

# Maritime RobotX Challenge

Chris Renton<sup>\*</sup>, Alejandro Donaire<sup>\*</sup>, Melanie Jackson<sup>\*</sup>, Nicolas Weightman<sup>\*</sup>, Thomas Hudson<sup>\*</sup>, Richard Ormerod<sup>\*</sup>, Tristan Robinson<sup>\*</sup>, Josiah Walker<sup>\*</sup>, Trent Houlston<sup>\*</sup>, Nathan Carter<sup>\*</sup>, Stephanie O'Donnell<sup>\*</sup>, Nathan Bartlett<sup>\*</sup>, George Wakeham<sup>\*</sup>, Damian Wong<sup>\*</sup>, James Buttenshaw<sup>\*</sup> and Matthew Polo<sup>\*</sup>

<sup>\*</sup> Faculty of Engineering and Built Environment,  
The University of Newcastle, Callaghan, Australia

6<sup>th</sup> October, 2014

## Abstract

The past decade has seen considerable interest and progress in the development of unmanned and autonomous vehicles for civilian applications. Further increases in the degree of autonomy face a number of challenges, both societal and technological, since autonomous vehicles must share operational spaces with human-operated vehicles.

Maritime RobotX is an international competition with fifteen university-student teams competing in a number of autonomous tasks. The competition tasks include basic navigation and control, hydroacoustic localisation, image recognition and docking, optical signal relaying and obstacle avoidance. The provided WAM-V is a standard platform from which a functional USV is developed.

The propulsion system consists of two electric outboard motors that are actively steered using linear actuators. This enables the USV to be fully actuated for manoeuvring in the horizontal plane and simplifies the control task. Three dedicated computers are used

for image processing and machine vision, control and state estimation, and hydroacoustic localisation. Navigation is accomplished using data sources including GPS, IMU and machine vision. Obstacle navigation is accomplished using a 2.5D grid SLAM solution so the USV can build a map of the obstacle field and find its position within that map. The grid map utilised by the guidance system finds a safe path through the obstacle field. This paper outlines the objectives and technical approach taken for the UON RobotX team.

## 1 Introduction

Several factors are increasing the desire for autonomous unmanned marine systems, including increased complexity and duration of maritime operations, as well as the removal of human factors to increase safety and reduce error. Unmanned marine vehicles (UMVs) have already been employed in cartography operations, from mapping and surveying to resource exploration and scientific research

(Roberts and Sutton, 2012). It is foreseen that increasing degrees of autonomy will be implemented in these, and other applications yet to be identified.

In order to foster student interest and inspire future research in autonomous marine systems, the Maritime RobotX Challenge (MRC) has been established. Each student team is provided with a standard unmanned surface vessel (USV), which must be equipped with propulsion units, sensor suites and control systems needed to complete the challenge tasks.

The challenge tasks consist of a variety of maritime surface vessel missions that the USV must complete autonomously. The USV is outfitted with several wide angle cameras with overlapping fields of view in order to detect and classify navigation markers, pontoons, docking symbols, a light buoy and floating obstacles. Hydroacoustic signals are detected by a hydrophone array, from which the source location is inferred. An inertial measurement unit (IMU) provides acceleration and orientation information to augment navigation using GPS. Visual, acoustic, inertial and GPS measurements are used to localise the vessel and to provide goal and obstacle information for the guidance system.

## 2 Technical Approach

The design and development of the USV aligns with the core strengths of the team in the areas of mechanical engineering, mechatronics and computer science. These disciplines provide the necessary framework to design and select mechanical and electrical hardware, and develop computer systems. The fol-

lowing sections outline the hardware architecture and the key facets of the guidance, navigation and control systems, and the mathematical models used therein.

### 2.1 Mechanical

To transform the provided platform into a fully functional USV, several major hardware components are required, such as a propulsion system, batteries, power and control electronics.

The propulsion system includes electric outboard motors and linear steering actuators as shown in Figure 1. The linear actuators are



Figure 1: The linear actuator (foreground) attached to the electric outboard motor. An external encoder is mounted to the top of the outboard motor.

mounted directly to the motors using custom brackets shown in Figure 2. The outboard motors were selected based on the favourable power requirements, monolithic construction and short lead time.

The propulsion system enables each motor to steer independently yielding a fully-actuated system for manoeuvring in the horizontal plane. Industrial grade linear actuators were selected based on stroke length and slew rate requirements.

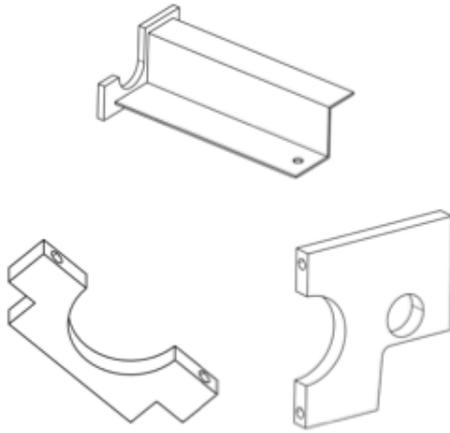


Figure 2: 3D model of linear actuator mounts. Manufactured from aluminium plate.

Figure 3 shows an assembled propulsion unit. This arrangement enables one person to install both units within minutes, by mounting them on a simple aluminium bracket designed to replace the existing hinge tongue plate assembly.



Figure 3: Propulsion unit with electric motor, linear actuator and motor mount.

The two propulsion units add 30kg of extra mass directly astern, causing the bow to pitch upwards due to the change in CoG. This is compensated by mounting the batteries on the USV ski fronts. The suspension system was reinforced to prevent overloading of the ball joints during launch and recovery by crane.

The onboard electronics are housed in IP67 rated enclosures ([National Electrical Manufacturers Association, 2004](#)) with cabling through IP68 rated glands. The cameras are mounted in waterproof enclosures as shown in Figure 4.

## 2.2 Electrical and computer hardware

The main power connections use sealed Anderson connectors<sup>1</sup>. Fuses are installed inside the battery cases to isolate the power supply in case of a connector fault. Each case contains eight 60AH lithium iron phosphate (LiFePO<sub>4</sub>) cells connected in series, rated for 10C burst and 3C continuous.

Figure 5 shows the electrical distribution box that feeds power to all ancillary systems. This

<sup>1</sup><http://www.andersonpower.com/products/multipole-sb.html>

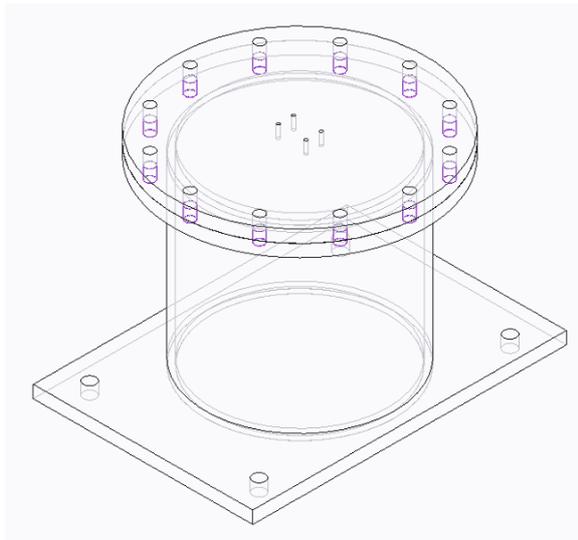


Figure 4: Isometric drawing of the camera housing, to be manufactured in perspex. The inner surface of the cylinder is to be lined with linear polarising film to reduce sun-glare.

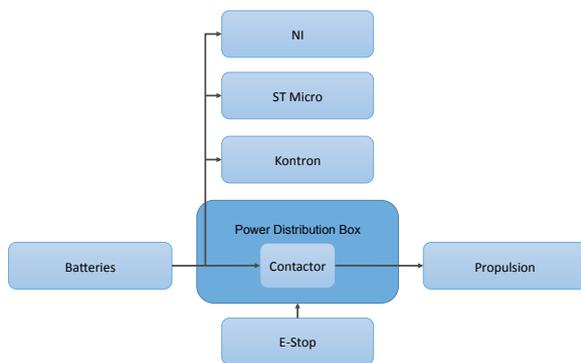


Figure 5: Power distribution system including batteries, ancillary systems, emergency stop and propulsion system.

box also contains the USV emergency stop system; a 400A DC contactor that interrupts power to the propulsion system. A hand-held remote e-stop continuously transmits heartbeat messages to the USV-mounted receiver. Pressing the e-stop button or a loss of signal causes the contactor to disengage, disabling the motors. This complies with the MRC safety requirements (AUVSI, 2014a).

