Nessie VI Autonomous Underwater Vehicle

Miroslav Radojević, Mohamad Motasem Nawaf, Francesco Maurelli, Tummas Tomassson, Chengjia Wang, Jonathan Clay and Yvan Petillot

***Abstract*—Nessie VI is the entry for the SAUC-E 2011 competition from the Nessie Team of Heriot-Watt University. The team is composed of 4 MSc students, one PhD and one Faculty member. In this edition of the Nessie AUV family, the platform has undertaken a significant upgrade from the Nessie V vehicle both in hardware and software levels. In terms of mechanical design, the new design was done by an MEng Student of the mechanical engineering department and implemented by student of the Ocean Systems Laboratory. In terms of software, the Robotic Operating System (ROS) from Willow Garage has been adopted and all existing code has been ported to this new middleware supporting layer. This has proven very beneficial enabling better debugging and replay capabilities as well as added functionalities such as on-board camera calibration. The robotic platform is equipped with state-of-the-art sensors, providing both the vehicle internal state such as position, speed and orientation as well as situation awareness (position of targets, and obstacles, mission goals). Using a robust hardware and software design the vehicle is ready to compete at this year competition, as well as being an excellent platform for research activities in the University. This paper describes the new vehicle hardware design and the novel software algorithms implemented to perform the different task challenges of the competition.**

I. Introduction

This paper describes the Nessie VI autonomous underwater vehicle (AUV) developed at the Ocean Systems Laboratory at Heriot-Watt University. The vehicle has been designed to compete in the 2011 Student Autonomous Underwater Challenge Europe (SAUC-E) competition held at the NATO Underwater Research Centre (NURC) in La Spezia, Italy. This paper will first describe the hardware design of the vehicle and then proceed to the software architecture. Innovations with respect to previous works in our lab are then highlighted, together with the conclusions.

II. Hardware Design

The vehicle draws on the strengths of previous incarnations while eliminating a series of identified weakness. The overall hardware design can be best described as a hover-capable torpedo-shaped vehicle.

It is made up of a cylindrical aluminium pressure hull surrounded by a Delrin® polymer frame that creates a skeleton. This skeleton acts as a mounting point for sensors and a PVC shell. The PVC shell was outsourced to a company specialised in thermoforming and was realised based on our design in solid edge. The supporting delrin frame has been updated when compared to 2010 to improve its strength and the sensor positioning. Initial tests of the PVC used have shown that it was mostly transparent to acoustic waves. As a consequence the sonar heads (Gemini and Micron Sonars) are now embedded inside the outer skin providing better protection. The outer skin not only creates a hydro-dynamic shell which protects the contained devices from impacts but also adds strength and rigidity to the overall design. The overall length of the vehicle is 1740mm and the main body has a diameter of 280mm. The design in solid edge from different view points can be seen on Fig. 1 as well as its equivalent real incarnation.

All authors are from the Ocean Systems Laboratory, Heriot-Watt University, Edinburgh, EH14 4AS, Scotland, UK. (http://osl.eps.hw.ac.uk)

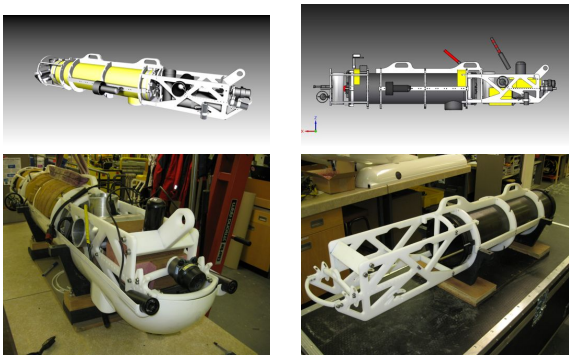


Fig. 1: Nessie VI AUV design and realisation

The robotic platform is equipped with a series of state-of-the-art sensors. These sensors include a Doppler Velocity Log (DVL), a forward and a downward looking sonar, four video cameras, a temperature sensor, a pressure sensor, acoustic modem, GPS, and a fibre optic gyro. These provide information on both vehicle state and information on the surrounding environment.

The use of 6 high powered thrusters gives it five degrees of freedom (Forward, Lateral, Vertical, Yaw, Pitch).

The 2.25kw power source, and ultra low power components combined with the high processing power give the AUV the ability to hold a position in low tidal currents for up to 20 hours.

A. PC104 embedded computer

The computers used in the vehicle are industrial MSM2000 PC104 embedded PCs. This model was chosen for its small form factor and its low power consumption which results in longer battery life and less heat. It has a Intel Atom Processor, running at 1.66 GHz with 1 Gb of onboard RAM, 4 USB 2.0 ports, 4 serial ports, GPS and one 1 Gbit Ethernet port. Two of these embedded PCs are used in the vehicle; one for sensing and control, and another for video capture and processing. These are connected via Ethernet. This split ensures that that primary control PC is never starved of resources by the image processing algorithms.

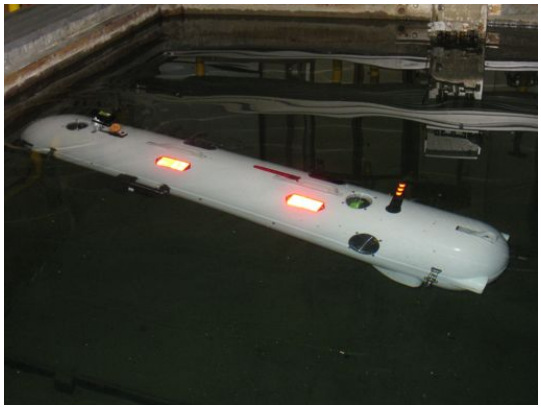


Fig. 2: Nessie VI AUV

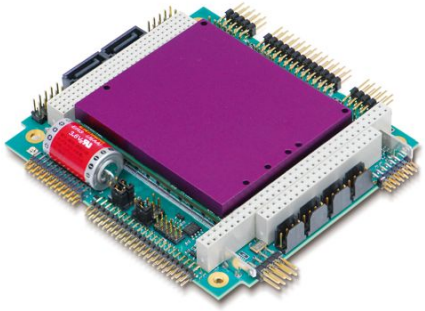


Fig. 3: A single Digital Logic pc104 MSM200X board

To make the computer more robust, flash based solid-state hard disks are used instead of standard magnetic disks. These further reduce the power requirements of the system and make it more robust to bumps and jerks as are expected in a vehicle during transport, deployment and operation. An image of a single PC104 is shown in figure 3, and the internal structure in figure 4. In case of boot faults, an Ethernet accessible dual serial port (N-Port) is connected to the serial console of each PC104, which allows the boot progress to be monitored seconds after the vehicle is powered on. This enables diagnosis and repair of boot faults over an Ethernet link (wired or wireless), without opening the hull.

