

SNAME103 Autonomous Robotic Boat 2014 Competition

Advisor: Professor Yung-Yu Chen

I-Pin Cheng, Wei-Ting Chen
Yu-Ching Lin, Pei-Yu Chang

*Department of System and Naval Mechatronics Engineering,
National Cheng Kung University, Taiwan*

1. Abstract

The SNAME 103, designed by NCKU Roboboat Team, is on the purpose of achievement of AUVSI Foundation and ONR's 7th International Roboboat Competition and it is the 5th generation from the prototype SNAME 99.

For enhancing the overall performance of this new version Roboboat, we redesign the boat via using pontoon and enlarging the deck area. Based on this modification, the capabilities of straight sailing and maneuverability are highly improved. In the programming of the functions of Roboboat for achieving competition, the famous software: LabView is used to calculate algorithms of the new missions of ONR's 7th International Roboboat Competition.



Electronic systems



GPS



Radar



Webcam



Hydrophone

In this investigation, we separate Roboboat into two big parts. One of the Roboboat is relative the mechanical and electronic systems, the other part is about missions we have to pass through. Secondly, four electrical devices: GPS, radar, webcam, hydrophone, are utilized for obstacle detecting and navigating purpose. GPS is the most important part of the guidance system of our development for navigating our designed

Roboboat to waypoints, radar which is used for obstacle avoidance, identification function of symbols is developed by using on shelf webcam for saving total cost purpose, hydrophone here is used for the mission of acoustic beacon positioning, and as to the underwater light identification, the same webcam is used but is with different identification algorithm.

2. Mechanical and Electronic Systems

2.1 Ship design

Last year, a multihull trimaran is designed for the competition, although that design could bring some advantages, we finally felt the package still too large to bring on board. Our team referred to other teams which used pontoons to provide buoyancy effectively from the previous competition, hence Hobie's the SideKick Ama Kit is adopted in our new generation Roboboat. The SideKick Ama Kit is usually used to provide additional buoyancy for kayak, one SideKick Ama Kit can support a weight which is almost ten kilograms, and it's convenient to carry. The estimated total weight of our equipments is about forty kilograms, so at least four SideKick Ama Kits must be utilized to support the deck.

To light the weight of the Roboboat, a foldable table made of aluminum alloy is used to set up deck of Roboboat. By this design, the weight of deck is only 2.75 kg, and we can bring it easily. The most difficult part is to link deck to the SideKick Ama Kits. For solving this problem, we adjust the poles of the SideKick Ama Kit, and drilled holes to fix the deck.

2.2 Electronic System

GIGABYTE GA-H77M-D3H main board with INTEL CORE i5 3470 CPU is utilized to be the major control unit for this boat robot and the usage of TRANSCEND JM1333KLN-4G memory, is to increase the responses of feedback signals that includes as: GPS, Camera and Compass, where communications between NEMA of GPS and heading angle of the used Compass and the controlled main board are translated by UART Protocol to USB.

In this investigation, control commands for guiding the boat robot are performed by the following steps: 1. Heading angle calculated by software LabVIEW first and then, 2. The feedback linearization based control in (16) is discretized and translated with UART Protocol to Micro controller: Arduino-Mega2560 and a series of PWM signals drives DC motors to generate torques to rotate propellers. Besides, micro controller: Arduino- Mega2560 is also adopted to control apparatuses in the boat robot as the servo motor of robot arm, and magnetic switch of shunting machine, and so on.

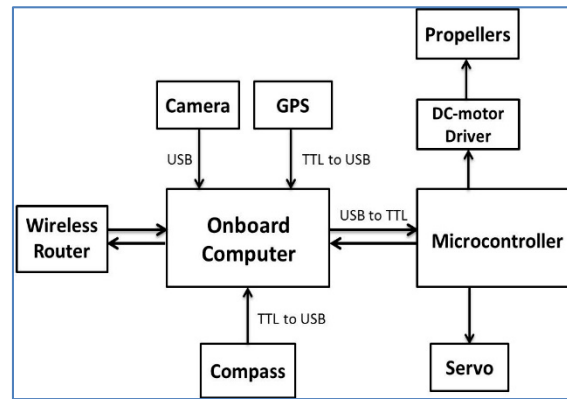


Fig.2.1 Electrical System

2.3 Power System

For the purpose of saving the space and increasing the usage duration, a lithium ion battery with type number: AMP20 Lithium Ion Prismatic Cell can steadily offer a nominal voltage: 3.3V and nominal capacity: 20Ah is taken into consideration based on the excellent advantages: high power and flimsy proof. By serial connecting 4 pieces of the above battery, a total voltage 13.2 can be realized to supply the boat robot with a full payload for running over one working hour.

Adjustable voltage regulators that are on the shelf and cheap generally are utilized to offer stable voltages for the main board with 12V and wireless router with 9V, and the advantage of this design is: adjustable voltage regulators can eliminate the sudden high voltage peaks or a huge of decrease in supplied voltage and avoid the preventable shut-down of the power system of boat robot as operation modes of propellers suddenly changes which always needs large power immediately to achieve the mission, and this behavior will result in a shut-down of the overall boat robot system in practical design. This power management module overcomes those two problems and steadily and continuously offers voltages to the whole system.

As to the thruster control design, a DC motor driver which can be precisely controlled to be a switch or average the input voltage with PWM Pulse is selected for the purpose of rotating propeller counterclockwise or clockwise.

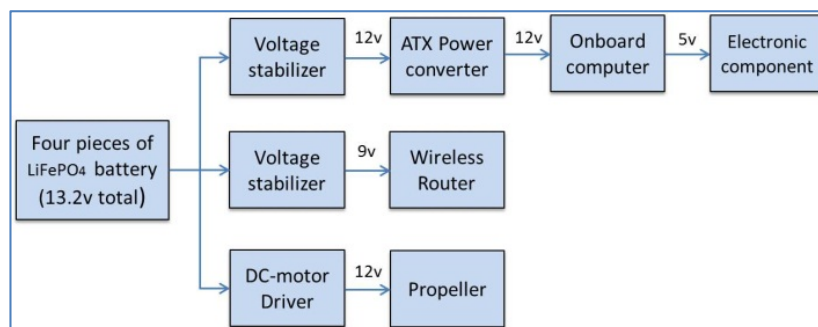


Fig.2.2 Parse tree diagram of power system

2.4 Propulsion

A pair of outboard motors is installed in our ship's stern which can provide nearly 36 pounds thrust in maximum and also possesses better maneuverability. This makes us controlling the ship motion more directly than using a rudder structure.

One outboard thruster provides nearly 18 lbs (8.16 kg) when a 180W power is

supplied. However, we only use half of its full power to propel our ship to increase the ship's cruising distance and also the endurance of the propellers.



Fig. 2.3 Outboard motor

3. Programming

3.1 GPS Module

In the cause of accurate orientation, we add Fastrax UP501 Antenna GPS module to our sensory system. Fastrax UP501 GPS receiver module with embedded GPS antenna enables high performance navigation in the most stringent applications and solid fix even in harsh GPS visibility environments.

3.1.1 Design Procedures

Control Law Design of Unmanned Surface Vessels

In this investigation, guidance law design of autonomous unmanned surface vessel design is divided into two parts. The first part is the tracking design of desired predefined waypoints. As shows in Figure 3.1, designers predefined a set of waypoints that will be tracked are given as, and the real-time vessel's positions are assumed to be measured by an embedded GPS. Based on these two messages, the real-time tracking error can be calculated until the tracking mission achieved.

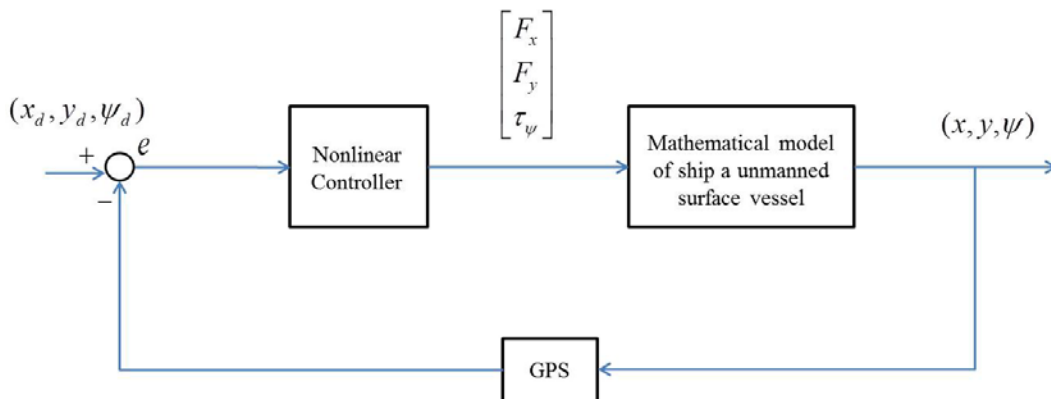


Fig. 3.1 Control flow chart of the unmanned surface vessel

Mathematical model of the unmanned surface vessel

General speaking, equation of motion of surface vessels are always expressed by a 3 DOF (degree of freedom) mathematical model. The body frame coordinate and earth frame coordinate of an unmanned surface vessel is illustrated as Figure 3.2.