The machine vision system, the embedded control system, and the hydroacoustic localisation system are networked via a man-

aged switch (see Figure 6), which communicates with the ground station via an 802.11g wireless gateway. This complies with the

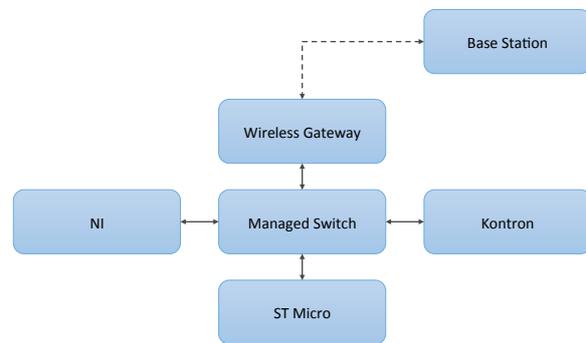


Figure 6: Network topology for the wireless communication system.

MRC communications requirements (AUVSI, 2014e).

The machine vision system has a dedicated computer system that processes images from four cameras arranged as shown in Figure 7. There are wide peripheral cameras and a

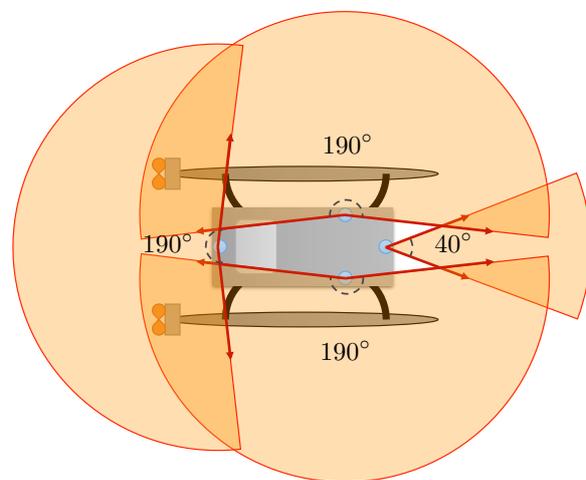


Figure 7: Four cameras (three wide-angle view, one long-focused) are arranged on the USV in a cross-like formation. Light orange arcs represent the field of view for each camera, with darker orange segments depicting field of view crossover.

forward-facing camera that has a long focal-length which improves object recognition directly in front of the vessel. The machine vi-

sion computer has a quad-core Intel processor and 4GB of DDR3 RAM, but only requires 10W.

The STM32F4 Discovery microcontroller was used to perform low-level tasks and interface with sensors and actuators. The controller and state estimator are implemented on the microcontroller.

Since the competition rules forbid the use of RTK GPS, a PPP GPS receiver is used in conjunction with an IMU for primary localisation.

The hydroacoustic localisation system uses a dedicated National Instruments CompactRIO<sup>2</sup> processor module with an FPGA chassis and ADC that is configured as shown in Figure 8. The data from the hydrophone array

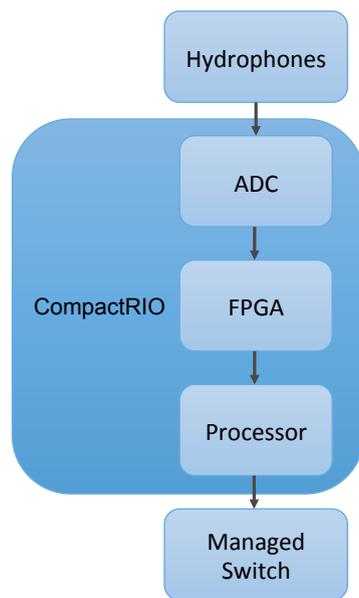


Figure 8: The hydroacoustic localisation system.

is sampled at 500kHz and all signal processing and beacon localisation is performed on the FPGA.

An overview of the sensors, actuators and computer hardware is shown in Figure 9. The

<sup>2</sup><http://www.ni.com/compactrio/>

following sections describe the software components in further detail.

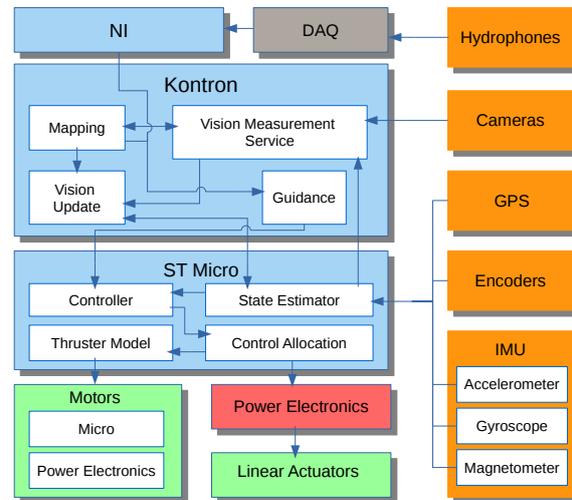


Figure 9: Overview of sensors (orange), actuators (green) and the computer system (blue).

### 2.3 Vehicle kinematics

For the case of marine vehicles, it is convenient to express the rigid-body dynamics in terms of body-fixed momenta or velocities and earth-fixed displacements by using a kinematic transformation (Fossen, 1994).

To describe the motion of the vessel, we consider two reference frames: *Earth-fixed* ( $\mathcal{N}$ -frame) and *body-fixed* ( $\mathcal{B}$ -frame). Associated with these frames are coordinate systems  $\{n\}$  and  $\{b\}$  respectively. The position of the vehicle is given by the relative position of a point of interest  $B$  in  $\mathcal{B}$  with respect to a point of interest  $N$  in  $\mathcal{N}$  (Perez et al., 2012). The components of this vector are North, East and Down positions. The orientation of the vessel is described by the rotation matrix  $\mathbf{R}_b^n \in \text{SO}(3)$ , which is parameterised by the vector<sup>3</sup>

<sup>3</sup> Note that  $\Theta_b^n$  is simply a column vector of parameters which describe the orientation. There exists no physical vector,  $\vec{\Theta}$ , associated with  $\Theta_b^n$ .

of Euler angles  $\Theta_b^n = [\phi, \theta, \psi]^T \in \mathbb{S} \times \mathbb{S} \times \mathbb{S}$ . These components are the roll, pitch and yaw angles respectively.

The *generalised-position vector* is defined by

$$\boldsymbol{\eta} \triangleq \begin{bmatrix} \mathbf{r}_{B/N}^n \\ \Theta_b^n \end{bmatrix}. \quad (1)$$

The velocities are conveniently expressed in terms of body-fixed coordinates and denoted by the *body-fixed velocity vector* defined by

$$\boldsymbol{\nu} \triangleq \begin{bmatrix} \mathbf{v}_{B/N}^b \\ \boldsymbol{\omega}_{B/N}^b \end{bmatrix}. \quad (2)$$

The linear-velocity vector  $\mathbf{v}_{B/N}^b = \mathcal{N} \dot{\mathbf{r}}_{B/N}^b = [u, v, w]^T$  is the time derivative of the position vector as seen by an observer in the frame  $\mathcal{N}$  with components expressed in  $\{b\}$ . These components are the surge, sway, and heave velocities respectively. The vector  $\boldsymbol{\omega}_{B/N}^b = [p, q, r]^T$  is the angular velocity of the body with respect to the  $\mathcal{N}$  frame with components expressed in  $\{b\}$ . These components are the roll, pitch, and yaw rates respectively.

The time derivative of the generalised position vector  $\dot{\boldsymbol{\eta}}$  is related to the body-fixed velocities  $\boldsymbol{\nu}$  according to the following kinematic differential equation:

$$\dot{\boldsymbol{\eta}} = \mathbf{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \quad (3)$$

where

$$\mathbf{J}(\boldsymbol{\eta}) = \begin{bmatrix} \mathbf{R}_b^n(\Theta_b^n) & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_b^n(\Theta_b^n) \end{bmatrix} \quad (4)$$

and

$$\mathbf{R}_b^n(\Theta_b^n) = e^{\psi \mathbf{S}(\mathbf{e}_3)} e^{\theta \mathbf{S}(\mathbf{e}_2)} e^{\phi \mathbf{S}(\mathbf{e}_1)}, \quad (5)$$

$$\mathbf{T}_b^n(\Theta_b^n) = \begin{bmatrix} \mathbf{e}_1, & e^{-\phi \mathbf{S}(\mathbf{e}_1)} \mathbf{e}_2, \\ & e^{-\phi \mathbf{S}(\mathbf{e}_1)} e^{-\theta \mathbf{S}(\mathbf{e}_2)} \mathbf{e}_3 \end{bmatrix}^{-1}, \quad (6)$$

where  $\mathbf{e}_i$  is the unit vector in the  $i^{\text{th}}$  coordinate direction,  $e^{(\cdot)}: \mathfrak{so}(3) \rightarrow \text{SO}(3)$  is the matrix exponential and  $\mathbf{S}: \mathbb{R}^3 \rightarrow \mathfrak{so}(3)$  is the skew operator<sup>4</sup>. These kinematic transformations are well defined for  $\theta \in (-\pi, \pi] \setminus \pm \frac{\pi}{2}$ , i.e., whenever the vessel is not at the extreme pitch angles.