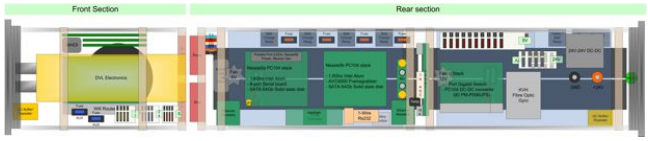


Fig. 4: An overview of the physical layout of the electronic systems

B. Power supply

The power supply is built around lithium ion technology. Nessie VI contains four 543.9 Wh battery packs (Fig. 5). Each unit has a nominal voltage of 25.9V and a capacity of 21Ah. These units were selected as they integrate well into the physical design, four such units can be positioned in the lower half of the pressure hull to keep the centre of gravity low while utilising most of this space along with leaving room above for the electrical and electronic systems.

Fig. 5: One of the four battery packs supplying the power to Nessie VI.

C. 1-Wire

The vehicle contains a 1-wire bus to allow easy addition of low level sensors, this bus has been utilised by temperature and voltage sensors. The 1-wire bus is attached to PC104A via a RS232 to 1-wire converter.

D. I^2C Interface

The primary PC104 employs a USB I^2C interface (Total Phase Aardvark) to control the thruster H-bridges and the bespoke relay boards.

E. Thrusters and H-Bridges

The vehicle has been equipped with SeaBotix thrusters (Fig. 6). These thrusters run at a maximum of 4.25 A and 28V. They weigh 700g in air and 350g in water. The six thrusters are arranged as following: forward port and starboard, lateral front and back lateral, and vertical front and back. This configuration provides 5 degrees of freedom in control: surge, sway, heave, pitch and yaw. We have demonstrated in experiments that we can move reliably in all directions whilst controlling pitch in the $[-35, 35]$ degrees range. Each thruster contains an individual SeaBotix H-Bridges controller which is controlled using the I^2C interface. We have moved the H-bridges controller inside the Hull as the thrusters have been prone to small leaks that damage the controller boards. The port and starboard forward thrusters are Seabotix Brushless HPDC1502 whilst the other 4 thrusters controlling depth, yaw and pitch are Seabotix BTD150 brushed thrusters.

F. Submersible switches and visual feedback device

A bespoke submersible module combining reed switches and LEDs was constructed for basic system control and visual feedback. The reed switch is activated by a magnetic key placed in the socket. This connects to the main relays which can switch the vehicle on and off. Another switch is used to control the power to the motors. This enables the operator to kill the motors in case of unexpected behaviours or safety risks whilst keeping the processors alive. Four LEDs (green,

Fig. 6: SeaBotix thruster

red, yellow and green) are controlled via the parallel interface of PC 104A to display visual feedback to the operator. The first Green LED shows power state of the vehicle, and yellow and red provide feedback on PC status as well as emergency situations. The final green led controls the power to the motors and can be used to start a mission. Fast action relays are used within the hulls, which allow the vehicle to be switched from tether to battery power on the fly, with no PC downtime. This facility is extremely practical during testing. The main reed switch also acts as a system kill switch. This means that a diver can easily stop the system should the need arise. It should be noted that the risk to divers from the vehicle is already minimal, given that each thruster has a protective shroud, and four of the thrusters are enclosed within the outer hull of the vehicle.

G. Doppler Velocity Logger

The Teledyne Explorer PA (Fig. 7) is a very versatile unit that can provide a weather of data on the velocity and position of the vehicle, it provides the system with altitude data along with surge, sway, and heave velocity data. This unit was chosen due to its small size and high level of accuracy, it can operate at an altitude of between 0.5m and 80m with an accuracy of $\pm$0.7cm/s while moving at 1m/s.

Fig. 7: Teledyne Explorer PA

H. Depth Sensor

A Keller Series 33X depth sensor (Fig. 8) is used to measure the distance between the vehicle and the surface.

It is connected to the PC through a RS485 serial interface and a guaranteed range from 0 to 10 bar (0-90 m in water). Absolute accuracy is 0.005 bar, equal to 0.05 m, with a precision of 0.0002 bar, equal to 0.002 m.

Fig. 8: Keller series 33X depth sensor

I. Compass

A TCM 6 compass (Fig. 9), kindly donated by PNI Corporation, measures the vehicle's heading with a precision of 0.5 degrees. It is able to compensate for the vehicle's tilt up to 80 degrees. Firmware calibration routine is provided for hard and soft iron compensation. Connection to the PC is through RS232 serial communication. Supply voltage of the sensor ranges between 3.6 and 5 V, while power usage is less than 22 mA.

Fig. 9: TCM 6 compass

J. FyberOptic Gyroscope

The KVH DSP-3000 was chosen to compliment the DVL as it can provide accurate angular rates. The unit itself is compact and light weight combined with lower power consumption.

1) Forward looking sonar: A Tritech Gemini 720i Multibeam Imaging sonar (Fig. 10) has been used for this project, it offers real-time imaging over a 120° sweep, meaning objects can be identified and tracked even if they are not stationary. The unit had an operational range of 0.2m right up to 120m, and the system can change the range sampled dynamically to give only the information required to aid processing.

K. Cameras

The vehicle is fitted with four 100 meter rated underwater colour cameras, with optics specifically designed for minimal distortion underwater. They interface to the second PC104 via a AVC2000-V PC/104-Plus Video Frame Grabber, this capture card allows the four camera feeds to be combined into one single stream where each camera takes up one quarter of the image. Two cameras are arranged in pairs, looking forward, for stereoscopic vision. They will be used during the competition in tandem with the sonar systems to confirm the identity of an

Fig. 10: Tritech Gemini 720i Multi-beam Imaging sonar

object. The stored images will be processed after the mission is complete to show the behaviour of the vehicle during the mission.

L. Wireless/wired Communication

A D-Link 802.11G Wireless Access Point is used for wireless communication with the vehicle when on the surface. This was stripped of its case and unnecessary connectors and moulded with a high gain aerial in rubber compound to make it waterproof, this unit is mounted in the fount section of the hull and acts as a switch for anything needing access to the network on this side of the main connector.

