Testing the Control Architecture of a Micro-Aerial Vehicle for Visual Inspection of Vessels

Francisco Bonnin-Pascual*, Alberto Ortiz,
Emilio Garcia-Fidalgo, and Joan P. Company-Corcoles

Dep. of Mathematics and Computer Science, Univ. of the Balearic Islands (Spain)
xisco.bonnin@uib.es

Abstract. Vessels constitute one of the most cost effective ways of transporting goods around the world. Despite the efforts, maritime accidents still occur, with catastrophic consequences. For this reason, vessels are submitted to periodical inspections for the early detection of cracks and corrosion. These inspections are nowadays carried out at a great cost. In order to contribute to make ship inspections safer and more cost-efficiently, this paper presents a novel Micro-Aerial Vehicle devised as a flying camera that can virtually teleport the human surveyor through the different structures of the vessel hull. The control software has been designed following the Supervised Autonomy paradigm, so that it is in charge of safety related issues such as collision avoidance, while the surveyor, within the main control loop, is supposed to supply displacement commands while he/she is concentrated on the inspection at hand. The paper provides an extensive evaluation of the platform capabilities and usability, both under laboratory conditions and onboard a real vessel, during an inspection campaign.

Keywords: micro-aerial vehicle, vessel inspection, supervised autonomy

1 Introduction

The importance of maritime transport for the international commerce is unquestionable. Different types of vessels are used depending on the kind of product that is carried: oil tankers, bulk carriers, container ships, etc. All of them can be affected by different kinds of defects that may appear due to several factors, such as structural overload, problems in the vessel design, the use of sub-standard materials/procedures, normal decaying of the metallic structures in the sea, etc. Regardless of its cause, cracks and corrosion are the two main defective situations that appear in vessel structures. These are indicators of the state of the vessel hull, and an early detection can prevent major problems such as wreckages. For

* This work has been partially supported by EU-FP7 project INCASS (MOVE/FP7/605200 /INCASS), by project number AAEE50/2015 (Govern de les Illes Balears, DGIR) and by FEDER funding.

this reason, Classification Societies impose periodical inspection to assess the structural integrity of vessels.

Nowadays, to perform the inspection of a vessel, this must be situated in a dockyard (and sometimes in a dry-dock) where high scaffoldings are installed to allow the surveyors to reach all the plates and structures of the vessel. This procedure, together with the lost-opportunity costs due to the fact that the ship is not being operated, represents an elevated expense for the ship owner/operator. Furthermore, during this process, vessel surveyors have to reach high-altitude areas or even hazardous environments, putting at risk his/her own integrity.

In line with the aforementioned, the INCASS project pursues to develop new technological tools with the aim of contributing to the re-engineering process of vessel inspection. This paper focuses on the aerial robotic platform that has been designed for the visual inspection of the inner vessel hull. The idea behind this device is to allow the surveyor to perform a proper inspection from a safe and comfortable position.

The robotics literature contains a number of contributions for vessel hull inspection involving robotic platforms. The majority of the existing approaches make use of underwater vehicles to inspect the submerged part of the hull. Some of them are based on the use of Remotely Operated Vehicles (ROV) (see for example [10, 12]), while other approaches are based on the use of Autonomous Underwater Vehicles (AUV) which estimate their position with regard to the vessel hull using different devices and/or techniques. Apart from solutions based on free-floating AUVs (see for example [7, 13]), in this group we can also find some approaches for hull-crawling vehicles which are attached to the hull by means of suction (see [1, 11]). The robotics literature also contains a reduced number of robotic platforms magnetically attached to the vessel hull, what makes feasible the inspection above the water line (e.g. [2, 8]).

To the best of the authors' knowledge, the only contributions about flying robots specifically devised for vessel hull visual inspection result from our research. In [3] we presented our first approach. This was based on a fully-autonomous vehicle which made use of a laser-scanner to estimate its position by means of Simultaneous Localization and Mapping (SLAM), and assuming vertical structures inside the vessel. This platform was operated via waypoint navigation, so that the position estimation was critical and the usability of the platform was limited. In our second approach [5], the user/surveyor was introduced in the position control loop of the platform so that he/she could indicate the displacement commands to the vehicle by means of a simple interaction device such as a joystick/gamepad. The control architecture based on the Supervised Autonomy (SA) paradigm [6] was in charge of alleviating the stress on the user/surveyor, so that he/she could concentrate on the inspection at hand while all the safety-related issues (e.g. obstacle avoidance) fell on the robotic platform. The velocity of the platform was estimated by means of two optical-flow sensors, so that it could operate only in well illuminated environments. In this paper we present the hardware and control architecture of a new design, together with an extensive evaluation of its capabilities and usability both under laboratory conditions and performing an inspection campaign on board a real vessel.