## 2.4 Control system

A tracking controller is designed to follow a desired trajectory  $\mathbf{q}_d(t)$  generated by the guidance system. The guidance system produces desired displacement, velocity and acceleration references. The tracking control law is given by

$$\mathbf{u} = -\mathbf{B}^{-1} \mathbf{A} \dot{\mathbf{q}}_d + \mathbf{B}^{-1} \ddot{\mathbf{q}}_d - \mathbf{K}_1(\mathbf{q} - \mathbf{q}_d) - \mathbf{K}_2(\dot{\mathbf{q}} - \dot{\mathbf{q}}_d), \quad (7)$$

where  $\mathbf{A}$  and  $\mathbf{B}$  are invertible matrices derived from the dynamic model of the vessel, and  $\mathbf{q}$ ,  $\dot{\mathbf{q}}$ ,  $\mathbf{q}_d$ ,  $\dot{\mathbf{q}}_d$  and  $\ddot{\mathbf{q}}_d$  are column vectors containing the states  $x$ ,  $y$  and  $\psi$  and their derivatives. The controller gain matrices  $\mathbf{K}_1$  and  $\mathbf{K}_2$  are designed using LQR.

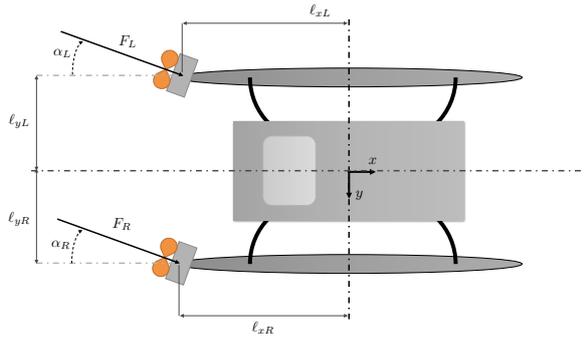


Figure 10: Steerable thrusters enable full actuation for manoeuvring in the horizontal plane. The thrusters produce forces  $F_L$  and  $F_R$  and are actively steered using a linear actuator to produce angles  $\alpha_L$  and  $\alpha_R$ .

### 2.4.1 Control allocation

The generalised control forces on the vessel are given by

$$\boldsymbol{\tau} = \mathbf{B}(\boldsymbol{\alpha})\mathbf{f}, \quad (8)$$

where the vector of steering angles  $\boldsymbol{\alpha}$  and thruster forces  $\mathbf{f}$  is given by

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_L \\ \alpha_R \end{bmatrix}, \quad \mathbf{f} = \begin{bmatrix} F_L \\ F_R \end{bmatrix}, \quad (9)$$

and the actuator configuration matrix is given by

$$\mathbf{B}(\boldsymbol{\alpha}) = \begin{bmatrix} \cos \alpha_L & \cos \alpha_R \\ \sin \alpha_L & \sin \alpha_R \\ B_{31}(\alpha_L) & B_{32}(\alpha_R) \end{bmatrix}, \quad (10)$$

where

$$B_{31}(\alpha_L) = -l_{yL} \cos \alpha_L + l_{xL} \sin \alpha_L, \quad (11)$$

$$B_{32}(\alpha_R) = -l_{yR} \cos \alpha_R + l_{xR} \sin \alpha_R. \quad (12)$$

The control allocation problem is posed as the

<sup>4</sup>Also known as the hat map, this operator produces a skew symmetric matrix such that  $\mathbf{S}(\mathbf{u})\mathbf{v} = \mathbf{u} \times \mathbf{v}$  for all  $\mathbf{u}, \mathbf{v} \in \mathbb{R}^3$ .

following constrained optimisation problem, which is adapted from [Johansen et al. \(2004\)](#); [Fossen et al. \(2009\)](#):

$$\min_{\mathbf{f}, \boldsymbol{\alpha}, \mathbf{s}} \left( \mathbf{f}^T \mathbf{P} \mathbf{f} + \mathbf{s}^T \mathbf{M} \mathbf{s} + (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0)^T \boldsymbol{\Omega} (\boldsymbol{\alpha} - \boldsymbol{\alpha}_0) \right) \quad (13a)$$

subject to

$$\mathbf{B}(\boldsymbol{\alpha})\mathbf{f} + \mathbf{s} = \boldsymbol{\tau} \quad (13b)$$

$$\mathbf{f}_{\min} \leq \mathbf{f} \leq \mathbf{f}_{\max} \quad (13c)$$

$$\boldsymbol{\alpha}_{\min} \leq \boldsymbol{\alpha} \leq \boldsymbol{\alpha}_{\max} \quad (13d)$$

$$\Delta \boldsymbol{\alpha}_{\min} \leq \boldsymbol{\alpha} - \boldsymbol{\alpha}_0 \leq \Delta \boldsymbol{\alpha}_{\max} \quad (13e)$$

where  $\mathbf{s}$  is a slack vector introduced to ensure a feasible solution,  $\mathbf{P}$  and  $\boldsymbol{\Omega}$  are positive definite matrices used to penalise thruster and steering effort and  $\mathbf{M}$  is a large positive definite matrix used to ensure that the optimal solution is  $\mathbf{s} \approx \mathbf{0}$ .

Since the equality constraint (13b) is nonlinear, the solution is obtained iteratively by sequential quadratic programming (SQP) ([Nocedal and Wright, 2006](#)). The solution at the  $k^{\text{th}}$  iteration is given by

$$\mathbf{f}_k = \mathbf{f}_{k-1} + \Delta \mathbf{f}_k, \quad (14)$$

$$\boldsymbol{\alpha}_k = \boldsymbol{\alpha}_{k-1} + \Delta \boldsymbol{\alpha}_k, \quad (15)$$

where  $\boldsymbol{\alpha}_0$  and  $\mathbf{f}_0$  are the control allocation solution at the previous time step, and  $\Delta \mathbf{f}_k$  and  $\Delta \boldsymbol{\alpha}_k$  are the solution to the following QP:

$$\min_{\Delta \mathbf{f}_k, \Delta \alpha_k, \mathbf{s}} \left( (\mathbf{f}_{k-1} + \Delta \mathbf{f}_k)^\top \mathbf{P} (\mathbf{f}_{k-1} + \Delta \mathbf{f}_k) + \mathbf{s}^\top \mathbf{M} \mathbf{s} + \left( \sum_{i=1}^{k-1} \Delta \alpha_i + \Delta \alpha_k \right)^\top \Omega \left( \sum_{i=1}^{k-1} \Delta \alpha_i + \Delta \alpha_k \right) \right) \quad (16a)$$

subject to

$$\begin{aligned} & \mathbf{B}(\alpha_{k-1}) \Delta \mathbf{f}_k \\ & + \frac{\partial}{\partial \alpha} \left( \mathbf{B}(\alpha) \mathbf{f}_{k-1} \right) \Big|_{\alpha=\alpha_{k-1}} \Delta \alpha_k + \mathbf{s} \\ & = \boldsymbol{\tau} - \mathbf{B}(\alpha_{k-1}) \mathbf{f}_{k-1} \end{aligned} \quad (16b)$$

$$\mathbf{f}_{\min} - \mathbf{f}_{k-1} \leq \Delta \mathbf{f}_k \leq \mathbf{f}_{\max} - \mathbf{f}_{k-1} \quad (16c)$$

$$\alpha_{\min} - \alpha_{k-1} \leq \Delta \alpha_k \leq \alpha_{\max} - \alpha_{k-1} \quad (16d)$$

$$\Delta \alpha_{\min} - \sum_{i=1}^{k-1} \Delta \alpha_i \leq \Delta \alpha_k \leq \Delta \alpha_{\max} - \sum_{i=1}^{k-1} \Delta \alpha_i \quad (16e)$$

#### 2.4.2 Propeller thrust model

The motors are commanded by sending a reference propeller angular velocity,  $\omega$ , which is regulated by an onboard controller. To compute the thrust produced by each propeller, the following signed quadratic function of propeller axial and angular velocities is used:

$$F = k \left| \frac{\ell}{2\pi} \omega - u_a \right| \left( \frac{\ell}{2\pi} \omega - u_a \right), \quad (17)$$

where  $k$  [kg.m<sup>-1</sup>] is a constant related to the mass of water displaced per unit axial distance travelled,  $\ell$  [m.rev<sup>-1</sup>] is the lead constant that describes the natural axial advance of the pro-

PELLER per revolution,  $\omega$  [rad.s<sup>-1</sup>] is the propeller angular velocity and  $u_a$  [m.s<sup>-1</sup>] is the axial component of the propeller translational velocity.

The constants  $k$  and  $\ell$  are obtained from static thrust and steady-state velocity tests for the full range of positive and negative propeller angular velocities.