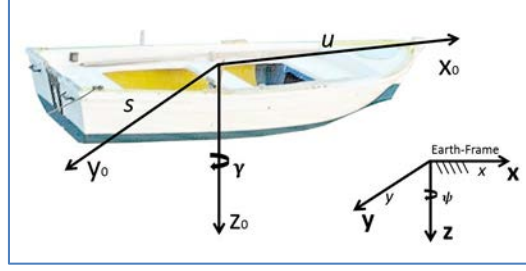


Fig. 3.2 Illustration of body frame and earth frame of the unmanned surface vessel

By choosing system state variables as: $v = [u \ s \ r]^T$ is defined as the vector of linear velocity and angular velocity of vessels in body frame, $\eta = [x \ y \ \psi]^T$ is the position and heading angle of the controlled unmanned surface vessels in the earth frame, and $\tau = [F_x \ F_y \ \tau_\psi]^T$ is the control force and torque acted in surface vessels, respectively. Based on the above definition and Lagrangian approach, dynamics of the controlled unmanned surface vessel can be expressed in body frame as:

$$M\dot{v} + C(v)v = \tau \quad (1)$$

M : Inertia matrix;

$C(v)$: Coriolis force;

τ : Control input;

Where the parameter matrices of system model of the controlled surface vessel can be further parameterized as:

I. Inertial matrix of the controlled unmanned surface vessel:

$$M = \begin{bmatrix} m & 0 & 0 \\ 0 & m & mX_G \\ 0 & mX_G & I_z \end{bmatrix} \quad (2)$$

Where

X_G : Distance from gravity center to x axis

I_z : Moment of inertia with respect to z axis

m : Weight of the unmanned surface vessel

And M possesses the following property:

$$M = M^T > 0$$

II. Parameterization of Coriolis force and central force matrix.

$$C(v) = \begin{bmatrix} 0 & 0 & -ms - mX_G r \\ 0 & 0 & m \\ ms + mX_G r & -m & 0 \end{bmatrix} \quad (3)$$

And $C(v)$ has the following property.

$$C(v) = -C^T(v)$$

The above vessel's model is described by body frame, and for practical design purpose, this model should be then transformed into earth frame. The kinematic relationship between body frame and earth frame of the unmanned surface vessel can be introduced by the following equation:

$$\dot{\eta} = J(\eta)v \quad (4)$$

Earth-fixed Vector Representation: Based on the transformation equation $J(\eta)$, surface vessel's coordinate will be transformed into earth coordinate by a relationship as:

$$\dot{\eta} = J(\eta)v \Leftrightarrow v = J^{-1}(\eta)\dot{\eta} \quad (5)$$

Where

$$J(\eta) = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (6)$$

$s \cdot = \sin(\cdot) \quad ; \quad c \cdot = \cos(\cdot)$

By the definitions in (4) and (5), we have

$$\ddot{\eta} = J(\eta)\dot{v} + \dot{J}(\eta)v \quad (7)$$

Then

$$\dot{v} = J^{-1}(\eta)[\ddot{\eta} - \dot{J}(\eta)J^{-1}(\eta)\dot{\eta}] \quad (8)$$

The controlled dynamics in earth frame coordinate of the unmanned surface vehicle is then represented by a formulation as:

$$M_\eta(\eta) = J^{-T}(\eta)MJ^{-1}(\eta) \quad (9)$$

$$= \begin{bmatrix} m_{11}c\psi^2 + m_{22}s\psi^2 & m_{11}c\psi s\psi - m_{22}c\psi s\psi & -m_{23}s\psi \\ m_{11}c\psi s\psi - m_{22}c\psi s\psi & m_{22}c\psi^2 + m_{11}s\psi^2 & m_{23}c\psi \\ -m_{23}s\psi & m_{23}c\psi & m_{33} \end{bmatrix}$$

And $m_{11} = m$, $m_{22} = m$, $m_{33} = I_z$ and $m_{23} = m_{32} = mX_G$. X_G is the center of gravity of the unmanned surface vessel. Besides, the transformed Coriolis force can be further described as follows:

$$C_\eta(v, \eta) = J^{-T}(\eta)[C(v) - MJ^{-1}(\eta)\dot{J}(\eta)]J^{-1}(\eta)$$

$$= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \quad (10)$$

Where

$$\begin{aligned} a_{11} &= -\dot{\psi}s\psi c\psi(m_{11} - m_{22}) \\ a_{12} &= \dot{\psi}(m_{11} - m_{11}s\psi^2 + m_{22}s\psi^2) \\ a_{13} &= c_{13}c\psi - c_{23}s\psi \\ c_{13} &= -c_{31} = -ms - mX_G r \end{aligned}$$

And

$$\begin{aligned} c_{23} &= -c_{32} = mu \\ a_{21} &= \dot{\psi}m_{11}c\psi^2 - \dot{\psi}m_{11} - \dot{\psi}m_{22}c\psi^2 \\ a_{22} &= \dot{\psi}s\psi c\psi(m_{11} - m_{22}) \\ a_{23} &= c_{23}c\psi + c_{13}s\psi \\ a_{31} &= c_{23}s\psi - c_{13}c\psi - \dot{\psi}m_{23}c\psi \\ a_{32} &= -c_{23}c\psi - c_{13}s\psi - \dot{\psi}m_{23}s\psi \\ a_{33} &= c_{33}, \text{ and } c_{33} = 0 \end{aligned}$$

Based on the above arrangements, an equation of motion for an unmanned surface vessel will be expressed in earth frame as:

$$M_\eta(\eta)\ddot{\eta} + C_\eta(v, \eta)\dot{\eta} = \tau_\eta \quad (11)$$

Where

$$\tau_\eta = \begin{bmatrix} F_x & F_y & T_z \end{bmatrix}^T$$

Define the tracking errors between controlled unmanned surface vessel and waypoints as:

$$e = \eta - \eta_d$$

$$= \begin{bmatrix} x \\ y \\ \psi \end{bmatrix} - \begin{bmatrix} x_d \\ y_d \\ \psi_d \end{bmatrix} \quad (12)$$

Where η is the predefined position of desired waypoints and (i.e., is at least twice differentiable) is the light-of-sight angle (LOS) between the unmanned surface vessel and waypoints. As the previous definition, η_d is the position and heading angle of the unmanned surface vessel.

Remark 1: When $\psi - \psi_d \rightarrow 0$, it means unmanned surface vessel and the predefined waypoint in the head on condition. Therefore, the unmanned surface vessel will eventually hit the predefined waypoint if tracking error e is driven to zero before the unmanned surface vessel crosses the predefined waypoint.

Design Objective

Consider system of the controlled unmanned surface vessel in (13). The design objective is mixed is to find a control law such that the following optimal performance can be achieved:

$$\min_{\tau_\eta(t) \in [0, t_f]} [e^T(t_f)e(t_f)] \quad (13)$$

For all $t_f \in [0, \infty]$

Control Law Design

For achieving the above design objective and guarantee the tracking performance, the easy-to-implemented feedback linearization method will be utilized and be mathematically derived in the following. Let us denote the system output as the tracking error e and τ_η is the control input; hence, we obtain the input-output relationship by the following mathematical derivations:

Step 1: Differentiate tracking error equation in (12) with respect to time, and then we have

$$\dot{e} = \dot{\eta} - \dot{\eta}_d$$

$$= J(\eta)v - \dot{\eta}_d \quad (14)$$

From (14), it is obvious that no control messages τ_η can be found.

