

國立臺灣大學工程科學及海洋工程學研究所

碩士論文

Department of Engineering Science and Ocean Engineering

College of Engineering

National Taiwan University

Master Thesis

高速轉彎自主式水下載具動力模型鑑定與

單眼視覺導航研究

Dynamic Modeling and Monocular Image-Based Pose

Tracking for an AUV in Power Turn

邱奕倫

Yi-Lun Chiu

指導教授：郭振華 博士

Advisor: Jen-Hwa Guo, Ph.D.

中華民國 105 年 4 月

April, 2016

國立臺灣大學碩士學位論文  
口試委員會審定書

高速轉彎自主式水下載具動力模型鑑定與  
單眼視覺導航研究

Dynamic Modeling and Monocular Image-Based Pose Tracking for an AUV  
in Power Turn

本論文係 邱奕倫 君 (f97525068) 在國立臺灣大學工程科學及海洋工程學系完成之碩士學位論文，於民國 105 年 3 月 31 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

高行華 (指導教授)

鄧達瑞 \_\_\_\_\_

江永雄 \_\_\_\_\_

系主任 江永雄 \_\_\_\_\_

## 誌謝



首先感謝的是郭振華教授這幾年來的照顧，提供充份的資源與舒適的研究環境，以及生活上的關心，也提供機會讓我參與研討會與國外活動，擴展了我的視野。此外，十分感謝這段期間任何幫助過我的實驗室夥伴，不論是在我研究與實驗過程中的幫忙或是在日常生活中，一起打球、打屁、玩遊戲、參加比賽，幫助我渡過這苦悶的日子。特別感謝一起攻讀博班的丹林，雖然努力的結果是我未能如願以償，但我不會忘記這過程中我們互相的勉勵與打拼。

最感謝的始終是一直支持著我的爸爸媽媽，以及我人生的伴侶宜容，我的求學過程中一直享受著你們的溺愛，對你們的感謝無法用隻言片語來形容，是你們支撐著不成才的我，才讓我能夠繼續堅持下去，我愛你們！

## 摘要



本論文探討兩項水下機器人基本問題：動力模型的建立以及載具位置的追蹤定位。本載具使用轉軸推進器，可於高速下執行橫向轉彎，並具備胸鰭進行急煞與深度控制動作。為了能夠預先估測載具動作，我們基於拉格朗日原理推導本載具動力模型。本文研究利用上方攝影系統來追蹤紀錄載具船體上的位置標記。進而取得載具的縱移、橫移與轉角速度等等數據，並應用非線性最佳化方法估測係數。經由比較模擬與實驗可知此動力模型具有相當的準確度與可靠性。

在載具的追蹤定位方面，我們採用單眼視覺方法來追蹤載具的位置與方向。我們使用前視鏡頭進行觀測，並提出一個新穎的即時最佳化估算方法。在已知地圖的環境中，使用粒子濾波器方法估測載具位置。特別的是將擴增實境的技術應用於量測模型中，這個量測方法為濾波器提供重要因子的計算方式。我們的方法已經過長時間水下巡航的驗證，實驗顯示該方法為穩定且高效能，即時提供水下載具位置與姿態。

關鍵字：自主式水下載具、水下導航、動力模型、單眼視覺、序列式蒙特卡羅定位演算法

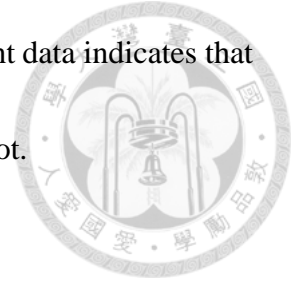
## ABSTRACT



This work investigates a development of a highly maneuverable AUV that has a high maneuverability to perform power turns. Two fundamental problems are addressed in this paper, which are the dynamic modelling of this AUV and pose tracking method by vision system. The vehicle has a rotatable stern propeller for horizontal turning at high speed, two paddles for the braking and ascending/descending. A motion model is firstly derived to predict the motion of the body. The dynamic equations are derived based on the Lagrange principle. Added mass coefficients are estimated using the equivalent ellipsoid method. A tank environment with an overhead camera system is utilized to record marker positions on the vehicle body. The iterative Lucas-Kanade method is applied for the tracking of the AUV.

To track the vehicle's position and orientation for autonomous navigation, we introduce a monocular image-based approach. Our approach is developed for an underwater environment which with fewer features and low visibility. We present a novel real-time optimizing estimation method which bases on the forward-looking camera for observing. The sequence Monte-Carlo method is used for estimating the pose of body. In particular, the augmented reality technique is involved to the measuring process, this measuring method provide the reliable estimation for importance factor. Our

approach was verified by long time cruise in a water tank. Experiment data indicates that it is robust and efficient for the real-time position tracking of the robot.



Keywords: autonomous underwater vehicle, dynamic modeling, monocular vision, Sequential Monte Carlo algorithm, augmented reality.

# CONTENTS



摘要 .....	II
ABSTRACT .....	III
CONTENTS .....	V
LIST OF FIGURES.....	VIII
LIST OF TABLES.....	XII
LIST OF SYMBOLS .....	XIII
<b>Chapter 1 Introduction .....</b>	<b>1</b>
1.1 Motivation .....	1
1.2 Related Work.....	6
1.3 System Architecture .....	9
<b>Chapter 2 Computer Vision Background .....</b>	<b>12</b>
2.1 Camera Projection Model.....	12
2.2 Extrinsic Matrix Estimation .....	16
2.3 Homography Transform .....	18
2.4 Lab color space.....	20



<b>Chapter 3 Vehicle Motion Model</b> .....	<b>22</b>
3.1 Vehicle Dynamic Modeling.....	22
3.2 Parameters Estimation .....	27
3.2.1 Thrust Estimation .....	27
3.2.2 Added Mass .....	29
3.2.3 Motion Data Gathering.....	34
3.2.4 Nonlinear Grey-Box Model Identification .....	41
3.3 Simulation.....	43
3.3.1 Test 1: L-Shaped Path .....	44
3.3.2 Test 2: S-Shaped Path.....	46
3.3.3 Comparison to the experimental data .....	48
<b>Chapter 4 Pose Tracking</b> .....	<b>52</b>
4.1 Particles Filter Algorithm .....	52
4.2 Measurement Comparing Template .....	55
4.3 Observation Model .....	57
4.4 Summary.....	59
<b>Chapter 5 Experiment</b> .....	<b>60</b>



5.1	Experiment 1 – The Comparison of EKF and PF.....	61
5.1.1	Extended Kalman Filter.....	62
5.1.2	Particle Filter Localization .....	65
5.2	Testing in NMMST’s Tank .....	67
5.3	The Kidnapped Problem.....	69
<b>Chapter 6 Conclusion.....</b>		<b>72</b>
<b>Appendix .....</b>		<b>74</b>
<b>Reference .....</b>		<b>75</b>



## LIST OF FIGURES



Figure 1.1 AUV pictured from the side and top with horizontal fins and a rotatable propeller.  
 ..... 10

Figure 1.2 AUV system architecture. .... 11

Figure 2.1 Relationship between two image coordinate representations. .... 13

Figure 2.2 The projected relationship of Homography..... 19

Figure 3.1 The reference frames of AUV with associated forces and moments. .... 23

Figure 3.2 The relationship between force and voltage..... 29

Figure 3.3 The free-running model test system. The distance  $D$  is the designed position  
 where the AUV should begin the test motion.  $A$  is the observable area, and the  
 blue dashed line represents the water surface..... 34

Figure 3.4 Experiment surrounding and the EM calibration pattern. .... 35

Figure 3.5 Calibration pattern captured by the top camera. Green circles are the  
 reprojection of corners..... 35

Figure 3.6 (a) The tracking of the two makers. The two trajectories of makers will be  
 described as two polygon functions (b) The trajectories in global frame which  
 through the HM mapping. The grid size is  $1m*1m$ ..... 37

Figure 3.7 The (a) surge, (b) sway and (c) yaw rate during this test in three different speed input values.....	40
Figure 3.8 The motor input value for dynamic model simulation. For our system, the motor input contains the direction and speed of the propeller.....	44
Figure 3.9 The simulated L-shaped trajectory.....	44
Figure 3.10 The change of (upper) sway and yaw rate in this simulation.....	45
Figure 3.11 The motor input value for S-shaped trajectory.....	46
Figure 3.12 The simulation result of S-shaped trajectory.....	46
Figure 3.13 The change of (upper) sway and yaw rate in this simulation.....	47
Figure 3.14 Comparison of surge, sway and yaw rate at speed inupt 30. ....	49
Figure 3.15 Comparison of surge, sway and yaw rate at speed inupt 40. ....	50
Figure 3.16 Comparison of surge, sway and yaw rate at speed inupt 50. ....	51
Figure 4.1 The process of particles filter localization .....	53
Figure 4.2 Sampled densities obtained for incremental movement of an AUV motion. Path are L-shpaed (left) and S-shpaed (right) which same as the simulation in section 3.3.....	54
Figure 4.3 The landmark extraction. Red rectangular represent first step - the color segmentation. After that, the center of LED will search in the region, which is represented by yellow cross.....	56

Figure 4.4 The Landmark detecting result and its corresponding histogram.....	56
Figure 5.1 The underwater robot IF and its peripheral equipment.....	60
Figure 5.2 Stage arrangement of the experiment 1. While turning, two buoys are used for guiding the AUV. ....	61
Figure 5.3 Result of EKF localization.....	64
Figure 5.4 x-position change of covariance with time .....	64
Figure 5.5 y-position change of covariance with time .....	65
Figure 5.6 Result of SMC localization .....	66
Figure 5.7 Changes of sample weight with time .....	66
Figure 5.8 The environment of the testing water tank in National Museum of Marine Science & Technology, Keelung, Taiwan .....	68
Figure 5.9 The long cruise testing environment, red line is the reference path for one trip, two landmarks are mounted each side. The initial position of AUV was located at the middle of two landmark and the AUV must keep 1m distance to the side walls.....	68
Figure 5.10 The tracking result of the cruise testing. (a) X-psition tracking result. (b) Y-position tracking result. (c) Yaw angle tracking result. ....	69
Figure 5.11 Re-localization for the kidnap problem.....	71
Figure 5.12 Changes of information entropy.....	71



## LIST OF TABLES



Table 3-1 Loading vs. voltage (control input) experiment results.....	28
Table 3-2 EM Parameters in our experiment environment (T in cm; $\theta$ in degree).....	36
Table 3-3 Parameters of dynamic model .....	43
Table 5-1 Extended Kalman filter localization algorithm. ....	63
Table 6-1 Specifications of the propeller:.....	74

## LIST OF SYMBOLS



symbols	Description
$(x, y)$	The original point coordinates of distorted images
$(\hat{x}, \hat{y})$	The corrected point coordinates of undistorted image
$(x_c, y_c)$	The center of the camera distortion model
$\mathbf{k} = (k_0, k_1, \dots, k_n)^T$	Distortion parameters
$C(r)$	The calibration model
$\{(\hat{x}_{\ell,i}, \hat{y}_{\ell,i})\}$	The corrected points (undistorted) using the distortion model
$\hat{S}^\ell$	The covariance matrix of each line
$\overline{\hat{x}_{\ell,i}} = \frac{\sum_{i=1}^{N_\ell} \hat{x}_{\ell,i}}{N_\ell}$	The average of respective variables
$\overline{\hat{y}_{\ell,i}} = \frac{\sum_{i=1}^{N_\ell} \hat{y}_{\ell,i}}{N_\ell}$	The average of respective variables
$\hat{E}(k)$	A forth-order polynomial in the coefficient of $\mathbf{k}$
$f$	The focal length of the camera
$\theta$	The skewness of camera coordinate system
$\mathbf{A}$	The homogenous camera intrinsic matrix

$P_c$	A coordinate position with respect to the cameracoordinate system
$X_c - Y_c - Z_c$	The original point of camera coordinate system
$X_w - Y_w - Z_w$	The original point of the world coordinate system
$R_{3 \times 3}$	A rotation matrix
$t_{3 \times 1}$	A translational vector
$\tilde{m}$	A homogeneous coordinate
$\tilde{M}$	A homogeneous coordinate
$s$	An arbitrary scale factor
$H$	A perspective projection
$\lambda$	An arbitrary scale factor
$h_i$	The i-th column of $H$
$B$	A symmetric matrix
$(\hat{u}, \hat{v})$	The corrected pixel coordinates
$(u, v)$	The original pixel coordinates
$(u_0, v_0)$	The principal point of images
$A_c$	Intrinsic parameter matrix
$R_n$	Extrinsic parameter
$X_n, Y_n$ and $Z_n$	The white value of the parameter





$M(x, y)$	Gradient magnitude of the image
$H(x, y)$	Gradient direction of the image
$(x, y)$	The coordinates of the points on the straight line
$\rho$	The normal distance from the image origin to the line
$\theta$	The angle the normal line makes with the $x$ -axis
$x_t$	The robot pose
$bel(x_t)$	The belief of the vehicle's state
$z_t$	The observation
$u_t$	The control inputs
$Bel(x_{t-1})$	The belief of pervious vehicle's state
$p(x_t x_{t-1}, u_t)$	The probabilistic motion model
$p(z_t x_t)$	The probabilistic observation model
$bel(x_0)$	The initial distribution of the vehicle's state
$p(z_t x_t^{[i]}, m)$	The probability of the measurement probability
$X_g - Y_g$	The global coordinate system
$X_b - Y_b$	The vehicle's body coordinate system
$\mathbf{X} = [x \ y \ \theta]^T$	The state of vehicle
$\mathbf{U} = [u \ v \ \omega]^T$	The control input of the vehicle
$u$	The $X_b$ -axis velocity



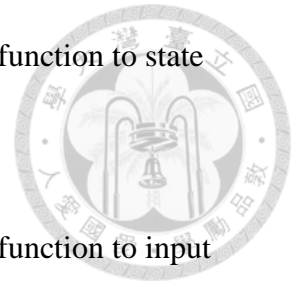
$v$	The $Y_b$ -axis velocity
$\omega$	The angular velocity
$X_t$	The predicted state at time $t$
$X_{t-1}$	The present state at time $t-1$
$U_t$	The current control input at time $t$
$\Delta t$	A time interval
$X_t^{[i]}$	The $i_{th}$ particle state in time step $t$
$\omega_t^{[i]}$	The $i_{th}$ particle weight in time step $t$
$V$	Velocity of vehicle
$\alpha$	Acceleration of vehicle
$\psi$	Angular velocity of vehicle
$m$	Map value
$n_{eff}$	Effective sample size
$K$	Particle number
$\rho$	Correlation coefficient
$\eta$	Constant
$m_{x,y}$	The local map value
$m_{x,y,local}(X_t)$	The global map value
$\bar{m}$	The average map value



$N$	The number of overlap cell between local and global map
$O_c$	The original point of camera coordinate system
$O$	Principal optical axis
$D_{po}$	The pixel distance from the center of the target to the principal optical axis
$D_f$	The distance from the center of the target in the image to the principal optical axis
$\theta$	The target relative camera the error angle
$D_{curmethreading}$	The angle recorded by the electronic compass
$D_{goalheading}$	The compensated angle which is continuously updated for the next tracking
$D_{p_t}$	The pixel distance of target
$D_t$	The real distance of target
$D_{d_t}$	The distance between $O_c$ and the plane of target
$g$	State transition function
$\varepsilon$	The noise of state
$\bar{X}_t$	Predicted of EKF algorithm's state
$\bar{\Sigma}_t$	Covariance



$\mathbf{G}_t$	The derivative of the state transition function to state variables
$\mathbf{V}_t$	The derivative of the state transition function to input variables
$\mathbf{R}_t$	Control input covariance
$h$	Observation function
$\delta_t$	The noises of observation
$\bar{z}_t$	Measurement prediction
$m_p$	The target position
$h_t$	The coordinate transformation between the world frame and the sensor frame
$\mathbf{H}_t$	The derivative of the measurement function to state variables
$\mathbf{Q}_t$	Observation covariance
$\mathbf{K}_t$	Kalman gain
$I$	An unit vector



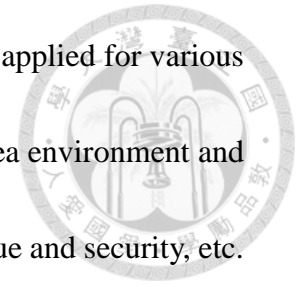
# Chapter 1 Introduction



## 1.1 Motivation

The ability of self-localization is the prerequisite for intelligent robots, it due to the fact that the knowledge of position is the key information for carrying out various tasks. Today, many powerful sensors have come out such as *Global Position System* (GPS) or laser range scanner; they provide a reliable measurement for estimating vehicle's position and perceiving environment. However, for underwater environment, acoustic-base sensors are still the most common used approach. Global Position System is not suitable here due to their rapid attenuation. The *Doppler velocity log* (DVL), which gets the motion information by calculating the time varying between the several sound wave cones, provides the vehicle's trajectory. Similar measurement also can be got by the *inertial navigation system* (INS). In recently years, variety of vision-based methods for estimating robots position is presented [9] [10] [24]. Compare to other sensors, vision base approach provides good estimation on short range measuring. In addition, camera has fascinated characteristics for under water application such as smaller size and cheaper price. The information of vision also contributes other tasks such as object detect or path planning. Over the past years, camera has become an indispensable sensor for the underwater vehicle. Today,

numerous *autonomous underwater vehicles* (AUV) are ready to be applied for various missions: undersea infrastructures construction and maintenance, sea environment and life investigation, military missions, prospection, sunken ships rescue and security, etc.