## 2.5 Image Processing

### 2.5.1 Camera Geometry Models

To enable the system to work efficiently with any type of camera sensor, a spherical camera geometry model was used to calculate ray projections for image scanning. In this model, all rays are cast from the focal point of the camera outward. The following equations describe conversion to and from radial camera pixel space, with  $\vec{p}$  as the pixel coordinates from the centre of the camera,  $\vec{c}$  as the unit vector camera ray, and  $\vec{d}$  as the angular pixel pitch of the camera:

$$\vec{p}' = \left[ \vec{p}_x \cdot \vec{d}_x \vec{p}_y \cdot \vec{d}_y \right] \quad (18)$$

$$\vec{p}' = \begin{bmatrix} \vec{c}_x \cdot \cos(\vec{c}_z) \\ \vec{c}_y \end{bmatrix} \quad (19)$$

$$\vec{c} = \begin{bmatrix} \vec{p}'_x \cdot \sin(\|\vec{p}'\|) / \|\vec{p}'\| \\ \vec{p}'_y \cdot \sin(\|\vec{p}'\|) / \|\vec{p}'\| \\ \cos(\|\vec{p}'\|) \end{bmatrix} \quad (20)$$

Rays projected in this space need only be transformed by the current IMU state before being converted into pixels to search for areas of interest. This allows the majority of image scanning calculations to be performed at start-up time. By using this technique, it is possible

to maintain high vision frame-rates with modest hardware requirements.

### 2.5.2 Scanning for Objects

When scanning images for relatively large objects only a small number of pixels need to be sampled. An efficient sample scheme is implemented; sampling less than 1% of all pixels to identify objects of interest.

The sample spacing will depend on the distance from the camera ground-plane on which objects of interest lie. In this case, the ground-plane is the water surface. Using the angle of declination of rays from the camera and its height, an intersection with the ground-plane can be calculated. Using the distance acquired from the ground-plane intersection, the object apparent angular height and width are calculated, assuming the ray intersects the lowest visual part of the object. This assumption is conservative, as a ray intersection through any other part of the object implies it would be closer to the camera, and therefore have a larger angular size.

Neighbour rays are then generated using the angular height and width at the calculated point; process repetition produces a sample set guaranteeing intersection with objects of a chosen size. While this process alone will guarantee detection if 100% accurate, a sample rate of half the angular width is used ensuring multiple intersections. Intersected areas are then densely scanned to recover information for the classification stage. Figure 11 and Figure 12 demonstrate the efficient sampling scheme and the dense scan respectively.

### 2.5.3 Image Segmentation

Colour-based classification is used for image segmentation, as buoys and markers have colours distinctive from the water environment. Look-Up Tables (LUTs) are used to define colours of interest for detecting objects. This technique has proven fast and robust in many past robotics challenges (Quinlan et al., 2004). Two significant improvements on previous systems are made to increase robustness of the system and reduce the need for human interaction for training: automatic LUT extension; and finite state machine based post-processing.

As lighting conditions can change, and differences in colour can occur, an algorithm for automatically extending LUTs is used. This creates an adaptive feedback loop, harvesting data about miss-classified pixels from the shape detectors and modifying the LUT in real time to maximise the correctly classified regions within objects. This process allows most object classes to be robustly and autonomously detected using 1--2 hints from a user.

Previous systems define simple rules to filter some noise during colour classification, before handing data on to shape classifiers. This system iteration replaces these rules with a finite state machine built from regular expressions, allowing many types of noise to be removed during pre-processing. Repeating colour patterns can be detected as areas of interest at no extra processing cost using the same regular expressions.

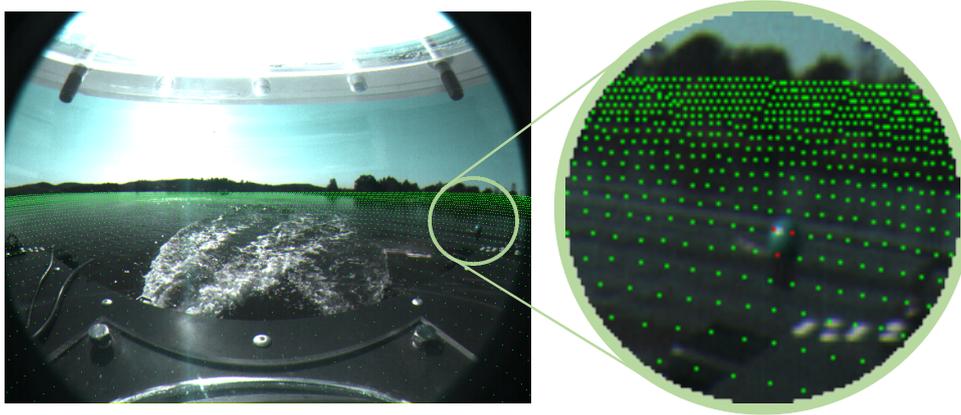


Figure 11: The green dots shown on the left and right images represent unoccupied area. The red dots on the close-up image show where the object-scanner has detected an obstacle in the water.

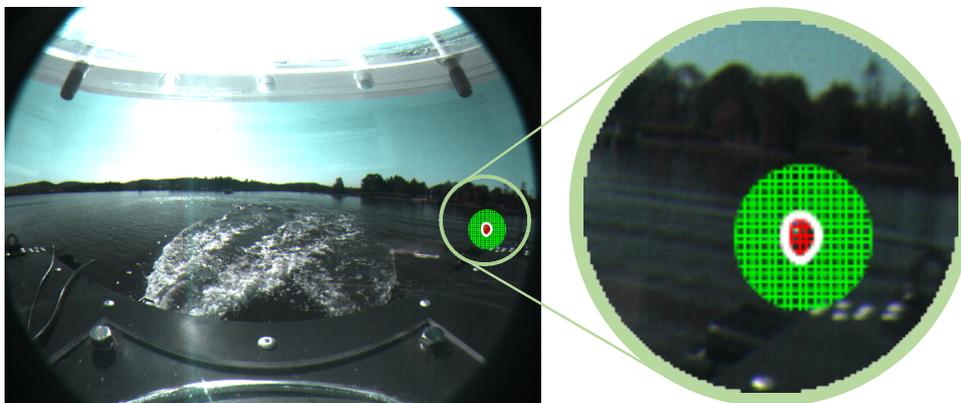


Figure 12: After the coarse scan has identified areas of interest, the fine scan draws segments in the area to find the edges of the object. These edges are then used in RANSAC to fit shapes to the image.

#### 2.5.4 Shape Detection

The random sample consensus (RANSAC) algorithm was developed by [Fischler and Bolles \(1981\)](#) for finding objects in computer vision and associated problems. Algorithm 1 shows a basic application of the RANSAC algorithm for computer vision. This involves generating many possible models using random sets of candidate points, and finding the one with the smallest fitting error to some close subset of candidate points. RANSAC is applied to finding buoys, field markers, and docking pontoons as geometric models. Model generation using RANSAC provides the accurate detection of

the radius of buoys, and the width and height of markers. Extra information generated by RANSAC allows for more accurate localisation of objects.

A significant issue with applying RANSAC to higher order geometry is that the algorithm has a polynomial dependence on the number of model parameters needed. Normally, if there are  $k$  candidate objects represented by several candidate points each, and  $n$  model parameters required, there are on the order of  $k^n$  sets of model parameters to generate and test; only one being correct. A constrained version of RANSAC is used to improve perfor-

mance by utilising the continuous colour regions which identify objects of interest. Using these hints, edges either side of continuous colour segments are used to define a matching pair of points for input into candidate model generation. Constraining model generation using matched segments halves the polynomial order of the RANSAC run-time, and aids in removing false detections. This approach reduces detection complexity for the detection of vertical marker buoys from  $k^4$  to  $k^2$ , allowing far more objects to be processed.

---

**Algorithm 1** RANSAC model fitting algorithm
 

---

**Require:** The candidate point set  $P$

**Require:** The maximum fitting distance,  $d$

**Require:** The geometric *model* we wish to fit, requiring  $n$  sample points to uniquely define

$bestmodel \leftarrow NULL$

$bestfit \leftarrow \infty$

**for**  $i \leftarrow 0$  **to** 100 **do**

$newmodel \leftarrow (p_1, \dots, p_n) \in P$

$newfit \leftarrow 0$

**for**  $p_i \in P$  **do**

$dist \leftarrow newmodel.fit(p_i)$

**if**  $dist \leq d$  **then**

$newfit \leftarrow newfit + dist$

**end if**

**end for**

**if**  $newfit \leq bestfit$  **then**

$bestfit \leftarrow newfit$

$bestmodel \leftarrow newmodel$

**end if**

**end for**

$P' \leftarrow P \setminus \{p \in P : bestmodel.fit(p) \leq d\}$

**return**  $bestmodel, P'$

---

### 2.5.5 Distance Detection

The shape recognition stage of the system produces several pieces of data that when fused, produce a reliable monocular distance

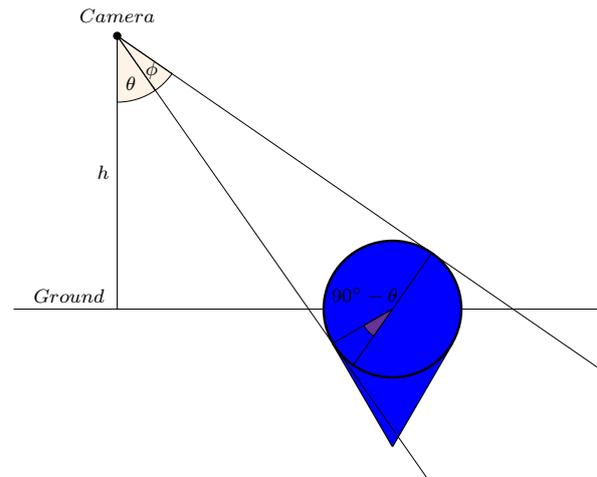


Figure 13: Geometry for estimation of distance of a marker buoy to the camera.

estimate for objects of a limited number of known sizes. The information received from the shape classifier is a camera ray which intersects the base of the object; the angular width and height in radians.

