

The Endeavour ASV: Hardware, Sensor & Software Overview

Queensland University of Technology

*Riki Lamont, Scott Nicholson, Zoë Renando, Christopher Dirkis, Peter Smith, David Jakes,
Jaynesh Vanmali, Scott Veitch, Amit Chong Bang
Dr Matthew Dunbabin*

Abstract

This paper provides an overview of the hardware, sensor and software systems of the *Endeavour* Autonomous Surface Vehicle (ASV) developed by the Queensland University of Technology for entry in the 2014 Maritime RobotX Challenge. The design features include a modular sensor and computing hardware arrangement to facilitate rapid assembly and transportation between field sites, and a unique dual electric propulsion system. A minimal, yet capable sensor suite and software architecture built on the Robotic Operating System (ROS) has been developed to allow completion of all five of the challenge tasks. The ASV's control, mapping and task specific vision and acoustics-based algorithms have been evaluated in both simulation and through field experiments with results presented demonstrating their capabilities.

1 Introduction

The Queensland University of Technology (QUT) presents the *Endeavour* Autonomous Surface Vehicle (ASV) for competing in the 2014 Maritime RobotX Challenge. The undergraduate team's goal was to produce a simple, yet capable solution for reliably undertaking all of the Challenge's tasks [1].

Previous international robotic boat competitions have focused on fully customised small-scale platforms. However, for the 2014 Maritime RobotX Challenge, the organisers have provided each team with a standardised platform, the WAM-V developed by Marine Advanced Research Inc. Each team is responsible for outfitting the platform with propulsion, sensing, computing and software systems.

A particular challenge in ASV development is selecting appropriate sensors, supporting software systems, and algorithms to undertake often complex tasks. The Maritime RobotX Challenge requires the ASVs to complete five tasks; (1) Navigation and Control, (2) Underwater Search and Report, (3) Identify Symbol and Dock, (4) Observe and Report, and (5) Detect and Avoid Obstacles. The *Endeavour* ASV systems have been selected, developed and evaluated to attempt and complete all five of these tasks.

The team's development strategy was to start field evaluation of the ASV as soon as possible, with the first on-water tests starting January 2014. This urgency was driven by the Australian tertiary academic calendar which conflicted with competition dates. An incremental development and testing schedule was taken for both



Figure 1: The Endeavour ASV showing the location of the sensing, computing, propulsion, safety and power systems (note the hydrophones are not shown but are mounted on motor brackets).

hardware and software systems. As such, the ASV underwent a number of design and sensor upgrades throughout its development with particular emphasis on safety, mapping, path-planning and vision system development.

The remainder of the paper is structured as follows; Section 2 provides an overview of the Endeavour ASV including sensor, propulsion and safety systems, with Section 3 describing its software architecture. Section 4 outlines the approach to executing the individual mission (task) components presenting experimental results. Section 5 describes the collaborative relationships forged throughout the development cycle, with Section 6 concluding the paper.

2 ASV Endeavour overview

Figure 1 illustrates all the sensors, computing, safety and propulsion systems as installed on the Endeavour ASV. The

following sections describe in more detail each of these systems.

2.1 Sensor Systems

A key requirement for any autonomous system is an appropriate sensor suite. These sensors are required to provide the necessary perception, navigation and control capabilities for the ASV to undertake each task. On the Endeavour ASV, sensors were chosen to provide complimentary and redundant data. Figure 2 shows an early testing configuration of the sensor suite mounted on the payload tray of the ASV.

2.1.1 Perception sensors

The perception sensors provide most of the fundamental information for executing the Challenge tasks. They also provide the situational awareness for navigation and obstacle avoidance. The Endeavour ASV has scanning laser rangefinders, cameras and hydrophones for observing above and below the water.



Figure 2: Early configuration of the Endeavour ASV sensor suite showing a camera, spinning and nodding laser scanning range finders, GPS, IMU and weather station.

Endeavour has two scanning laser rangefinders, both attached to rotating bases (see Figure 2). The first, a Hokuyo UTM-30LX, has a scan sector of 270 degrees and an effective outdoor detection range of approximately 15 m. This scanner is rotated on the vertical axis to provide close range object detection. Whilst proportions of the scans are pointing towards the sky and not used in this competition, this configuration allows for detection of overhanging hazards such as trees and bridges. The second scanning laser rangefinder is a SICK LMS291. On Endeavour, this scanner is mounted facing forward and nods up and down providing obstacle detection out to 80 m in front of the ASV.

Many competition tasks require computer vision to interpret parts of the environment (e.g. symbols, buoy colours). Therefore, Endeavour has two Logitech HD 720p USB web cameras. These cameras are processed as 480 x 640 pixel images to allow real-time on-board processing.