Step 2: Repeat the same mathematical operation as that in Step 1, then (14) becomes

$$\ddot{e} = \ddot{\eta} - \ddot{\eta}_d$$

$$= M_\eta(\eta)^{-1} \{-C_\eta(v, \eta)\dot{\eta} + \tau_\eta\} - \ddot{\eta}_d \quad (15)$$

Suppose $M_\eta(\eta)$ and $C_\eta(v, \eta)$ are exactly known, a feedback linearization control law can be represented in the following form

$$\tau_\eta = M_\eta(\eta) \{M_\eta(\eta)^{-1} C_\eta(v, \eta) \dot{\eta} + \sigma\} \quad (16)$$

Where σ is given by $\sigma = [\sigma_1 \ \sigma_2 \ \sigma_3]^T$ and

$$\sigma_1 = \ddot{x}_d - \alpha_1(\dot{x} - \dot{x}_d) - \alpha_2(x - x_d) \quad (17a)$$

$$\sigma_2 = \ddot{y}_d - \alpha_3(\dot{y} - \dot{y}_d) - \alpha_4(y - y_d) \quad (17b)$$

$$\sigma_3 = \ddot{\psi}_d - \alpha_5(\dot{\psi} - \dot{\psi}_d) - \alpha_6(\psi - \psi_d) \quad (17c)$$

By substituting the control law (16) into (15), we get the following characteristic equation for tracking error dynamics of the closed-loop system as

$$\ddot{e} + A\dot{e} + Be = 0 \quad (18)$$

Where

$$A = \begin{bmatrix} \alpha_1 & 0 & 0 \\ 0 & \alpha_3 & 0 \\ 0 & 0 & \alpha_5 \end{bmatrix}, \quad B = \begin{bmatrix} \alpha_2 & 0 & 0 \\ 0 & \alpha_4 & 0 \\ 0 & 0 & \alpha_6 \end{bmatrix}$$

The coefficient $\alpha_i > 0$, for $i=1,\dots,6$ in (17) have to be selected so that the characteristic equation in (18) is a Hurwitz polynomial (i.e., roots of (18) are all in the left-half complex plane) and this implies that the overall system is asymptotically stable and tracking error e can be proven to converge to zero asymptotically with a convergence rate that depends on the choice of α_i

3.1.2 Simulation Result

The dynamical model parameters of the unmanned surface vessel have been given by Fossen [4], and the system parameters needed for simulation are listed as follows:

$$M = \begin{bmatrix} 25.8 & 0 & 0 \\ 0 & 33.8 & 1.0115 \\ 0 & 1.0115 & 2.76 \end{bmatrix}$$

And

$$C(v) = \begin{bmatrix} 0 & 0 & 33.8s - 1.0115r \\ 0 & 0 & 25.8 \\ -33.8s + 1.0115r & -25.8 & 0 \end{bmatrix}$$

The proposed control law is realized with

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

For verifying the control performance of the proposed method to the unmanned surface vessel, two cases with predefined waypoints that are arbitrary chosen are given as below:

In this case, twelve predefined waypoints are given to test the tracking performance of this proposed method as following:

$\{(0,0) (0,160) (60,80) (120,150) (180,240) (260,150) (300,0) (346,-100) (250,-150) (150,-150) (90,-60) (0,0)\}$

Fig. 3.3 shows the course of the unmanned surface vessel with respect to waypoints in Case and the controlled unmanned surface vessel precisely approaches every predefined waypoint. The unmanned surface vessel's positions and heading angle tracking errors in Earth-fixed coordinate are shown as Figs. 3.4-3.6, and from these figs., the tracking errors are simultaneously convergent to zero asymptotically for each tracking period. Control commands included control forces in X-Y axis and control torque are shown as Figs. 3.7-3.9 for this simulation.