Distances can be calculated using the intersection of the camera ray with the ground plane, the angular width, and the angular height of the object. In the case of vertical markers, this produces three independent measures of distance, while for blue buoys there will be two---angular width and height are equal. Docking pontoons will also generate length based and ground plane intersection estimates.

Fusing these estimates reduces the variance in distance estimation, allowing more accurate mapping. Figure 13 shows the geometry used for estimating the distance of a blue marker buoy by both ground-plane distance and angular height ( $\phi$ ). Note that once ground-plane distance is established, in most cases this will disambiguate the relevant buoy size and allow angular height to generate a single size candidate estimate.

## 2.6 State estimation

Consider the following discrete-time nonlinear state-space model with additive Gaussian noise:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{w}_k, \quad (21a)$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k, \quad (21b)$$

where the current state is  $\mathbf{x}_k$ , the process noise  $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$  and the measurement noise  $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$ . An Unscented Kalman filter (UKF) (Van Der Merwe and Wan, 2001) is used to estimate the state using process and measurement models described in the following sections.

### 2.6.1 Kinematic process model

To account for unknown accelerations that occur due to hydrodynamic forces and disturbances in addition to the control forces, a discrete white noise acceleration process model is used for state estimation. For a single degree of freedom, if  $a_k$  is the constant acceleration during the  $k^{\text{th}}$  sampling period (of length  $T$ ), the increment in velocity during this period is  $Ta_k$ , while the effect of this acceleration on the position is  $\frac{1}{2}Ta_k^2$  (Bar-Shalom et al., 2001). This principle is extended to the Euler discretisation of the rigid-body kinematics (3), which yields the following model driven by body-fixed white noise acceleration:

$$\begin{bmatrix} \boldsymbol{\eta}_{k+1} \\ \boldsymbol{\nu}_{k+1} \\ \mathbf{b}_{k+1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\eta}_k + T\mathbf{J}(\boldsymbol{\eta}_k)\boldsymbol{\nu}_k \\ \boldsymbol{\nu}_k \\ \mathbf{b}_k \end{bmatrix} + \underbrace{\begin{bmatrix} \frac{1}{2}T^2\mathbf{J}(\boldsymbol{\eta}_k) \\ T\mathbf{I}_{6 \times 6} \\ \mathbf{0} \end{bmatrix}}_{\boldsymbol{\Gamma}_a(\boldsymbol{\eta}_k)} \mathbf{a}_k, \quad (22)$$

where  $\mathbf{b}_k$  is a vector of sensor bias and the body-fixed acceleration  $\mathbf{a}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_v)$ ,

where

$$\mathbf{Q}_v = \begin{bmatrix} \mathbf{Q}_{\dot{\mathbf{v}}} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_{\dot{\boldsymbol{\omega}}} \end{bmatrix}, \quad (23)$$

$$\mathbf{Q}_{\dot{\mathbf{v}}} = \text{diag}(\sigma_{\dot{u}}^2, \sigma_{\dot{v}}^2, \sigma_{\dot{w}}^2), \quad (24)$$

$$\mathbf{Q}_{\dot{\boldsymbol{\omega}}} = \text{diag}(\sigma_{\dot{p}}^2, \sigma_{\dot{q}}^2, \sigma_{\dot{r}}^2). \quad (25)$$

The process model parameters  $\sigma_{\dot{u}}$ ,  $\sigma_{\dot{v}}$  and  $\sigma_{\dot{w}}$  have units of translational acceleration, while  $\sigma_{\dot{p}}$ ,  $\sigma_{\dot{q}}$  and  $\sigma_{\dot{r}}$  have units of angular acceleration. Each of these parameters is chosen to be of the order of the maximum acceleration magnitude that the vessel can experience in that degree of freedom.

The model (22) is in the form (21a) with state  $\mathbf{x}_k = [\boldsymbol{\eta}_k, \boldsymbol{\nu}_k, \mathbf{b}_k]^T$  and process noise  $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ , where

$$\begin{aligned} \mathbf{Q}_k &= \mathbb{E} [\boldsymbol{\Gamma}_a(\boldsymbol{\eta}_k) \mathbf{a}_k \mathbf{a}_k^T \boldsymbol{\Gamma}_a^T(\boldsymbol{\eta}_k)] \\ &\approx \boldsymbol{\Gamma}_a(\boldsymbol{\eta}_k) \underbrace{\mathbb{E} [\mathbf{a}_k \mathbf{a}_k^T]}_{\mathbf{Q}_v} \boldsymbol{\Gamma}_a^T(\boldsymbol{\eta}_k), \end{aligned} \quad (26)$$

where the approximation holds assuming that the uncertainty in the orientation states is much smaller than the uncertainty in the input accelerations.

### 2.6.2 Inertial measurement model

The IMU contains a 3-axis accelerometer, a 3-axis MEMS gyroscope and a 3-axis magnetometer. These sensors are collocated at the point  $M$  with the common coordinate system  $\{m\}$ .

The accelerometer measures the combined effect of sensor acceleration and gravity. The sensor acceleration is written in terms of the body acceleration at  $B$  and the body angular velocity, which yields the following measure-

ment model:

$$\mathbf{y}_{\text{acc}} = \mathbf{R}_b^m \left( \mathbf{a}_{B/N}^b + \mathbf{S}(\boldsymbol{\omega}_{B/N}^b) \mathbf{r}_{M/B}^b - (\mathbf{R}_b^n(\boldsymbol{\Theta}_b^n))^T \mathbf{g}^n \right) + \mathbf{v}_{\text{acc}}, \quad (27)$$

where  $\mathbf{a}_{B/N}^b \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{\dot{\mathbf{v}}})$  is the body-fixed translational acceleration input and  $\mathbf{v}_{\text{acc}} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{\text{acc}})$  is the accelerometer sensor noise. This measurement model is in the form (21b) with covariance given by

$$\mathbf{R} = \text{cov}(\mathbf{R}_b^m \mathbf{a}_{B/N}^b + \mathbf{v}_{\text{acc}}) \quad (28)$$

$$\begin{aligned} &= \mathbf{R}_b^m \mathbf{E} \left[ \mathbf{a}_{B/N}^b (\mathbf{a}_{B/N}^b)^T \right] (\mathbf{R}_b^m)^T \\ &\quad + \mathbf{E} \left[ \mathbf{R}_b^m \mathbf{a}_{B/N}^b \mathbf{v}_{\text{acc}}^T \right] + \mathbf{E} \left[ \mathbf{v}_{\text{acc}} (\mathbf{R}_b^m \mathbf{a}_{B/N}^b)^T \right] \\ &\quad + \mathbf{E} \left[ \mathbf{v}_{\text{acc}} \mathbf{v}_{\text{acc}}^T \right] \end{aligned} \quad (29)$$

$$= \mathbf{R}_b^m \mathbf{Q}_{\dot{\mathbf{v}}} (\mathbf{R}_b^m)^T + \mathbf{R}_{\text{acc}}, \quad (30)$$

where the cross terms vanish since the input acceleration and the accelerometer sensor noise are uncorrelated.

The gyroscope measures the body angular velocity up to a constant offset bias,  $\mathbf{b}$ . The measurement model is given by

$$\mathbf{y}_{\text{gyro}} = \mathbf{R}_b^m (\boldsymbol{\omega}_{B/N}^b + \mathbf{b}) + \mathbf{v}_{\text{gyro}}, \quad (31)$$

where  $\mathbf{v}_{\text{gyro}} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{\text{gyro}})$ .

The magnetometer measures the strength of the ambient magnetic field and the local field line direction. The measurement model is given by

$$\mathbf{y}_{\text{mag}} = \mathbf{R}_b^m (\mathbf{R}_b^n(\boldsymbol{\Theta}_b^n))^T \mathbf{m}^n + \mathbf{v}_{\text{mag}}, \quad (32)$$

where  $\mathbf{v}_{\text{mag}} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{\text{mag}})$ .

