - search$ (BS) method is used when the NS method fails, mainly as a consequence of abrupt horizontal and vertical oscillations during the pipeline following.

The INI method consist in choosing a pair of coordinates for each region, corresponding to the edge of the right and left side of the pipeline, the first point provided by the method INI corresponds to the first peak found watching graphs from left to right, and the first peak viewing graphs from right to left.

The NS method replaces the first one after that duct detection was performed successfully. This second method consists in looking for potential points forming the boundary of the pipeline in an area near to the coordinates of the previous points. We propose to use the optical flow, in order to estimate the displacement of the duct edges. The Lucas-Kanade (LK) algorithm [4] is implemented in order to estimate the optical flow; the basic idea of this algorithm is based on two assumptions: brightness constancy and small displacements, it means we assume that the brightness of a pixel does not change as it is tracked from image to image, and that the magnitude of displacement between two images remains small. It leads to the well known optical flow equation :

$$I(w, h, t) = I(w + \triangle w, h + \triangle h, t + \triangle t) \tag{13}$$

where $I(w, h, t)$ is the brightness of the previous image, $I(w + \triangle w, h + \triangle h, t + \triangle t)$ is the brightness of the current image, $\triangle w$, $\triangle h$ are the incremental changes of w and h, respectively, corresponding to the increment in time $\triangle t$. Then, using the LK algorithm we obtain the values of $\triangle w$ and $\triangle h$, which determine the search area for the NS method, this allows the artificial vision algorithm to be more robust to sudden movements of the vehicle.

The BS method is very similar to the INI method, with the advantage that additional information is known for

the pipeline detection. This method considers the distance, measured in pixels, between the points of the duct edge defined in the previous image. Once all the points are selected, we use the least squares method for curve fitting and we obtain the equations that represent the location of the points corresponding to the left and right edge of the duct. We also compute a linear correlation coefficient (ccl), which indicates the correlation between the points found and the line obtained by the curve fitting. Therefore, the value of ccl is used as an evaluation criterion for deciding which of the three methods (INI, NS, BS) has to be applied. Finally, if $ccl \geq 0.8$, we estimate the ψ_R value from the equations of the straight lines representing the edges of the duct. The resulting artificial vision algorithm is described in Figure 4.

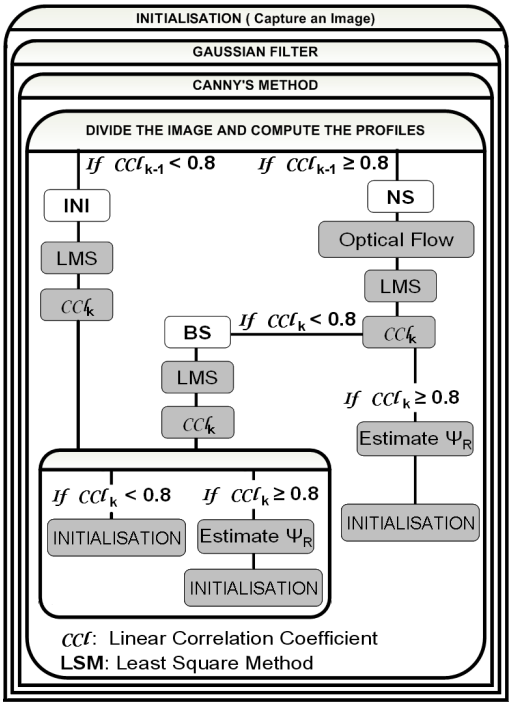


Figure 4: Description of the artificial vision algorithm.