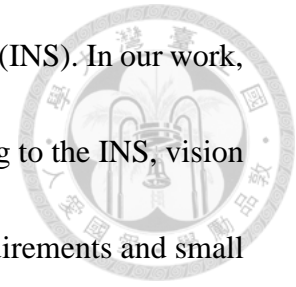


Vision has become essential for all these applications. Hence, there exists a good motivation to improve AUVs navigation techniques by expanding their autonomy, capabilities and their usefulness.

However, although various sensors are maturely applying in the AUV navigation, dynamic model of vehicle is still indispensable on pose tracking problem. Different to kinematics, the dynamics of a robot vehicle describes how the robot moves in response to these actuator forces. Dynamic model provides the prior estimation for vehicle's motion. Once the equations of motion for a vehicle are known, a simulation of vehicle motion can be used to compare to real experimental data. The vehicle localization is further enhanced with fusion of model with onboard depth and yaw angle measurements through a particle filter.

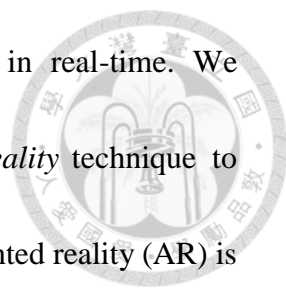
In this paper, the identification approach is based on the free-running method [3]. Common used method for acquiring the motion data is using the *Planar Motion Mechanisms* (PMM). However, PMM needs a huge water tank and a towing tank carriage, and it is a heavy price for maintaining it. On the one hand, free-running experiment allowing AUV to swim for a range without any constrain. The motion of

vehicle was measured by sensors such as inertial navigation system (INS). In our work, we use vision system for measuring the motion of AUV. Comparing to the INS, vision system has the merit that has no drift problem, lower hardware requirements and small experimental site.



Localization and tracking problem can be classified into two categories: *relative* and *absolute*. The relative localization method measures the on-board sensors such as odometry, gyroscope and accelerometer. Camera also could be applied on relative localization, this method called “visual odometry” and it has numerous researches and implementations [10] [13] [25]. Dead-reckoning is the navigation technique which employs the odometry data in the motion model of robot, and compute the position relative the start position. Although the relative localization can be applied in different environment, the localization error will grow with time. In contrast, absolute localization obtains the error-constrained position by beacons, satellite-based signal and landmarks. Absolute-base method could localize the robot even the robot suffer an abrupt movement. However, in practice, the robot cannot observe any landmarks sometime, so usually a robot user will fuse both of the information of landmarks and the odometry data by a Kalman filter or particle filter [22] for getting more reliable estimations.

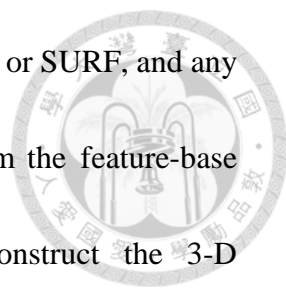
We address a monocular image-base method which robustly continues estimating



the vehicle's absolute position, and reconstructs the trajectory in real-time. We introduce a novel method which is inspired by the *augmented reality* technique to provide the non-Gaussian distribution measurement model. Augmented reality (AR) is the opposite of the virtual reality (VR), which puts user into a completely synthetic environment. The augmented reality system makes the real and virtual objects coexist in a same space. There are many successful application of augmented reality [14] [15] [16]. The augmented reality system tries to enhance the user's perceptual ability, the beginnings could date back to Sutherland's research in 1960s. Early researches uses the device which called head mounted display (HMD) to see through and receive the augmented images. Recently, due to various kinds' developments of display device, AR no longer is constrained on heavy devices. Today, augmented reality is showing its potential on different applications like medical, robot path planning, commercial and military.

The idea of the AR-base tracking method is explained as follows: by pre-rendering what will be seen by the augmented reality technique, if the expectation is similar to the observing image which is captured from the camera, the uncertainty should be decreased. The pose estimating method will try to find the pose with best alignment between the projecting landmarks and the observing landmark on the appearance image. For detecting the observation landmark, the color segmentation

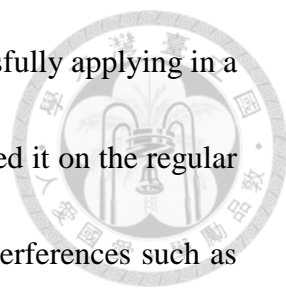




technique is adopted, more complex feature descriptor such as SIFT or SURF, and any feature correspondence algorithm are not used here. Different from the feature-base method, the appearance measurement method does not reconstruct the 3-D environment; the measurement is relative to the re-projected image. This appearance measurement is not only non-Gaussian but is also a non-parametric problem.

We employed the sequential Monte Carlo method, or called particle-filter, in state estimation to cope with the measurement information. In general, more landmarks provides more accurate estimation. However, even there is just only one observed landmark, experimental data shows that our approach is still able to constraint the tracking error. In practical applications, two landmarks (LEDs) at each side of the testing tank were placed, and the error of localization could be less than 1 meter. All measuring processes are only referring to the current appearance image, so our method does not need to solve the data association problem. The data association problem is that correspond the measurement to the existence map, it is unavoidable for calculating the movement estimation. However, wrong associations will cause the localization disastrous. Our algorithm omits this expensive stage by simplifying the environmental features, and continually provides the robust measurements.

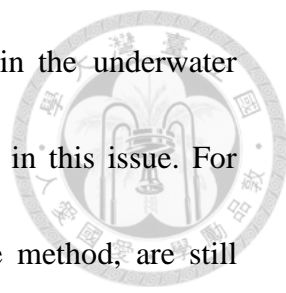
Compare to other landmark-based method, our approach does not need to recognize the artificial landmark in the observing process. The prior information in our



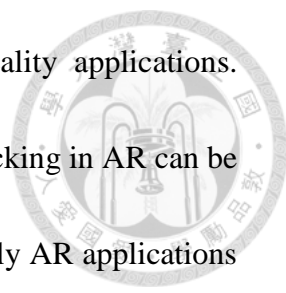
method is the position of the landmark. Our system has been successfully applying in a real autonomous underwater robot for the marine museum. We tested it on the regular exhibition days when the environment contains full of dynamic interferences such as images of tourist and the change of luminance. It shows that this method is robust enough for such a noisy environment and it only depends on a single camera. The remainder of this paper is organized as following: Chapter 2 introduces the background of the image based method for navigation related to our research. Chapter 3 talks about the modeling of our robot. A novel method for seeking the parameters will also address in this chapter. The particle-filter localization method is introduced in Chapter 4. Following the experiment result and conclusion will be shown in Chapter 5 and in Chapter 6, respectively.

## 1.2 Related Work

This paper we investigate the dynamic model of underwater vehicle. Similar researches which derive model by Newton's second law [5] [6] [8] or Lagrangian equations [4], they are basically the same but expressed in different formats. Our dynamic model is close to V. Kopman develop in [4], V. Kopman reduce degree of freedom to a planar motion, and use nonlinear optimization method in parameter identification.



Tracking and pose estimation are quite a complex problem in the underwater navigation community. Until now there is no single best solution in this issue. For underwater applications, traditional approaches, i.e. acoustic-base method, are still indispensable. However, acoustic systems usually are limited in resolution and size. Although the range of visibility is seriously limited in waters, more and more researchers using vision systems for their advantages of small size, low cost and better in the resolution. Recently, the major technique of localization in undersea area is the combination of the position estimation with map construction techniques. Haywood [26], Marks *et al.* [27] mosaicked the bottom image of sea and extracted the frame-to-frame displacement information for navigation purposes. Mosaic-base method usually assumes that the bottom of sea is plane and static, and the vehicle moves on the plane which is almost parallel to the target plane. Eustice and Pizarro [13] employed the combination of camera with other sensors to get the six-degree-of-freedom movement of a vehicle. It improves the robustness of image registration than camera-only approach because the pose had been constrained by multiple sensors. Gracias [10] developed a method for mosaicking that did not constrain the camera motion or its relative orientation to the target plane. The method was based on self-calibration technique by robust matching algorithm between pairs of consecutive images of a video sequence.



Position tracking plays an essential role for augmented reality applications. Depend on tracking features is artificial or not, the vision-based tracking in AR can be divided into the marker approach and the markerless approach. Early AR applications in prepared environments usually use the fiducially markers [16] or light emitting diodes (LEDs) to find out the relationship between the local and world coordinate. In contrast, markerless-base AR was developed for less-constrain. The markerless-base AR tracks camera by natural features [28]. Recently an outstanding work is by Schöps [28] who presented a direct monocular visual odometry algorithm. In addition to estimate the camera pose, the technique maps the environment by pixel-wise filtering over many small-baseline stereo comparison. This method is fundamentally different from feature-based method, it allow for fast and sub-pixel camera tracking. Even though the robustness and the computational costs need to be improved, pure vision-based approaches for pursuing general-case, real-time tracking are very promising.

Han [29] applied fisheye lens to observe the surrounding environment. The ground robot could obtain more information by the omnidirectional vision system. They also employ landmark-based global localization by particle filter, and developed the measurement model for the landmark. Wolf [17] also presented visual global localization based on particle-filter. They compared the possible view-points to

database by color image retrieval system. To overcome that small corresponding view-point problem, an invariant feature was defined.



### 1.3 System Architecture

The tested AUV has 90 cm overall length and 19.5 Kg dry mass, see Fig.1.1. The streamlined shell of AUV is made of acrylic. In front of the AUV retains two holes for observing cameras. A water-proof aluminum box contains the battery and all electric systems are mounted inside the shell. Particularly, the AUV equipped a propeller which is mounted on a rotatable fixture in the rear side, the AUV then has the thrust vector control ability. In general, the system is divided into three partitions, which include:

(1) Computer: An Advantech highly-efficient industrial PC is equipped. It receives the pre-command from operator through wireless communication and subsequently arranges the tasks. All movements of the lateral fins and thrusters are controlled by the PC. The control command is designed by the PID control to increase the robustness and the smoothness in navigation.

(2) Sensors: The perception system includes two ARTCAM-022MINI cameras which has a frame rate up to 50 fps, a pressure transducer and an electronic compass. All sensors provide the feedback information to the computer for controlling the depth and heading of vehicle. The measurement update rate is 10 Hz, which is sufficient for

motion control in our application.

(3) Propeller and Control Surface: This module executes the final decision of task command. Two FAULHABER 2657\_024CR series motors with the maximum torque of 1.324 N-m are used to drive the angle of the lateral fin and the propeller.

Compared to traditional rudders, the vectoring thrust provides sufficient heading control ability even in quite slow speeds, also decreases the drag force during swing moment. The propeller of the AUV can rotate in the horizontal plane, so the depth is controlled by the lateral fins. Additionally, two Lithium-ion batteries and three voltage transformers are placed ahead of the waterproof box to balance the gravity of the rear bracket and the propeller.



Figure 1.1 AUV pictured from the side and top with horizontal fins and a rotatable propeller.

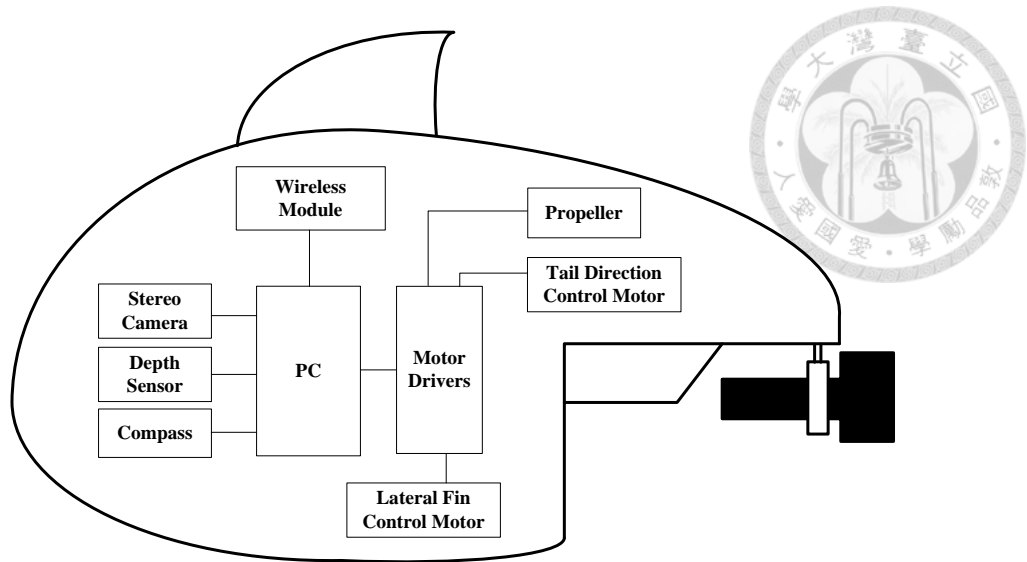


Figure 1.2 AUV system architecture.

## Chapter 2 Computer Vision Background



In this chapter, we briefly introduce the background knowledge about computer vision. These image mapping techniques will be mentioned in this chapter, include the following parts:

- (1) *Motion data Measuring (section 3.2.3)*
- (2) *The observation model of particles filter (section 4.3)*
- (3) *Landmark detection (section 4.2)*

### 2.1 Camera Projection Model

The purpose of camera calibration is to find the model that describes the transformation relationship among the image coordinate system, the camera coordinate system, and the world coordinate. There are two type coordinate representations for image coordinate. The one is described by using metric units and the other is described by using pixel units. The  $x - y$  coordinate system is used to describe the first one and the  $u - v$  coordinate system to describe the second one. The relationship of the two descriptions can be shown as:



$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2.1)$$



Where  $O_1(u_0, v_0)$  is the principal point of image,  $\alpha_u$  and  $\alpha_v$  are the scale factors in image of  $u$  and  $v$  axis respectively. The relationship between two coordinate representations is shown in following figure.