### 2.6.3 GPS measurement model

The position of the body  $B$  relative to the Earth centre  $O$  may be expressed in the ECEF Euclidean coordinate system  $\{e\}$  with components  $\mathbf{r}_{B/O}^e = [x^e, y^e, z^e]^T$ . The coordinate transformation from  $\{e\}$  to the geodetic ECEF curvilinear coordinate system  $\{g\}$  is achieved using the nonlinear map  $\Phi_e^g: \mathbf{r}_{B/O}^e \mapsto \mathbf{r}_{B/O}^g$ . Expressing the body position in geodetic coordinates has components  $\mathbf{r}_{B/O}^g = [\phi^g, \lambda^g, h^g]^T$ , where  $\phi^g$  is the latitude,  $\lambda^g$  is the longitude and  $h^g$  is the altitude.

The mapping  $\Phi_e^g$  is given in closed form by Hofmann-Wellenhof et al. (2007); Noureldin et al. (2013) as follows:

$$\phi^g = \arctan \frac{z^e + (e')^2 b \sin^3 \theta}{p - e^2 a \cos^3 \theta}, \quad (33)$$

$$\lambda^g = 2 \arctan \frac{y^e}{x^e + \sqrt{(x^e)^2 + (y^e)^2}}, \quad (34)$$

$$h^g = \frac{p}{\cos \phi} - N, \quad (35)$$

where

$$e' = \sqrt{\frac{a^2 - b^2}{b^2}}, \quad (36)$$

$$\theta = \arctan \left( \frac{z^e a}{p b} \right), \quad (37)$$

$$p = \sqrt{(x^e)^2 + (y^e)^2}, \quad (38)$$

$$N = \frac{a^2}{\sqrt{a^2 \cos^2 \phi + b^2 \sin^2 \phi}}, \quad (39)$$

and  $a = 6378137.0\text{m}$  is the semimajor axis,  $b = 6356752.3142\text{m}$  is the semiminor axis and  $e = 0.08181919$  is the eccentricity for the Earth ellipsoid as defined by the WGS-84 standard.

The measurement model is given by

$$\mathbf{y}_{\text{GPS}} = \Phi_e^g(\mathbf{R}_n^e \mathbf{r}_{B/N}^n + \mathbf{r}_{N/O}^e) + \mathbf{v}_{\text{GPS}}, \quad (40)$$

where  $\mathbf{v}_{\text{GPS}} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{\text{GPS}})$  is the measurement noise.

#### 2.6.4 Landmark measurement model

The measurement model for landmark (gate) buoys and docking pontoons is given by

$$\mathbf{y}_L = \frac{\mathbf{r}_{L/C}^c}{\|\mathbf{r}_{L/C}^c\|} + \mathbf{v}_L, \quad (41)$$

where  $\{c\}$  is the coordinate system for the camera with optical centre located at  $C$  and

$$\begin{aligned} \mathbf{r}_{L/C}^c = \\ \mathbf{R}_b^c \left( (\mathbf{R}_b^n(\Theta_b^n))^T (\mathbf{r}_{L/N}^n - \mathbf{r}_{B/N}^n) - \mathbf{r}_{C/B}^b \right), \end{aligned} \quad (42)$$

and  $\mathbf{v}_L \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_L)$ .

Obstacle mapping is described in the following section and the measurement model for obstacle localisation is introduced in Section 2.7.2.

## 2.7 Obstacle mapping and localisation

For navigation in the presence of obstacles, an occupancy grid is used to represent the obstacle field. This was chosen since the preliminary competition rules (AUVSI, 2014b) specified that moored obstacles will be floating on the surface, visible, and of various shapes and colours<sup>5</sup>. A grid representation is able to readily encode such obstacles. The map occupancy state is used in the path planner in order to reach a desired goal location while avoiding obstacles.

<sup>5</sup> Later rules (AUVSI, 2014c) specified that all obstacles are spherical and of the same colour.

The obstacles also provide navigation cues for localisation, which aids navigation when one or more gate buoys are not observed by any camera. Simultaneous localisation and mapping (SLAM) allows the vessel to build a map of the environment while determining its position within that map (Thrun et al., 2005). A 2D grid SLAM implementation is adapted to suit the information available from the vision system. The process of obstacle mapping and the measurement model required for the localisation are described in the following sections.

### 2.7.1 Map update

Let  $P_k$  be the centre of a detected obstacle, and  $Q_{ik}$  be the point on a unit sphere centred at  $C_i$  such that  $\vec{r}_{Q_{ik}/C_i} = \frac{\vec{r}_{P_k/C_i}}{\|\vec{r}_{P_k/C_i}\|}$ . The vision system provides the unit direction vectors and apparent angular sizes of the obstacles observed in  $i^{\text{th}}$  camera image as the set of pairs  $\{\mathbf{r}_{Q_{ik}/C_i}^c, \theta_{ik}\}$  for  $k = 1, 2, \dots$

The position of each obstacle centre,  $\mathbf{r}_{P/N}^n$ , is given by

$$\mathbf{r}_{P_k/N}^n = \mathbf{R}_c^n \left( \underbrace{\frac{r_k}{\sin \theta_{ik}} \cdot \mathbf{r}_{Q/C}^c}_{\mathbf{r}_{P_k/C}^c} \right) + \mathbf{r}_{C/N}^n, \quad (43)$$

where  $r_k$  is the radius of the  $k^{\text{th}}$  obstacle.

The change in occupancy probability of a grid cell centred at  $G$  is positive if

$$\|\mathbf{r}_{P'_k/N}^n - \mathbf{r}_{G/N}^n\| < r_k \quad \text{for any } k, \quad (44)$$

where

$$\mathbf{r}_{P'_k/N}^n = \text{diag}(1, 1, 0) \mathbf{r}_{P_k/N}^n \quad (45)$$

and  $P'_k$  is the projection of  $P_k$  onto the water

surface. This change in occupancy probability is repeated for each camera that observes the obstacle.

If the condition (44) is false, two cases remain. If the grid centre  $G$  is not occupied according to (44) and satisfies

$$\mathbf{r}_{Q_{ik}/C_i}^c \cdot \frac{\mathbf{r}_{G/C_i}^c}{\|\mathbf{r}_{G/C_i}^c\|} > \cos \theta_{ik}, \quad (46)$$

then the change in occupancy probability is negative for the  $i^{\text{th}}$  camera since it observes the water surface. If neither conditions (44) or (46) are satisfied, no change in occupancy probability for  $G$  is performed from the  $i^{\text{th}}$  camera, since  $G$  is occluded by an obstacle. The three cases are shown in Figure 14.

### 2.7.2 Obstacle measurement model

Since the cameras observe the obstacle field from an elevated position, depth cues may be inferred from the angle of declination to the obstacle on the water surface. This enables a beam-based measurement model to be used with a 2D rather than a 3D occupancy grid. While the beam model casts rays to the nearest obstacle in 2D, the map is updated based on all available 3D information, since the cameras can see over the top of small obstacles.

The emulated beam model for the  $i^{\text{th}}$  camera employs a measurement vector. The obstacle measurement model for the  $i^{\text{th}}$  camera is given by

$$\mathbf{y}_{C_i} = \begin{bmatrix} \rho_{H_{i1}/C_i} \\ \rho_{H_{i2}/C_i} \\ \rho_{H_{i3}/C_i} \\ \vdots \end{bmatrix} + \mathbf{v}_{C_i}, \quad (47)$$

where  $\mathbf{v}_{C_i} \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_{C_i})$  is the measurement noise and  $\rho_{H_{ij}/C_i} \triangleq \|\vec{r}_{H_{ij}/C_i}\|^{-1}$  is the inverse depth from the camera optical centre  $C_i$  to the  $j^{\text{th}}$  ray termination point  $H_{ij}$  in the horizontal plane. The ray angles span the horizontal field of view of each camera.

The use of inverse depth parameters is well established in the monocular visual SLAM literature (Montiel et al., 2006; Civera et al., 2008; Sola et al., 2012), but does not receive as much attention in grid SLAM applications. By using a Gaussian distribution to represent uncertainty in inverse depth parameters (IDPs), rays may effectively terminate at infinity while the IDPs remain bounded (Marzorati et al., 2009).

The obstacle measurement vector is computed from a 2D parametric map that consists of obstacle centres  $P'_k$  and radii  $r_k$  for all  $k$  as shown in Figure 15. This map is constructed using (43), (45) and the measured  $\{\mathbf{r}_{Q/C}^c, \theta\}$  data that the vision system provides. The measurement prediction vector is computed from the 2D occupancy grid as shown in Figure 16.

## 2.8 Guidance

The guidance system consists of two phases. First, obstacles are grouped if necessary, then the path planning algorithm is executed.