The rest of the paper is organized as follows: Section 2 describes the hardware configuration of the robotic platform and details its control architecture; Section 3 firstly reports on a number of laboratory experiments aiming at the validation of the proposed design (Section 3.1) and then presents the performance of the vehicle during an inspection campaign on board a real vessel (Section 3.2); to finish, Section 4 draws some conclusions on the work described.

2 The Micro-Aerial Vehicle

The requirements of the robotic platform have been defined taking into account the target task. The idea is to obtain an aerial robotic device to teleport the vessel inspector through the different structures of the vessel, so that he/she can perceive an appropriate view of the hull state. The requirements to fulfil this task include the capability to allow a close-up view of the inspected surface, even when operating in dark environments, while obeying the commands of the user/surveyor. At the same time, the vehicle must implement self-preservation functions, such as prevent collision with the surrounding obstacles, and provide some autonomous behaviours to alleviate the mental workload of the user/surveyor, especially when performing repetitive operations or those prolonged in time.

To fulfil the previous requirements, we have developed a Micro-Aerial Vehicle (MAV) based on a multirotor device with capabilities for hovering and Vertical Take-off and Landing (VTOL). The vehicle is equipped with cameras to take high resolution pictures and videos from the vessel hull surface. The inspection in dark spaces, such as ballast tanks or closed cargo holds, is possible thanks to the use of a high power LED that illuminates the inspected surface. All the pictures are tagged with the estimated pose of the vehicle to perform an effective inspection of the vessel and to allow revisiting the area if necessary.

The control software has been configured to be hosted by any of the research platforms developed by Ascending Technologies (the quadcopters Hummingbird and Pelican, and the hexacopter Firefly), although it could be adapted to other systems. The AscTec platforms are equipped with an Inertial Measuring Unit (IMU) and two ARM7 microcontrollers. The primary ARM7 microcontroller is in charge of the attitude stabilization and thrust control loops (provided by the manufacturer), while the secondary microcontroller is left free so that the user can implement its own position/velocity controller. The design presented in this paper is based on the Pelican platform, since it features the higher payload capacity among the three AscTec research platforms. This vehicle has been fitted with an additional processing board to run all the state estimation and high-level control procedures on board. More precisely, it is an Intel NUC board combining an Intel Core i5-4250-U 2×1.3GHz processor and 8 GB RAM.

As indicated before, the vehicle is devised to be operated by a vessel surveyor from a ground station, so that he/she can provide the desired displacement

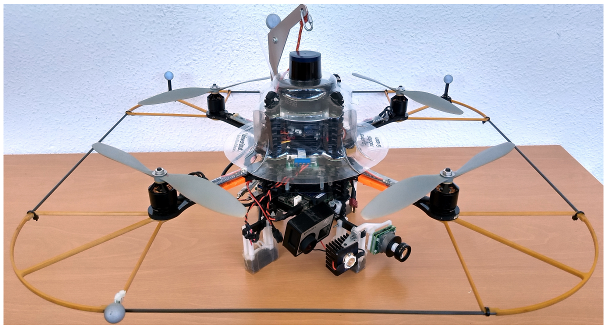

Fig. 1. The MAV for vessel visual inspection.

commands by means of a joystick/gamepad. In other words, the user/surveyor is introduced in the position control loop, so that the vehicle control architecture does not require the position of the platform. Hence, the platform state includes the velocities and accelerations along the three axes, the vehicle altitude, and its orientation. The corresponding estimates are used by the secondary ARM7 microcontroller, which implements the velocity and height PID controllers that map input speed commands into roll, pitch, yaw and thrust orders.

To estimate the platform state, the vehicle has been fitted with a laser-scanner Hokuyo UST-20LX and a Lidar-Lite optical range sensor. The first device is used to get the displacement of the vehicle in the horizontal plane, while the latter provides the distance to the floor. The vehicle velocities are estimated by means of a Kalman Filter which combines the measures provided by these two devices with the accelerations delivered by the IMU. All these procedures run on the onboard computer as a collection of ROS nodes. Figure 1 shows the MAV fitted with the aforementioned navigation sensor suite.