IV. CONTROL STRATEGY

For the design of the controller, it is common to assume that the hydrodynamic parameters involved in the dynamical model of the underwater vehicle are unknown. Indeed, they depend on effects and properties that are difficult to model or estimate, like added mass, skin friction, vortex shedding, characteristics of fluid, etc. Therefore, we propose to use a Proportional Derivative (PD) controller. Since in our case we know the gravity vector $\boldsymbol{g(\eta)}$, we apply a gravity compensation within the PD controller. Although there are many other control strategies for underwater vehicles, in this paper, we have chosen a $PD+\boldsymbol{g(\eta)}$ controller as the main objective is to estimate the ability of the vehicle to perform vision based autonomous pipeline following. Considering the nonlinear system given by equation (1) and (2), we propose the following control input:

$$\boldsymbol{\tau = g(\eta) - J^T(\eta)\tau_{PD}} \tag{14}$$

with

$$\boldsymbol{\tau_{PD} = K_p e(t) + K_d \frac{de(t)}{dt}} \tag{15}$$

where $\boldsymbol{e = \eta - \eta_d}$ represents the state error, and the gains of the controller are denoted $\boldsymbol{K_p = K_p^T > 0}$ and $\boldsymbol{K_d = K_d^T > 0}$. Then, introducing equation (14) into (1), the closed loop system can be written:

$$\boldsymbol{M\dot{\nu} + [C(\nu) + D(\nu)]\nu = -J^T(\eta)\tau_{PD} + w_e} \tag{16}$$

now considering that $\boldsymbol{\eta_d} = cte$,gives:

$$\boldsymbol{M\dot{\nu} + [C(\nu) + D(\nu)]\nu = -J^T(\eta)[K_p e + K_d\dot{\eta}] + w_e} \tag{17}$$

introducing equation (2) into (17) and introducing $\boldsymbol{K_{dd} = J^T(\eta)K_d J(\eta)}$, we obtain:

$$\boldsymbol{M\dot{\nu} + [C(\nu) + D(\nu) + K_{dd}]\nu + J^T(\eta)K_p e = w_e} \tag{18}$$

A Lyapunov candidate function for system (18) is

$$\boldsymbol{V = \frac{1}{2}\nu^T M\nu + \frac{1}{2}e^T K_p e} \tag{19}$$

and since in our applications the vehicle will only be allowed to move at low speed, we assume that $\boldsymbol{M = M^T > 0}$. This suggests that V is globally positive definite function. Now, the time derivative of this function is expressed as:

$$\boldsymbol{\dot{V} = \nu^T M\dot{\nu} + \dot{e}^T K_p e} \tag{20}$$

since $\boldsymbol{\eta_d} = cte$, then $\boldsymbol{\dot{e}^T = \dot{\eta}^T = \nu^T J^T(\eta)}$, therefore:

$$\boldsymbol{\dot{V} = \nu^T[M\dot{\nu} + J^T(\eta)K_p e]} \tag{21}$$

Now introducing equation (18) into the above equation, we have:

$$\boldsymbol{\dot{V} = \nu^T[w_e - [C(\nu) + D(\nu) + K_{dd}]\nu]} \tag{22}$$

As $\boldsymbol{C(\nu)}$ is a Skew-symmetric matrix, then $\boldsymbol{\nu^T C(\nu)\nu = 0}$, $\forall\ \boldsymbol{\nu}$, and we assume that $\boldsymbol{w_e \approx 0}$, therefore:

$$\boldsymbol{\dot{V} = -\nu^T[D(\nu) + K_{dd}]\nu} \tag{23}$$

We can notice that $\dot{V}$=0 when $\boldsymbol{\nu}$=0. Now we can use LaSalle's theorem in order to prove that closed loop system is globally asymptotically stable, therefore:

$$\boldsymbol{\Omega = [(\nu, e) \in \mathbb{R}^n : \dot{V}(\nu, e) = 0]} \tag{24}$$

then

$$\boldsymbol{\Omega = [e \in \mathbb{R}^n, \nu = 0 \in \mathbb{R}^n]} \tag{25}$$

From equations (18) and (25), we can notice that: $\boldsymbol{(\nu, e) = (0, 0)}$ is the only initial condition in $\boldsymbol{\Omega}$ for which $\boldsymbol{(e, \nu)} \in \boldsymbol{\Omega}$ for all t≥0. This means that the equilibrium point is globally asymptotically stable according to LaSalle's theorem. In the case where the integral action is included, it is also possible to prove the local asymptotic stability, see [3].