In the rear section there is a Netgear ProSafe 5-port Gigabit switch that is connected to not only the main PCs but also a wet-mateable connector on the rear hull bulkhead. This connector can be used to attach an industrial 100m Ethernet tether instead of the wireless access point, if communication is desired for submerged testing.

M. GPS

The GPS module is embedded on the PC104, though this system is only effective when at the surface it does offer the opportunity to locate the system in case of any rogue behaviour. If the vehicle found itself drastically out of position, it could send its GPS position back via either wireless link or via the acoustic modem. The GPS antenna has been potted and the cable passed through hull via a potted connector.

N. Water and temperature senses

There is a number of sensors to help detect both the ingress of water and high temperatures within the vehicle. The water sensors are connected to the parallel port of PC104A. The temperature sensors are attached to the 1-wire bus and are located at critical locations through out the hull, there locations can be seen in Fig.11 and Fig. 12.

III. Software Design

A. Architecture

The software architecture for Nessie VI is illustrated in Fig. 13. All modules are implemented as separate processes, and communicate using a mixture of the OceanSHELL system developed in the Ocean Systems Laboratory and the Robotic

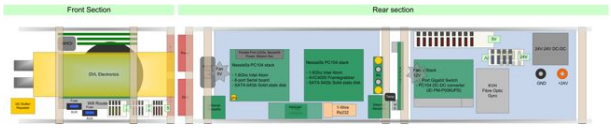


Fig. 11: A top view of the physical layout of the electronic systems

Fig. 12: A Bottom view of the physical layout of the electronic systems

Operating System (ROS) developed by Willow Garage. Both are lightweight UDP/TCP based communications protocols for distributed, modular systems. Each module process is monitored by a *watcher* process, which will restart the module if it fails. Adjustable parameters for the modules are stored in configuration files, to simplify modification. The software modules used in the vehicle will be described in more detail in the following sections.

B. Autopilot

The motion of an underwater vehicle is described by a combination of six velocities (surge, sway, heave, yaw, pitch and roll). These constitute the vehicle's six degrees of freedom. In most hover capable underwater vehicles, only surge, sway, heave and yaw are controlled, leaving the vehicle's centre of mass and buoyancy to maintain the pitch and roll approximately zero. This happens also with Nessie VI, except that we can also control pitch. The operation of the autopilot control system is illustrated in Fig. 14. It is comprised of a bank of cascading Proportional, Integral, Derivative (PID) controllers, and an axis force to thrust the controller. In each of the controlled axes, the current position and set point are fed into the position PID controller which outputs the velocity set point. This combines with the current velocity to feed the velocity PID, which outputs the desired thrust to meet that velocity. The gains within the system have been determined experimentally to yield a controlled response with little overshoot and acceptable response time. The performance of the autopilot control system for the surge, sway, heave and yaw of the vehicle is presented in Fig. 15, Fig. 16, Fig. 17 and Fig. 18 respectively. The control system functions as the core of the Autopilot software system. It is the job of the Autopilot to accept commands from the mission controller and manoeuvre the vehicle to a series of way-points within the tank. Communication between the two modules consists of a waypoint request from the mission controller, followed by continued updates from the autopilot specifying whether or not the waypoint has been achieved (within a given tolerance). There is also a new mode added this year enabling to control the speed directly. This is useful for the visual servoing module presented later which requires direct speed control.