As in [5], the control software has been designed around the SA paradigm. This defines a framework for human-robot interaction which allows an operator inside the main control loop, though assisted by the robot through a number of autonomous functions oriented towards making simpler the intended operations.

In line with the SA paradigm, speed commands are generated through a set of autonomous behaviours organized in a hybrid competitive/cooperative framework. These behaviours are implemented as ROS nodes which are executed on the onboard computer. Figure 2 details the behaviour-based architecture, grouping the different behaviours depending on its purpose. A total of four general categories have been identified and defined: (a) behaviours to accomplish the user intention, which propagate the user desired speed command, attenuating it towards zero in the presence of close obstacles, or keeps hovering until the communication with the ground station is restored after an interruption; (b) behaviours to ensure the platform safety within the environment, which prevent the robot from colliding or getting off the safe area of operation; (c) behaviours to increase the autonomy level, which provide higher levels of autonomy to both simplify the vehicle operation and to introduce further assistance during inspec-

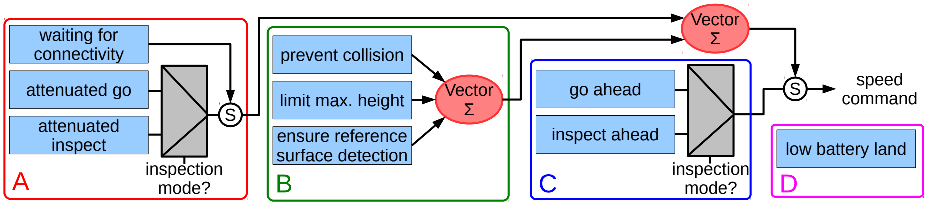

Fig. 2. MAV behaviours: (A) behaviours to accomplish the user intention, (B) behaviours to ensure the platform safety within the environment, (C) behaviours to increase the autonomy level, and (D) behaviours to check flight viability.

tions; and (d) behaviours to check flight viability, which checks whether the flight can start or progress at a certain moment in time. Some of the behaviours in groups (a) and (c) can operate in the so-called *inspection mode*. While in this mode, the vehicle moves at a constant and reduced speed (if it is not hovering) and user commands for longitudinal displacements or turning around the vertical axis are ignored. In this way, the platform keeps at a constant distance and orientation with regard to the front wall, for improved image capture.

Finally, in order to tag the pictures with the position of the platform, the robot integrates the laser-based SLAM method *GMapping* [9]. This process, which is also executed as a ROS node on the onboard computer, provides the 2D position of the platform, which is later combined with the estimated altitude of the MAV to get a 3D position.

3 Experimental Evaluation

This section provides the experimental assessment of the MAV. In first place, Section 3.1 assesses the capability and usability on the platform through laboratory experiments. Secondly, Section 3.2 reports on the platform performance during an inspection campaign on board a real vessel.

3.1 Platform Validation

During the experiments in the laboratory, we have made use of a motion capture system to estimate the position, orientation and velocity of the platform. They are used as ground truth (GT) to evaluate the performance of the system.

In a first kind of experiment, the behaviour of the MAV is evaluated while the user is issuing displacement commands. The plots corresponding to this experiment can be found in Fig. 3. As shown in Fig. 3 [D], the system is able to estimate the vehicle speed and follow accurately the user commands. Due to lack of space, only motion along the vehicle longitudinal axis is provided.

A second kind of experiments evaluate performance at the robot behaviour level. Due to space limitations, in this paper, we can only report on the results for five experiments involving selected behaviours.