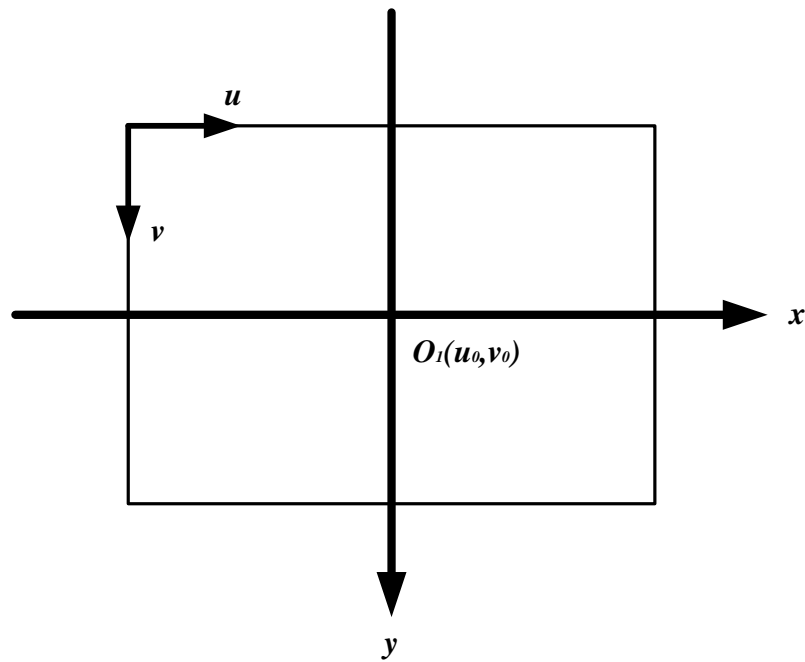
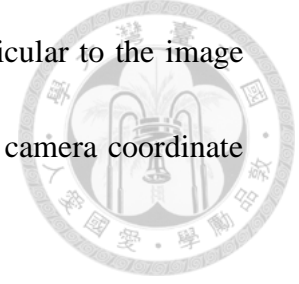


Figure 2.1 Relationship between two image coordinate representations.

The original point of camera coordinate system ( $X_c - Y_c - Z_c$  coordinate system) is at the optical center of the camera lens. The  $X_c$  axis and  $Y_c$  axis are parallel to the  $x$

axis and  $y$  axis respectively. Additionally, the  $Z_c$  axis is perpendicular to the image plane. The relationship between the image coordinate system and camera coordinate system can be written as:



$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{z_c} \begin{bmatrix} f & -f \cos \theta & 0 & 0 \\ 0 & f \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (2.2)$$

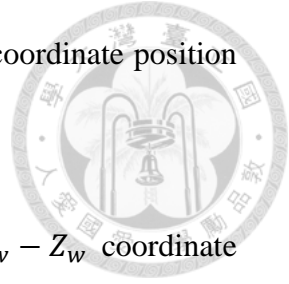
Where  $f$  is the focal length of the camera,  $\theta$  is the skew factor of camera coordinate system. Because of the excellent modern camera manufacturing technology, the skew factor almost is zero for the camera models. Hence, the  $\theta$  should be equal  $90^\circ$ , the equation could be rewritten as follow:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{z_c} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (2.3)$$

Substitute equation 2.3 to equation 2.1, the equation could be shown as:

$$\begin{aligned} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} &= \frac{1}{z_c} \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \\ &= \frac{1}{z_c} \begin{bmatrix} f \alpha_u & 0 & u_0 & 0 \\ 0 & f \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \equiv \frac{1}{z_c} AP_c \end{aligned} \quad (2.4)$$

where  $A$  is the homogenous camera intrinsic matrix,  $P_c$  is a coordinate position with respect to the camera coordinate system.



The original point of the world coordinate system ( $X_w - Y_w - Z_w$  coordinate system) is at any point of the three-dimensional space. The relationship between the camera coordinate system and the world coordinate system can be depicted as follow:

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R_{3 \times 3} & T_{3 \times 1} \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2.5)$$

where  $R_{3 \times 3} = R_{x_c} R_{y_c} R_{z_c} = (r_x \ r_y \ r_z)$  is a rotation matrix which can be obtained from matrix multiplication of the three basic rotation matrices about the  $X_c$ ,  $Y_c$ , and  $Z_c$  axis,  $r_x$ ,  $r_y$ , and  $r_z$  are the three columns unit vectors of  $R$ ,  $T_{3 \times 1}$  is a translational vector which relates the world coordinate system to the camera coordinate system, and  $0^T = [0 \ 0 \ 0]^T$ .

Substitute equation 2.5 into equation 2.4, a linear transformation between the world coordinate system and the image coordinate system can be derived as follow:

$$s\tilde{m} = A_c(R \ T)\tilde{M} \quad (2.6)$$

where the  $A_c = \begin{bmatrix} f\alpha_u & 0 & u_0 \\ 0 & f\alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$ ,  $\tilde{m} = [u \ v \ 1]^T$  is a homogeneous

coordinate,  $\tilde{M} = [X_w \ Y_w \ Z_w \ 1]^T$  is a homogeneous coordinate in the three-dimensional space,  $s$  is an arbitrary scale factor.



## 2.2 Extrinsic Matrix Estimation

Recall the eq. 2.6, matrix  $[R|t]$  represents the extrinsic parameters which denote the coordinate system transformations from 3-D world coordinates to 3-D camera coordinates. The extrinsic parameters define the position of the camera center and the camera's orientation in world coordinates.  $T$  is the position of the origin of the world coordinate system expressed in coordinates of the camera-centered coordinate system.

The parameters of intrinsic matrix (IM) usually are known from the specifications of camera. Following we explain how to estimation the extrinsic parameters. We expand eq. 2.6:

$$s\tilde{m} = \begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2.7)$$

Combine intrinsic matrix and extrinsic matrix into  $M$ , and the resulting matrix is called the camera projection matrix.

$$M = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}$$



(2.8)

Hence the projection can be express as:

$$\begin{aligned} u &= \frac{m_{11}X_w + m_{12}Y_w + m_{13}Z_w + m_{14}}{m_{31}X_w + m_{32}Y_w + m_{33}Z_w + m_{34}} \\ v &= \frac{m_{21}X_w + m_{22}Y_w + m_{23}Z_w + m_{24}}{m_{31}X_w + m_{32}Y_w + m_{33}Z_w + m_{34}} \end{aligned}$$

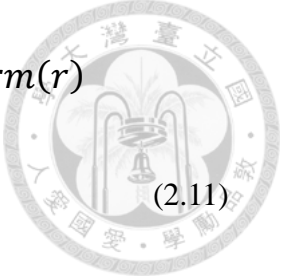
(2.9)

$M$  has 11 independent entries, so it needs at least 11 equations for computing  $M$ ,  
Namely we need at least 6 world-image point correspondences. Levenberg-Marquardt  
optimizer is used here to solve  $M$ . In order to extract  $R$ ,  $T$  from  $M$ , we decompose  $M$   
matrix and we get:

$$M = \begin{bmatrix} f_x r_{11} + u_0 r_{31} & f_x r_{12} + u_0 r_{32} & f_x r_{13} + u_0 r_{33} & f_x T_x + u_0 T_z \\ f_y r_{21} + v_0 r_{31} & f_y r_{22} + v_0 r_{32} & f_y r_{23} + v_0 r_{33} & f_y T_y + v_0 T_z \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}$$

(2.10)

However, even we now have rotation matrix  $R$ , to get  $R$ ,  $T$  we need to converts a  
rotation matrix to a rotation vector through:

$$\sin(\theta) \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix} = \frac{R - R^T}{2}, \quad \theta = \text{norm}(r) \quad (2.11)$$


Equation 2.11 is called the Rodriguez transform.

### 2.3 Homography Transform

Two views of the same 3-D plane are related by a homography [11] (also referred to as a planar transformation), which is represented by a 3\*3 matrix defined up to a scale factor. A homography matrix performs a point-to-point mapping between the homogeneous coordinates of the image points  $x'$  and  $x$ , such that  $x' = Hx$ . From [25] we know it has, at most, eight degrees of freedom. Therefore, estimating  $H$  requires at least four pairs of corresponding points; in the case of more than four correspondences, it can be estimated by least squares. Matrix form is express as following:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2.12)$$

Therefore we can get:

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}$$

$$y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}$$



(2.13)

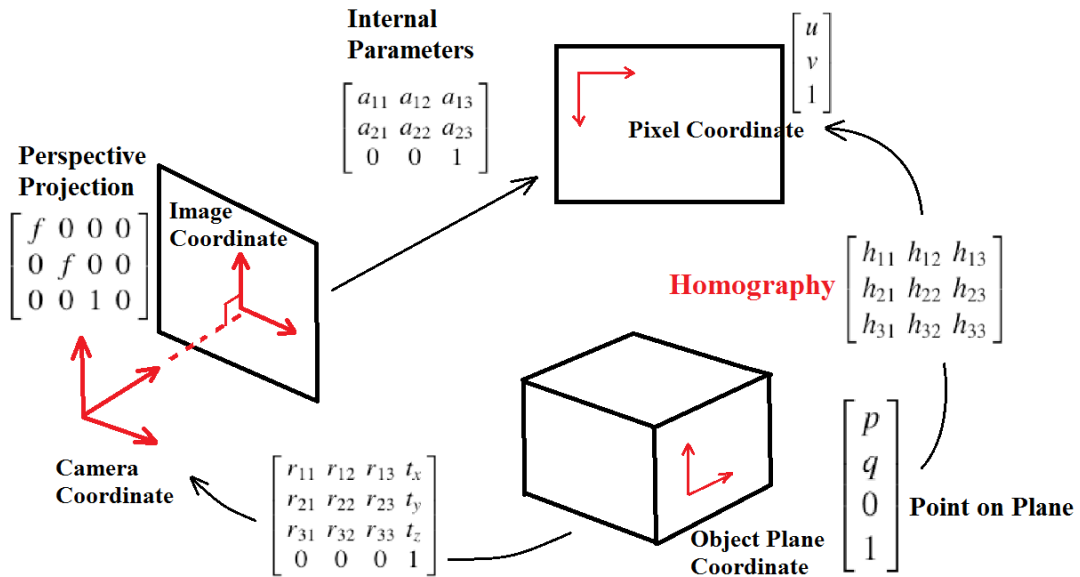


Figure 2.2 The projected relationship of Homography.

If we have enough pairs of corresponding points, Levenberg-Marquardt (LM) algorithm can be used to refine the Eq.2.14, Random sample consensus (RANSAC) algorithm will be used here for providing initial condition of nonlinear optimization.

$$\min(H) \sum \|Hx'_i - x_i\|^2$$

(2.14)



## 2.4 Lab color space

The  $L$ - $a$ - $b$  color space is a color opponent space with dimension  $L$  for lightness and  $a$ ,  $b$  for the color-opponent dimensions, based on a nonlinearly compressed CIE XYZ color space coordinate [2]. The nonlinear relations for  $L$ ,  $a$ , and  $b$  are intended to mimic the nonlinear response of the eye. Furthermore, uniform changes of components in the  $L$ - $a$ - $b$  color space aim to correspond to uniform changes in perceived color, so the relative perceptual differences between any two colors in  $L$ - $a$ - $b$  can be approximated by treating each color as a point in a three-dimensional space. An  $L$ - $a$ - $b$  color space creates a space which can be computed from the XYZ space, but is more perceptually uniform than XYZ. Perceptually uniform means that a change of the same amount in a color value should produce a change of about the same visual importance. Both  $L$ - $a$ - $b$  spaces are relative to the white point of the XYZ data they were converted from.  $L$ - $a$ - $b$  values do not define absolute colors unless the white point is also specified.

The solution to convert digital images from the RGB space to the  $L$ - $a$ - $b$  color space is given by the following formula.

$$L^* = 116f\left(\frac{Y}{Y_n}\right) - 16 \quad (2.15)$$

$$a^* = 500 \left[ f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right) \right] \quad (2.16)$$

20



$$b^* = 200 \left[ f\left(\frac{Y}{Y_n}\right) - f\left(\frac{z}{z_n}\right) \right] \quad (2.17)$$



$X, Y, Z, X_n, Y_n$  and  $Z_n$  are the coordinates of the CIEXYZ color space. The solution to convert digital images from the RGB space to the CIEXYZ color space is in the following formula.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.608 & 0.174 & 0.201 \\ 0.299 & 0.587 & 0.114 \\ 0 & 0.066 & 1.117 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2.18)$$

$X_n, Y_n$ , and  $Z_n$  are respectively corresponding to the white value of the parameter.

$$f(x) = \begin{cases} x^{\frac{1}{3}}, & x > 0.008856 \\ 7.787x + \frac{16}{116}, & x \leq 0.008856 \end{cases} \quad (2.19)$$

Color space conversion is the translation of the representation of a color from one basis to another. This typically occurs in the context of converting an image that is represented in one color space to another color space.

## Chapter 3 Vehicle Motion Model



### 3.1 Vehicle Dynamic Modeling

In order to simplify the modeling procedure, the following assumptions are considered must be pre-declared:

1. *The shape of body and the distribution of mass are symmetric about the  $Y_b$ - $Z_b$  plane.*
2. *The AUV has a small heave motion, so our model to be considered as planar motion which remains surge, sway, and yaw.*
3. *The center of mass and the center of buoyancy have same location in horizontal plane and has sufficiency static margin.*

To describe the motions of AUV, we use  $[X_w, Y_w, Z_w]^T$  to represent the world-fix coordinate system, and  $[x_b, y_b, z_b]^T$  indicate the body-fix coordinate system and the original point locates on the center of buoyancy. Figure 3.1 shows the coordinate and all related forces act on AUV. The thrust  $T$  with an angle  $\theta$  and moment reaction  $M$  acts on the hinge of propeller. Hydrodynamic effects of the AUV's body can be considered as drag  $f^D$  and lift forces  $f^L$ , which acting against and normal respectively to the AUV's

velocity  $\vec{V} = [u, v, w]^T$  direction through the center of buoyancy  $B$ .

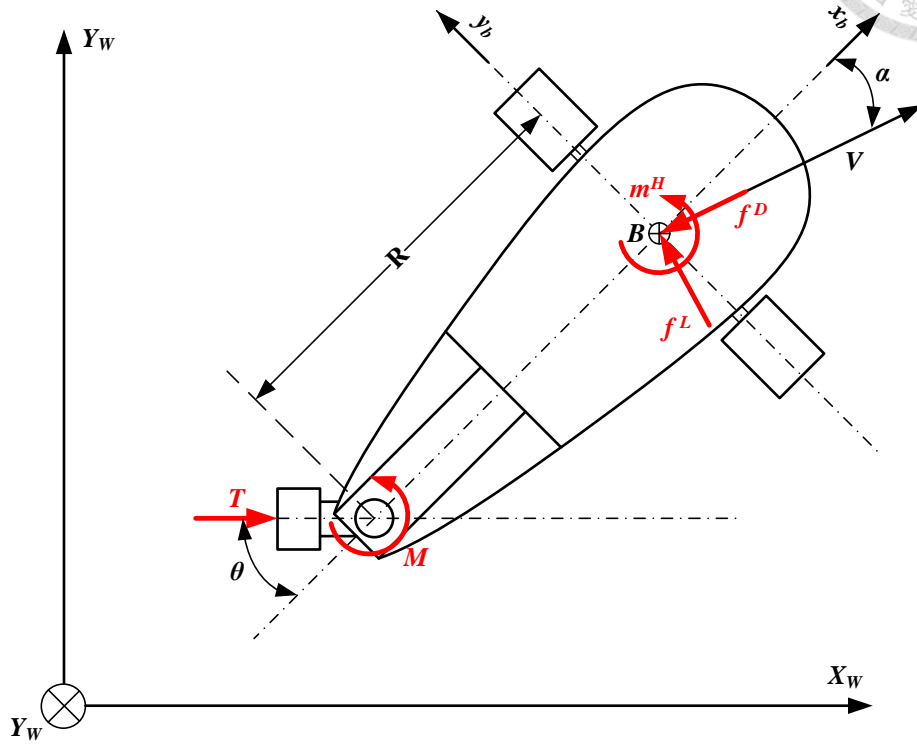


Figure 3.1 The reference frames of AUV with associated forces and moments.