To measure the underwater acoustic environment, Endeavour has two Reson

TC4032 hydrophones connected to a Roland Quad-Capture USB Audio Interface Device capable of sampling at 192 kHz.

2.1.2 Navigation Sensors

Primary navigation sensing for Endeavour is from a GPS receiver and magnetic compass. A Hemisphere A325 GPS receiver with OmniSTAR differential correction provides position updates at up to 20 Hz.

As the ASV typically operates in a relatively flat environment (low sea-state), a low-end magnetic flux compass was selected for providing compass/heading angle (9DOF Razor Inertial Measurement System). In addition to heading angle, this sensor provides roll, pitch and angular rates which are all used in the navigation solution. In practice it was found to be very sensitive requiring recalibration after any hardware position changes on the payload tray. However, following calibration it proved reliable for real-time navigation and control of the ASV.

2.1.3 Meteorology

Experience with previous ASV systems [2] found that particularly at low-speed, guidance can be improved by incorporating wind data into the controller. It was decided early in the integration process to install a Vaisala WTX-520 weather station on the ASV to provide real-time wind-speed and direction data (at 1 Hz) to the on-board control system.

2.1.4 System Sensors

Endeavour has a range of current and voltage sensors for monitoring the motors

and batteries which provides real-time feedback to the main software safety system. Monitoring these variables is critical for battery management and detecting any system fault.

2.2 Payload tray

All hardware, apart from the motors and emergency stop buttons, is mounted to the payload tray. This strategic decision facilitated the repeated assembly and disassembly for transport to the team's remote field sites.

The placement of hardware mounted on the payload tray (sensors, computing and power) was carefully chosen to; assist with load distribution when on the water, ensure that each sensor maximises its field-of-view, and to ensure obstruction-free lifting of the vessel via the pre-installed lift points.

2.3 Propulsion System

The Endeavour ASV has two independent propulsion systems: *primary* and *auxiliary*. These systems were developed in such a way to maximise field testing time as well as the final system performance during the competition.

2.3.1 Primary propulsion system

The first propulsion system, known as the primary system, consists of two Osapian 12 volt electric trolling motors (see Figure 3) each capable of producing 40 lb of thrust. This relatively inexpensive system was implemented due to their ease of mounting and robustness in prior work [2]. The motors are driven with reprogrammed Botbitz 85 Amp brushed Electronic Speed

Controllers (ESC) and powered off a 105 Amp hour deep cycle lead acid battery.

Whilst the primary motor system proved reliable throughout the entire development campaign, the shallow propeller pitch and its power rating restricted the maximum achievable speed of the ASV to around 3 knots. Therefore, an auxiliary propulsion system was built and integrated to work in conjunction with the primary system.

2.3.2 Auxiliary propulsion system

The auxiliary system is a unique and elegant modification to a gasoline driven outboard motor (see Figure 3). The basis of the system is a Parsun 2 horsepower outboard motor with an aluminium propeller. The power head (engine) was removed exposing the drive shaft and a RotoMax 50cc Size Brushless Outrunner electric motor attached via a flexible coupling. This dramatically simplified the design requiring only three parts to be built in-house; (1) aluminium spacer, (2) cylindrical plastic motor housing, and (3) a stainless steel top plate. The auxiliary motors are controlled by reversible Turnigy 200 Amp sensorless ESCs.

Due to the experimental nature of the auxiliary propulsion system, its purpose here is to provide significant on-demand thrust for manoeuvring and higher-speed sprints. Therefore, each motor has a smaller independent lithium polymer battery system with a capacity of 15 Amp hours at 28.8 volt. This gives each motor a continuous maximum power run time of approximately 10 minutes. This is considered sufficient for the competition, as many tasks only require the primary system.

The auxiliary motor system achieved fully laden ASV speeds of 6 knots.



Figure 3: The motors used on the Endeavour ASV. (Left) Two Osapian 12V electric trolling motors provide the primary propulsion system. (Right) Custom developed 28.8V electric motors provide a high-speed auxiliary propulsion system (shown attached to the propulsion mounting system).

2.3.3 Propulsion mounting system

The propulsion systems were mounted to the ASV via custom built removable transom plates (Figure 4). The motor mounting system exploited the existing mounting mechanisms of the WAM-V without the need to alter the pontoon structure or fixing points in any way. The motor mounts are made from aluminium square sections riveted together with wooden spacers. Their profile creates insignificant additional drag to the vehicle. This design allows for easy removal and stowage of the motors as a complete assembly. Additionally, the system also provides the mounting

structure for the hydrophones used in the acoustic search task.