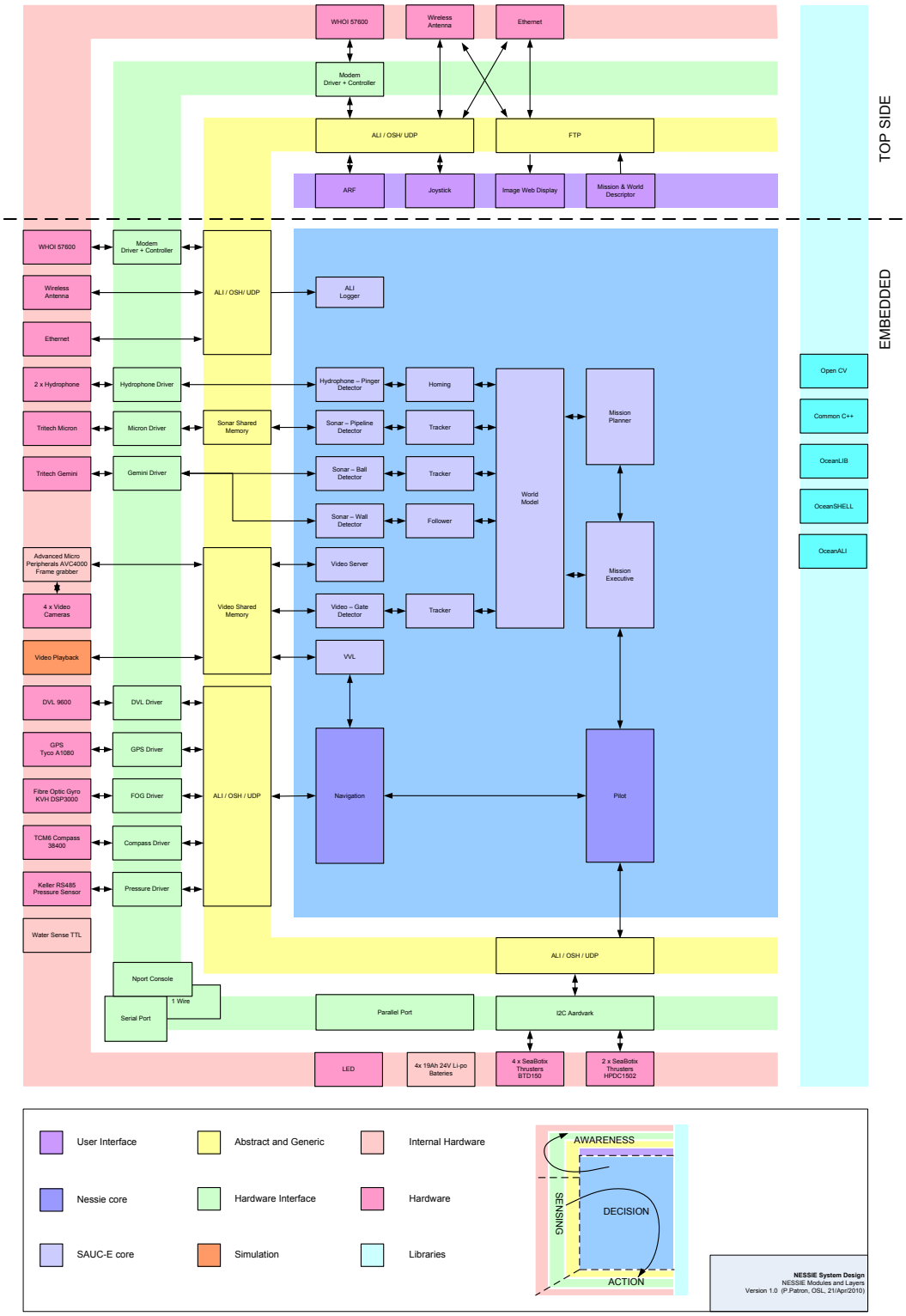


Fig. 13: Software design.

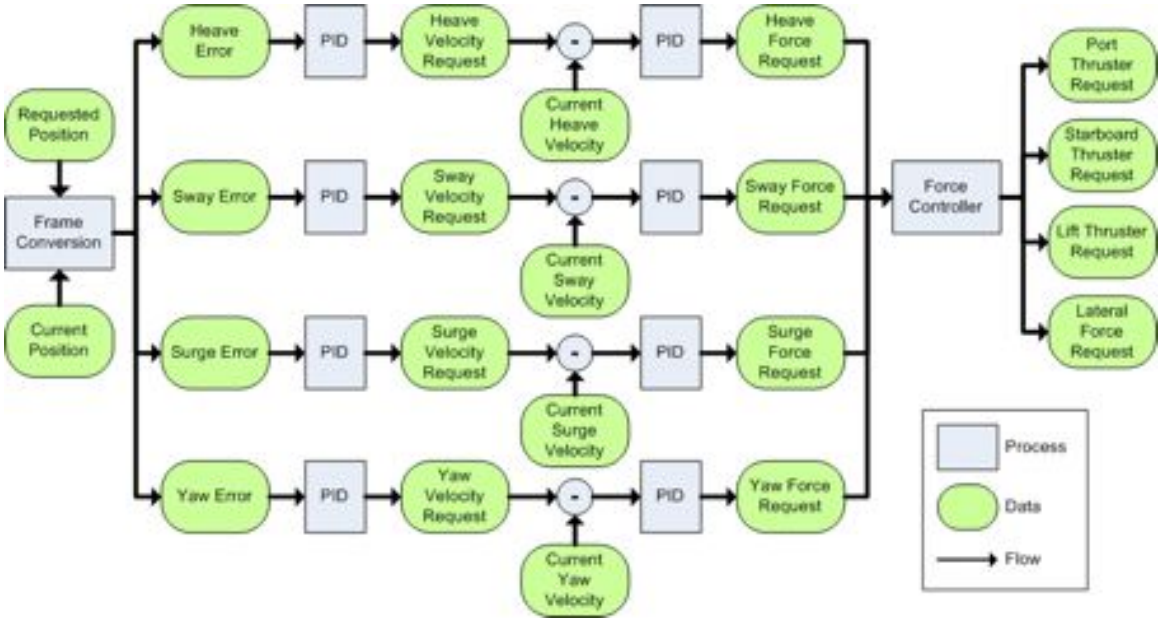


Fig. 14: Autopilot Position Control System

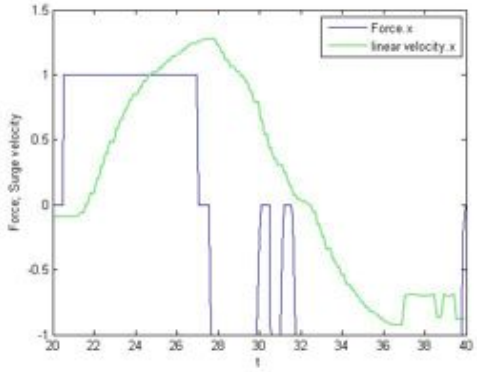


Fig. 15: Closed loop control of the autopilot showing the axis forces and velocity responses for the surge motion.

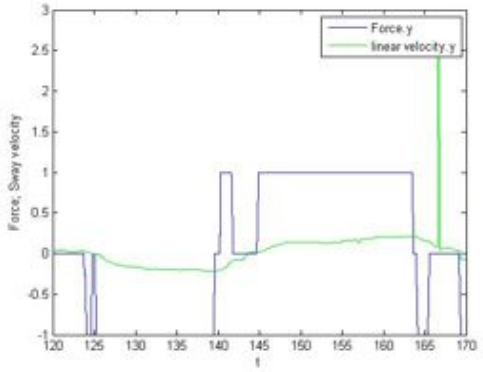


Fig. 16: Closed loop control of the autopilot showing the axis forces and velocity responses for sway.

C. Navigation

Position of Nessie VI is corrected with GPS signal when it surfaces. Once it is submerged, the vehicle uses all the information collected from the motion and rotation sensors. The DVL measures the linear velocity of the vehicle with respect to the sea bottom along the surge, sway and heave axis. Fusion of measurements from TCM compass and FOG is used to estimate vehicle heading as well as the pitch and yaw angular velocities. Measurements of heading rate obtained from FOG are highly accurate while TCM contributes with absolute heading information. Depth measurement is assigned to the pressure sensor.

The navigation system designed for Nessie VI is based on an Extended Kalman Filter (EKF) algorithm [1] which aims at estimating the position of Nessie VI within the competition area using the measurements from the sensors on board. EKF is a recursive estimator whose cycle consists of two stages called the state prediction and the state correction (Fig. 19). Prediction stage employs the state-transition model, while correction employs the measurement model.

Five degrees of freedom "constant velocity" state transition mathematical model [2] is used to carry out the prediction of the vehicle's location assuming it maintains its linear and angular velocities within the time scale of the updates thus following the smooth curved trajectory.

The concept of sensor fusion relies on the usage of EKF's measurement model to combine together different measurements. Filter periodically creates an observation by gathering together information from different sensors keeping the most recent measurements. Observation is used to correct the estimation obtained after the prediction stage, as a standard stage of Kalman filtering.