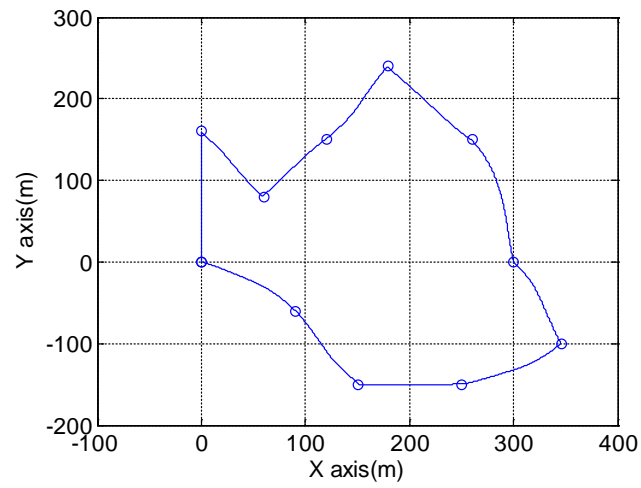


Fig. 3.3 Trajectory Plot

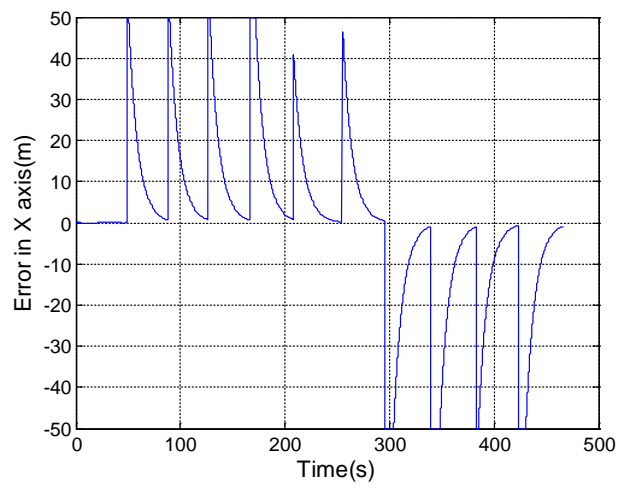


Fig. 3.4 Tracking error in X axis

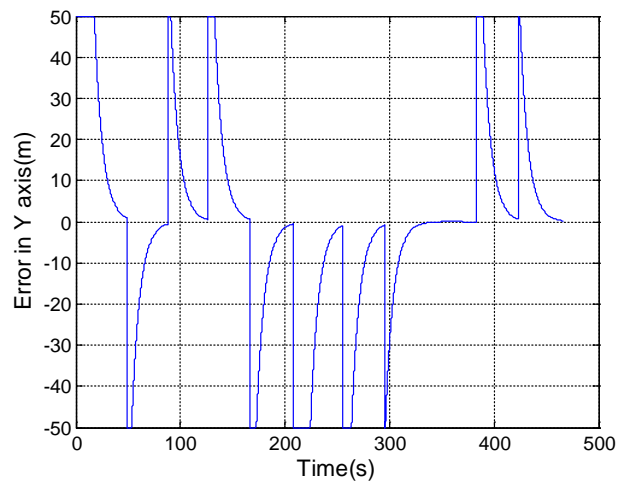


Fig. 3.5 Tracking error in Y axis

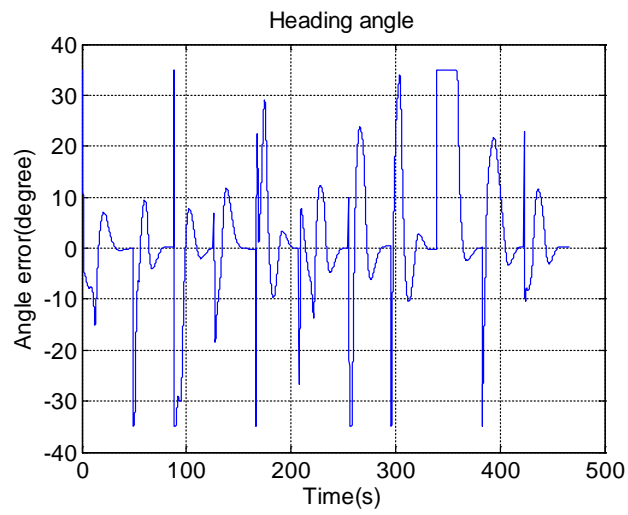


Fig. 3.6 History of heading angle error

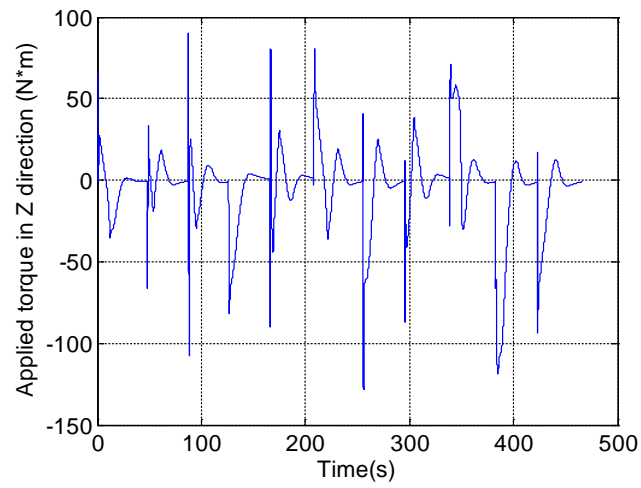


Fig. 3.7 Applied torque T_z

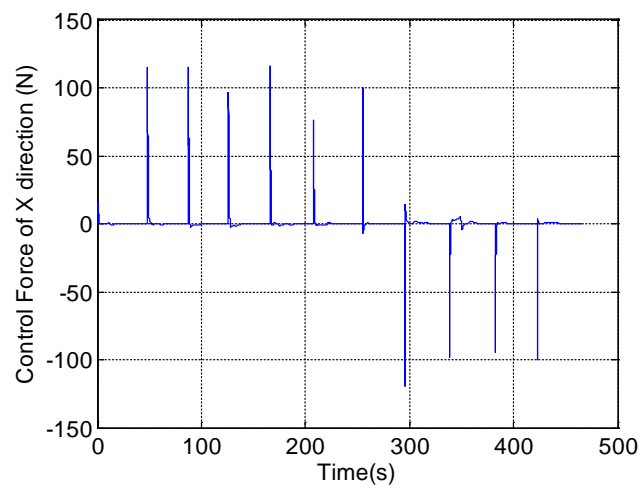


Fig. 3.8 Control history of F_x

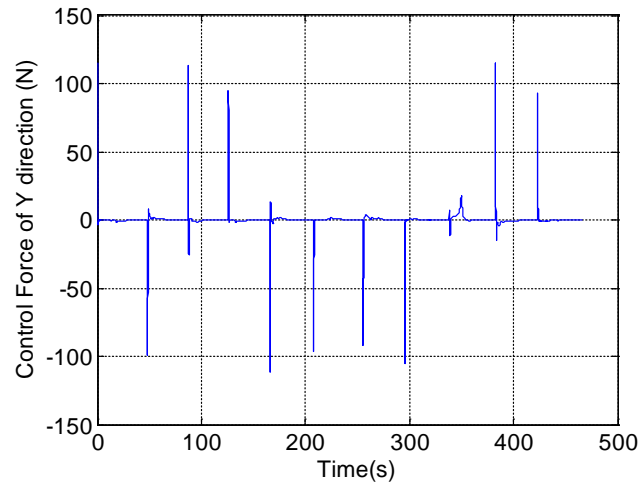


Fig. 3.9 Control history of F_y

3.1.3 Real Test

We use the above developed control method to guide Robotboat, and run experiments at An-ping open sea in Tainan (the south part of Taiwan). Fig. 3.10 shows us the surrounding of An-ping open sea. In Fig. 3.11, a red point represents our design Roboat, and green points represent the predefined waypoints. In this experiment, ten waypoints are given to approach, and as that shown in Fig. 3.12, each yellow point means the current position of the controlled Robotboat. When the Roboat arrives at the predefined waypoints, the yellow point will jump to next point. In this development, a real-time display function which could record each point of the Roboat locus is shown as Fig. 3.13.

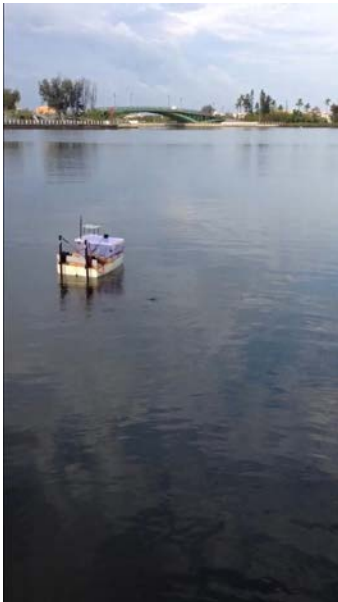


Fig. 3.10 Test at open water



Fig. 3.11 Trajectories



Fig. 3.12 Real-time display

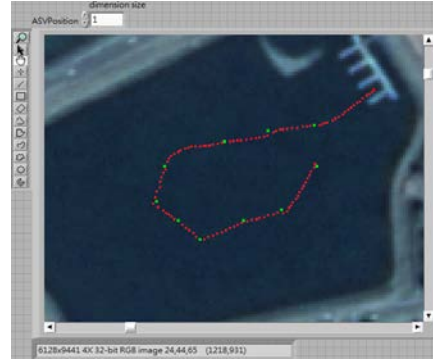


Fig. 3.13 Roboat's locus

3.2 Obstacle avoidance

For this mission, first, in front of the entrance gate made of four buoys, we use web camera to capture the image message of the entrance gate and use our designed image recognition method to analyze the colors of the buoys and drive the propellers to force the Roboat to approach the designated destination gate which is predefined by AUVSI. Then a radar with the type, SICK LMS100, connecting computer with Ethernet, will be used to detect the obstacles' (Polyform A-0 buoys) distances and collision angels and these sensing messages will feedback to control the propellers to avoid hitting the buoys. Fig. 3.14 is the flow chart for obstacle avoidance, Fig.3.15 is the flow chart for image processing.

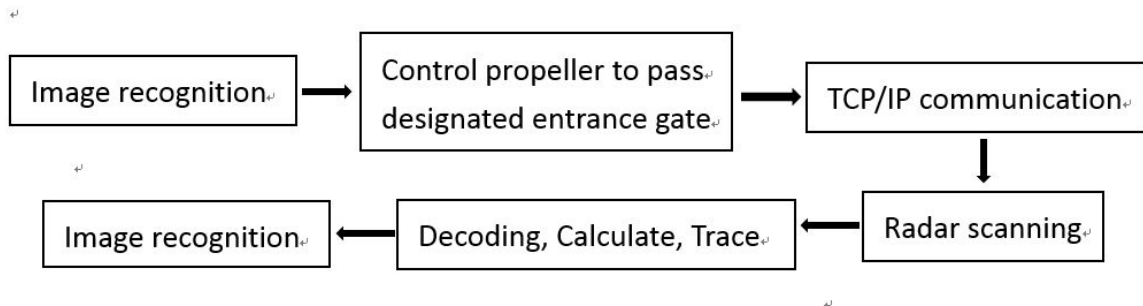


Fig. 3.14 Flow chart of mission's program

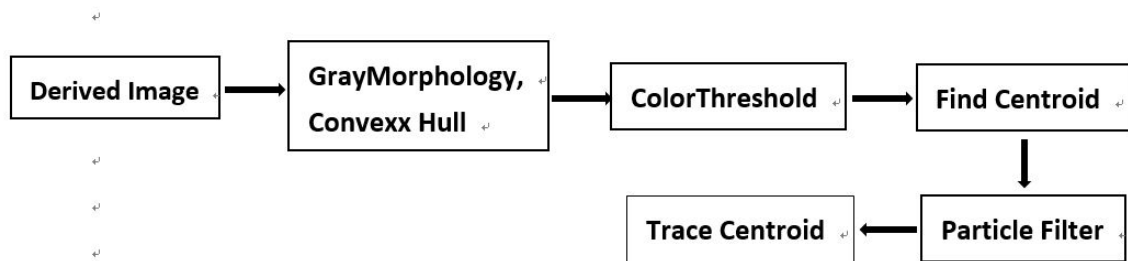


Fig.3.15 Flow chart of image processing

3.2.1 Propeller control

After image recognition and radar scanning in the competition area of the second stage, an array of values for the target image with the standard binary form will be generated. By the comparing the patterns of color storage in ARDUINO MEGA 2560 as Fig.3.16, a match pattern will be recoiled, and a PWM based on the recognition result

between the input image and stored patterns will further delivered. The 50A-Dual-channel-H-bridge-Motor-Driver-Module as Fig.3.16 is used practically for converting the PWM signal into voltage and this physical signal feeds electrically to the propeller to achieve the purpose of controlling sailing speed and steering the direction of Roboboat by the difference rotation between two individually controlled propellers.