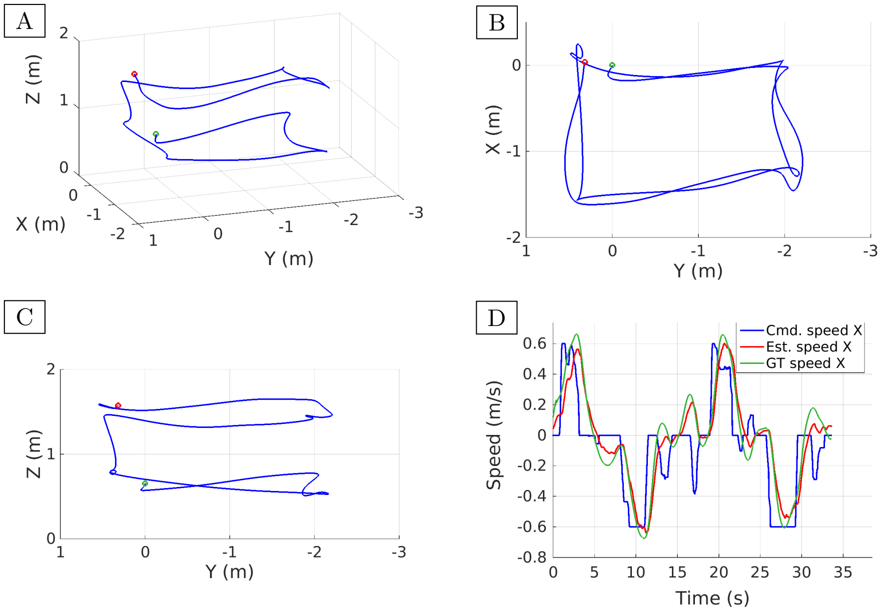

Fig. 3. Results estimated for the MAV commanded to perform a double-square trajectory: (A) plot of the trajectory by the motion tracking system (the green and red dots indicate the initial and final points), (B-C) 2D projections of the trajectory, (D) reactions of the MAV to the velocity commands in the longitudinal axis.

A first experiment, reported in Fig. 4 [right], checks the *go_ahead* behaviour. At the beginning, the user indicates a longitudinal desired speed of 0.4 m/s and then activates the *go_ahead* behaviour (instant A). At this moment, in accordance to the behaviour definition, the speed command keeps at 0.4 m/s although the user-desired speed returns to zero. This value is kept until the wall in front of the vehicle becomes closer than 1.2 m (instant B), which is the minimum distance allowed. Then, the *prevent_collision* behaviour cancels the *go_ahead* command and stops the platform. This behaviour is also in charge of producing the negative speed command that separates the platform from the wall until it is again at the safety distance (instant C). Figure 4 [left] shows the vehicle trajectory, indicating when the *go_ahead* behaviour is active. Notice that the vehicle has been initially oriented facing the wall so that the positive longitudinal speed command moves the platform towards the negative direction of the Y axis.

In a second experiment, we check the performance of the *limit_max_height* behaviour. Figure 5 [right] shows how the MAV ascends following the vertical user-desired speed until the platform reaches a height of 3 m (instant A), which was set as the maximum height for this experiment. From time instants A to B, the behaviour prevents the platform from going higher ignoring the vertical user-desired speed until it becomes zero (instant B). Next, the platform descends

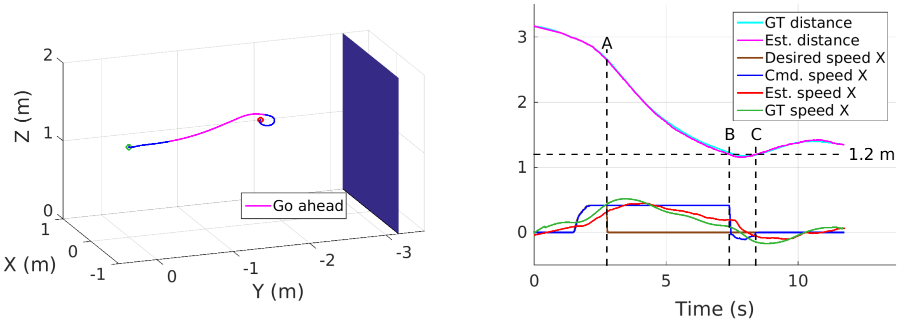

Fig. 4. Performance of the *go_ahead* and the *prevent_collision* behaviours: (left) vehicle trajectory and wall position as detected by the motion capture system, (right) the user-desired speed is sustained while the wall is at enough distance (A→B), it is cancelled and even forced to be negative to prevent an imminent collision (B→C) until the platform is again at the safety distance (C→). All units are in SI (m or m/s accordingly).

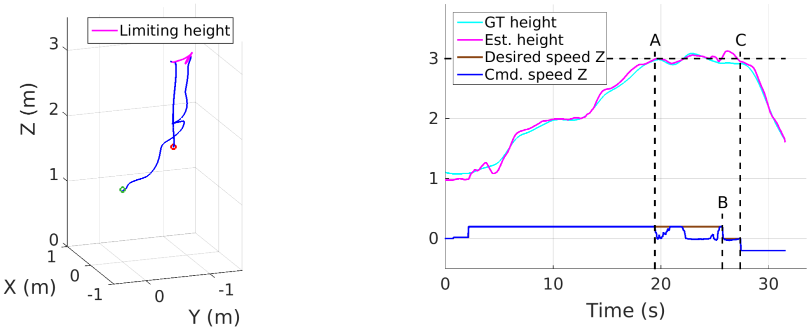

Fig. 5. Performance of the *limit_max_height* behaviour: (left) vehicle trajectory, (right) the user-desired vertical speed is obeyed until the platform reaches the maximum allowed height (→A), then the desired vertical speed is ignored (A→B) until it becomes zero (B→C), and finally the platform descends following again the desired speed (C→). All units are in SI (m or m/s accordingly).

since the user asks for a negative vertical speed (instant C). Figure 5 [left] shows the vehicle trajectory, indicating when the vehicle reaches the limiting height.