In this article, we use Newton-Euler method to derive the dynamic model. First, expression of the Newton's second law of motion is written as:

$$\begin{aligned}
 F_E &= \frac{dB}{dt_I} \\
 H_E &= \frac{dK}{dt_I}
 \end{aligned}
 \tag{3.1}$$

where  $I$  indicates the inertial reference frame, here it equals to the world-fix

frame. The symbols  $B$  and  $K$  are the linear and the angular momentum,  $F_E$  and  $H_E$  are the sum of external body forces and moments acting on the AUV. In order to derive the momentum, suppose  $r_G$  is the vector of AUV's center of mass, so the absolute speed of center of mass  $V_G$  can be derived as follow:

$$V_G = V + \dot{r}_G + \omega \times r_G \quad (3.2)$$

$$V = ui + vj + \omega k \quad (3.3)$$

Here  $V$  is the absolute speed of the origin of the body-frame, which is shown in Fig.3.1. For most vehicles, the consuming rate of fuel also can be neglected, that is  $\dot{m}$  equal to zero. Thus the differentiation of linear momentum  $B$  becomes:

$$F_E = m \frac{d}{dt_I} (V + \dot{r}_G + \omega \times r_G) \quad (3.4)$$

If the vehicle does not equip any trimming weight system,  $r_G$  will not change with time. Therefore, differentiate each term of Eq. 3.4 and we will get the relationship between external force and the rigid body motion:

$$F_E = m \left( \dot{V} + \omega \times V + \dot{\omega} \times r_G + \omega \times (\omega \times r_G) \right) \quad (3.5)$$

Similarly, the rotating motion can be derived as:

$$H_E = mr_G \times (\dot{V} + \omega \times V) + I_O \dot{\omega} + \omega \times (I_O \omega) \quad (3.6)$$



where  $I_O$  is the inertia matrix:

$$I_O = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix}$$

Notice that the external force and moment can be decomposed as  $F_E = X_E i + Y_E j + Z_E k$  and  $H_E = K_E i + M_E j + N_E k$ , so we expanded Eq. 3.5 and Eq. 3.6 and combine both in concise way, then the six-degree-of-freedom rigid body motion is represented as follows.

$$M_{RB} \dot{v} = C(v)v + \tau_{RB} \quad (3.7)$$

Where the matrix  $M_{RB}$  is the inertial matrix and  $C(v)$  is the Coriolis and centripetal matrix.  $\tau_{RB}$  include all the external forces such as trust from propeller, restoring moment and hydrodynamic force. However, since we already restricts our motion model to the planar motion, the pitch, roll, and heave motion are all be considered to be zeros. Therefore, the restoring moment which is from the interaction between gravity and buoyancy is neglected. In addition, we select the origin of

body-frame fit to the center of buoyancy, so that  $r_G = [0 \ 0 \ 0]^T$  and  $I_O$  remains the principal axis component. Under these assumptions and combine the added mass effect, Eq. 3.7 reduces to [5]:

$$\begin{aligned}
 (m - X_{\dot{u}})\dot{u} &= (m - Y_{\dot{v}})vr + X_E \\
 (m - Y_{\dot{v}})\dot{v} &= -(m - X_{\dot{u}})ur + Y_E \\
 (J_z - N_{\dot{r}})\dot{r} &= (Y_{\dot{v}} - X_{\dot{u}})uv + N_E
 \end{aligned}
 \tag{3.8}$$

Where  $m$  is the mass of vehicle body,  $J_z$  is the inertia moment about the z-axis,  $X_{\dot{u}}$ ,  $Y_{\dot{v}}$  and  $N_{\dot{r}}$  represent the effect of added mass about the x-directions, y-directions and the rotating axis z-axis, usually,  $-X_{\dot{u}}$ ,  $-Y_{\dot{v}}$  and  $-N_{\dot{r}}$  are positive numbers. The hydrodynamic drag, lift and moment acting on the body are merged with thrust force into  $\tau_{RB}$ .

According to Fig. 3.1, the external force depicted from Eq. 3.8 can be expressed as following:

$$X_E = T \cos \theta - f^D \cos \alpha + f^L \sin \alpha \tag{3.9}$$

$$Y_E = T \sin \theta - f^D \sin \alpha - f^L \cos \alpha \tag{3.10}$$

$$N_E = -T\alpha \sin \theta + M + m^H \tag{3.11}$$

Here,  $\alpha$  is the attack angle of the vehicle, and refer to [4], the hydrodynamic damping terms are:

$$f^D = \frac{1}{2}\rho V^2 S C_D \quad (3.12)$$

$$f^L = \frac{1}{2}\rho V^2 S C_L \alpha \quad (3.13)$$

$$m^H = \frac{1}{2}\rho V^2 S (2R) \left[ C_{M\alpha} \alpha + \frac{2R}{V} C_{Mr} r \right] \quad (3.14)$$

Above equations involve several hydrodynamic damping coefficients, which are the coefficient of drag  $C_D$ , coefficient of lift  $C_L$ , coefficient of hydrodynamic restoring moment  $C_{M\alpha}$ , and coefficient of hydrodynamic viscous moment  $C_{Mr}$ . In addition,  $\rho$  is the mass density of water and  $S$  is the reference surface area of the vehicle. The reference surface area has several definitions in the literature, such as wet surface area or platform area. Naturally, different definition yields different values for the reference surface area. We also must note that the value  $V$  and  $\alpha$  are time-varying.

## 3.2 Parameters Estimation

### 3.2.1 Thrust Estimation

An experiment is set up for getting the relationship between thrust and voltage. A load scale was attached to the tested AUV to measure the thrust. We gave the AUV an

input value to make a forward motion, and motion was constrained in one direction.

The input value, which is a control value between -128 to 128, is proportional to voltage. Negative values mean reverse rotation which is not considered, so we just treat the positive control value 0-128. Table 3-1 shows the experiment results of thrust measurements.

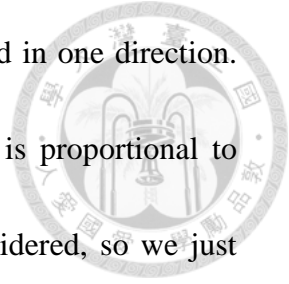


Table 3-1 Loading vs. voltage (control input) experiment results

Voltage Command ( $V_c$ )	Initial Loading (g)	Final Loading (g)	Difference (g)	Force (N)
15	2005	1860	145	1.421
15	2005	1900	105	1.029
15	2005	1890	115	1.127
25	2005	1760	245	2.401
25	2005	1720	285	2.793
25	2005	1710	295	2.891
35	2006	1480	526	5.1548
35	2006	1480	526	5.1548
35	2006	1550	456	4.4688
40	2006	1400	606	5.9388
50	2006	1040	966	9.4668
50	2006	1100	906	8.8788
60	2004	844	1160	11.368
60	2004	822	1182	11.5836
60	2004	912	1092	10.7016
80	2003	500	1503	14.7294
80	2003	480	1523	14.9254
80	2003	610	1393	13.6514
100 (max)	2003	601	1402	13.7396



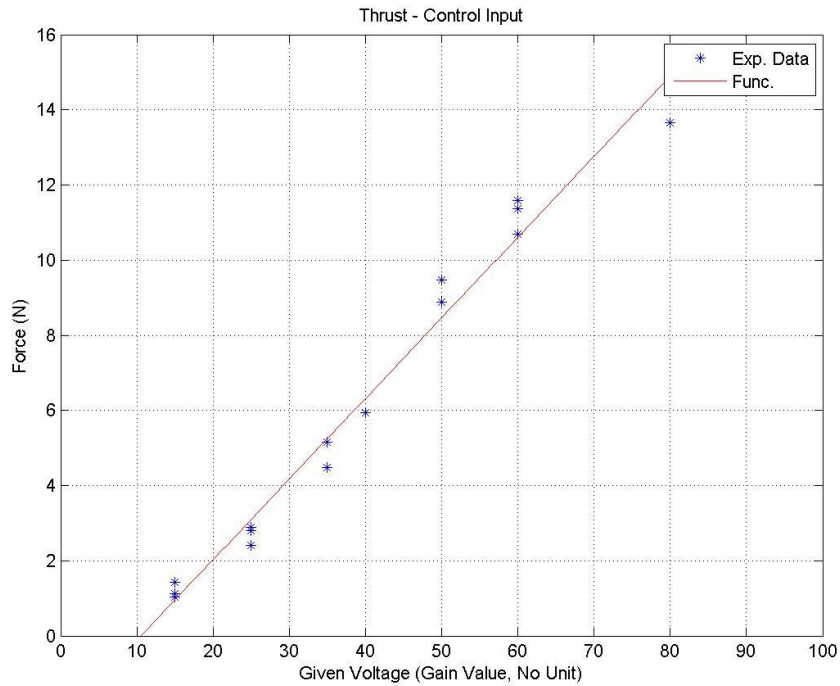


Figure 3.2 The relationship between force and voltage.

Through the experiment data, the  $V_c$  – thrust relationship can be obtained by linear regression. The equation is:

$$T(v) = 0.2146v - 2.2635 \quad (3.15)$$

### 3.2.2 Added Mass

When a solid body moved in fluid, it removed the surrounding fluid simultaneously. If the body accelerated, it also has to accelerate those surrounding fluid. Hence, the inertial force of those removed fluid can be seen as masses (or moments of inertia) that are added to the solid body. From the hydrodynamic theory, we can know an object

moving in boundless, ideal fluids, the containing kinetic energy is:



$$T = \frac{1}{2} \sum_{i=1}^6 \sum_{j=1}^6 \lambda_{ij} v_i v_j$$

(3.16)

where  $\lambda_{ij}$  is a positive definite matrix and has 6\*6 dimension:

$$\lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} & \lambda_{13} & \lambda_{14} & \lambda_{15} & \lambda_{16} \\ \lambda_{21} & \lambda_{22} & \lambda_{23} & \lambda_{24} & \lambda_{25} & \lambda_{26} \\ \lambda_{31} & \lambda_{32} & \lambda_{33} & \lambda_{34} & \lambda_{35} & \lambda_{36} \\ \lambda_{41} & \lambda_{42} & \lambda_{43} & \lambda_{44} & \lambda_{45} & \lambda_{46} \\ \lambda_{51} & \lambda_{52} & \lambda_{53} & \lambda_{54} & \lambda_{55} & \lambda_{56} \\ \lambda_{61} & \lambda_{62} & \lambda_{63} & \lambda_{64} & \lambda_{65} & \lambda_{66} \end{bmatrix}$$

(3.17)


By the potential flow theory, we can prove:

$$\lambda_{ij} = \lambda_{ji}$$

(3.18)

That means there are just 21 independent elements in the added mass matrix. Here the added mass is generalization, they contain different dimensions. When  $i, j = 1, 2, 3$ ,  $\lambda_{ij}$  is mass, if  $i, j = 4, 5, 6$ ,  $\lambda_{ij}$  is moments of inertia. The indexes '1', '2', '3' represent the translation in x, y, z axis, '4', '5', '6' represent the rotation with respect to x, y, z axis. For understanding how much masses we must consider additionally, we transfer the kinetic energy come with the removing fluid into the kinetic energy of equivalent masses.

Expand Eq. 3.19 we can get:



$$\begin{aligned}
 T = \frac{1}{2} [ & \lambda_{11}u^2 + \lambda_{22}v^2 + \lambda_{33}w^2 + 2\lambda_{12}uv + 2\lambda_{23}vw + 2\lambda_{13}wu \\
 & + \lambda_{44}p^2 + \lambda_{55}q^2 + \lambda_{66}r^2 + 2\lambda_{45}uv + 2\lambda_{56}vw + 2\lambda_{46}wu \\
 & + 2(\lambda_{14}u + \lambda_{24}v + \lambda_{34}w)p \\
 & + 2(\lambda_{15}u + \lambda_{15}v + \lambda_{15}w)q \\
 & + 2(\lambda_{16}u + \lambda_{16}v + \lambda_{16}w)r ]
 \end{aligned}$$

(3. 19)

From the flow disturbance, the momentum and kinetic energy of fluid have the following relation:

$$B_i = \frac{\partial T}{\partial V_i} \quad (i = 1, 2, \dots, 6)$$

(3. 20)

Notice that  $i = 1$  represents the motion in  $u$  direction, as  $V_2 = v, V_3 = w, V_4 = p, V_5 = q, V_6 = r$ . Substitute Eq. 3.19 into Eq. 3.20, we can get the expression of momentum in each direction.

$$\left\{ \begin{array}{l}
 B_1 = B_x = \frac{\partial T}{\partial u} \\
 B_2 = B_y = \frac{\partial T}{\partial v} \\
 \vdots \\
 B_6 = K_z = \frac{\partial T}{\partial r}
 \end{array} \right.$$

(3. 21)

The fluid inertia forces and torque on the vehicle are:

$$\begin{cases} F_I = -\frac{dB}{dt} \\ M_I = -\frac{dK}{dt} \end{cases}$$



(3. 22)

Simplifying our model into planar motion, we have that  $w = q = \dot{w} = \dot{q} = p = \dot{p} = 0$ . Experience shows  $\lambda_{26}$  is small and negligible, therefore, substitute Eq. 3.21 into Eq. 3.22, the fluid inertia forces projection on body-fix coordinate system for our vehicle is expressed as following:

$$\begin{cases} X_I = -\lambda_{11}\dot{u} + \lambda_{22}vr \\ Y_I = -\lambda_{22}\dot{v} - \lambda_{11}ur \\ N_I = -\lambda_{66}\dot{r} + (\lambda_{11} - \lambda_{22})uv \end{cases}$$

(3. 23)

Refer to Eq. 3.8, notice that  $\lambda_{11} = X_{\dot{u}}$ ,  $\lambda_{22} = Y_{\dot{v}}$ , and  $\lambda_{66} = N_{\dot{r}}$ . Added mass  $\lambda_{ij}$  of a solid body in fluid can be computed if the potentials of velocity fields is known which arise from the motion along and around coordinate axes. Therefore, to compute added masses, we need involve the Laplace equation problem with given boundary conditions. However, for an arbitrary body of revolution the solution is quite complicated [7]. In general, approximate methods of computation of added masses are adopted by modeling the vehicle in some simple bodies which could obtain the exact solutions of a Laplace equation. It is proper for modeling our vehicle to an ellipsoid

which is one of few cases can be analyzed explicitly [7]. By the simplification of body shape,  $\lambda_{ij}$  will become zero if  $i \neq j$ , and the diagonal terms  $\lambda_{11}$ ,  $\lambda_{22}$  and  $\lambda_{66}$  are:

$$\begin{aligned}\lambda_{11} &= \frac{4}{3}\pi\rho abc \frac{A_0}{2 - A_0} \\ \lambda_{22} &= \frac{4}{3}\pi\rho abc \frac{B_0}{2 - B_0} \\ \lambda_{66} &= \frac{4}{15}\pi\rho \frac{abc(a^2 - b^2)^2(B_0 - A_0)}{2(a^2 - b^2) + (A_0 - B_0)(a^2 + b^2)}\end{aligned}\tag{3. 24}$$

where  $A_0$ ,  $B_0$  are:

$$\begin{aligned}A_0 &= abc \int_0^\infty \frac{du}{(a^2 + u)\sqrt{(a^2 + u)(b^2 + u)(c^2 + u)}} \\ B_0 &= abc \int_0^\infty \frac{du}{(b^2 + u)\sqrt{(a^2 + u)(b^2 + u)(c^2 + u)}}\end{aligned}\tag{3. 25}$$

Eq. 3.25 is the complete elliptic integral of the second kind. In above formulas,  $a$ ,  $b$ ,  $c$  are the half-axes of the three-axial ellipsoid we approximate. Hence, considering the contour of our vehicle, we got  $A_0 = 0.2614$  and  $B_0 = 1.1674$  by using the software *Mathematica* for the integration.