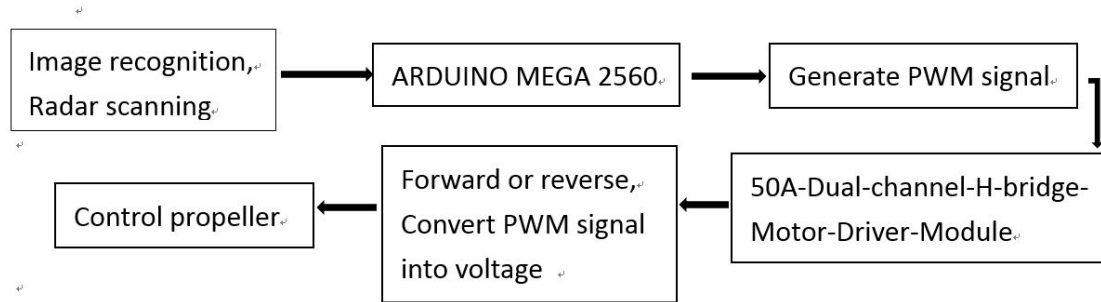


Fig.3.16 flow chart of propeller control

● Description of Pulse-width modulation (PWM)

It is a modulation technique that controls the width of the pulse, formally the pulse duration, based on signal information of the modulator. Although this modulation technique can be used to encode information for transmission, its main function is to supply the control of the power to electrical devices, especially to inertial loads such as motors.

3.2.2 Description of Adopted Devices

The following are the brief description of the used devices such as ARDUINO MEGA 2560 and 50A-Dual-channel-H-bridge-Motor-Driver-Module for proportionally transferring the binary code of detected image data to control voltages of propellers of Roboboat.

1. ARDUINO MEGA 2560

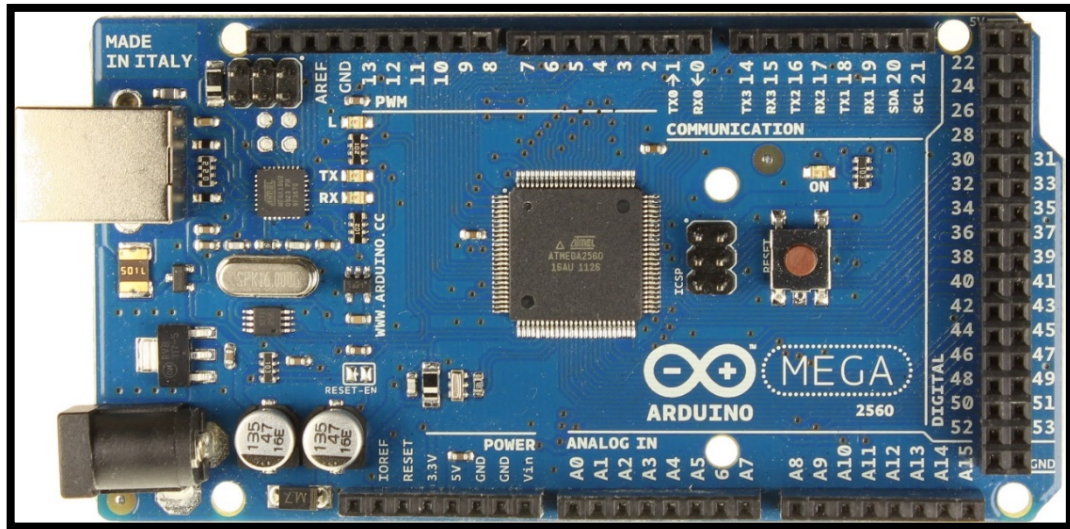


Fig3.17 ARDUINO MEGA 2560

A table listed below is the specification of ARDUINO MEGA 2560.

Performance parameters:

Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

2. 50A-Dual-channel-H-bridge-Motor-Driver-Module

Fig.3.18 is the picture of 50A-Dual-channel-H-bridge-Motor-Driver-Module, and the task of this module is used control the rotational motors of propellers.



Fig.3.18 50A-Dual-channel-H-bridge-Motor-Driver-Module

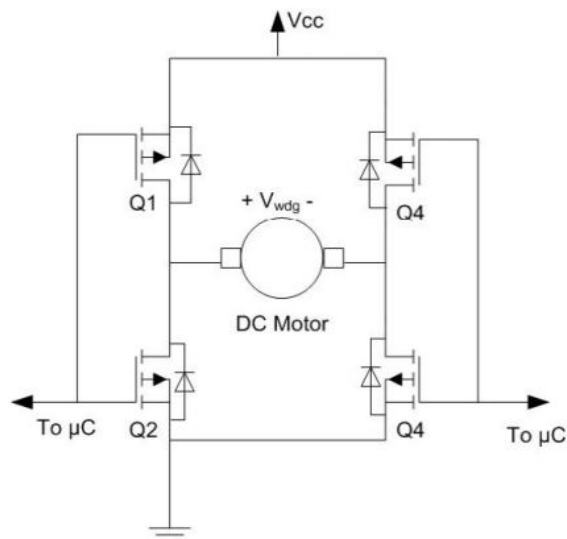


Fig.3.19 the circuit diagram of H-Bridge

A performance list as below gives the specification and basic functions of 50A-Dual-channel-H-bridge-Motor-Driver-Module

Performance parameters:

- Brand new and high quality.
- Rated voltage: 3V-15V.
- Rated current: 50A.
- Peak Current: 100A.
- Dimensions: length 6.05cm, Width 4.55cm.
- Installation: screw connection. Mounting hole spacing: 5.5cm / 4cm (length-wise/width-wise).

3. Radar (SICK LMS100)

The following Table is the features of the used sensing radar with a type LMS 100, and Fig. 3.20 is the picture of LMS 100.



Fig.3.20 SICK LMS100

Table 2 Special features of the LMS100

Variant	Special features
All	<ul style="list-style-type: none"> ● Field of view maximum 270 degree ● Scanning range up to 20m(65.62ft) with 13% object remission(18m (59.05ft) with 10% object remission) ● Resolution of the angular step with 0.25/0.50 degree ● Rotation frequency 25/50 Hz ● Flexible system configurations ● Configuration/measured value request using messages(command string) ● Data interfaces CAN, Ethernet, RS-232 Measured value output(raw data) ● Electro-sensitive, active measurement technique ● Measurement of objects with almost any shape ● Measured value output for a second reflection pulse(e.g in rain or on measurement through a window) <p>Integrate field application</p> <ul style="list-style-type: none"> ● 10 configurable evaluation fields ● Monitoring of the evaluation contours ● Switching of the evaluation field using digital inputs ● Encoder inputs

Operating principle of the SICK LMS100

The LMS100 is an electro-optical laser measurement system that electro-sensitively scans the perimeter of its surroundings in a plane with the aid of laser beams as shown in Fig. 3.21. The LMS100 measures its surroundings in two-dimensional polar coordinates. If a laser beam is incident to an object, the position is determined in the form of distance and direction.

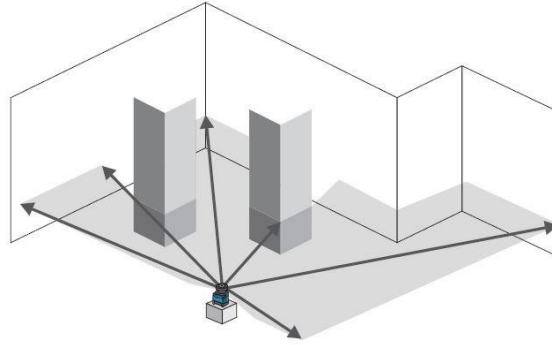


Fig.3.21 LMS100 measurement range

Scanning takes place in a sector of 270° in each sensing period. The scanning range of the LMS100 is maximum 20 m (65.62 ft) on light, natural surfaces with an object remission $> 13\%$ (e.g. a white house wall).

Distance measurement

The LMS100 emits pulsed laser beams using a laser diode. If such a laser pulse is incident to an object or a person, it will reflect to LMS 100 from the object's surface. The reflection is detected in the laser measurement system's receiver using a photodiode.

The distance to the object is calculated from the propagation time that the light requires from emission to reception of the reflection at the sensor. This behavior is indicated as Fig. 3.22.

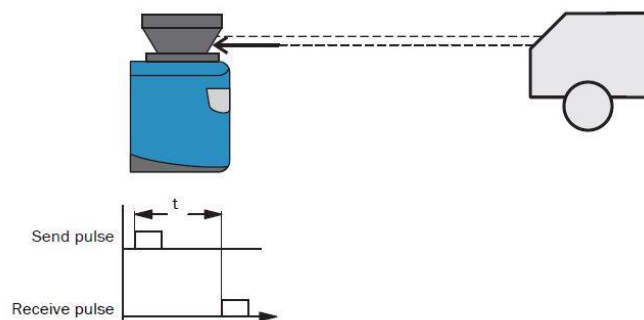


Fig.3.22 LMS100 measurement range

4. Webcam

In this part, we install the same device as SNAME102. USB camera as Fig. 3. 23 which only need little power to maintain the image captured function. Besides, it is very convenient to settle in any kind of systems because using USB camera, we do not have to load extra weight.