Results for a third experiment are plotted in Fig 6 [right]. This case involves the *waiting_for_connectivity* behaviour. At the beginning of the experiment, the user orders a negative longitudinal speed to move the platform. During the displacement, the communication with the base station is lost (instant A), so that the user-desired speed signal is no longer available at the platform. As a consequence, the behaviour takes control and makes the vehicle hover while it waits for a reconnection. After 5 seconds (instant B), the communication link has not been restored and the behaviour decides to make the platform land. Figure 6 [left] shows the vehicle trajectory, where different colours are used to

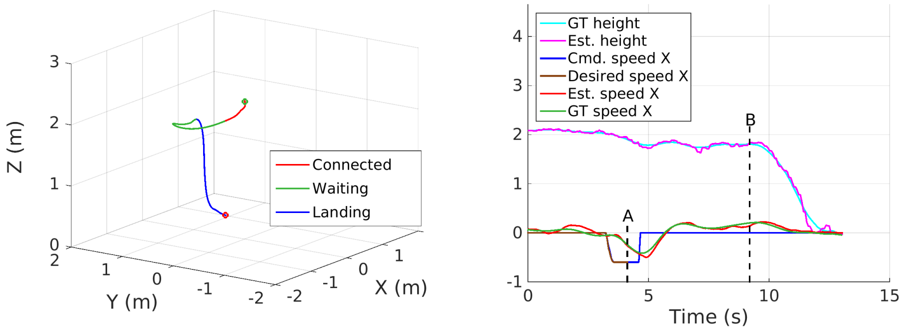

Fig. 6. Performance of the *waiting_for_connectivity* behaviour: (left) vehicle trajectory indicated by the motion capture system, (right) the communication with the base station is lost during the flight (→A), what makes the behaviour force a hovering manoeuvre (A→B) while the vehicle tries to reconnect; after waiting for five seconds without success, the behaviour makes the platform land (B→). All units are in SI.

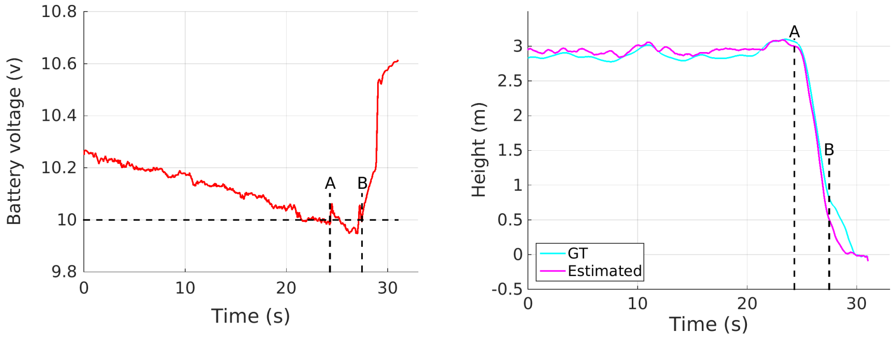

Fig. 7. Performance of the *low_battery_land* behaviour: the platform hovers at 3 m until the battery voltage is below 10 V (→A), what makes the behaviour initiate the descending (A→B) and landing manoeuvres (B→).

indicate when the vehicle is receiving the desired command, when it is waiting and trying to reconnect, and when the vehicle performs the landing manoeuvre.

Figure 7 corresponds to a fourth experiment, aiming at checking the performance of the *low_battery_land* behaviour. During this experiment, the MAV is left hovering at almost 3 m until the battery voltage becomes lower than 10 V (instant A). At this moment, the behaviour takes control of the platform to make it land. The landing manoeuvre starts descending the platform up to 0.5 m, and finishes with the deceleration of the motors thrust (instant B).