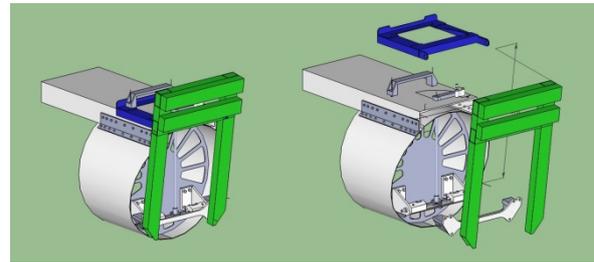


Figure 4: Schematic showing the custom designed propulsion mounting plate which holds both the primary and auxiliary motor systems as well as the hydrophones.

The combined primary and auxiliary motors with the quick release mounting system has proven reliable in practice. It also provides a flexible and redundant propulsion system for mission planning and task completion.

2.4 Safety Systems

The Endeavour ASV has both hardware and software safety systems. Both systems are interlinked to ensure that all motors are powered off when requested.

There are two manually activated emergency stop buttons on the port and starboard sides of the ASV's rear arch. Additionally, there is a wireless emergency stop (kill switch). This consists of two XBee Pro Series 1 RF Modules. The XBee in the handset provides a heartbeat signal via a digital line to the specified address of the on-vehicle receiving XBee. If the receiving device is not toggled every 0.5 seconds it will automatically de-activate the motor power relays (default power off).

An independent manual control link is provided by a commercially available Spektrum RC system. This system allows

switching between manual and autonomous control of the ASV. A loss of signal from this link also activates the emergency stop system.

To assist the development and remotely visualise the status of the ASV, a set of three flashing status lights are attached to the weather station mast (shown in Figure 1). These indicate the control status of the ASV; emergency shutdown mode (red), manually driven mode (yellow) and autonomous mode (blue).

2.5 Radio Reporting System

To facilitate system development and provide the necessary communications to the judges' network, two wireless communication systems are employed. The general data communications is via a TD-Link TD-W8968 Wireless Router selected to operate at the maximum allowable transmit power of 200 mW EIRP. This was evaluated at two different field locations with an effective communication range of 150-200m.

The communications to the judges' network is via the CPU's on-board wireless module (Intel Dual Band AC7260).

3 Software Architecture

The software architecture developed for the Endeavour ASV is shown in Figure 5. It is an event driven, cross-platform structure built on the Robotic Operating System (ROS) [3]. ROS was chosen as it provided seamless integration and extension of the individual software modules representing sensing, localisation, planning and control. It also

facilitates more rapid software development as it caters to the programming languages preferred by the individual team members (C++, Java, Python and Matlab™). Additionally, it facilitates both in-field and laboratory development due to the ability to use both "live" and "replayed" sensor streams to evaluate and refine individual modules and algorithms.

The software platform executes actions in parallel and asynchronously. The individual sensor modules provide data asynchronously to the state estimator and image, laser and acoustic classifiers. The *State Estimator* maintains the pose of the ASV for localisation and control. The *State Estimator*, laser and vision classifiers (for buoys) all feed the *Map Manager* which maintains an up-to-date, globally referenced map of real and virtual obstacles as the ASV traverses the courses (see Section 3.1.1). The *Path Planner* produces viable (efficient and obstacle free) trajectories for the ASV based on the obstacle map and the desired waypoints provided by the *Mission/Task Controller* (see Section 3.1.2).

The execution of the missions is performed by the *Mission/Task Controller* block. This block maintains a series of state-machines for achieving high-level task objectives and overall system function. The decision support structure within the state machines is parameterised allowing the controller to retry, abort or skip specific tasks, or components therein. The *Mission Controller* also determines the required motor control inputs sent to the *Motor Controller* for driving both the primary and auxiliary propulsion systems.

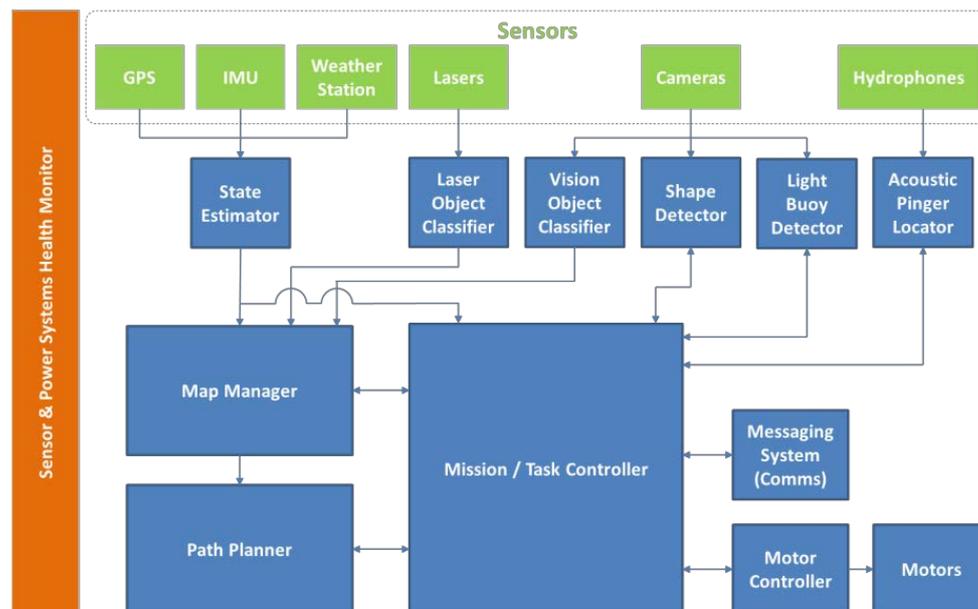


Figure 5: The Endeavour ASV software architecture.