Fig.3.23 Logitech HD WebCam C525s

3.3 Automated docking

There are three main steps for achieving this mission: automated docking. First, the Roboboat would arrive at the specified waypoint by the propeller control System of the Roboboat. Second, the vision is captured by the USB WebCam, Logitech HD WebCam C525s, and then processed via LabView Vision Assistant which will display the results (judgment of the designated shape of dock by AUVSI) in every calculated step. Finally, the relative distance and desired turning angle between the Roboboat and target are calculated, and then Arduino board, Mega-2560 receives the delivering data pairs (relative distance and desired turning angle) to control the propeller of the Roboboat. Fig.3.20 is the block diagram of achieving the automated docking.

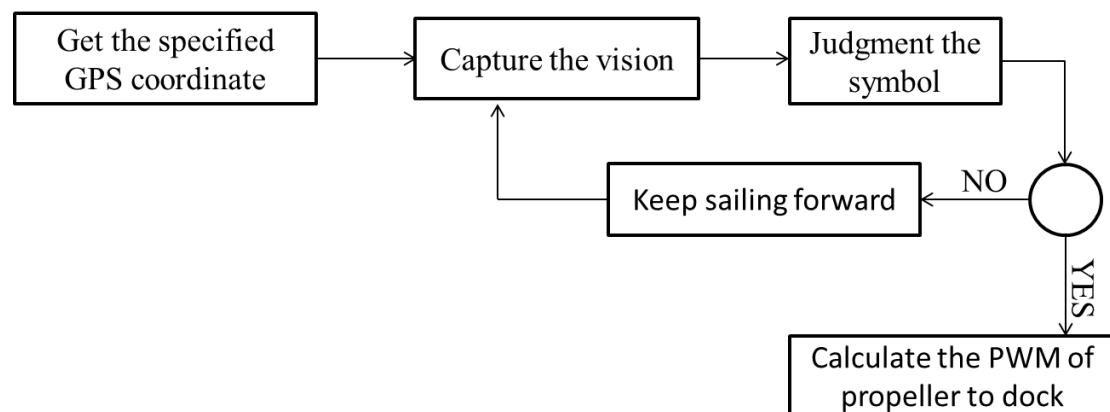


Fig.3.20 Flow chart of automated docking

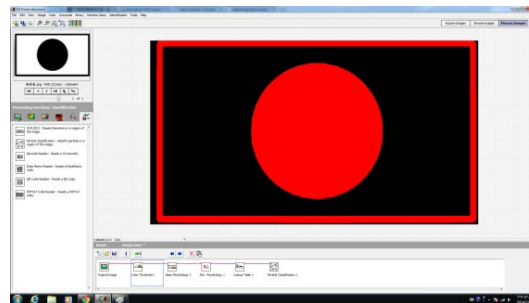
In the following subsection, each block for constructing auto decking ability for Robotboat will introduced.

3.3.1 Identification of Symbols

The vision captured via WebCam is processed in LabView Vision Assistance Vis, such as Color Threshold, Basic Morphology, Gray..., all of that are included in the Assistant of LabView. Following are the details about how the VIs processes the captured vision. In our design, three steps are included as color threshold, basic morphology, and particle classification.

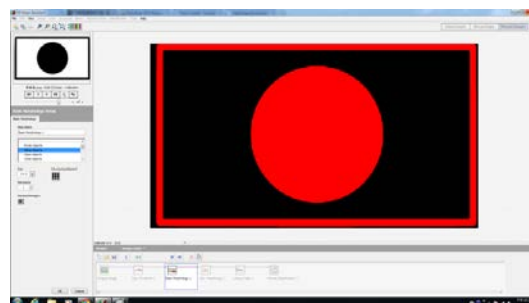
Step 1: Color Threshold

The Color Threshold module is used to remove part of the image to fit the specified color range. There are three forms: HSL, HSV, RGB, can be chosen, each of them has an appropriate application environment based on the intensity of hue, saturation and luminance of the testing environment.



Step 2: Basic Morphology

The Morphology VI is used to perform morphology operations on the vision, and this operation should perform in binary image. The main functions of this module are touching the border of an image, removing noisy particles, and removing unwanted particles based on the user characteristics.



Step 3: Particle classification

Based on the specified application, a set of training samples that represent the properties of the entire population of samples wanted to be classified is labeled. The flow chart of this process is illustrated as Fig. 3.21

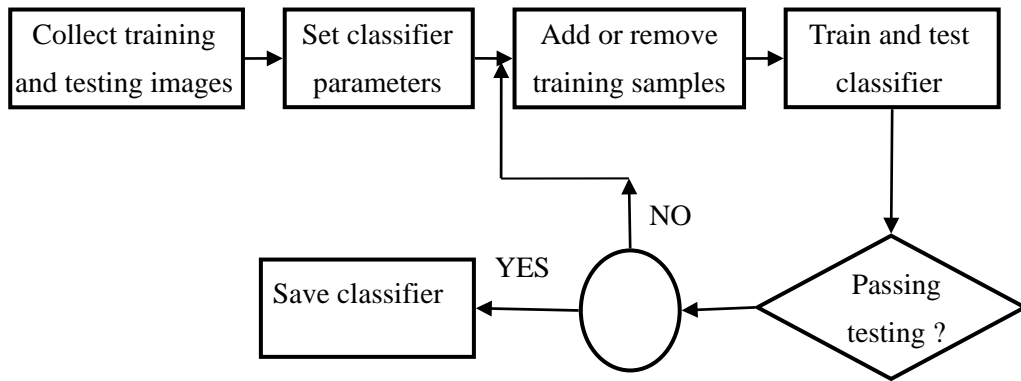
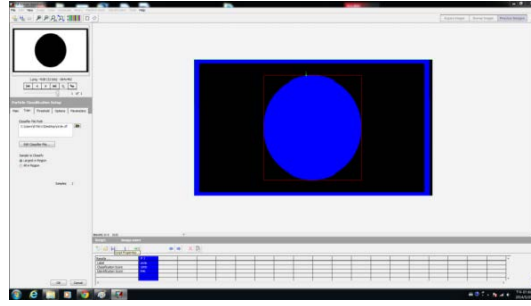


Fig.3.21 Flow chart of VIs process

Assume the shape of designated dock is exactly identified by the above Vis process, the controlled Roboboat will be guided to the desired dock with the messages of relative distance and desired turning angle by adjusting PWM outputs of installed propellers.

3.3.2 Calculate the PWMs

After identifying the correct symbol that AUVSI specified from the four symbols which each of them accompanies a ship dock near the bank, Roboboat utilizes the messages of distance and heading angle measured by Radar to calculating the heading angle error: $Error^\circ$ in the ENU Coordinate Fig 3.22, e.g., a clockwise rotation can be decided by the following equation:

$$Clockwise^\circ = \theta_{ENU}^\circ + 360^\circ - \theta_{Compass}$$

By analogue error decision process, a counterclockwise rotation decision can be made.

Next, a proposed fuzzy control method which is with linguistic variables $Error^\circ$ and \dot{Error}° is applied to inferring the corresponding PWM for controlling the two propellers.