### 2.8.1 Obstacle cell clustering

Convex hull clustering is used to ensure the boat does not attempt to pass through a gap between obstacles that is too small for the boat to pass safely. If the distance between

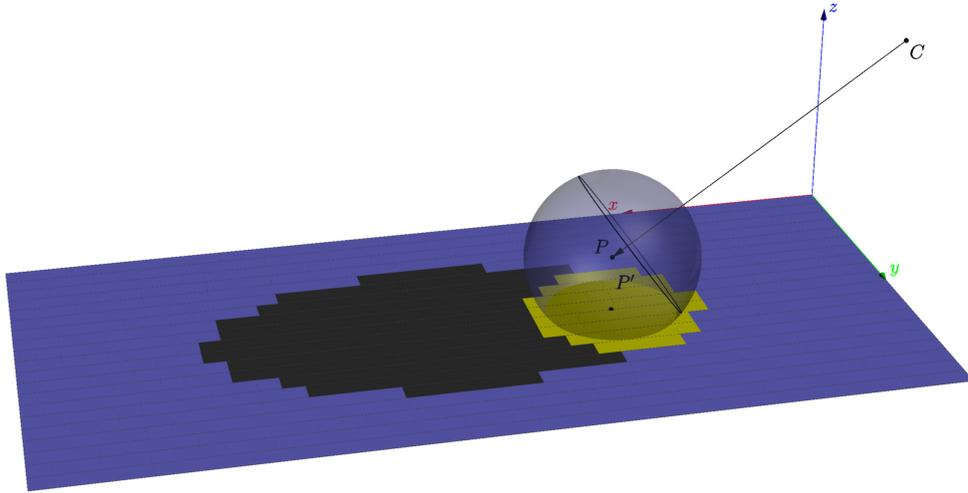


Figure 14: Spherical obstacle of radius  $r$  with centre at  $P$  observed from a perspective camera at  $C$ . The map grid cells unobserved by the camera (dark grey) are bounded by a conic section, while an orthographic projection from above (yellow) represents the occupancy state used to map the obstacle.

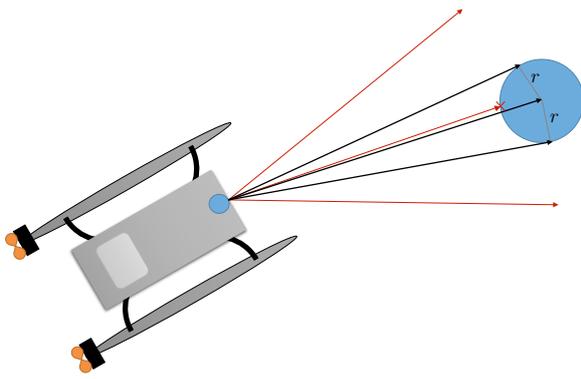


Figure 15: Virtual beam measurements are obtained by casting rays from each camera until they terminate at a circle centred at  $P'$  with radius  $r$  at some inverse depth  $\rho_{H/C} > 0$ . If the ray does not intersect any circles, then  $\rho_{H/C} = 0$ . The obstacle position  $P'$  is determined using (43), (45) and the  $\{r_{Q/C}^c, \theta\}$  pairs obtained from the vision system.

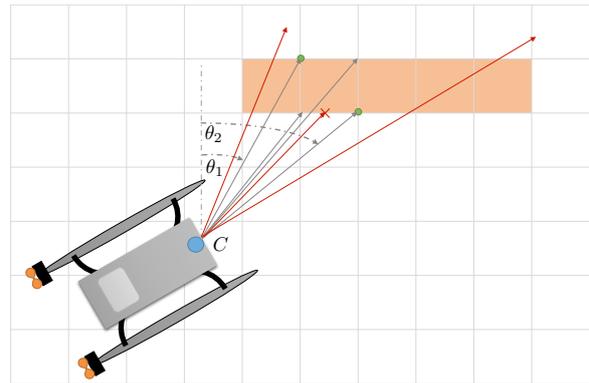


Figure 16: Beam measurements are predicted by casting rays from each camera until they terminate at the boundary of an occupied grid cell at some inverse depth  $\rho_{H/C} > 0$ . If the ray does not intersect any occupied cells, then  $\rho_{H/C} = 0$ .

### 2.8.2 Path Planning

any two occupied cells on the grid map is less than a threshold corresponding to the maximum dimension of the boat, the convex hull algorithm will group those cells together as a single cluster. These clusters are then treated as large obstacles by the guidance algorithm, as shown in Figure 17.

Discrete grid-based path planning solutions offer a number of advantages in terms of simplicity and performance over their continuous counterparts. The  $A^*$  and  $D^*$  family of path planning algorithms are chosen for the guidance of the USV due to the robust predictability, given accurate cost models.

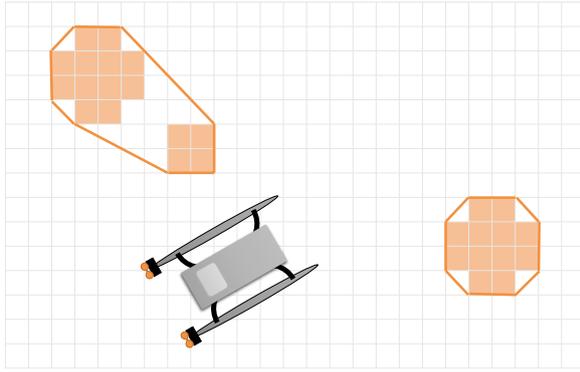


Figure 17: Occupied grid cells (light orange) are clustered if the distance between cells is less than the vessel maximum dimension. The path planner uses the convex hull of each cluster (dark orange) as an obstacle to avoid.

The  $D^*$  algorithm (Koenig et al., 2004) is a dynamically updated version of the  $A^*$  exhaustive path-finding algorithm. Field  $D^*$  improves upon the original by adding continuous path interpolation to produce a path quality similar to continuous planners, while maintaining high efficiency (Ferguson and Stentz, 2005).

A simpler approximation of Field  $D^*$  is implemented by taking the original  $D^*$  algorithm and allowing transitions to non-immediate neighbours, summing the fractional cost of transition through intermediate cells. The maximum radius of these allowed transitions represents the quality of approximation to Field  $D^*$ , but comes at an increased performance cost.

## 2.9 Hydroacoustic localisation

One method to solve the acoustic source localisation problem is the Time Delay Of Arrival (TDOA) method, which uses an array of microphone pairs and simple geometry to locate the source as described by Mumolo et al. (2003). Combining this method with another reduces

design complexity, and required microphones resources---usually  $\geq 6$ .

A reduced TDOA method using an S-array is used. This will introduce errors to the prediction but uses only 4 hydrophones (HP) for 3D localisation. This design is shown in Figure 18, where the horizontal pair will be used to approximate X-Y position and the offset will get an angle to approximate the Z position (depth).

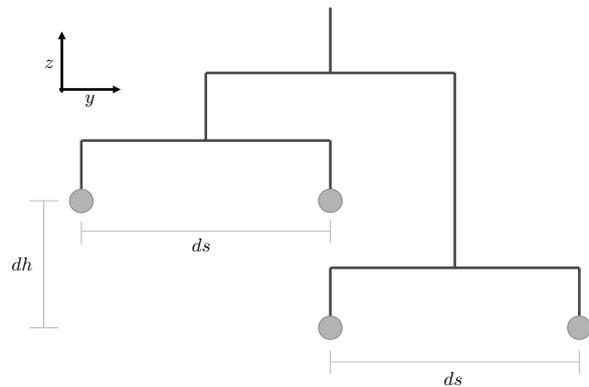


Figure 18: Hydroacoustic microphones are mounted in offset pairs to maximise the information obtained from measurements.

This array also allows the testing of the newly developed algorithm which uses TDOA information as well as Difference In Signal Amplitude (DISA). An acoustic signal will lose its strength as a wave propagates from its source in either 2D or 3D shells, it will further be diminished by dissipation (Lurton, 2002). The following equation describes the geometric spreading of a wave:

$$A_1 = \frac{A_0}{R_1^n}, \quad (48)$$

where  $A_1$  is the signal strength,  $A_0$  is the original strength,  $R_1$  is the distance from the source to the hydrophone and  $n$  can be either 1 or 2 for planar and spherical spreading.

To determine the source location, the length

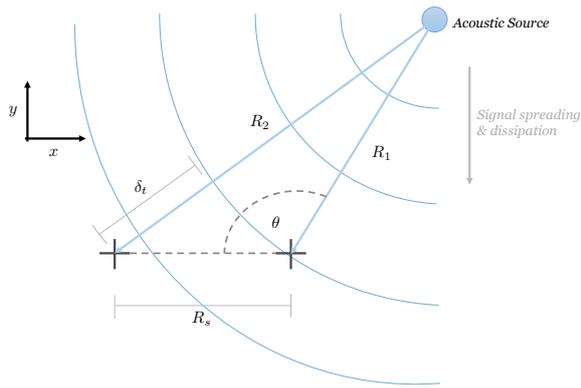


Figure 19: The angle from a hydrophone pair to an acoustic source may be determined from the time difference of the approaching wave front. The reflection ambiguity is resolved by introducing an additional microphone that is not collinear with the existing pair.

of the rays  $R_1$  and  $R_2$ , shown in Figure 19, need to be found. The following equation calculates the distance from the source to each HP:

$$R_1 = \frac{\delta_t \sqrt{\frac{A_2}{A_1}}}{1 - \sqrt{\frac{A_2}{A_1}}} \quad (49)$$

Knowing all distances, the cosine rule is used to determine the angle from the source to each HP as opposed to each HP pair, as used in the TDOA method. Position predictions can then be made. This method requires only 3 HPs for 3D localisation as opposed to 8 for a standard TDOA system.