Benefits of the usage of EKF include the ability to make the prediction and combine together currently available sen-

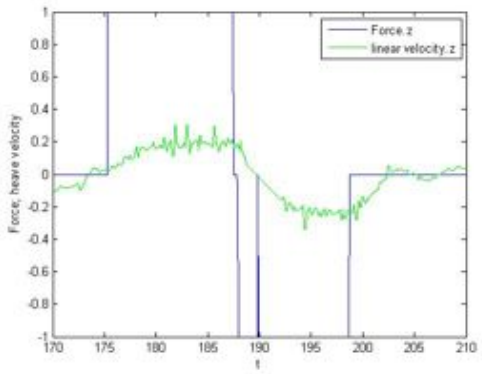


Fig. 17: Closed loop control of the autopilot showing the axis forces and velocity responses for heave.

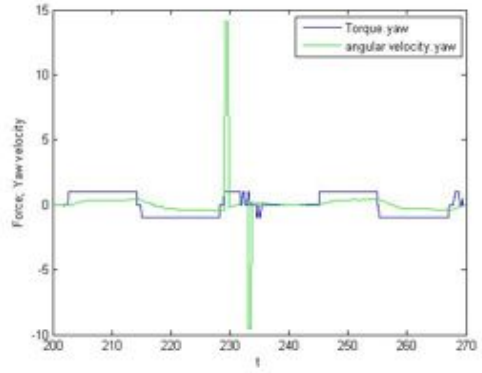


Fig. 18: Closed loop control of the autopilot showing the axis forces and velocity responses for yaw.

sory data, thus compensating potential sensor interruptions. Besides, filtering contributes in making an overall estimate with less drift (Fig. 20) and smoother trajectory. Finally, estimation strategy can be modified by tuning the elements of prediction and measurement error covariances, resulting in ability to set different levels of confidence in particular sensor measurements or aspects of system model. Final estimate tends to be optimal with respect to set expectations.

Comparison of the localisation algorithm using EKF and dead-reckoning when driving the vehicle roughly towards the north within the pool of 12 m length is presented in Fig. 20. EKF results in less drift.

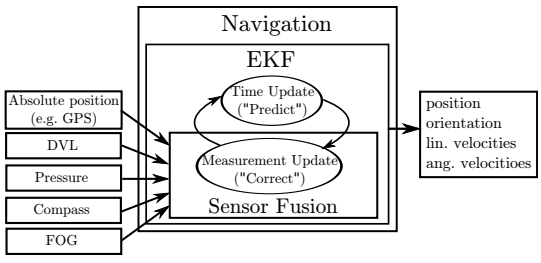


Fig. 19: Sensor fusion & navigation using EKF.

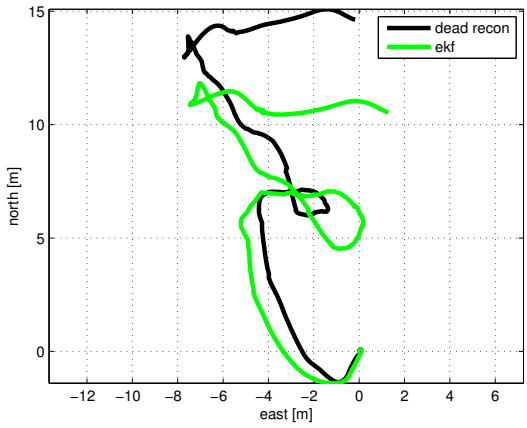


Fig. 20: An example of the navigation: North-east localisation of the vehicle.

D. Visual/Sonar Object Detection

To provide the vehicle with object detection capabilities our architecture distributes visual sensor data using a client-server approach. For each sensor, the video server provided by ROS is used and manages run-time configuration, image acquisition, and the image distribution. Conceptually, captured images are treated as frames of video data, allowing streams of sonar or camera video to be distributed to client modules. ROS enables up to 10fps video transfer rates to be achieved with very low computational overhead. The same rate is achieved even when saving to disk as SSD hard drives are used. This comes at the expense of large log files. Each video stream has allocated a unique ROS topic where the images are published. Several clients can access the same data at the same time by registering to the topic. A zero-copy mechanism enables to share the data efficiently when the clients are located on the same computer.

1) Buoy detection and tracking: The mid-water orange ball target is detected with the forward camera. Three detection methods are possible: colour-based detection, detection of circles, and a Mean-Shift Algorithm method. One of these methods will be selected for use in the competition, depending on the conditions experienced on the first day in the tank. Considering the experience of the past years, the most likely candidate is the colour-based detection.

a) Colour: As for the colour detection, the image colour space is mapped from BGR (Blue, Green, Red) into YCrCb (Luma, Red Chroma, Blue Chroma). For the red ball, the red chroma is selected and a threshold applied to highlight the orange colour of the ball. An open morphological transformation using a disk-shape structuring element is then applied (an erosion followed by a dilation), in order to remove the noise. A more robust method consists of subtracting the green channel from the red one, instead of considering the red one only. Finally, the circularity of each remaining element is calculated, and those above a given threshold are taken to be orange balls. This method is very simple, robust and efficient.

b) Hough Transform: Besides colour, the other main feature of the balls is their shape. A Hough Transform may be used to detect the circles, but applied naively this can result

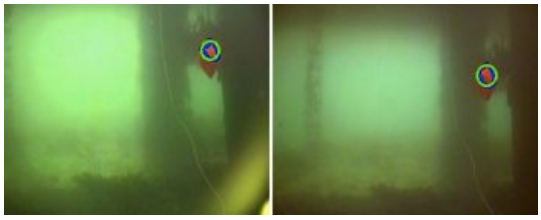


Fig. 21: An example of ball detection.

in many false alarms. To avoid these errors, temporal tracking is performed in which circles not present over a number of frames are disregarded. This can still result in some false positive detections, but it is a very fast operation.

c) Mean Shift Algorithm: The Mean-Shift algorithm allows tracking of elements using a reference histogram in a given colour space component. The orange ball is very saturated compared with the background, hence a reference histogram may be used in the saturation component of the HSV colour space to detect it. This method is robust and efficient, requiring a only a small amount of computation.

d) Stereo Vision: By combining object detections from two images taken from differing viewpoints, it is possible to obtain estimates of relative 3D position. Firstly, an object is tracked in both images using the detection algorithms mentioned above, which give the centre of the object. Epipolar geometry is used to reject outliers, and application of the intrinsic matrices for each camera converts from pixels to metric values. Finally, the triangulation method is used to calculate the estimated 3D positions. In practice, the 3D position estimation is good, with an error of between 0.01 and 0.05 metres at a distance of 2.5 metres from the cameras. The process is described in more detail in [3].