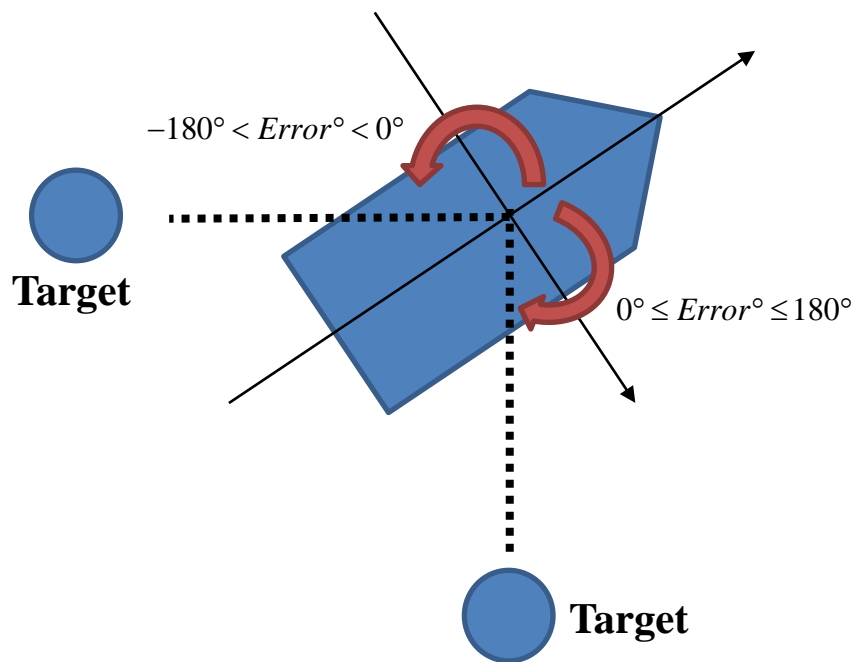


Fig 3.2

A brief expression for our proposed fuzzy control system is given here as Fig.3.23. In our fuzzy control system, it has crisp inputs and outputs as they are of real values instead of fuzzy sets. Fuzzification uses membership function to convert crisp inputs into fuzzy sets. The inference mechanism produces fuzzy conclusions based on the fuzzy rules from the rule-base, and the defuzzification block converts these fuzzy conclusions into crisp outputs.

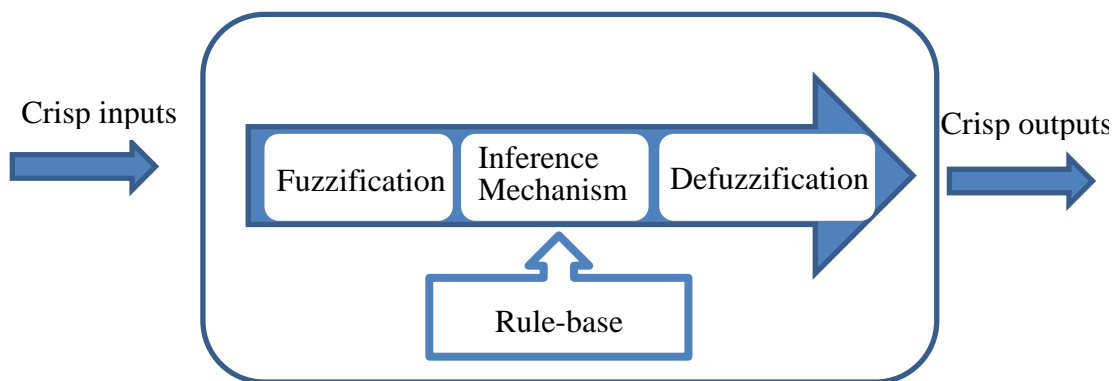


Fig 3.23

Generally, the rule-base can obtain through experiencing expert or samples training. And in the program the method of defuzzification includes the discrete

center of gravity. The formula for this method is portrayed
$$output = \frac{\sum_{i=1}^n p_i(y)\alpha_i}{\sum_{i=1}^n \alpha_i},$$

where α_i are the strengths, $p_i(y)$ are the fuzzy conclusions and $output$ are the crisp values.

3.4 Acoustic beacon positioning

For achieving the mission of precisely detecting the location of the voiced pinger, Robotboat is firstly guided by our deigned controller which is navigated by GPS to the waypoints where the position of mission arranged in priori. In this mission, the design objective is to detect the direction of an arranged pinger with a colored buoy which is set up under water. In order to identify the location of this pinger, we use a pair of hydrophones to receive the phase difference of incoming frequencies in different channels. Based on algorithms of beamforming Delay-and-Sum technique, we could get the direction and distance of voiced pinger, and then control the propeller of Roboboat to get there. When Roboboat arrives at the waypoint where the location of candidate pinger is, the webcam will capture the color of buoy, and WiFi is then used to send back to headquarter.

3.4.1 Application of Devices

a. Frequency Detecting Sensor

To detect the frequency of pinger under the water, the Brüel & Kjær Hydrophone Type 8103 which is with a small-size, high-sensitivity transducer for making absolute sound measurements over the frequency range 0.1Hz to 180 kHz are chosen. This sensor has a really high sensitivity relative to its size and good all-round characteristics which make it generally being applicable to laboratory, industrial and educational usage purpose. The 8103's high-frequency response is especially valuable when applying in acoustic investigations of marine animals and in the measurement of the pressure-distribution patterns in ultrasonic-cleaning baths. Table 2 indicates the major features of the 8103.

Table 2 Hydrophone Type 8103 Data Sheet

Type	8103
Voltage sensitivity*: (with cable) at 20 C	30 V/Pa □ 8 V (-211 dB re 1 V/Pa -2dB)
Charge sensitivity	0.12 pC/Pa
Capacitance (with integral cable)	3850 pF

Frequency range* (re 250 Hz)	(+ 1.5 dB) 0.1 Hz to 100 kHz (- 6.0 dB) (+ 3.5 dB) 0.1 Hz to 180 kHz (-12.5 dB)
Horizontal directivity (radial xy plane)	± 2 dB at 100 kHz
Vertical directivity (axial xz plane)	± 4 dB at 100 kHz
Leakage resistance (at 20 C)	$>2500\text{ M}\Omega$
Operating temperature range Short-term: Continuous:	-40°C to +120 °C -40°C to +80°C
Max. operating static pressure	252 dB = $4 \cdot 10^6$ Pa = 40 atm. = 400 m ocean depth
Dimensions Length: Body dia:	50 mm (1.97") 9.5 mm (0.37")
Weight: (including integral cable)	170 g (0.37 lb)

b. Amplifier

We use B&K 2692 as the amplifier of hydrophone 8103, the output signal will be sent to the signal control board: arduino. The following points are the features of B&K 2692:

- Highly flexible construction – 1, 2, 3 or 4 channel configurations, or a combination of acoustic and vibration transducer inputs is possible
- High input signal range. Low noise. Extensive overload facilities
- Available for: charge, microphone, sound intensity, CCLD and very high input
- Compact robust design and battery operation makes it suitable for use in the field (and in the laboratory)
- Serial control interface (RS-232) allows computer control of setups and test functions. A large number of amplifiers can be controlled from a single PC
- Wide range of filters that can be set up for specific tasks

3.4.2 Software Programming Process

The acoustic signal will be discretized and delivered from Arduino to the software program written by LabView. For dealing with this mission, Delay-and-Sum Technique which is introduced in the below section will be utilized to position the direction of

pinger, and output the direction message for controlling the embedded propeller to the location of the pinger. Delay-and-Sum method is a common method on signal processing technique in sensor arrays for directional signal transmission, and can be used detect the direction and location of a sound resource (pinger) by time difference between two used water proof microphones and is easy to implement. For practical experiments, the number of hydrophones is higher, and the accuracy of direction of pinger can be obtained. Fig.3.23 shows us the overall flow chart of acoustic beacon positioning method used in this investigation.

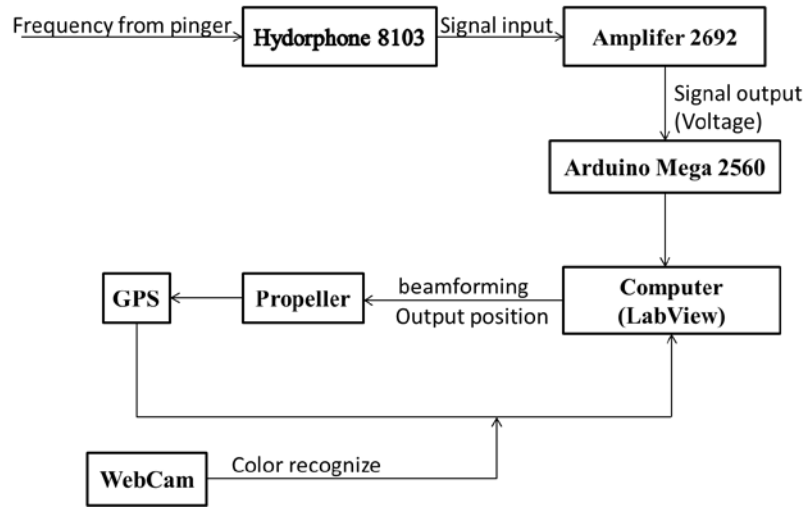


Fig.3.23 Flow chart of Acoustic beacon positioning

3.4.3 Introduction of Delay-and-Sum Technique

Delay-and-Sum technique is one of beamforming technique. This method is used to calculate the direction of pinger. Assuming there is an arrays consisting of N sensors, the array outputs, at time k , are expressed as

$$\begin{aligned}
 y_n(k) &= \alpha_n s[k - t - F_n(\tau)] + v_n(k) \\
 &= x_n(k) + v_n(k), \quad n = 1, 2, \dots, N
 \end{aligned} \tag{19}$$

where $0 < \alpha_n (n = 1, 2, \dots, N) < 1$ is the attenuation factors due to propagation effects, $s(k)$ is the unknown pinger signal, t is the propagation time from the unknown pinger to sensor 1, $v_n(k)$ is an additive noise signal at the n th sensor, τ is the relative delay between sensors 1 and 2, and $F_n(\tau)$ is the relative delay between sensors 1 and n with $F_1(\tau) = 0$ and $F_2(\tau) = \tau$. By processing the array observations

$y_n(k)$, we can acquire the angle of pinger.