### 3.2.3 Motion Data Gathering

This section introduces the motion data recording method using a free running test for identifying the vehicle's dynamic model. We focused on the drift motion; this is because most localization error comes from the inaccurate estimation of the yaw rate. Motion data gathering strategy is shown in Fig. 3.3, a landmark was arranged to guide the AUV and indicate the position where the AUV should begin turning motion. Several angles of the propeller jet were tested in our experiment. They would be constant during the rotation moment for a while until the speed had been stable. The AUV's motion were captured by the top-view camera. The camera has 4 m×2.4 m observable area on the AUV's running plane.

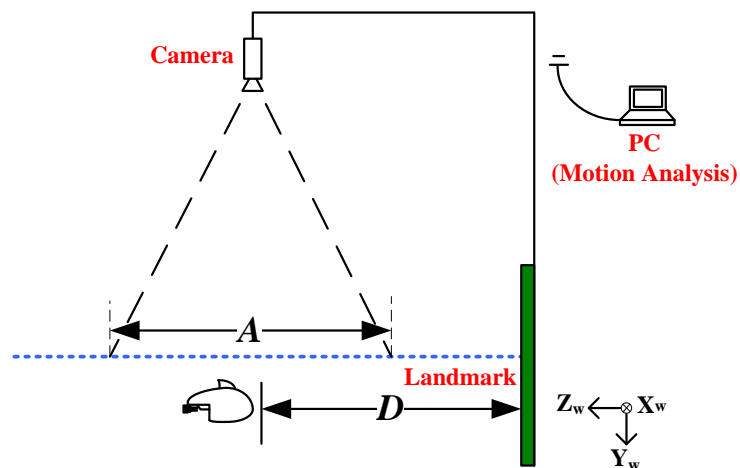


Figure 3.3 The free-running model test system. The distance  $D$  is the designed position where the AUV should begin the test motion.  $A$  is the observable area, and the blue dashed line represents the water surface.

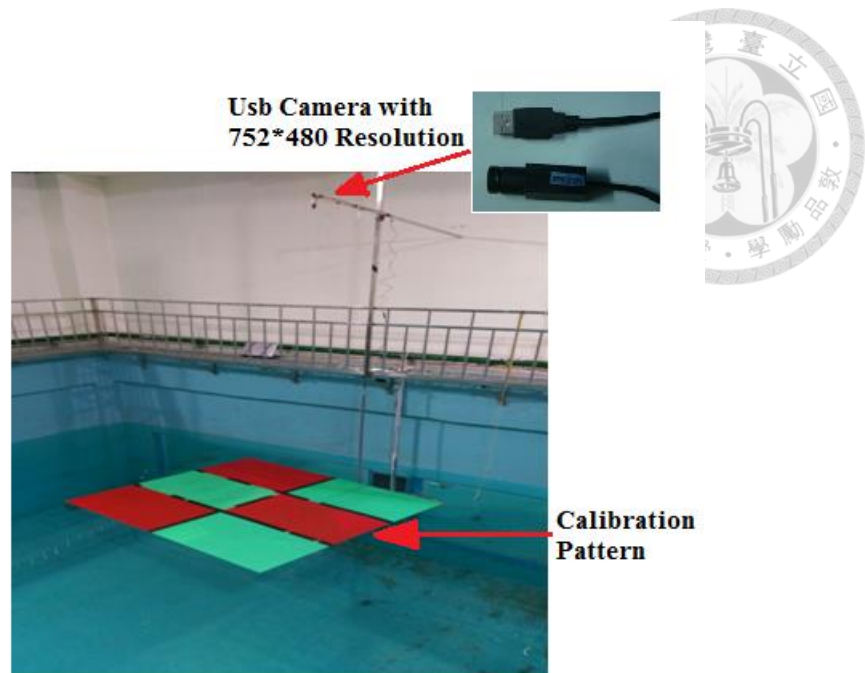


Figure 3.4 Experiment surrounding and the EM calibration pattern.

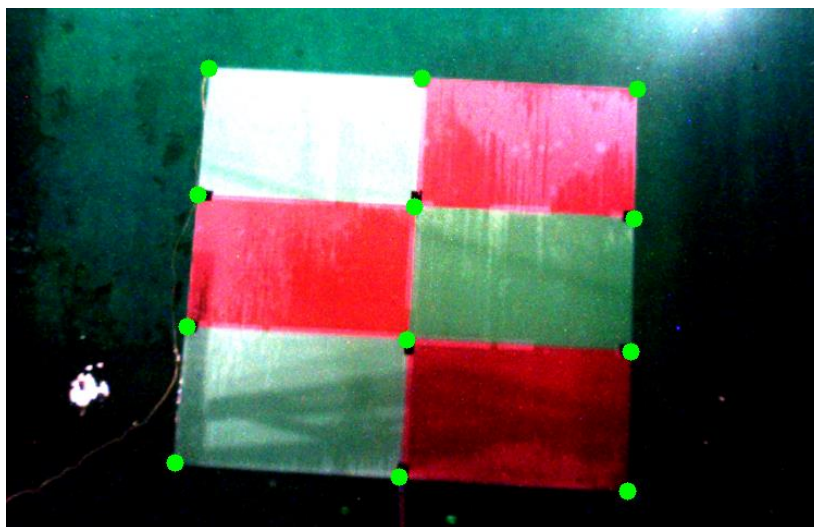


Figure 3.5 Calibration pattern captured by the top camera. Green circles are the reprojection of corners.

A set of calibration step is conducted to seek the camera's extrinsic parameters. We use a known dimension calibration pattern (Fig. 3.4) and put it at designed place where the world coordinate is known. Notice that we assume that the intrinsic parameter of the camera has already been well calibrated before the experiment. The estimation method of extrinsic parameters is introduced in Chapter 2.2. Once the extrinsic and intrinsic parameters are acquired, we could define the mapping relationship between image plane and any surface in the real world. In this test, the surface is where our vehicle operated on.

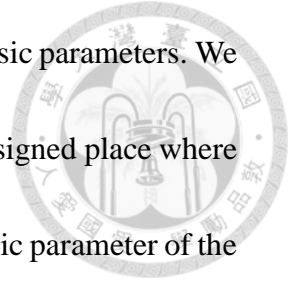


Table 3-2 EM Parameters in our experiment environment (T in cm;  $\theta$  in degree).

$T_x$	$T_y$	$T_z$	$\theta_x$	$\theta_y$	$\theta_z$
7.353762	-329.450	80.044741	84.476125	3.200259	-1.161266



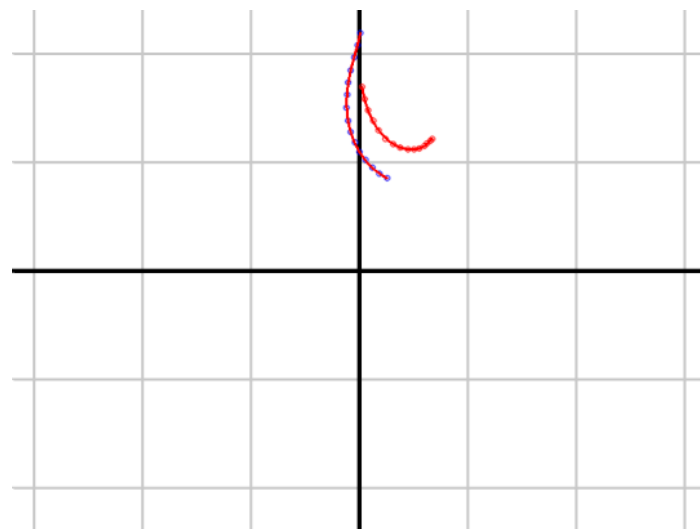
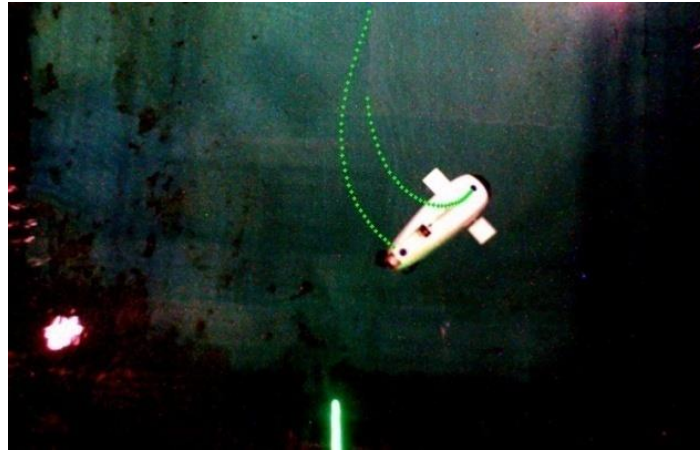
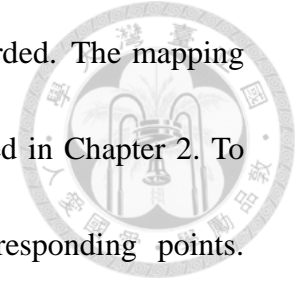


Figure 3.6 (a) The tracking of the two makers. The two trajectories of makers will be described as two polygon functions (b) The trajectories in global frame which through the HM mapping. The grid size is 1m\*1m.

In order to stably track the vehicle, we put two makers both in the front and rear side of our AUV. Numerous methods for tracking feature on video stream were developed. Here we adopt the iterative Lucas-Kanade [1] method for the tracking motion of our AUV. The frame rate of the experimental video is about 30, with the resolution in 752×480 pixels. Current time of each frame is also recorded during the

experiment. Raw 2D position data on image sequence were recorded. The mapping procedure is via the homography transformation, which is discussed in Chapter 2. To estimate the homography matrix, we need at least four corresponding points.



Corresponding pairs is easy to obtain if we get the EM parameters, which are listed in Table 3-2. We can find innumerable point pairs via the camera model projection. The HM is estimated by RANSAC method, and is listed as following:

$$H = \begin{bmatrix} 0.543193 & 0.031264 & -209.908227 \\ 0.033975 & -0.528892 & 211.217187 \\ 0.000032 & 0.000144 & 1.000000 \end{bmatrix} \quad (3.26)$$

These raw data are smoothed by the regression analysis. The n-orders time-varying polynomials described the trajectory  $x_t$  and  $y_t$  of vehicle, the instantaneous heading angle of vehicle  $h_t$  was obtained by calculating the relative angle of the two makers. Figure 3.6 shows an example of tracking the two makers and the reconstructed continued trajectory which was drawn by the polynomial function.

$$x_t = a_n t^n + \dots + a_1 t + a_0$$

$$y_t = b_n t^n + \dots + b_1 t + b_0$$

$$h_t = c_n t^n + \dots + c_1 t + c_0$$

(3.27)

The order  $n$  is arbitrary depends on the shape of trajectory. In our application, we used  $n=20$  to describe the trajectory. Therefore, by differential Eq. 3.27, we can get the velocity of vehicle in world space.



The velocity in world space will further be transferred into body frame. By tracking the two markers on the AUV from video, we also know the global heading of body. So the attack angle  $\alpha$  can be determined. Refer to fig.3.1, we decomposed global velocity  $V$  into surge and sway motion. The decomposed result are shown in Fig. 3.7 in various controlled thrust inputs. The relationship between the controlled gain value of voltage and thrust is described in Eq. 3.15. These motion data in body frame will be used in the next section.

Although using camera to observe the position of vehicle is convenient and intuitive, the uncertainty of AUV's position will increase with the distance between AUV and the camera. This is due to the quantization error. So in order to preserve the reliability of the motion data, the view of camera is better from the top view and testing area as near the camera as possible.

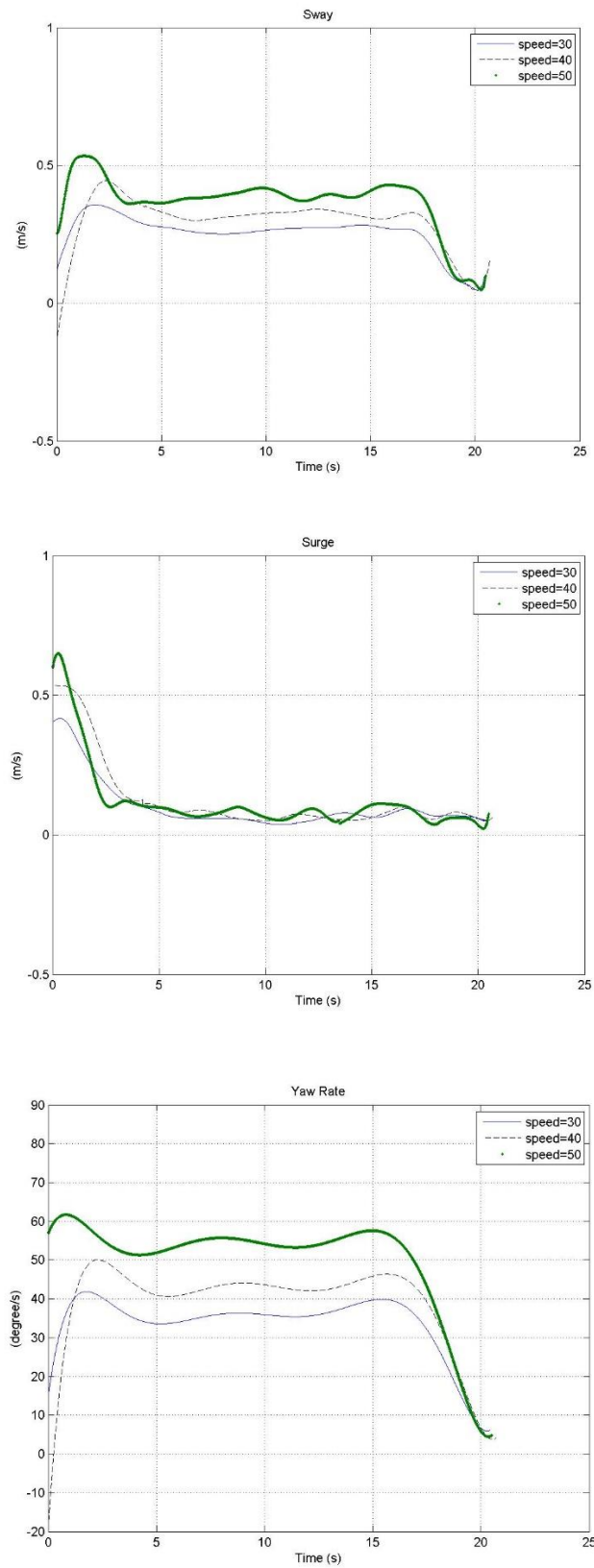


Figure 3.7 The (a) surge, (b) sway and (c) yaw rate during this test in three different speed input values.