In a fifth and last experiment, we show the performance of the platform during an inspection task. The operation starts when the operator makes the platform approach the wall under inspection. At more or less 1 m distance, the *inspection mode* is activated, and, hence, longitudinal motion as well as rotations in yaw are not allowed to ensure better image capture conditions.

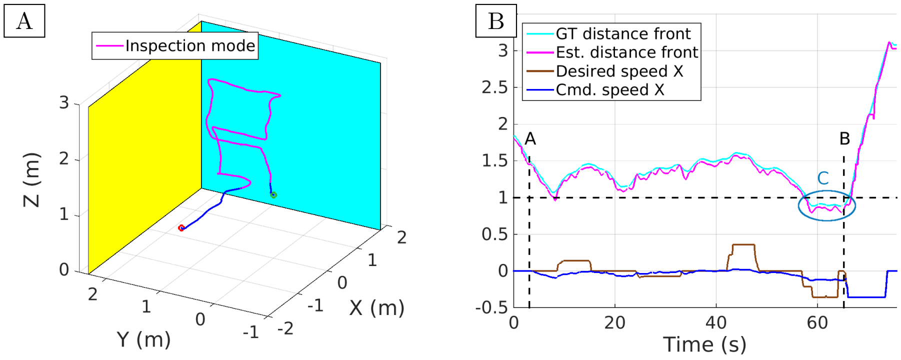

Fig. 8. Performance of the platform using the *inspection mode*: (A) walls and vehicle trajectory, indicating when the *inspection mode* is active, (B) longitudinal commands/displacements (see text for explanations). All units are in SI (m or m/s).

The operator next orders lateral and vertical motion commands to sweep the surface. Figure 8 [A] shows the vehicle trajectory, indicating when the *inspection mode* is active, while Fig. 8 [B] illustrates the full operation for the longitudinal motion. Notice that, when the *inspection mode* is enabled (between A and B), the longitudinal user-desired speed is ignored and a PID controller is in charge of keeping the distance to the inspected wall. The plots also show repulsive speed commands produced when the platform is below 1 m from the wall (instant C).

3.2 Field Tests

This section reports on the experimental results obtained using the MAV for the inspection of a real vessel. The field trials were performed on board a Handymax bulk carrier with dwt above 45000 tons, and whose size is 190 m (length)× 32 m (breadth)×16.5 m (height). During the test campaign, the MAV was operated in three different compartments: the cargo hold #4, the water ballast topside tank #3 and the forepeak tank.

The operating conditions in each compartment were very different. On the one hand, the cargo hold was a very large compartment where the light could be relatively adjusted, since the hatch could be opened and closed. On the other hand, the forepeak and topside ballast tanks were narrow and dark spaces accessible through a manhole-sized entry point, so that the onboard LED had to be used to allow for a proper visual inspection.

All the experiments were performed following the same procedure: (1) the vehicle is situated in a flat and obstacle-free area for the take-off, (2) the user sends the take-off command using a gamepad/joystick and the vehicle starts the flight, (3) the user approximates the platform to the area where the inspection has to take place, while the control architecture based on SA take care of the platform preservation, (4) the user can optionally enable the *inspection mode* to make the vehicle move smoothly and keep at a constant distance to the inspected

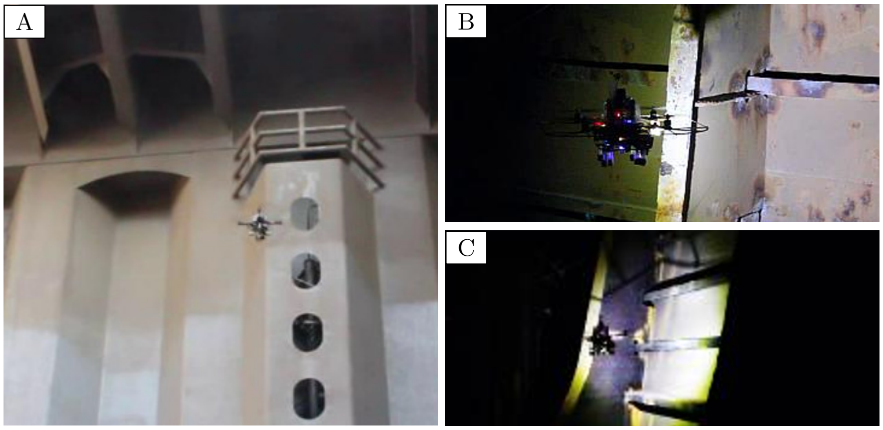

Fig. 9. Some pictures to illustrate the testing activities in the three vessel compartments: (A) the cargo hold, (B) the topside tank and (C) the forepeak tank.

surface, (5) a sequence of pictures can be started when desired, (6) the user can command the platform along the lateral and vertical axes (also longitudinally if the *inspection mode* is not enabled) to perform the inspection, (7) the sequence of pictures can be stopped when desired, (8) the *inspection mode* is disabled (if it was enabled), (9) the user commands the platform to an obstacle-free area for landing, and (10) he/she sends the command for landing.