V. REAL-TIME EXPERIMENTS

The goal of the real-time experiments is to observe the behavior of the vehicle, test the implemented artificial vision algorithm and analyze the response of the closed loop system. Figure 5 shows the prototype that we have developed for pipeline following. For this, the experiments have been performed with a constant thrust, this is $\tau_X = 4N$. This suggests that the vehicle´s speed along the x_b axis is constant ($u = cte$). The weight and the buoyancy force of the vehicle are approximately $186.3N$ and $191.2N$, respectively.

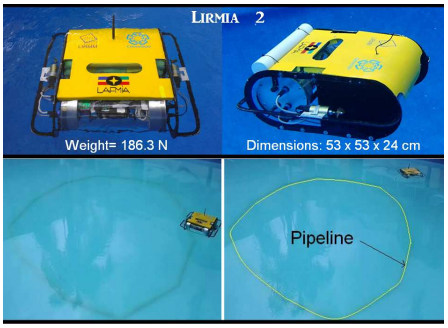


Figure 5: Lirmia 2 prototype and the experimental setup (pipe lying on the bottom of a pool).

Considering the design of the vehicle and its low speed, we have considered that the pitch and roll angle remain close to the equilibrium point ($\phi = 0, \theta = 0$) without a control action. Therefore, the control strategy is focused on immersion (z) and yaw motion (ψ). The values used for the controller were $kp_z = 130$, $kd_z = 80$, $kp_\psi = 1.9$, $kd_\psi = 4.1$ and $f_b = -4.9N$. To estimate the performance and the robustness of the system, a pipe has been placed on the bottom of a pool, drawing a quasi circular loop. The diameter of this loop was about 7 meters, as can be seen on Figure 5.

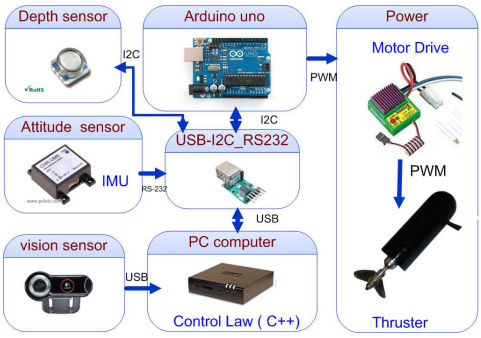


Figure 6: Hardware architecture of the Lirmia 2.

The embedded system of "Lirmia 2" is shown on Figure 6. It consists in an embedded computer with an Intel Atom Z550 2GHz CPU and a 1GB DDR2-533 RAM memory. This embedded system also includes an inertial measurement unit (UM6 Orientation Sensor, CH Robotics), a Logitech webcam Pro 9000, and a pressure sensor. The computer's operating system is Windows XP embedded. Using Visual C++, the computer processes the data from the sensors, and then computes and sends the control inputs to the actuators.

Figure 7 shows (in red) the vehicle´s trajectory along the z axis (depth). One can notice that the vehicle remains close to the reference (in blue) (z_d=0.25 m) with small oscillations induced by the translation movement.

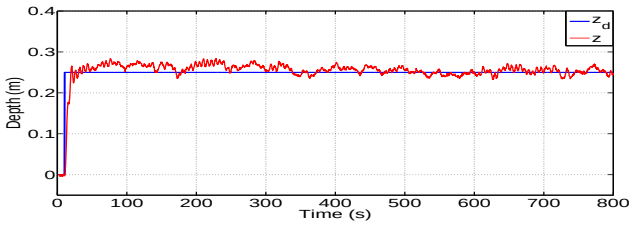


Figure 7: Desired (in blue) and measured (in red) depth trajectory of the vehicle.

The force to keep the vehicle close to z_d is shown on Figure 8. One can notice that the initial force is bigger than during the rest of the experiment. This effect is produced when the vehicle is close to the surface since the thrusters are not immersed enough and then need to turn faster in order to produce the same force as for the rest of the experiment.