### 3.2.4 Nonlinear Grey-Box Model Identification

The planar motion model of the AUV contains several states which couple to each other. It is a high-order nonlinear system. In this section, we estimate the parameters through a nonlinear grey-box identification algorithm [4] [30] based on the free-running testing. Nonlinear grey-box models are suitable for estimating parameters of systems that are described by nonlinear state-space structures in continuous or discrete time. The purpose of the nonlinear optimization is minimizing the prediction error  $\epsilon$  as:

$$\epsilon = \sum (S_{exp} - S_{pre})^2 \quad (3.28)$$

Where  $S_{exp}$  is the acquired experimental data, and  $S_{pre}$  represents the predicted result based on the testing model. The experimental data are the history of surge  $u(t)$ , sway  $v(t)$  and yaw rate  $r(t)$  of the vehicle, which have been shown in Fig.3.7.

To deal with the optimization problem, we utilize the nonlinear grey-box model object of the Matlab system identification toolbox. The testing model has to be represented as the first-order nonlinear difference or differential equations form represented as the following:



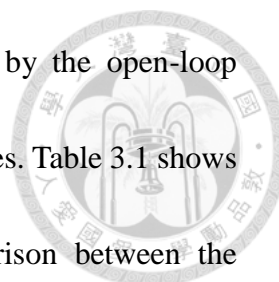
$$\begin{aligned}
 x(t) &= F(t, x(t), u(t), p_1, p_2, \dots, p_n) \\
 y(t) &= H(t, x(t), u(t), p_1, p_2, \dots, p_n) + e(t) \\
 x(0) &= x_0
 \end{aligned}$$

(3. 29)

The state vector  $x(t) = [u(t), v(t), \omega(t)]^T$  contains surge, sway and yaw rate. The argument  $u(t)$  is the input vector  $[T(t), M(t), \theta(t)]^T$  where are the thrust, the moment acting on the hinge and the angle of propeller, these control information of the vehicle would be recorded during whole experiment.  $p_1$  to  $p_n$  are those parameters needed to be determined. Refer to the model equations, we exclude those time-varying terms and combined the others, then we can define the parameters as following:

$$\begin{aligned}
 p_1 &= m - X_{\dot{u}} & p_5 &= \frac{1}{2} \rho S C_L \\
 p_2 &= m - Y_{\dot{v}} & p_6 &= \rho S R C_{M\alpha} \\
 p_3 &= J_z - N_{\dot{r}} & p_7 &= 2 \rho S R^2 C_{Mr} \\
 p_4 &= \frac{1}{2} \rho S C_D
 \end{aligned}$$

Note that nonlinear optimum always stop at local minimum. Hence the initial values of those parameters seriously affect the identification result. In order to simplify the task, first three parameters which describe the added mass effect are pre-estimated by consulting with the literature. Since the number of parameters without initial guess



had been reduced to four, we sought the remaining parameters by the open-loop simulation. Suitable set parameters were selected for the initial values. Table 3.1 shows the set of initial values and the estimated results. The comparison between the experimental data and model is shown in the Fig. 3.6. Fig. 3.7 shows comparison of simulation and practical trajectory of the AUV; both were fed with the same control inputs.

Table 3-3 Parameters of dynamic model

Parameter	$P_1$	$P_2$	$P_3$	$P_4$	$P_5$	$P_6$	$P_7$
Initial	24.691	72.937	2.3338	70.5	15.0	0.5	10.5
Estimation	24.691	72.937	2.3338	153.413	54.743	-14.831	24.07

### 3.3 Simulation

The simulation is conducted for the AUV to move in two paths. Two control inputs in serial were set to drive the AUV to execute an L-shaped and an S-shaped trajectory. The rotating speed of tail during the control was considered in the simulation. The swing speed of the stern propeller was 62.4 (deg/s). Three speed input of propeller were tested here, which are 30, 40, and 50. From eq.3.15, we can know it equal to 4.17, 6.32 and 8.47 (N).

### 3.3.1 Test 1: L-Shaped Path

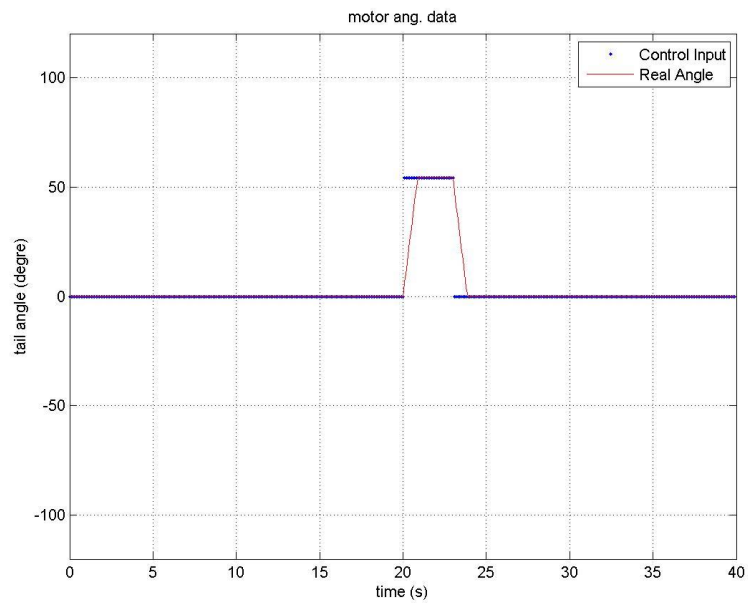
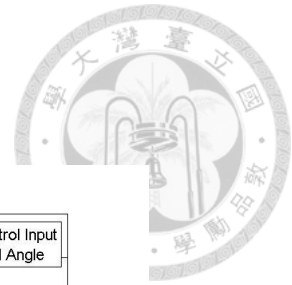


Figure 3.8 The motor input value for dynamic model simulation. For our system, the motor input contains the direction and speed of the propeller.

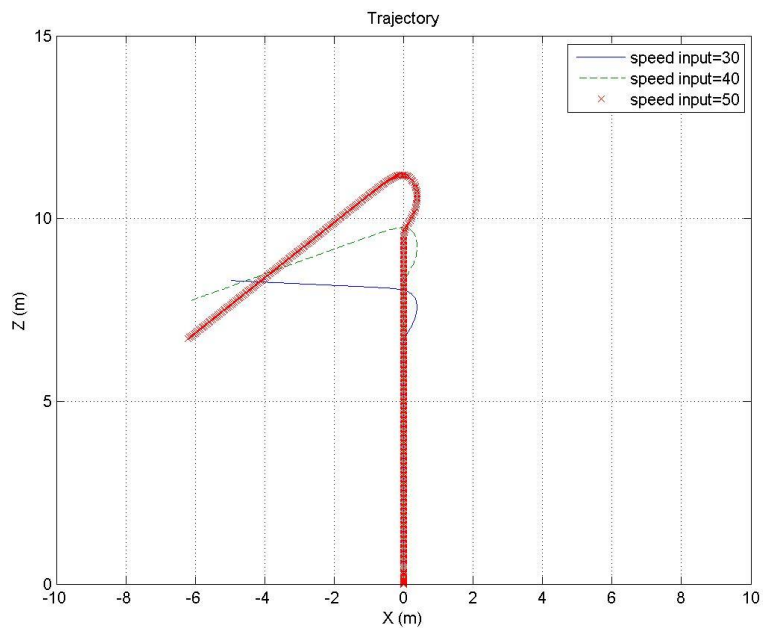


Figure 3.9 The simulated L-shaped trajectory.



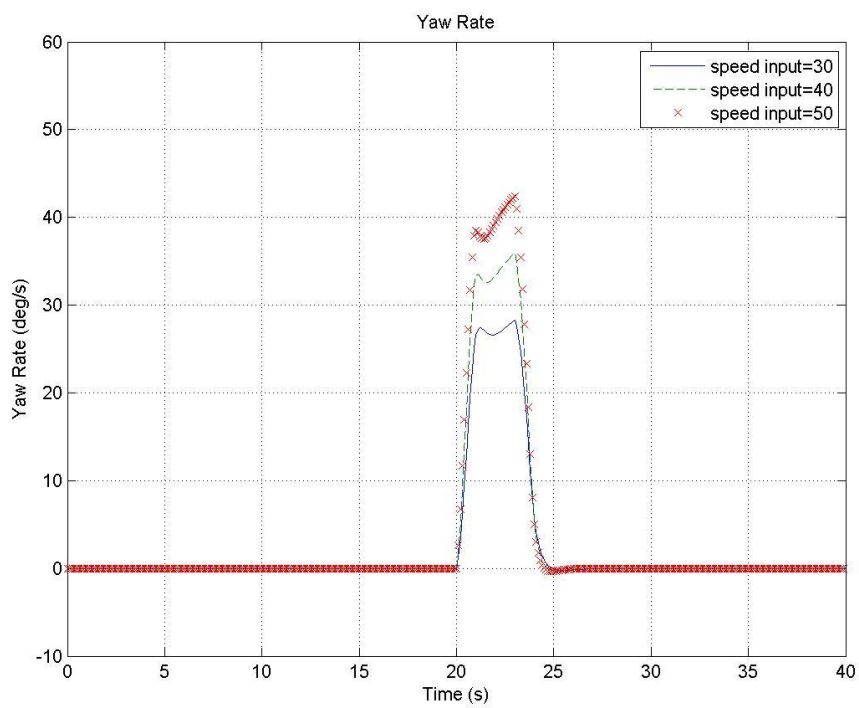
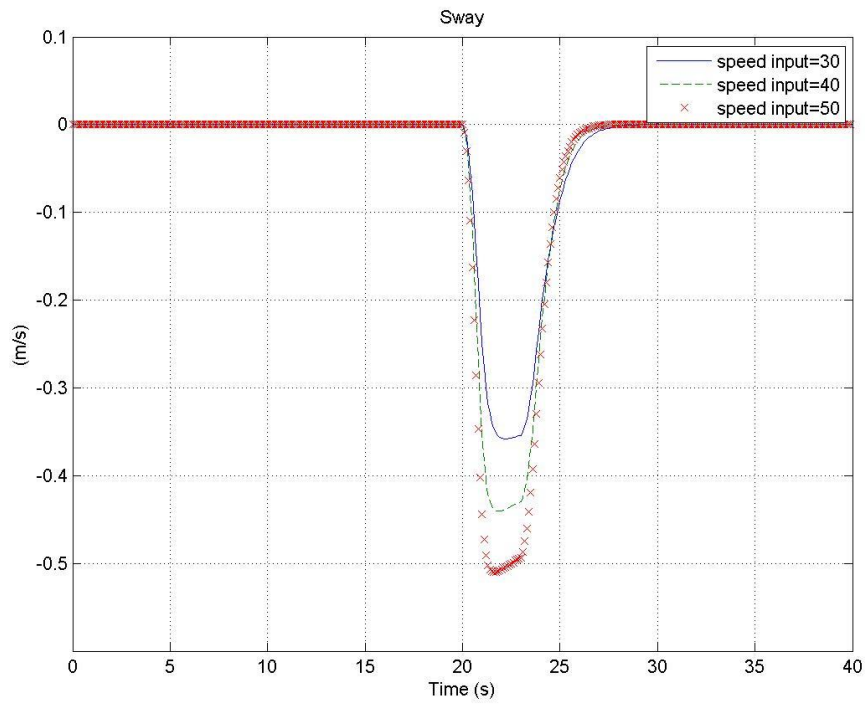


Figure 3.10 The change of (upper) sway and yaw rate in this simulation.

### 3.3.2 Test 2: S-Shaped Path

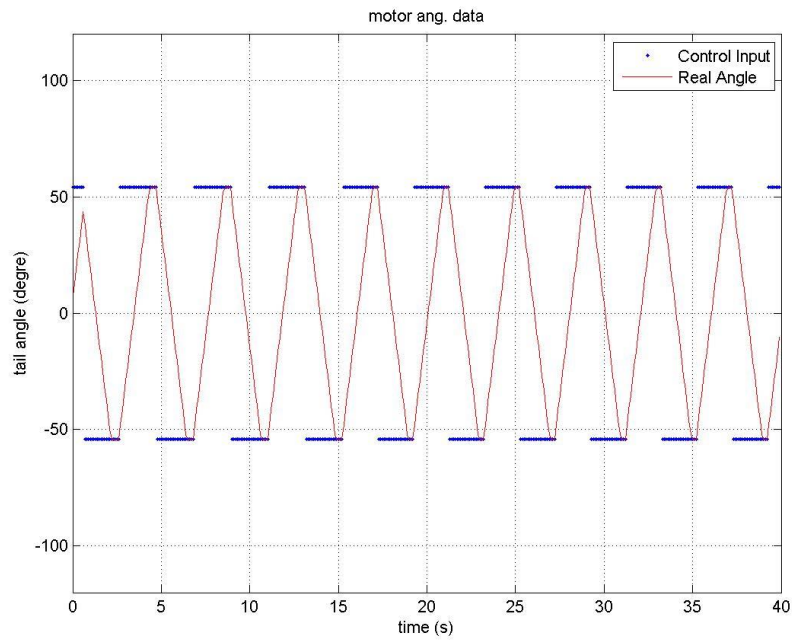
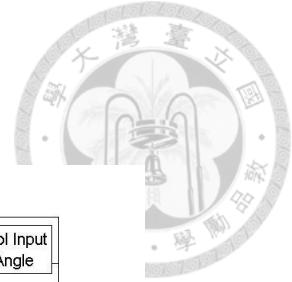


Figure 3.11 The motor input value for S-shaped trajectory.

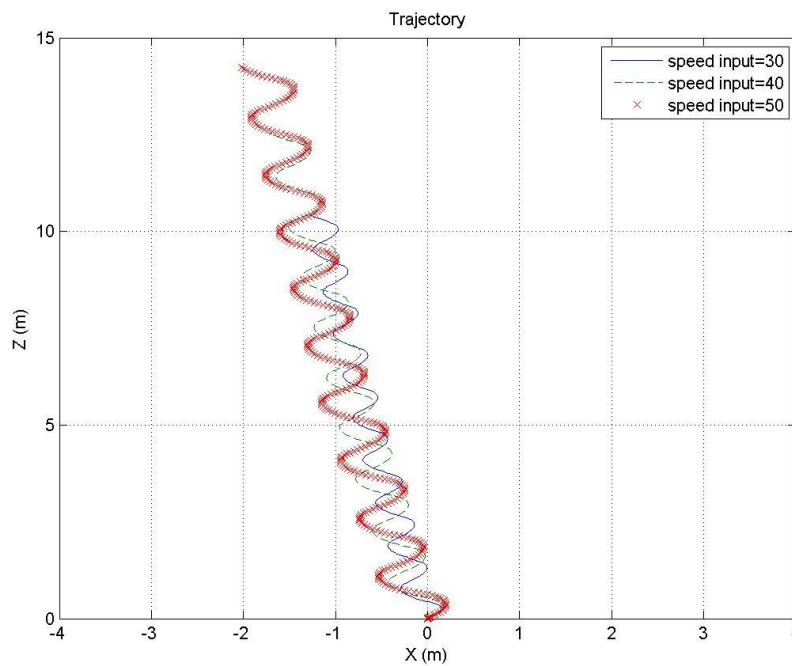


Figure 3.12 The simulation result of S-shaped trajectory.