To facilitate load management on the primary CPU (a Gigabyte GA-Z97N i7), particularly for computationally expensive computer vision tasks, the *Mission Controller* can put each classifier block into an “idle” state until required. Finally, the *Mission Controller* communicates with the *Messaging System* block which provides the heartbeat and task-specific messaging interface to the Judges’ wireless network.

The *Sensor & Power Systems Health Monitor* provides a system wide assessment of the operational state of the ASV. It is built on previous experience with marine robotic systems and includes monitoring motor and battery currents and voltages, and individual sensor performance with the ability to restart modules when necessary.

3.1 Mapping and Path Planning

A system was required to perform efficient obstacle detection and avoidance using data from multiple sources. The solution, commonly known as an occupancy grid, was chosen due to being fast and efficient to

manipulate and maintain across a modular system and integrated well with the chosen D-Star path planning (see Section 3.1.2).

3.1.1 Map Manager

The Endeavour ASV employs a custom scalable obstacle mapping system based on occupancy grids. This occupancy grid required integration of both actual and virtual obstacles (representing course boundaries and no-go zones) which could not be detected by on-board sensors. Additionally, through field experience it was determined that the occupancy grid requires a level of “forgetting” and confidence in its obstacle prediction. This is to account for spurious sensor readings, range estimation inaccuracies and temporally short-duration obstacles (e.g. birds flying within the sensor field or other vessels traversing the course).

The globally referenced occupancy grid was designed without bounds, being more flexible, robust and easy to maintain than a fixed size grid. This is achieved by dividing

the grid into chunks of 32x32 cells which are stored in a hashmap and created on demand. Each chunk has a coordinate, a change flag, and a 32x32 length byte array for the cell counts.

The occupancy grid is maintained by a client server system distributed over ROS for modularity. A server process handles laser scans and other input messages, populating the grid. A client process requests a copy of the grid from the server, and then receives messages updating chunks when the values change.

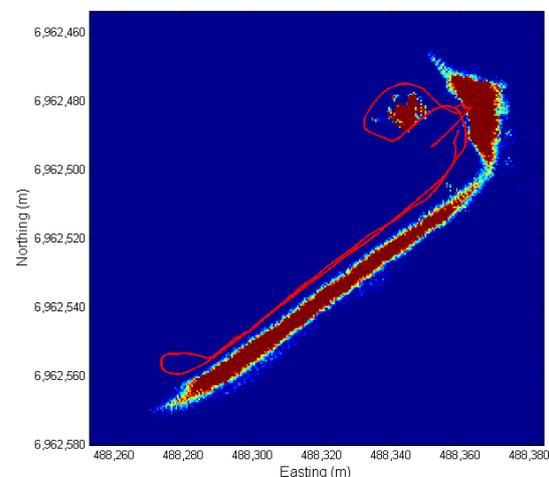
When a new laser scan message is received, the rays are translated to 3D coordinates, flattened to the ground plane and the cells within which they fall have their value incremented and capped. Rays reported with an unknown or infinite distances are traced to find their intersection with the ground plane, and if within a reasonable range of the laser, cells are decremented. This ensures that erroneous data or moving obstacles fade from the occupancy grid over time.

Cells can be assigned reserved values as a special flag for a course boundary and cannot be decremented. The server provides a service for drawing boundary lines which other systems call to mark known obstacles and course boundaries by coordinate.

On Endeavour, a cell size of 0.5 m was chosen as an appropriate trade-off between memory requirements and increasing path planning speed and trajectory granularity. Figure 6 shows an example of an occupancy grid produced by the ASV using laser scanning data.



(a)



(b)

Figure 6: (a) The Gold Creek Reservoir field test site, and (b) the associated occupancy grid produced by the ASV showing the trajectory of the boat (red line) and the grid colour representing the confidence of an obstacle (darker means more confident).

3.1.2 Path Planning

Path planning for Endeavour is based on a modification to the D-Star algorithm [4]. This algorithm is performed on the occupancy grid constructed above which represents both physical obstacles (e.g. buoys) and virtual obstacles such as course bounds.