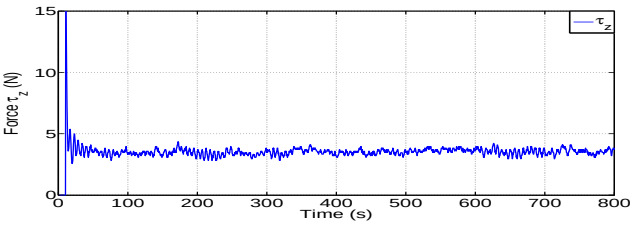


Figure 8: Control input of the thrusters along z axis.

Figure 9 introduces the performance of the relative angle ψ_R between the pipeline and the vehicle when applying the proposed control law. In our case, the goal is to keep the vehicle aligned with the pipeline while the AUV is moving forward. This suggests that the desired angle between the vehicle and the pipeline (ψ_{Rd}) should be equal to 0. One observes that the ψ_R angle remains bounded ($-22°$ to $12°$). Notice, that this behavior is generated because the vehicle is turning around the pipeline formed by straight ducts which draws a quasi circular loop. Indeed, this induces sudden changes in the ψ_R's measurement. This effect could be compared to frequent disturbances. Therefore the control action induces fast changes in the heading of the vehicle in order to remain aligned with the pipeline.

The torque τ_N (yaw) generated by the lateral thrusters along the axis (Oz_b) is shown on Figure 10. Notice that this torque is saturated in order to prevent damages on the thrusters. Most of the time the torque was positive because the pipeline was drawing a circular path, as can be seen on Figure 5.

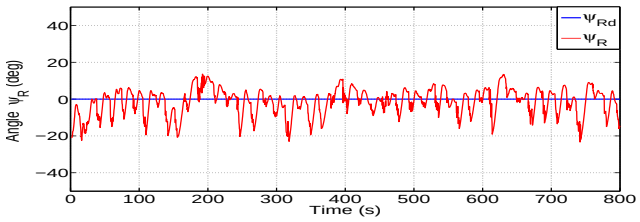


Figure 9: Relative angle ψ_R (in red) between the AUV (Ox_b) axis and the pipeline.

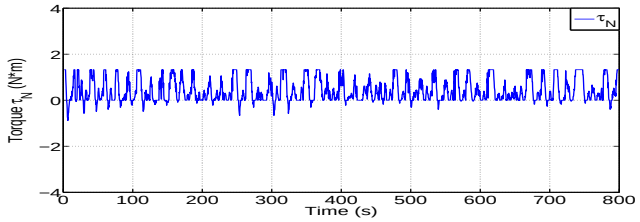


Figure 10: Torque N along the z_b axis during the pipeline following.

Figure 11 shows the pitch angle response when the translational velocity u is not null. One observes that before $t = 10s$ the pitch angle is close to zero as $u = 0$. After this time, the pitch angle is close to 4 deg, since the vehicle starts moving forward. Notice that the oscillations are induced by the movement and the immersion dynamic. This 4 deg angle cannot be avoided as the vehicle is non holonomic. In fact state variables z and θ are coupled (see, equations (9) and (11)).

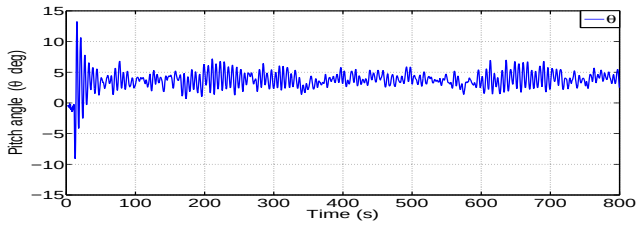


Figure 11: Pitch angle θ along the vehicle´s trajectory

Figure 12 shows the unactuated roll angle response when the vehicle starts moving forward. In this case, the roll angle is close to zero before $t = 10s$. After this time, the roll angle is close to 2.5 deg due to unmodeled hydrodynamic effects and the oscillations are smaller than for the pitch angle.