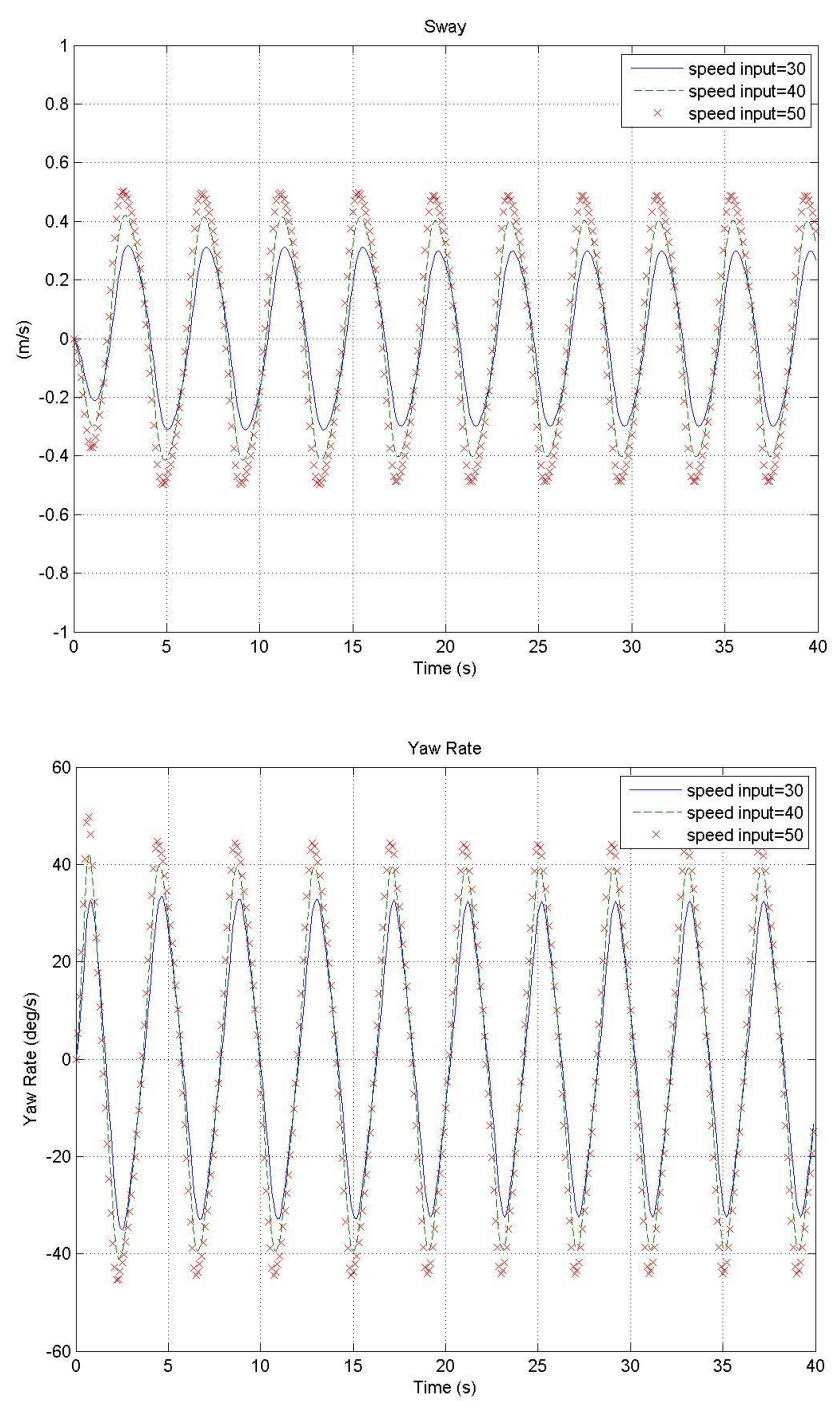


Figure 3.13 The change of (upper) sway and yaw rate in this simulation.

### 3.3.3 Comparison to the experimental data



Fig. 3.14 to Fig. 3.16 present comparisons between the experimental data used for identification of parameters, which was mention in Chapter 3.2.3. Simulations run using the identified parameters in Table 3-3. Also, the resulting trajectories out of three speed inputs of propeller are compared, and the swing angle of tail in all cases were set at position=30000, which means  $\theta$  (in Fig. 3.1) is 54 degree.

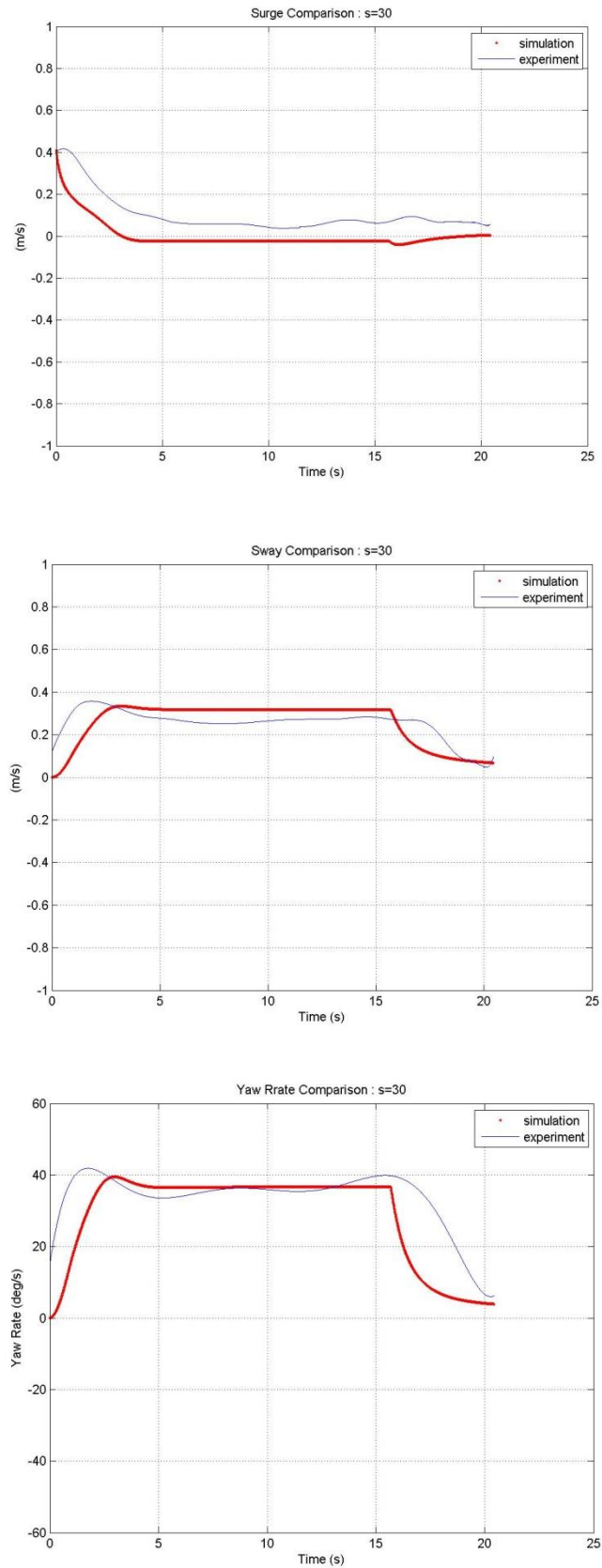


Figure 3.14 Comparison of surge, sway and yaw rate at speed inupt 30.

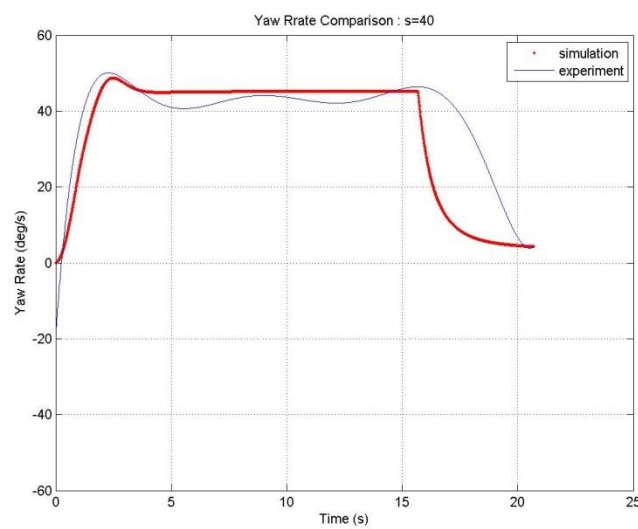
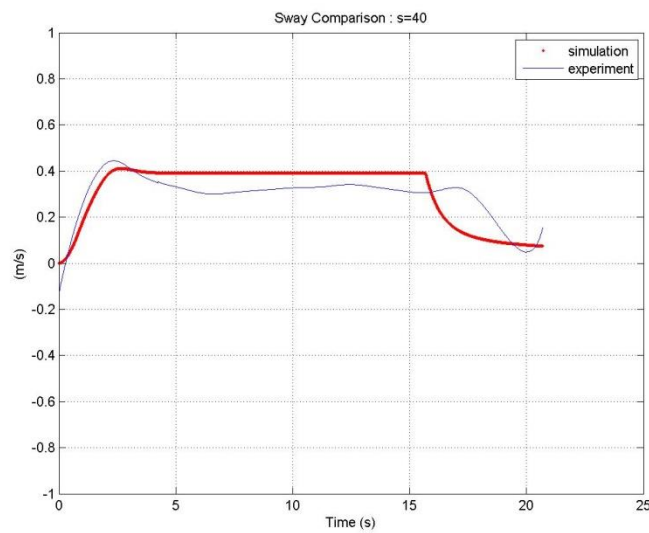
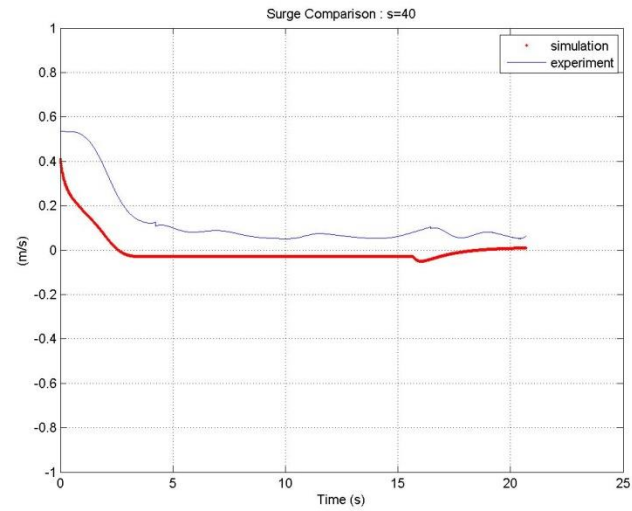


Figure 3.15 Comparison of surge, sway and yaw rate at speed inupt 40.

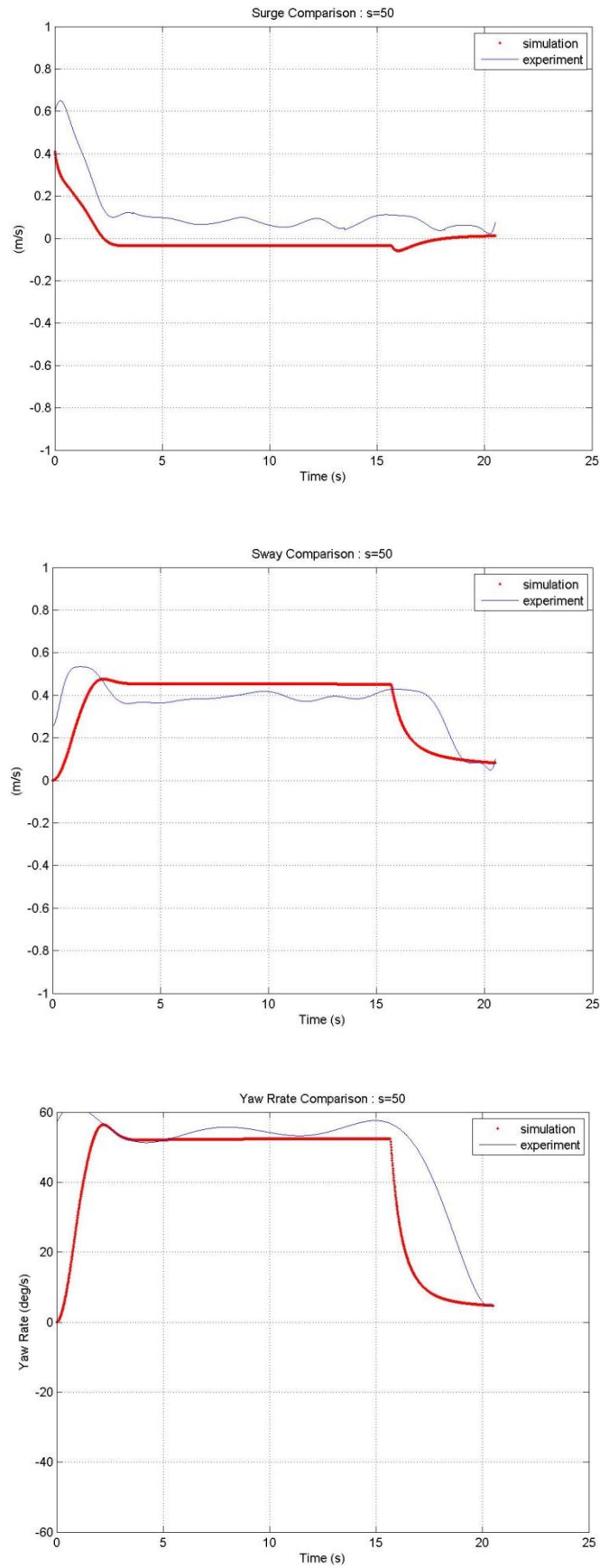


Figure 3.16 Comparison of surge, sway and yaw rate at speed inupt 50.

## Chapter 4 Pose Tracking



### 4.1 Particles Filter Algorithm

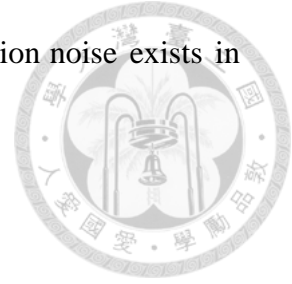
Particles filter represents the belief of state by a set of random samples. Particles can accommodate arbitrary noise distribution. In the re-sampling step, uncertainty of state estimation will be reduced. However, the re-sampling process needs to refer the result of weighting, so a precise perceptual model is critical. Particles filter is constituted with Monte Carlo sampling and recursive Bayesian filter, the main formula of recursive Bayesian filter is as follows:

$$Bel(x_t) = \eta p(z_t|x_t) \int p(x_t|x_{t-1}, u_t) Bel(x_{t-1}) dx_{t-1} \quad (4.1)$$

Here  $Bel(x_{t-1})$  is the belief which represents pervious state of the vehicle, the notation  $z_t$  is measurement of the vehicle,  $u_t$  is the control input, and the conditional probability distribution  $p(x_t|x_{t-1}, u_t)$  represents motion model of the vehicle. The likelihood  $p(z_t|x_t)$  plays an important role in the PF. In this work, we calculate  $p(z_t|x_t)$  using the AR technique to measure the similarity. The flow char of the PF localization is shown in Fig 4.1. For each particle,  $p(x_t|x_{t-1}, u_t)$  is a Markov random process. Since it represents each particle's motion model, the Gaussian noise



was introduced in the state transferring step. In our work, the motion noise exists in yaw rate  $\dot{r}$  and linear velocity  $V$ .



$$\begin{pmatrix} \hat{V} \\ \hat{\dot{r}} \end{pmatrix} = \begin{pmatrix} V \\ \dot{r} \end{pmatrix} + \begin{pmatrix} \varepsilon_{\alpha_1|V|+\alpha_2|\dot{r}|} \\ \varepsilon_{\alpha_3|V|+\alpha_4|\dot{r}|} \end{pmatrix} \quad (4.2)$$

where  $\hat{V}$  and  $\hat{\dot{r}}$  represent the motion in reality,  $\varepsilon_b$  is a zero-mean error variable with standard deviation  $b$ . Figure 4.2 illustrates the simulation of the motion model derived in Chapter 3 with noises. We can see due to the noise, the uncertainty of vehicle's pose will increase dramatically for long distance travelling.