Figure 9 shows some pictures taken during testing at the three compartments, while Fig. 10 shows the path estimated for some flights. During the experiments, the paths were successfully estimated thanks to the GMapping method (which makes use of the data supplied by the laser scanner) and the laser altimeter.

Finally, Fig. 11 shows some of the pictures taken during the inspection campaign. All these images have been used to create three image datasets (one for each compartment) that have been later used to detect the defective areas using the saliency-based defect detection method described in [4]. In this regard, examples of detection outputs can be found in Fig. 11 [bottom].

4 Conclusions

The hardware and control software architecture for a MAV intended for the visual inspection of vessels have been described. Following the SA paradigm, the robotic platform is devised to be operated as a flying camera that is in charge of all the safety related issues (such as collision avoidance or battery voltage monitoring), while the surveyor can concentrate on the inspection task.

Several experimental results have been reported to validate the capabilities and usability of the platform including results from an inspection campaign on board a real vessel. During these test trials, the robotic platform was used to create three datasets comprising images from three different vessel compartments.

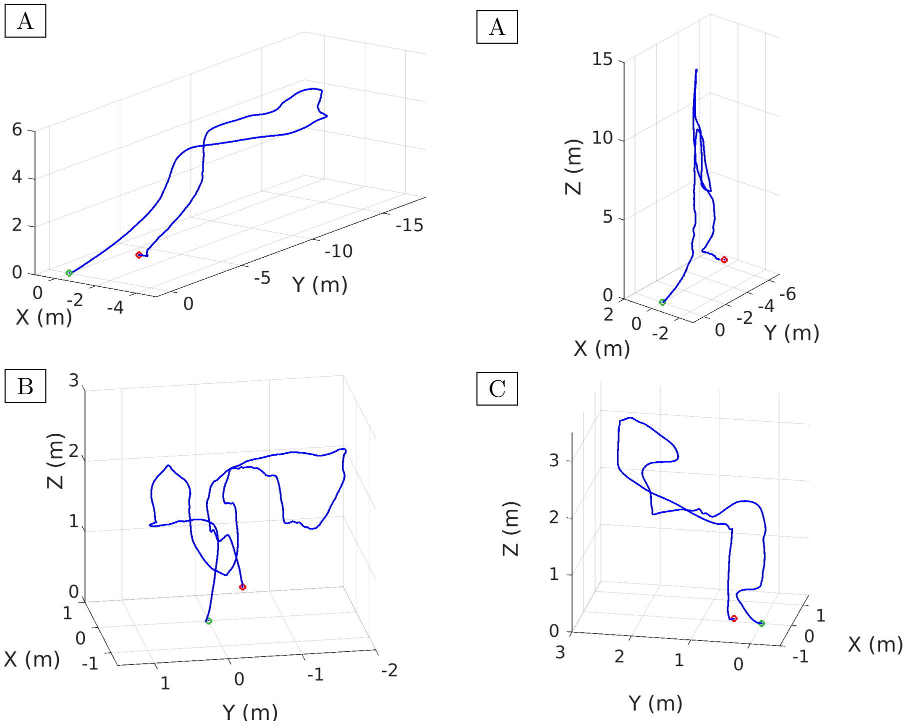

Fig. 10. Estimated paths followed by the aerial robot during four flights inside the vessel compartments: (A) cargo hold, (B) topside tank, and (C) forepeak tank. The green and red dots indicate the initial and final points respectively.

Thanks to the use of the *inspection mode*, the collected images are of good quality (i.e. no blurring) and can be used to feed defect detection algorithms to autonomously detect defective situations.

In comparison with the MAV presented in [5], the new approach is able to fly in dark environments thanks to the use of laser-based sensors for the estimation of the platform state. Furthermore, the approach presented in this paper allows tagging the images with the vehicle pose for an effective inspection and to be able to revisit the area if necessary.