The heuristic employed here varies from traditional D-Star approaches in that it

considers the speed and turning ability of the ASV in the cost function. The principle behind this is that a fast boat will be able to travel further in less time, but will not be able to turn sharply, or slow down quickly. Specifically, the planner assigns three different speeds that the ASV can travel - slow, medium and fast. The cost function then assigns a higher cost to slower speeds but with an ability to turn sharply. A moderate cost is assigned to medium boat speed but restricts its turning ability. Finally, a low cost is assigned to a fast boat speed but with minimal turning ability. Using this approach, straighter paths with higher speed are favoured but penalised when turning is required (e.g. avoiding obstacles).

To initiate a path search, the *Mission Controller* provides a set of goal points and if required a specified orientation at that goal (e.g. for docking or gate entry/exit). The path planner then searches for an optimal (minimal cost) path based on the implemented cost function and a predefined safety distance from obstacles, returning a series of waypoints and speeds for the boat to follow. During path execution, the planner continually monitors the map for new obstacles and replans the trajectory if necessary.

Figure 7 shows two example paths using a simplified cost function consisting of 3 speeds and 16 directions within a 1 m grid. This consists of both physical and virtual (boundary) obstacles and a desired keep-out region. This approach uses a bucket sort for linear time complexity leading to efficient path calculations. More direct, or less direct paths can be achieved with

modification of the cost function and/or cell granularity.

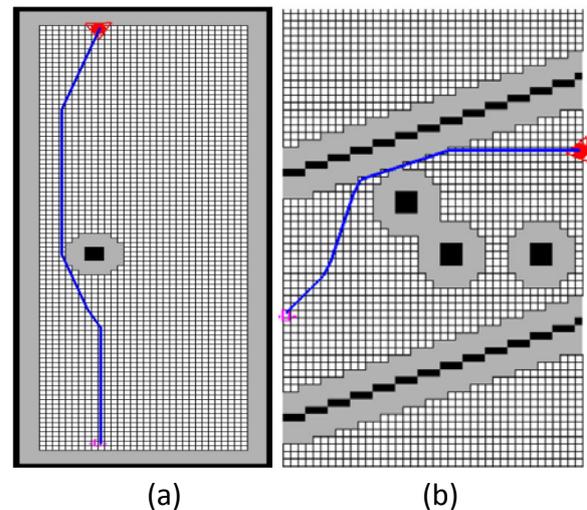


Figure 7: Examples of path planning for the ASV on a 1 m occupancy grid with a 3 m keep-out zone (shaded grey) around obstacles and course boundaries (thick black line). The start location is shown in pink, the end point in red. (a) Case with one obstacle. (b) Complex case whereby the ASV chooses a narrow gap between the course boundary and obstacle.

4 Mission Execution

Missions are performed by the combination of a set of control *primitives* (behaviours) and task specific *detectors*. The control primitives provide the basic movement of the ASV and include; (1) waypoint, (2) search, (3) station-keep, and (4) target follow. The sole purpose of the *detectors* are to process image and laser data for coloured buoys, symbols and light-buoy sequences, as well as estimate the position of the acoustic pingers. It is the role of the *Mission Controller* (Figure 5) to combine these primitives and detectors in order to execute each task using decision rules built

on experience obtained during vehicle testing. The following sections describe these primitives and detectors.

4.1 Control Primitives

Control primitives are the building blocks from which missions are conducted. The Endeavour ASV has four primitives (waypoint, search, station-keeping and target-tracking) representing the minimum set of behaviours determined to achieve robust and repeatable missions in the marine environment. The *Mission Controller* selects the most appropriate primitive to achieve each the task objectives.

Waypoint control uses the desired trajectory coordinates returned from the path planner (Section 3.1.2). The search behaviour performs a spinning search pattern. The target-tracking behaviour maintains a point of interest to be in the centre of the image and either approaches, or keeps a set to a distance from the object. The station keeping behaviour maintains the boat at its current GPS location. In the later three cases, the obstacle map is continually monitored to ensure no collisions will occur.

4.2 Task Specific Detectors

Task specific detectors consist of vision-based buoy, symbol and light-buoy detection, as well as an acoustic pinger localisation system. The following sections describe these detectors.