This beamformer consists of two basic processing steps. The first step is to time-shift each sensor signal by a value corresponding to the time difference of arrival (TDOA) between that sensor and the reference one. With the signal model given in (19) and after time shifting, we obtain

$$\begin{aligned}
 y_{a,n}(k) &= y_n[k + F_n(\tau)] \\
 &= \alpha_n s(k - t) + v_{a,n}(k) \\
 &= x_{a,n}(k) + v_{a,n}(k) \quad , \quad n = 1, 2, \dots, N
 \end{aligned} \tag{20}$$

where

$$v_{a,n}(k) = v_n[k + F_n(\tau)]$$

and the subscript ‘a’ implies an aligned copy of the sensor signal.

The second step consists of adding up the time-shifted signals, giving the output of a delay-and-sum beamformer:

$$\begin{aligned}
 z_{DS}(k) &= \frac{1}{N} \sum_{n=1}^N y_{a,n}(k) \\
 &= \alpha_s s(k - t) + \frac{1}{N} v_s(k)
 \end{aligned} \tag{21}$$

where

$$\begin{aligned}
 \alpha_s &= \frac{1}{N} \sum_{n=1}^N \alpha_n \\
 v_s(k) &= \frac{1}{N} \sum_{n=1}^N v_{a,n}(k) \\
 &= \sum_{n=1}^N v_n[k + F_n(\tau)]
 \end{aligned}$$

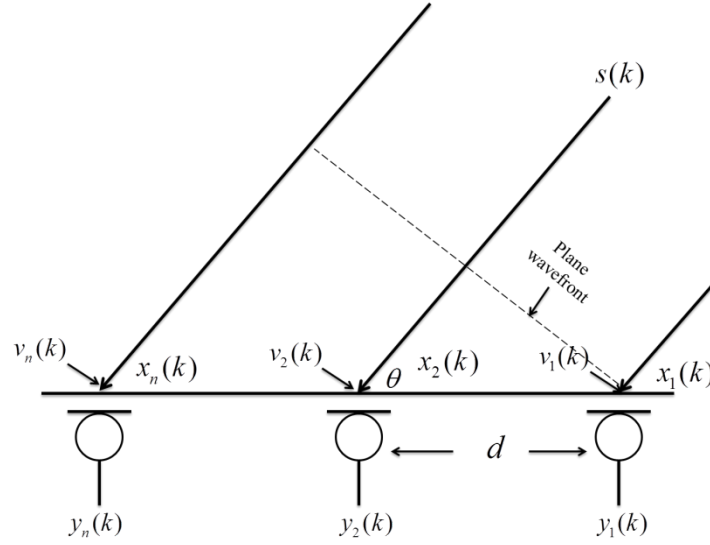


Fig.3.24 delay-and sum technique

In fig.3.24, suppose that we have an equispaced linear array, which consists of N sensors. If we donate the spacing between two neighboring sensors as d , and assume that the pinger is in the far field and the wave rays reach the array with an incident angle of θ , the TDOA between the n th and the reference sensors can be written as

$$F_n(\tau) = (n-1)\tau = (n-1)d \cos \theta / c \quad (22)$$

Where c denotes the sound velocity in air. In this case, the directional response of the DS filter, which is the spatial Fourier transform of the filter, can be expressed as

$$\begin{aligned} S_{DS}(\psi, \theta) &= \frac{1}{N} \sum_{n=1}^N [e^{j2\pi(n-1)fd \cos(\theta)/c}] e^{j2\pi(n-1)fd \cos(\psi)/c} \\ &= \frac{1}{N} \sum_{n=1}^N e^{-j2\pi(n-1)fd[\cos(\psi) - \cos(\theta)]/c} \end{aligned} \quad (23)$$

where $\psi (0 \leq \psi \leq \pi)$ is a directional angle. The beam pattern is then written as

$$\begin{aligned} A_{DS}(\psi, \theta) &= |S_{DS}(\psi, \theta)| \\ &= \left| \frac{\sin[N\pi fd(\cos \psi - \cos \theta) / c]}{N \sin[\pi fd(\cos \psi - \cos \theta) / c]} \right| \end{aligned} \quad (24)$$

From (23) and (24), we can get the phase of pinger.

3.5 Underwater light identification

For this objective, it is used color classification to identify samples based on the color of LED lights.

3.5.1 Methodology of color classifier

NI Vision's color classifier has a training phase and a classifying phase. In the training phase, we provide the classifier with known samples. A know sample consists of a region in the image containing the color that we want the classifier to learn and a label for the color. For every sample that is added during the training phase, the color classifier calculates a color feature and assigns the associated class label to the feature.

Eventually, all the trained samples (color feature with the label) added to the classifier are saved as a file which represents a trained color classifier.

After training the classifier, we can classify regions in an image into their corresponding classes for color identification and color inspection type machine vision applications. In the classifying phase, the classification engine calculates the color feature of the sample that we want to identify and classifies them among trained sample using one of the existing classification algorithms. Fig.3.25 is an practical example of applying the color classifier to identify a blue ball.

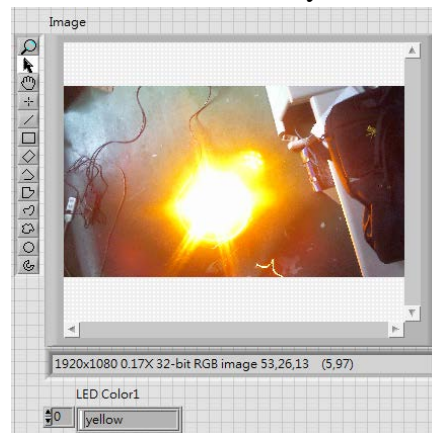


Fig.3.25 color identification example

4. Reference

- [1] Davidson, K.S.M., and Schiff, L.I., "Turning and course keeping qualities of ships," *Trans. Soc. Nav. Archit. Mar. Eng.*, Vol. 54 (1946).
- [2] Vukic, Z. "Improving fault handling in marine vehicle course-keeping systems," *IEEE Proc. Robotics and Automation Magazine*, Vol.6, Issue 2, pp. 39 – 52 (1999).
- [3] Holzhuter, T., "LQG approach for the high precision track control of ships," *IEEE Proc. Control Theory Appl.*, Vol. 144, Issue 2, pp. 121–127 (1997).
- [4] Fossen, *Guidance and Control of Ocean Vehicles*, John Wiley and Sons, New York, N.Y., pp. 19 (1994).
- [5] http://zone.ni.com/reference/en-XX/help/372916L-01/nivisionconcepts/color_classification/
- [6] <http://www.bksv.com/doc/bp1702.pdf>
- [7] <http://www.bksv.com/products/transducers/conditioning/charge/2692a0i1?cmd=Description>
- [8] http://www.sick-automation.ru/images/File/pdf/DIV05/LMS100_manual.pdf
- [9] Jacob Benesty, Yiteng Huang, Jingdong Chen, "Microphone Array Signal Processing", Springer, Berlin Heidelberg, 2008
- [10] http://i01.i.aliimg.com/wsphoto/v0/633648729_1/Free-shipping-50A-Dual-channel-H-bridge-Motor-Driver-Module-for-Robot-Chassis-Servo.jpg
- [11] http://eshare.stust.edu.tw/EshareFile/2010_4/2010_4_ccd9befb.pdf

- [12] <http://www.electroschematics.com/wp-content/uploads/2013/01/Arduino-Mega-2560-Pinout-550x267.jpg>