An example of mid-water target detection using stereo vision is given in Fig. 21.

2) Pipe detection and Tracking: Pipeline following is achieved through an image-based visual servoing scheme. The system consists of two procedures; feature extraction and control law. In feature extraction, the pipeline region is detected using a dynamic threshold applied to the yellow component of the image, an example of the this step is shown in Fig. 22(a). Then the two lines that delimit the pipeline are extracted by using a combination of statistical method and Hough transform to insure the accuracy. In the statistical method, the orientation and the center of the pipeline are calculated based on the distribution of pipeline's pixels resultant from the thresholding process. This information is used as prior knowledge to locate the two lines we are looking for in Hough space. Fig. 22(b) shows the results of detecting the two lines of the pipeline. In control law, an error is calculated as the difference between the current lines parameters and the line parameters at the goal position in which the pipeline has to be vertical at the centre of the image. The error is used then to produce a velocity screw by the help of image Jacobian for the line's features adopted to maintain 4 d.o.f. The *following behaviour* is achieved by introducing a secondary objective to the forward axis of the vehicle. In this case, the velocity component is formed as an inversely proportional function of the error, so it is activated gradually once the vehicle is aligned with the pipeline. Fig. 23 shows the architecture of the visual servoing system.

(a) Segemented image (b) Pipeline delimiting lines

Fig. 22: Pipeline detection.

3) ASV Tracking and following: The detection of the autonomous surface vehicle is achieved based on vision. Similar algorithm used for pipeline detection in the previous paragraph is applied to detect the ASV, then to align it with the vehicle using the same control scheme. The difference is that the secondary objective is to localise the ASV at the centre of the image. The centre is computed based on statistics; the velocity component of the forward movement of the vehicle is set to be proportional to the distance between the centre of the robot and such of the image. Since the velocity of the ASV is assumed to be slow, it is guaranteed that the alignment with the vehicle is done faster than the *following behaviour*.

E. Pinger localisation

This task will be achieved by visual means. The process is very similar to the one employed for the pipe detection and following but a different colour space will be chosen and multiple Hough transforms will be applied.

F. Wall inspection

One of the tasks of the competition is the inspection of a wall that mainly requires following the wall and facing it at the same time. Nessie VI uses the Tritech Gemini sonar looking forward and the deformable virtual zone (DVZ) principle to perform this task. A safety zone around the vehicle is defined within the sonar field of view. The intrusion of a wall in the safety zone results in a deformation of the safety zone that captures the orientation of the wall and the current position and orientation of Nessie VI. A motion vector is calculated from the deformed zone in order to follow the wall at a constant speed. The DVZ principle has several advantages. It does not require the arduous the implementation of switches or path planning when dealing with corners and is efficient in terms of computer calculations. The forward looking sonar has a field of view of 120 degrees and a maximal range of 50 meters. Only a range of 4 meters is used for the wall inspection task. The field of view is made of 256 beams from which points are extracted and converted in an intrusion profile. Nessie VI has to move parallel facing the wall, which requires controlling the rotation and the motion in the direction perpendicular to the wall. This results in an over-damped oscillator.

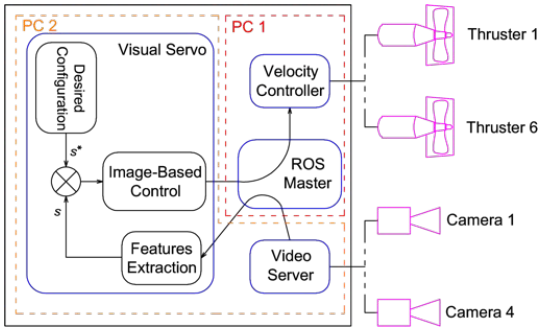


Fig. 23: Visual servo system architecture

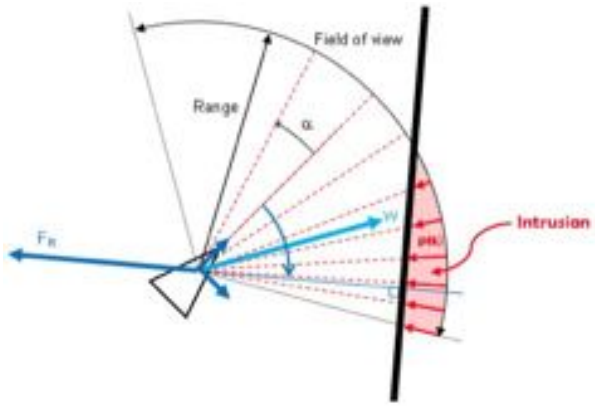


Fig. 24: Intrusion and reaction force

Simulations were performed with a wall made of planar faces with interior and exterior corner and show excellent results. The distance of Nessie VI to the wall converges to the desired 3 meters. It decreases in the interior corners due to the limited sonar field of view, which hinders the detection of changes in the wall profile. The distance to the wall diminishes temporarily. The temporal evolution of the surge, sway and rotational velocities as well as the evolution of the heading (yaw) shows the convergence of the kinematics to the desired ones. Nessie VI tends to face the wall and move parallel to it with sway velocity. Peaks and valleys are just transitory and they correspond to changes in the wall profile (Fig. 25- 27).

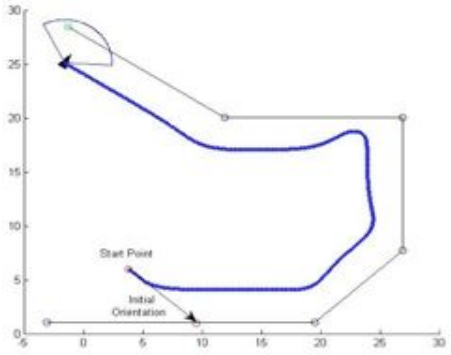


Fig. 25: Tracking of the trajectory.