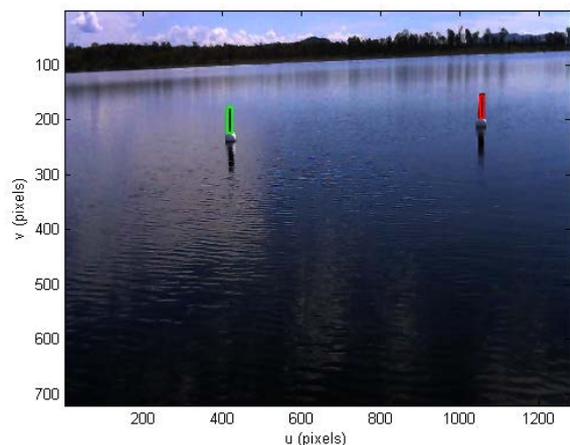
4.2.1 Buoy Identification

Computer vision was chosen as the means to detect buoys and identify their colour

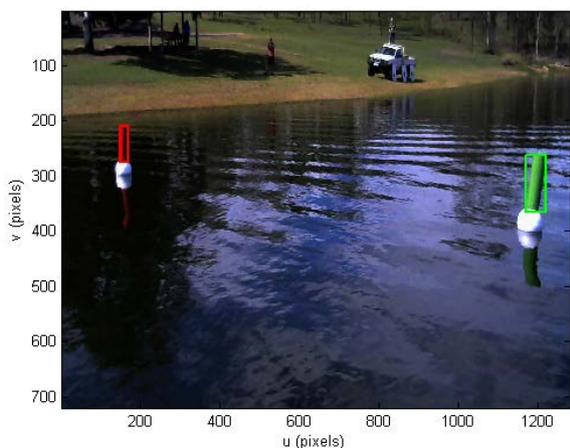
and shape. However, there are many challenges associated with computer vision in outdoor environments. These challenges include varying lighting conditions, reflections from the water, shadows and background noise from within or outside of the course.

The vision system on-board the Endeavour ASV uses a combination of tools from both Open Source Computer Vision (OpenCV) library and the Machine Vision Toolbox for Matlab [5] linked to ROS through a Beta version of the Matlab™ Robotics System Toolbox. OpenCV and the Matlab™ toolboxes provide a flexible and powerful framework for algorithm development, and image analysis tools for rapid parameter tuning and visualisation.

Buoys are first segmented from the image by performing an image correction procedure followed by multi-channel thresholding, with shape and size verification. The threshold parameters were selected from experimental data to be robust under a range of lighting conditions. Due to the potential for false positives, a temporal and spatial filter is employed. The temporal filter checks for consistency between multiple image frames, whilst the spatial filter estimates the buoy position in the global coordinate frame and compares it with the laser-based obstacle map via a likelihood function. If matched, the buoy position and colour are returned to both the *Mission Controller* and the obstacle map. The algorithm was found to be robust under varied range and lighting conditions. Examples of navigation buoy detection for varied lighting conditions are shown in Figure 8.



(a)



(b)

Figure 8: Example of correct buoy identification (colour and shape) from the ASV’s on-board camera in both (a) shadow, and (b) sunlit conditions.

4.2.2 Symbol Detection & Identification

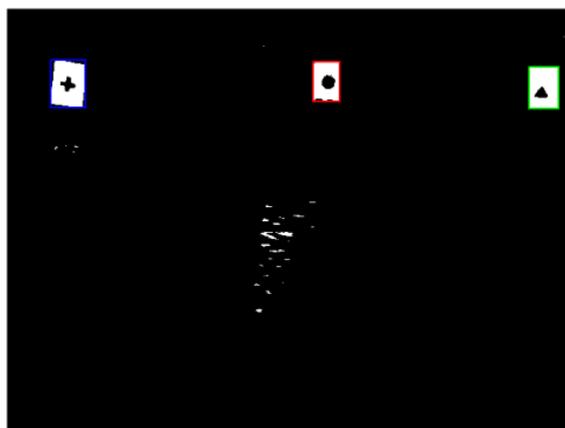
The docking task requires robust identification of one of three symbols (circle, triangle and cruciform). A number of vision-based image analysis approaches were evaluated during algorithm development to improve detection performance. The challenges again were varied lighting conditions, particularly shadows and low-resolution of the images. On the Endeavour ASV, the resulting symbol detection approach is based on image thresholding and template matching. To

improve target detection reliability a temporal filter is used to reject noise. To improve classification performance, additional similarity metrics are incorporated to the template matching.

Figure 9 shows an example of symbol detection and correct identification. This approach proved robust in practice to image reflections and background noise and provided detection up to 12 m with symbols only 0.3 m in height.



(a)



(b)

Figure 9: Example of correct symbol detection and classification from the ASV’s on-board camera. (a) Original image, and (b) classified image with the blue square denoting a cruciform, red square denoting a circle, and green square denoting a triangle.

4.2.3 Light Buoy Detection & Sequence Identification

The Observe and Report Task of the Maritime RobotX Challenge requires the detection of a “light buoy” and the decoding of the light sequence that is displayed. The light buoy is firstly detected using the laser-based obstacle detection system described above. Once an object is found in within the course boundary, the ASV manoeuvres to the obstacle for verification. Verification is performed using the same image thresholding approach as for symbol detection. Upon verification, the ASV uses the target tracking primitive to maintain a set distance from the light buoy and performs image-based colour segmentation at a region-of-interest around the buoy. Temporal filters are applied the segmented image sequence and observation is maintained until a predetermined level of confidence is obtained in the detected color sequence. Figure 10 shows an example segmented image from the ASV undertaking light buoy detection and the associated predicted light flash sequence.