### 3 Collaboration

The UON RobotX Team have developed their USV in collaboration with industry, government organisations, the community and between groups within the university.

Sponsorship was obtained primarily by approaching organisations directly. Support has been gained through in-kind donations of

hardware from industry and financial backing from the competition sponsors and organisers, the university and government initiatives.

Within the university, the Newcastle Robotics Laboratory (NRL) supports students participating in two international competitions--- Maritime RobotX Challenge (UON RobotX) and RoboCup (NUBot). NRL is an interdisciplinary research and training initiative of several robotics related research teams.

Kontron have provided in-kind donations of state-of-the-art embedded computers, for both competitions, which are running computationally expensive machine vision software. Hetricon Australia donated a wireless e-stop system, Altra Industrial Motion have contributed linear actuators for the USV steering system and Regional Development Australia (RDA) Hunter have provided funding for team member travel. The Australian Government, Defence Science and Technology Organisation (DSTO) have also provided financial support.

## 4 Conclusion

Student engagement in the science and engineering of autonomy has been paramount in developing the USV for the MRC. The machine vision system exceeds the requirements of the obstacle avoidance task. The system was designed to recognise obstacles of various shapes, colours and sizes as per [AUVSI \(2014b\)](#), however, this was later simplified in [AUVSI \(2014c\)](#) to three defined size buoys of consistent shape and colour. The broader approach may benefit subsequent challenges or future research opportunities.

Unfortunately, due to sponsorship withdrawal, the hydroacoustic search and localisation task was not being attempted. Since scoring guidelines (AUVSI, 2014d) were released in September, this task is being reinvestigated due to the large point allocation.

One of many other challenges that the UON team have faced include propulsion system development. Progress was slow until the propulsion system funding was received towards the end of June. The University of Newcastle kindly provided emergency funding to the team in order to procure this equipment. Despite these delays the team continued to write software components of the autonomous system.

The development of these and other critical systems have transformed the USV into an autonomous vessel with capabilities extending beyond the MRC. UON will continue to employ the USV in further research; using the boat for project tasks and further iterations of the competition. The USV is well equipped and has hub capabilities for multi-vehicle cooperation including vehicles in multiple domains (aerial, surface and underwater). Continued collaboration with external organisations helps facilitate research towards greater autonomy in marine systems. The team is looking forward to future research opportunities after the competition.

Ongoing research contributes towards producing safe and robust unmanned and autonomous vehicles. To be accepted by society at large, an unprecedented level of safety and reliability must be demonstrated before these vehicles can be operated in spaces shared by human-operated vehicles (Perez et al., 2013).

## A Hardware

Selected mechanical, electrical and computer hardware is listed in Table 1.

## References

- AUVSI, May 2014a. Maritime RobotX Challenge Kill Switch (Emergency Stop) Specification, v1.0. online.
- AUVSI, 2014b. Maritime RobotX Challenge Preliminary Rules and Task Descriptions (21st March 2014). Tech. rep., AUVSI Foundation.
- AUVSI, 2014c. Maritime RobotX Challenge Preliminary Rules and Task Descriptions (24th July 2014). Tech. rep., AUVSI Foundation.
- AUVSI, 2014d. Maritime RobotX Challenge Preliminary Scoring Guidance, v2.2 (7th September 2014). Tech. rep., AUVSI Foundation.
- AUVSI, Jul 2014e. Maritime RobotX Challenge Wireless Communications Protocol v4.0. online.
- Bar-Shalom, Y., Li, X. R., Kirubarajan, T., 2001. Estimation with applications to tracking and navigation: theory algorithms and software. John Wiley & Sons.
- Civera, J., Davison, A. J., Montiel, J., 2008. Inverse depth parametrization for monocular SLAM. *Robotics, IEEE Transactions on* 24 (5), 932--945.
- Ferguson, D., Stentz, A., June 2005. The field D\* algorithm for improved path planning and replanning in uniform and non-uniform

Table 1: Selected hardware components.

System	Part	Manufacturer	Model
Propulsion	Electric Outboard motor	Torqueedo	Cruise 2.0RL
	Linear actuators	Warner Linear	K2XP2.0G05-24V-12
Power	Battery Cells	Winston-Battery	WB-LYP60AHA
	Fuses	Cooper-Bussmann	CB185-150
	Contactors	Nanfeng	ZJW400A
ST Micro	Microcontroller board	ST Micro	STM32FDISCOVERY
	GSP receiver	U-Blox	EVK-6ppp
	IMU	ACME Systems	DAISY-7
Kontron	Computer on Module	Kontron	COMe-mBTc10-J1900-4GB
	Development Board	Kontron	COM Express Eval Type 10
NI	Processor Module	National Instruments	CRIO-9124
	Chassis w/ inbuilt FPGA	National Instruments	CRIO-9113
	ADC capture card	National Instruments	NI-9222
Networking	Wireless gateway	TP-Link	TL-WA7210N/TL-ANT2415D
	Switch	Netgear	GS105e

cost environments. Tech. Rep. CMU-RI-TR-05-19, Robotics Institute, Pittsburgh, PA.

Fischler, M. A., Bolles, R. C., Jun. 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM* 24 (6), 381--395.

Fossen, T., 1994. *Guidance and control of ocean vehicles*. John Wiley and Sons Ltd, New York.

Fossen, T. I., Johansen, T. A., Perez, T., 2009. A survey of control allocation methods for underwater vehicles. *Underwater vehicles*, 109--128.

Hofmann-Wellenhof, B., Lichtenegger, H., Wasle, E., 2007. *GNSS--global navigation satellite systems: GPS, GLONASS, Galileo, and more*. Springer.

Johansen, T. A., Fossen, T. I., Berge, S. P., 2004. Constrained nonlinear control allocation with singularity avoidance using se-

quential quadratic programming. *Control Systems Technology*, IEEE Transactions on 12 (1), 211--216.

Koenig, S., Likhachev, M., Furcy, D., 2004. Lifelong planning A\*. *Artificial Intelligence* 155 (1-2), 93--146.

Lurton, X., 2002. *An introduction to underwater acoustics: principles and applications*. Springer.

Marzorati, D., Matteucci, M., Migliore, D., Sorrenti, D. G., 2009. On the use of inverse scaling in monocular SLAM. In: *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*. IEEE, pp. 2030--2036.

Montiel, J., Civera, J., Davison, A., 2006. Unified inverse depth parametrization for monocular SLAM. *analysis* 9, 1.

Mumolo, E., Nolich, M., Vercelli, G., 2003. Algorithms for acoustic localization based on microphone array in service robotics.

- Robotics and Autonomous systems 42 (2), 69--88.
- National Electrical Manufacturers Association, 2004. ANSI/IEC 60529-2004 - Degrees of Protection Provided by Enclosures. Tech. rep., Approved American National Standard.
- Nocedal, J., Wright, S. J., 2006. Numerical optimization.
- Noureldin, A., Karamat, T. B., Georgy, J., 2013. Fundamentals of inertial navigation, satellite-based positioning and their integration. Springer.
- Perez, T., Donaire, A., Renton, C., 2012. Port-Hamiltonian control of fully-actuated underwater vehicles. In: Roberts, G., Sutton, R. (Eds.), Further advances in unmanned marine vehicles. IET.
- Perez, T., Williams, B., Clothier, R., 2013. Robust autonomy of intelligent vehicles. In: Intelligent Autonomous Vehicles. Vol. 8.
- Quinlan, M. J., Chalup, S. K., Middleton, R. H., 2004. Application of SVMs for colour classification and collision detection with AIBO robots. In: Advances of Neural Information Processing Systems (NIPS'2003). Vol. 16. The MIT Press, Cambridge, MA, pp. 635--642.
- Roberts, G. N., Sutton, R., 2012. Further Advances in Unmanned Marine Vehicles. Vol. 77. Inst of Engineering & Technology.
- Sola, J., Vidal-Calleja, T., Civera, J., Montiel, J. M. M., 2012. Impact of landmark parametrization on monocular EKF-SLAM with points and lines. International journal of computer vision 97 (3), 339--368.
- Thrun, S., Burgard, W., Fox, D., 2005. Probabilistic robotics. MIT press.
- Van Der Merwe, R., Wan, E. A., 2001. The square-root unscented kalman filter for state and parameter-estimation. In: Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on. Vol. 6. IEEE, pp. 3461--3464.