VI. CONCLUSION AND FUTURE WORK

This paper described the design and the experimental testing of pipeline following performed by an autonomous underwater vehicle. A new prototype with an embedded control system has been developed and presented. This vehicle performs real time embedded image processing in order to implement an artificial vision algorithm. The obtained results show the good ability of the vehicle to follow the pipeline, without losing it. In a next future, we will implement another

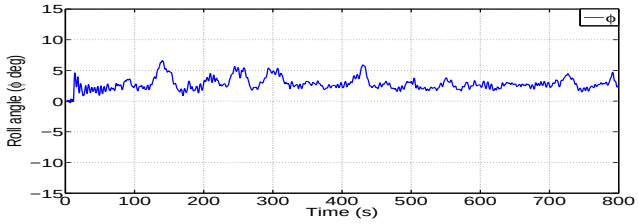


Figure 12: Roll angle ϕ along the vehicle´s trajectory.

control strategy aiming to improve the performances along z and ψ. Experiments in presence of external disturbances and with less visibility will be soon conducted in natural environment.

REFERENCES

[1] A. Stewart, L. Glegg, M. P. Olivieri, R. K. Coulson, and S. M. Smith. *A Passive Sonar System Based on an Autonomous Underwater Vehicle*. IEEE Journal of Oceanic Engineering, vol. 26, No. 4, October 2001, pp 700-710.

[2] Alberto Ortiz, Miquel Simó, Gabriel Oliver. *A vision system for an underwater cable tracker*. Machine Vision and Applications, vol 13: 129-140, 2002.

[3] Arimoto, S. and F. Miyazaki. *Satability and Robustness of PID feedback Control for Robot Manipulators of Sensory Capability.* In:Proc. of the 1st Int. Symp. On Robotics Research (M. Brady and R.Paul, Eds) pp 783-799, MIT Press, 1984.

[4] B.D. Lucas and T. Kanade, *An iterative image registration technique with an application to stereo vision*, Proceedings of the 1981 DARPA Imaging Understanding Workshop ($pp.121 - 130$), 1981.

[5] H. Goldstein, C.P. Poole and J.L Safko. *Classical Mechanics*. Addison Wesley Series in Physics, Adison-Wesley, USA, second edition, 1983.

[6] L. Jaulin and A. Bertholom *Localization of an AUV using set-membership methods.* SeaTechWeek. 16-19 october, 2006.

[7] L. Kleeman. *Advanced Sonar and Odometry Error Modeling for Simultaneous Localisation and Map Building*. Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, USA, 2003.

[8] Lamb, H.(1932). *Hydrodynamics*. Cambridge University Press. London

[9] Marsden, J.E. *Elementary Classical Analysis*, W.H. Freeman and Company, San Francisco, 1974.

[10] Muhammad Asif, Mohd. Rizal Arshad and Abid Yahya. *An Active Contour for Underwater Target Tracking and Navigation*, International Conference on Man-Machine Systems, September 15-16 2006, Langkawi, Malaysia.

[11] Newman, J.N. (1977). *Marine Hydrodynamics*. MIT Press. Cambridge, MA.

[12] Primo Zingaretti, Silvia Maria Zanoli. *Robust real-time detection of an underwater pipeline*. Engineerin Applications of Articial Intelligence, Pergamon, 1998.

[13] The Society of Naval Architects and Marine Engineers. Nomenclature for Treating the Motion of a Submerged Body Through a Fluid. IN: *Technical and Research Bulletin* No. 1-5.

[14] Thor I. Fossen. *Guidance and Control of Ocean Vehicles*. John Wiley and Sons, Norway, Second edition, 1999.

[15] Thor I. Fossen. *Marine control systems guidance, navigation, and control of shipd, rigs and underwater vehicles.* Marine Cybernetics, 2002.

[16] Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control.* John Wiley, 2011.