4.2.4 Acoustic Pinger Localisation

The Underwater Search and Report task of the Maritime RobotX Challenge requires the identification of an underwater acoustic pinger and the estimation of its position and depth. The Endeavour ASV has two Reson TC4032 hydrophones mounted to the propulsion mount and connected to a Roland Quad-Capture USB Audio Interface Device. Matlab™, operating on a Windows™ laptop and integrated to ROS via a Beta version of the Matlab™ Robotics

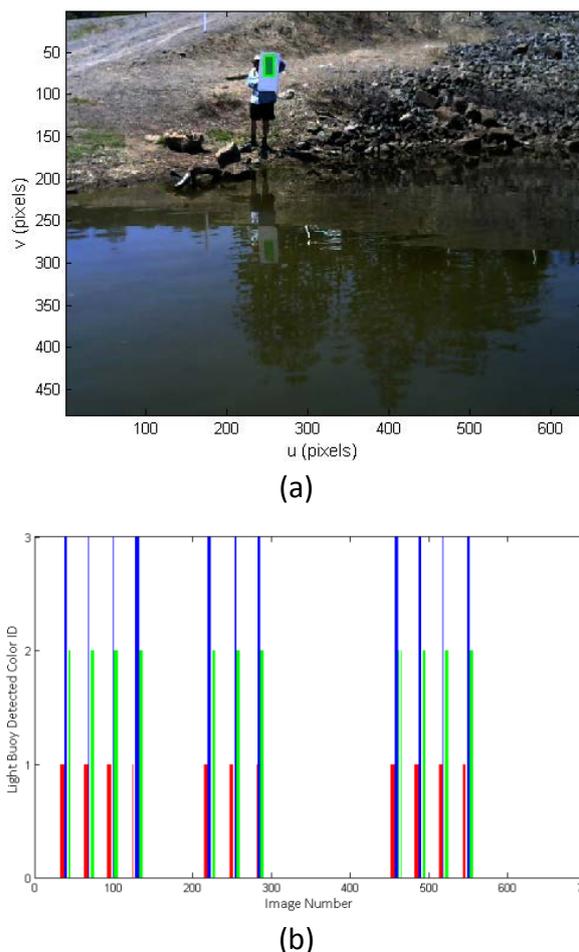


Figure 10: Example showing (a) light buoy segmentation from the ASV’s on-board camera, and (b) the associated detected colour for a sequence of images illustrating the light sequence of red-blue-green.

System Toolbox, is used to periodically capture audio data from both hydrophones for processing. Following high-pass filtering, the pinger frequency and the time difference of the ping signal received by each hydrophone is calculated.

Only two hydrophones are used on the Endeavour ASV. Ideally, three hydrophones simplifies the localisation problem to that of triangulation, however, it was deemed too difficult to mount three on-board the ASV.

The approach for pinger localisation is based on the Time Difference of Arrival (TDA). Using TDA, the potential location of the pinger resides on hyperboloids. Assuming the pinger is stationary and the ASV is moving, at each ping a new set of hyperboloids is obtained. Due to noise in the TDA estimation, a particle filter is used, weighting each particle by distance to the asymptotic cones formed by the hyperbola. A resampling algorithm is then applied to move low weight particles closer to those which currently possess a higher weight. On convergence, the pinger location can be reliably estimated even with significant noise and as few as 100 particles.

Figure 11 shows an example time evolution of the particles showing convergence to the actual pinger location. The buoy identification component for this task is conducted as described above.

5 Collaboration

The team gratefully acknowledges the generosity of our sponsors and supporters in the development of the Endeavour ASV. These are; the Queensland University of Technology, the AUVSI Foundation, the Defence Science and Technology Organisation (DSTO), GFB Robotics Pty Ltd, RoPro Design Inc., Planet Wrap Pty Ltd, T.M.C. Outboards and Small Engines Pty Ltd, Mathworks, and the Commonwealth Scientific and Industrial Research Organisation (CSIRO).