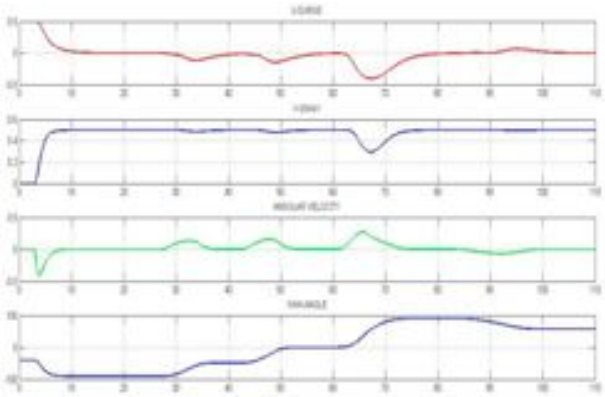


Fig. 26: Temporal evolution of the distance between the wall and NessieVI.

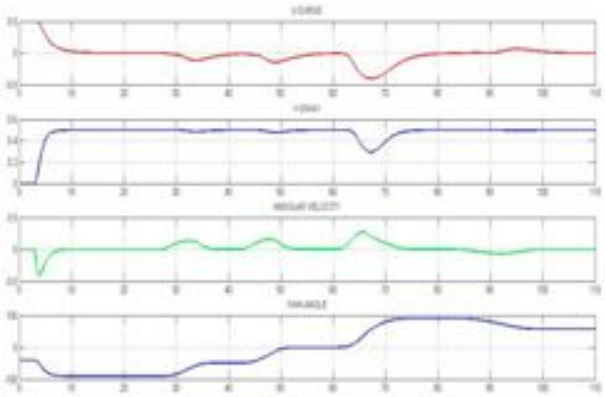


Fig. 27: Temporal evolution of the velocities

G. Mission Planning and Control

Execution of actions and communication of mission planner with other modules, such as *Autopilot* and *Navigation* module, are managed by Action Master. Each action was designed as a finite state machine based on a uniform template provided by action core which defines the control commands and representation of mission status. A novel priority-based approach for adaptive mission planning especially suitable for this competition was implemented and integrated into action master module to manage execution of the Nessie actions.

1) Action Master: Messages received from the other modules will be first verified and recognized. Information of current state and executed action is updated using messages from *Navigation* module. Action master checks the existence and availability of specified action by checking whether the preconditions are satisfied. Execution status and time consumption of each action are continuously monitored. Once an action succeeded or failed, action master will clear it and start another task.

2) Action Core: Execution state of actions are marked as *Stopped*, *Enabled*, *Running* or *Finished* which is attached by action master. State of expected action effects are marked as *Pending* or *Achieved* based on the messages received from *Navigation* and Object detection modules. Control commands are sent as hover request messages to *Autopilot* module with

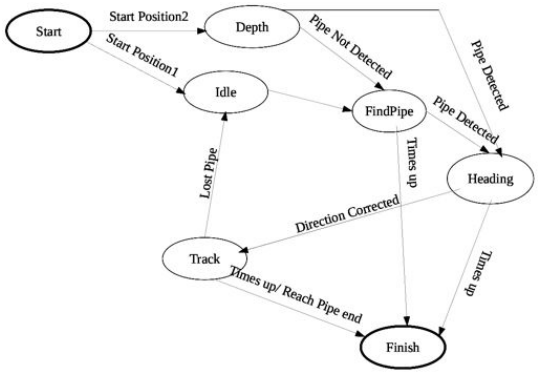


Fig. 28: Finite State Machine Defined for the PipeFollow Action

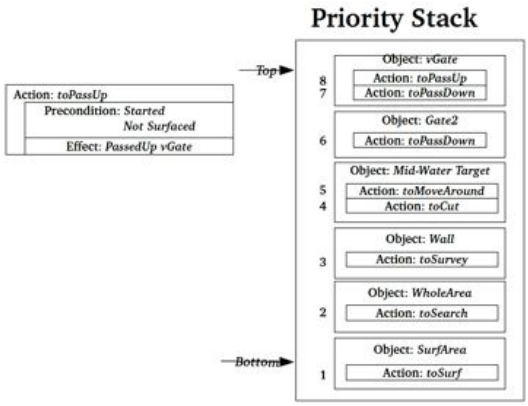


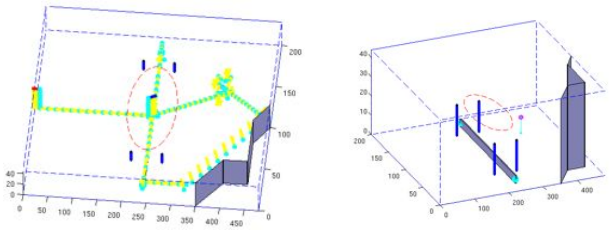
Fig. 29: The definition of priority stack. Structure of actions are shown on the left.

respect to world or local coordinates.

3) Action Implementation Example: When marked as *Enabled* and no other action is running, the execution state of one action is marked as *Running* and the vehicle starts the finite state machine defined for this action. As an example, the state machine of the action *PipeFollow* is shown in Fig. 28.

4) Mission Planning and Action prioritization: We proposed a novel planning method which is a simplified Markov decision process. All defined actions and objects were assigned a priority based on the scores of each task and succeed rate of each action. These actions are stored in a sub-stack in the main priority stack. The action labelled highest priority at the top. Fig. 29 shows the structure of the priority stack and actions indexed by different objects stored in it.

During the planning process, mission planner scans the stack from top to bottom and outputs the first action whose object exists and precondition satisfied. If the goal effects of the output action is reached within the assigned time limit, the action is marked as a succeeded task, otherwise failed. When the indexed actions of one object finished or the preconditions are not satisfied, the object is marked as *Stopped* again. This priority structure make the actions of low-priority objects breakable by actions of high-priority objects. When the actions of high-priority objects finished the actions of low-priority objects recover.



(a) reference plan (b) simulated scenario

Fig. 30: Trajectory generated by reference plan and simulated scenario: the red point of trajectory is starting position and blue point is surface position.

Using the Plan Proximity metric [4], the approach was evaluated under a partially-known dynamic scenario described by the SAUC-E 2011 mission rules. For evaluation purposes, the mission environment was not fully known a priori by the mission planner, but instead, it was discovered through the execution of the mission. A human generated mission plan was used as a reference ground truth for the evaluation of the different planning strategies. Fig. 30(a) shows the trajectory of reference plan in scenario shown in Fig. 30(b).

The proposed adaptive planning strategy was executed with different values of the planning horizon T, the discount factor β and the laziness factor. The different outcomes are evaluated by looking at their Plan Proximity to reference plan. The comparison results showed a high degree of similarity between our approach and the humanly driven adaptation.