References

1. Akinfiev, T.S., Armada, M.A., Fernandez, R.: Nondestructive Testing of the State of a Ship's Hull with an Underwater Robot. Russian Journal of Nondestructive Testing 44(9), 626–633 (2008)
2. Bibuli, M., Bruzzone, G., Bruzzone, G., Caccia, M., Giacopelli, M., Petitti, A., Spirandelli, E.: MARC: Magnetic Autonomous Robotic Crawler Development and Exploitation in the MINOAS Project. In: Int. Conf. on Computer Applications and Information Technology in the Maritime Industries. pp. 62–75 (2012)

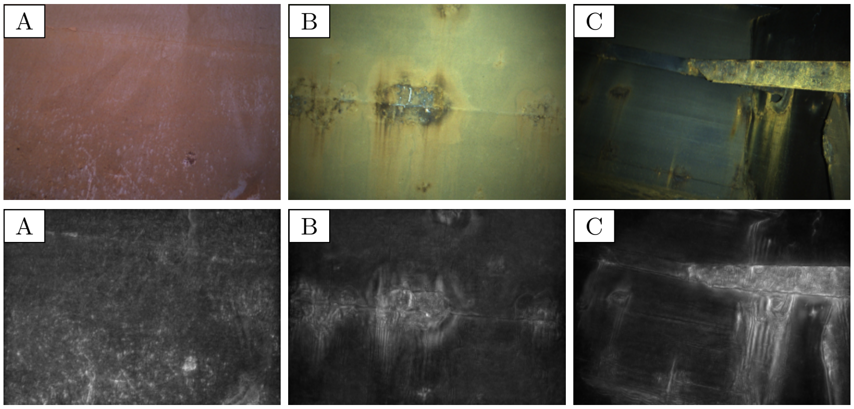

Fig. 11. (Top) Images taken by the MAV inside the vessel compartments: (A) cargo hold, (B) topside tank, and (C) forepeak tank. (Bottom) Defect maps resulting from the saliency-based defect detection algorithm described in [4].

3. Bonnin-Pascual, F., Garcia-Fidalgo, E., Ortiz, A.: Semi-autonomous Visual Inspection of Vessels Assisted by an Unmanned Micro Aerial Vehicle. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. pp. 3955–3961 (2012)
4. Bonnin-Pascual, F., Ortiz, A.: A Generic Framework for Defect Detection on Vessel Structures based on Image Saliency. In: IEEE Int. Conf. on Emerging Technologies and Factory Automation (2016)
5. Bonnin-Pascual, F., Ortiz, A., Garcia-Fidalgo, E., Company, J.P.: A Micro-Aerial Platform for Vessel Visual Inspection based on Supervised Autonomy. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems. pp. 46–52 (2015)
6. Cheng, G., Zelinsky, A.: Supervised Autonomy: A Framework for Human-Robot Systems Development. Autonomous Robots 10, 251–266 (2001)
7. Ferreira, C.Z., Conte, G.Y.C., Avila, J.P.J., Pereira, R.C., Ribeiro, T.M.C.: Underwater Robotic Vehicle for Ship Hull Inspection: Control System Architecture. In: Int. Cong. of Mechanical Engineering (2013)
8. Fondahl, K., Eich, M., Wollenberg, J., Kirchner, F.: A Magnetic Climbing Robot for Marine Inspection Services. In: Int. Conf. on Computer Applications and Information Technology in the Maritime Industries. pp. 92–102 (2012)
9. Grisetti, G., Stachniss, C., Burgard, W.: Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters. IEEE Trans. on Robotics 23, 34–46 (2007)
10. Ishizu, K., Sakagami, N., Ishimaru, K., Shibata, M., Onishi, H., Murakami, S., Kawamura, S.: Ship Hull Inspection using A Small Underwater Robot With A Mechanical Contact Mechanism. In: IEEE/MTS OCEANS Conf. pp. 1–6 (2012)
11. Narewski, M.: Hismar - Underwater Hull Inspection and Cleaning System As a Tool for Ship Propulsion System Performance Increase. Journal of Polish CIMAC 4(2), 227–234 (2009)
12. Newsome, S.M., Rodocker, J.: Effective Technology for Underwater Hull and Infrastructure Inspection. In: IEEE/MTS OCEANS Conf. pp. 1–6 (2009)
13. Ozog, P., Carlevaris-Bianco, N., Kim, A., Eustice, R.M.: Long-term Mapping Techniques for Ship Hull Inspection and Surveillance using an Autonomous Underwater Vehicle. Journal of Field Robotics 33(3), 265–289 (2016)