Those listed above provided financial support, equipment loans, materials and discounted products. Additionally, a critical

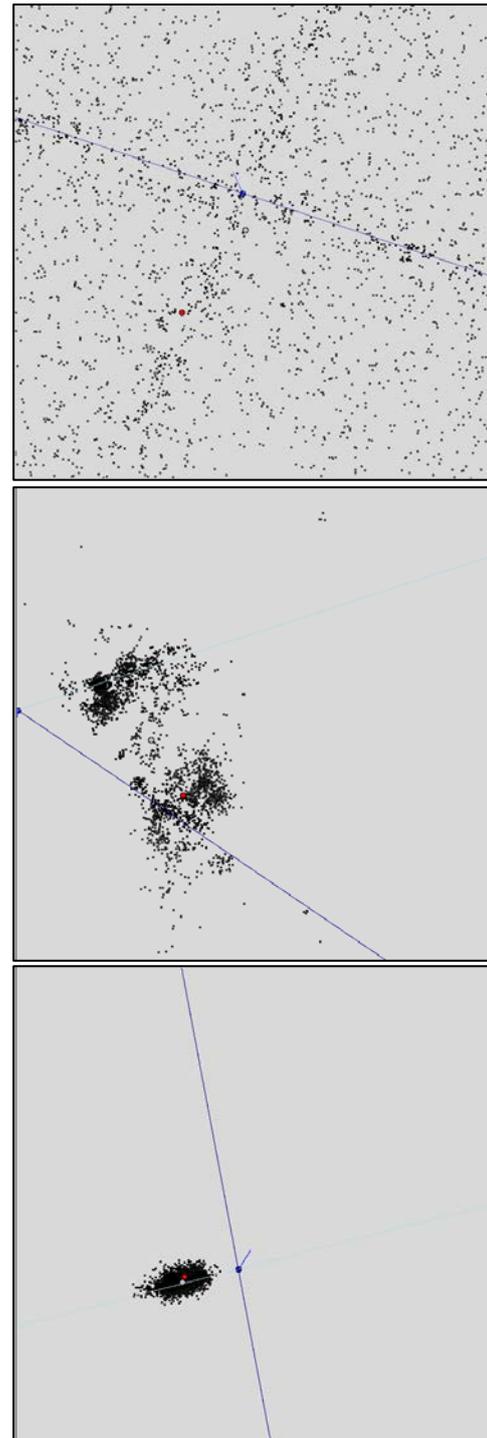


Figure 11: Example showing the convergence of the particle filter for acoustic pinger localisation using TDA. The ASV is represented by the blue dot and line indicating its heading, the red dot shows the actual pinger location and the cyan dot shows the estimated pinger location.

relationship was forged with Seqwater, the operators of the on-water test sites used by the team. By complying with Seqwater's comprehensive Health and Safety policies, the team gained repeated access to these sites. This relationship will facilitate future Research and Development of the ASV.

During development, the undergraduate team members negotiated with, and sought assistance and advice from the wide range of specialists and services offered by the university. These interactions covered areas from marketing and media strategies, to outsourcing of specialized fabrication. Important lessons learnt from this engagement included understanding the function of these specialists, how to use university resources, how to promote mutual benefit to stakeholders, and the role traditional and online media can have on event promotion.

The knowledge gained through the interaction with these collaborators and sponsors has been invaluable and the experience will help guide discussions for gaining support in the future.

6 Conclusion

This paper has presented the hardware, sensor and software systems that provide a complete autonomy solution for the Endeavour Autonomous Surface Vehicle developed by the Queensland University of Technology. The system components have been evaluated through simulation and on-water testing at two different water storages, showing that it is capable of undertaking and completing each of the

2014 Maritime RobotX Challenge tasks. During the ASV's development and associated field testing, a number hardware systems were upgraded to improve reliability and overall capability. Future work will refine and operationalise the auxiliary propulsion system for extended high-speed use as well as upgrade the sensor suite for low-light and night-time operations.

The Endeavour ASV, consisting of the WAM-V platform and the Queensland University of Technology's autonomy solution, has proved reliable during extensive field evaluation. As such, it has already been used for new research efforts into search and rescue operations, and following the Challenge, it will be upgraded with a suite of environmental sensors for monitoring coastal waters in WMO Sea-State 2 to 3 conditions.

References

- [1] Maritime RobotX Challenge: Preliminary Rules & Task Descriptions (Version 2.6, Updated 03 September 2014), accessed 5 October 2014 <<http://robotx.org/index.php/2014-01-05-21-55-32/rules-and-mission-tasks/robotx-preliminary-rules-task-descriptions>>.
- [2] Dunbabin, M., Grinham, A. and Udy, J. (2009). An autonomous surface vehicle for water quality monitoring, In Proc. *Australasian Conference on Robotics & Automation* (ACRA), Sydney, pp. 1-6 (online proceedings).

- [3] Quigley, M., Conley, K. and Gerkey, B.P., Faust, J., Foote, T., Leibs, J., Wheeler, R., and Ng, A.Y. (2009). ROS: an open-source Robot Operating System, In Proc. *ICRA Workshop on Open Source Software*.
- [4] Stentz, A. (1994) Optimal and Efficient Path Planning for Partially-Known Environments, In *Proc. International Conference on Robotics and Automation*, pp. 3310–3317.
- [5] Corke, P.I. (2011). *Robotics, Vision & Control*, Springer 2011, ISBN 978-3-642-20143-1.