IV. Innovation

The innovations concerning Nessie VI are related to both hardware and software. The new mechanical design has greatly improved hydrodynamics and speed of access to the key components of the platform.

Pipe tracking task is commonly performed using side-scan sonar technology or video cameras. Instead of replacing our existing micron sonar with a side-scan sonar, we decided to make use of existing resources and develop a new algorithm for pipe tracking. This task is now performed with a rotating sonar and uses a variety of image processing techniques to extract information about the pipe's position from the sonar image. Because the micron image only provides a cross-section of the pipe, a line-fitting algorithm was also developed to establish the orientation of the pipe.

Although locating a mid-water target has been an element of previous competitions, this year' addition of the line-cutting task has required an improved controlled system to be developed.

In order to perform the wall following task, this year we chose to utilise a new Gemini forward looking sonar. This required the full development of a Linux driver in order to integrate the sonar into the Nessie VI architecture.

V. Conclusion

Nessie VI is the new vehicle of the Ocean Systems Laboratory building upon the achievements of previous vehicle

while eliminating several shortcomings. The significant change in vehicle design required a new motion control mechanism to handle five degrees of freedom and several new hardware drivers were developed. A new vehicle maintains the ability to hover to close object inspection while making significant improvements in transit time.

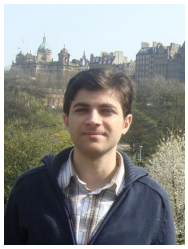
ACKNOWLEDGEMENT

Team Nessie would like to thank its sponsors, BP Ltd., PNI Corp. Tritech International Ltd., SeeByte Ltd., SeaBotix Inc. and Teledyne RD Instruments. We would also like to thank all the people at the Ocean Systems Laboratory for their support.

TEAM BIOGRAPHIES

Jonathan Clay is currently studying for a Masters in Robotics and Cybertronics, after which he hopes to find a PhD related to AUVs within the OSL lab. His interest has grow greatly with his involvement in the Nessie project and with his fourth year project mAUV. He is the current President of the Robotics society at Heriot-Watt University, during the last year he has been working closely with the staff in the electrical department to help produce a Jenga playing robot. When workloads he also works as a freelance events technician though an agency, the highlight of which has been working at T in the park for two and a half weeks last year. This was a great experience working across the site on a number of challenges with a wide range of people.

Francesco Maurelli is a PhD candidate at the Ocean Systems Laboratory, Heriot-Watt University (Scotland, UK). Following the Master program in Robotics and Artificial Intelligence at Sapienza, University of Rome (Italy), he prepared his thesis at Fraunhofer Institut IAIS (Sankt Augustin, Germany). His work was focused on a novel 3D laser scanner for autonomous vehicle navigation, used also in the framework of the Darpa Urban Challenge, with Team Berlin. He joined the European Marie Curie Research Network FREEsubNET in October 2007, at Heriot-Watt University. His work is focused on two main parts: standards for autonomous docking (manager of the work package), and AUV (Autonomous Underwater Vehicle) navigation, with an emphasis on sonar-based localisation and SLAM, using probabilistic techniques.



Mohamad Motasem Nawaf Received his bachelor degree in Computer Engineering from the University of Aleppo, Syria 2007. Then he studied a research master of computer since, specialized in information technology and web from the National Institute of Applied Science of Lyon (INSA de Lyon), France in 2009. He received a grant from European Commission as being selected for the Erasmus Mundus Masters in computer vision and robotics, a joint programme of the University of Burgundy(France), University of Girona(Spain), and Heriot-Watt University (United Kingdom). He did his master thesis at Ocean Systems Lab as a member of the Nessie team. His focus was on developing vision-based control for autonomous underwater vehicles.

Chengjia Wang studied Electronic Information Engineering and English Language in Qingdao University of Science and Technology, China, 2005. He received a bachelor of engineering (ranking 1/139) and a bachelor of art in 2009. The same year, he was granted with an Erasmus sholoarship and study at Heriot-watt University, University of Girona and University of Burgundy in the course of Erasmus Mundus Master in Vision and Robotics (Erasmus VIBOT). He joined Nessie team in Ocean System Laboratory of Heriot-watt University and wrote his master thesis in the field of adaptive mission planning of autonomous underwater vehicle (AUV) which will applied to NessieVI in SAUC-E 2011. A planning-control system was built up with planning algorithms based Markov decision process (MDP). He graduated from his master course with distinction in 2011. He's now looking for a PhD related to computer vision, medical image processing or autonomous robots.

Miroslav Radojević received a Bsc. degree in automatics at the Faculty of Electrical Engineering, University of Belgrade, Serbia in 2008. After graduating, he worked for a year developing algorithms for relay protection. In 2009 he joined Erasmus Mundus Masters Programme ViBot that specialises in computer vision and robotics which he graduated with distinction in 2011. He joined team Nessie of the Ocean Systems Laboratory in February 2011 where he prepared his masters thesis. His main activity in the team involved designing the EKF-based navigation module for Nessie VI.

Yvan Petillot is a Professor at Heriot Watt University. His main areas of interest are image understanding, sensor fusion and underwater robotics. He is an active member of the Oceans Systems Laboratory and the Vision Image and Signal Processing Group. Finally he is a director of SeeByte Ltd, a Spin-out of the Oceans Systems Laboratory commercialising some of the technologies developed in the Oceans Systems Laboratory. Yvan holds an Engineering degree in Telecommunications with a specialisation in Image and signal processing from ENSTBr. He also has a M.Sc. in optics and signal processing and a Ph.D. in image processing.

REFERENCES

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*, R. C. Arkin, Ed. MIT Press, ISBN: 0-262-20162-3, 2005.

[2] D. Ribas, "Underwater slam for structured environments using an imaging sonar," Ph.D. dissertation, University of Girona, Girona, Spain, September 2008.

[3] J. Cartwright, N. Johnson, B. Davis, Z. Qiang, T. L. Bravo, A. Enoch, G. Lemaitre, H. Roth, and Y. Petillot, "Nessie iii autonomous underwater vehicle for sauc-e 2008," in *Proceedings of UUVS 2008, Southampton, UK*, 2008.

[4] P. Patrón and A. Birch, "Plan proximity: an enhanced metric for plan stability," in *Workshop on Verification and Validation of Planning and Scheduling Systems, 19th International Conference on Automated Planning and Scheduling (ICAPS'09)*, Thessaloniki, Greece, September 2009, pp. 74–75.