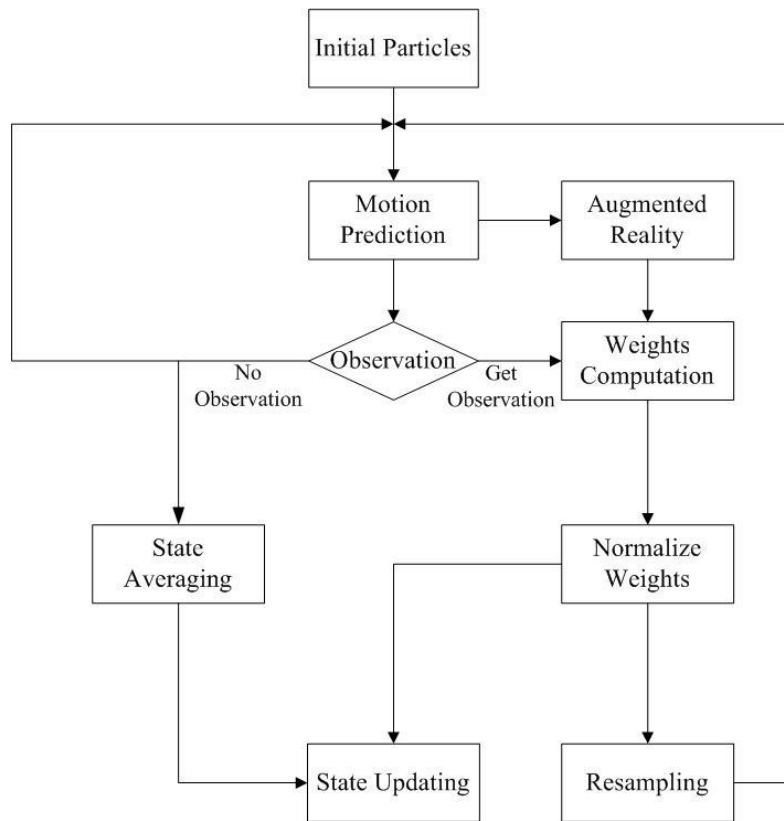


Figure 4.1 The process of particles filter localization

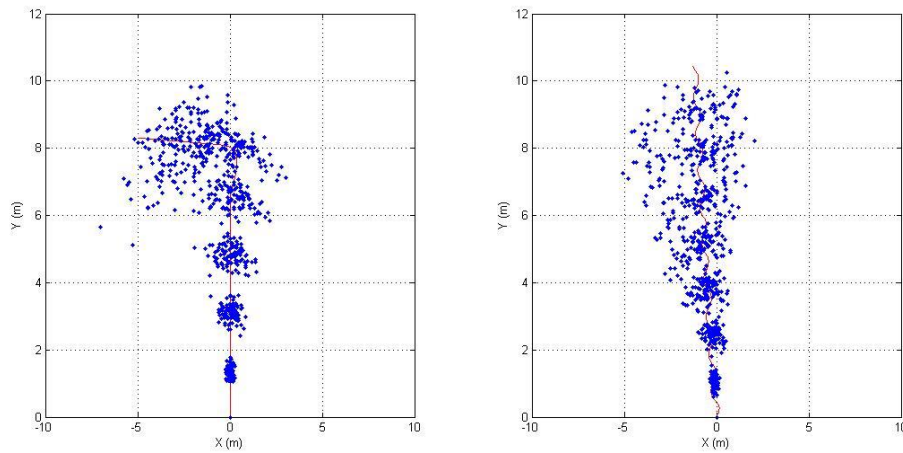


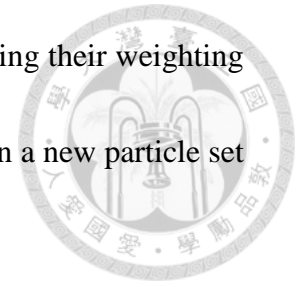
Figure 4.2 Sampled densities obtained for incremental movement of an AUV motion. Path are L-shpaed (left) and S-shpaed (right) which same as the simulation in section 3.3.

Refer to Fig. 4.1, if the robot didn't observe any features, the means of the predicted distribution will be the same as the distribution of the current state. Once observations are obtained, the particle which has the maximum likelihood will represent the current state. The weight of each particle is updated by the Eq. 4.3:

$$w_t^{[i]} = w_{t-1}^{[i]} \cdot p(z_t | x_t^{[i]}) \quad (4.3)$$

After a few iterations, the degeneration problem of the weights will arise. Some of them will have negligible value. Therefore in the re-sampling step, we eliminate the particles which has weighting less than 0.01. Also, the remained space from elimination will be

shared to the survived particles. The number of allotment is according their weighting score. Final step is normalizing the weight again, and we will obtain a new particle set which represents the new probability distribution of belief.



## 4.2 Measurement Comparing Template

Illuminants are suitable for guiding an underwater vehicle if there is large visibility. Water-proof LEDs of green color light were applied as the landmarks. To get the landmark's position from the robot's view, color segmentation process was utilized here. We adopted  $L^*a^*b^*$  color space here. In practice, the color of LED light is not invariable, when the robot close to the light, the brightness of the light's center will become white color in robot's view. This may cause misdetection of LED's position. To increase the accuracy of detection, after the green area was extracted by the color segmentation, we search the maximum brightness point within the area, which results in the detecting accuracy to be in pixel resolution.

In order to speed up the comparison process, after the positions of landmarks are obtained, the detecting result was transferred into one-dimension histogram, which is shown in Fig.4.4. Considering measurement error, the possible position of landmark is modeled to have the Gaussian distribution. Measurement error may come from camera calibration or color segmentation. This histogram will be used for calculating the

likelihood which is addressed in the next section.

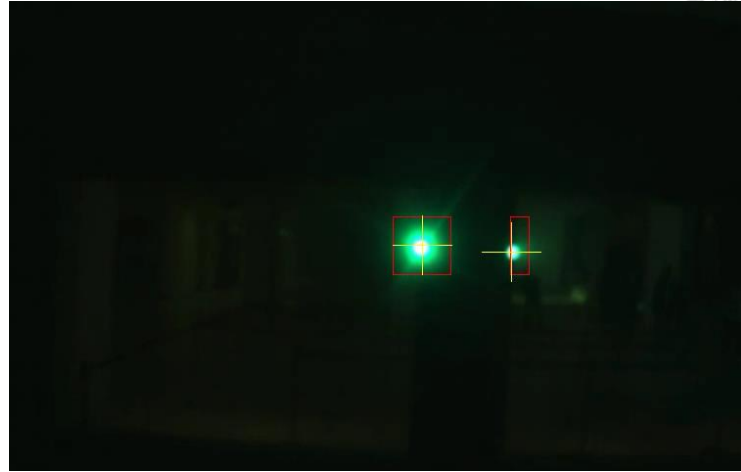


Figure 4.3 The landmark extraction. Red rectangular represent first step - the color segmentation. After that, the center of LED will search in the region, which is represented by yellow cross.

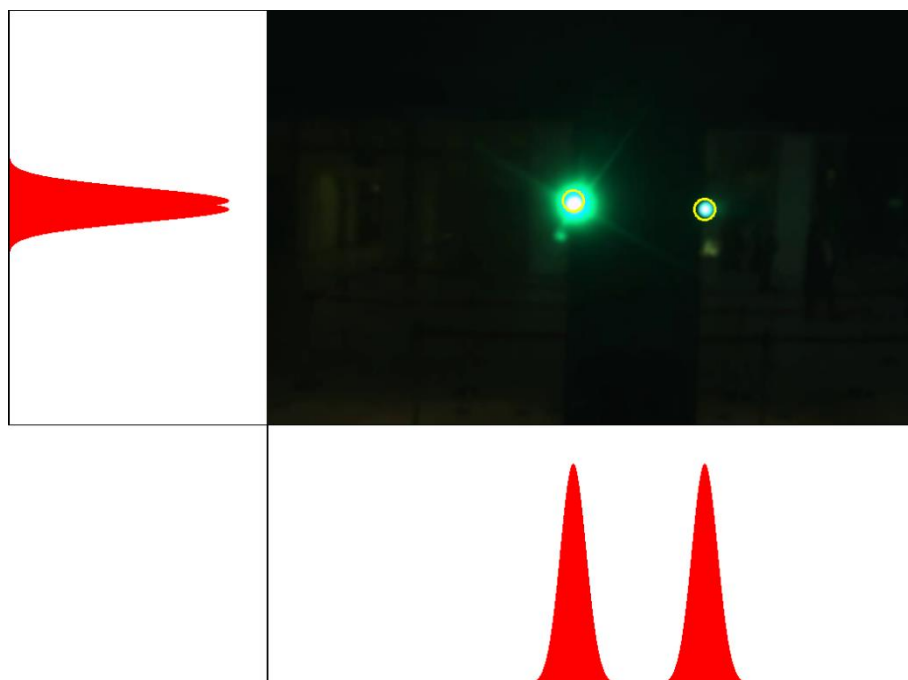
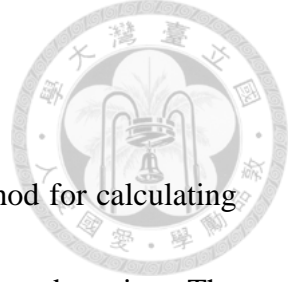


Figure 4.4 The Landmark detecting result and its corresponding histogram.



### 4.3 Observation Model

In this section, we will present an effective measurement method for calculating the weight of each particle which uses the monocular camera for observing. The observation model in particles filter algorithm is used to calculate the importance factor of each particle. The importance factor is obtained as quotient as following:

$$\eta p(z_t | x_t^{[i]}, m) = \frac{\eta p(z_t | x_t^{[i]}, m) p(x_t^{[i]} | x_{t-1}^{[i]}, u_t) bel(x_{t-1}^{[i]})}{p(x_t^{[i]} | x_{t-1}^{[i]}, u_t) bel(x_{t-1}^{[i]})} \quad (4.4)$$

Here the notation  $i$  indicate the  $i$ -th particle. Since  $\eta$  is a constant, Eq. 4.3 is proportional to the importance factor. The numerator and the denominator are also called the target distribution and the proposal distribution. The importance factor accounts for the mismatch between these two distributions.

To calculate the likelihood  $p(z_t | x_t^{[i]}, m)$ , here we use registration technique which is the key issue of augmented reality. The registration problem amounts to tracking the camera pose in a world frame. The term  $m$  represents the map. Here it stores the 3-D world coordinate of all the landmarks. For general augmented reality application, the pose information are assumed known to form the extrinsic matrix.

Back to calculation of likelihood, we re-project all the landmarks according to the

state  $bel(x_{t-1}^{[i]})$  of particle which implied an extrinsic matrix  $[R|t]$ . Because the multi-hypothesis of body poses, not all the landmarks will be projected to the virtual image plane. This re-projection result represents the prediction of measurement of each state of  $i$ -th particle. At any time step, we process this prediction for  $n$  times to weight all the particles, where  $n$  is the amount of particles.

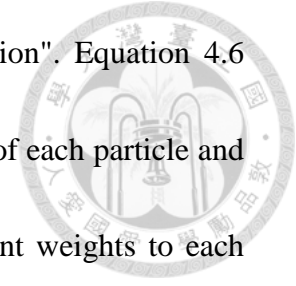
Suppose the position of landmark in map is the  $p_f$ . Depend on the hypothesis of pose, we use the  $i$ -th extrinsic matrix is  $[R|t]^{[i]}$ , then according to camera projection which has been discussed in Chapter 2:

$$I_p^{[i]} = K [R|t]^{[i]} p_f \quad (4.5)$$

$I_p^{[i]}$  is the imaged view of the  $i$ -th particle. It means that if the robot is located at that position, the expected appearance it will be seeing. Same as the observation, which transfer the landmark detecting result into 1-D histogram, the predicted image  $I_p$  was also transferred the 2-D landmark's position into the histogram form. We also use the Gaussian distribution to model it for correlation. Here we defined  $H(I_p^{[i]})$  represents the combination of two of the histograms in X and Y axis. Assume that we have the observation image  $I_o$  from camera. Therefore the weight is accounted by following equation:

$$w^{[i]} = p(z_t | x_t^{[i]}, m) = NCC(H(I_p^{[i]}), H(I_o)) \quad (4.6)$$

The function  $NCC$  is the well known "Normalized Cross Correlation". Equation 4.6 calculates the similarity between predicted measurement  $H(I_p^{[i]})$  of each particle and real measurement  $H(I_o)$ . Through this equation we assign different weights to each particle.



#### 4.4 Summary

The sequential Monte Carlo method (SMC), also called particles filter (PF) is utilized for estimating the global pose of AUV. In general, particle filters can accommodate arbitrary sensor characteristics, motion dynamics, and noise distributions. However, in practice, real-time application always restricted by its heavy computer loading, or we must decrease the number of samples. In this chapter we present a pose tracking method without 3D reconstruction procedure (like stereo vision or structure from motion algorithm), and do not need any complex object reorganization or data association issue, which often appear in SLAM problem. The measurement method is quite simple but useful and reliable. Benefitting from the multi-hypothesis of PF, temporary mismatch of landmark can be overcome, PF will track all the hypothesis for seeking the best state.

## Chapter 5 Experiment



Our pose tracking method was tested using the robot Iron Fish (IF) which is an autonomous underwater vehicle. IF equip with a digital compass, a pressure gauge and two cameras, see Fig. 5.1. The AUV is equipped with the highly-efficient industrial computer, among which CPU adopts Intel Core i7-2710QE, RAM has 8 gigabytes. Basic heading and depth control were accomplished by PID controller, and the PID algorithm is given by

$$u_{PID} = k_p e + k_i \int e + k_d \dot{e} \quad (5.1)$$

where,  $u_{PID}$  is the output of the PID controller,  $k_p$  is the proportional gain,  $k_i$  is the integral gain,  $k_d$  is the derivative gain and  $e$  is the error signal.

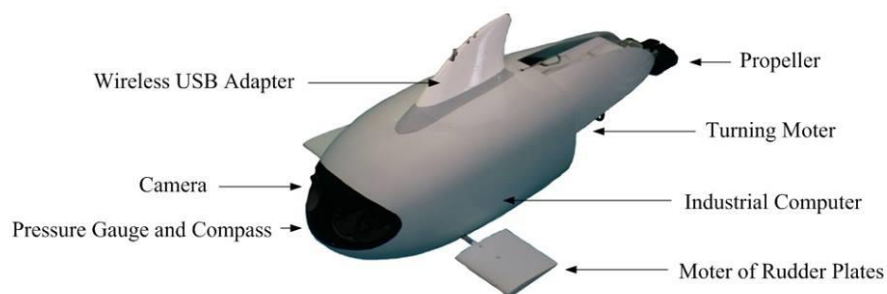


Figure 5.1 The underwater robot IF and its peripheral equipment.



## 5.1 Experiment 1 – The Comparison of EKF and PF



In this section, an experiment was introduced that verified the tracking algorithm which is developed in this work. The testing environment is shown in Fig.5.2. Two landmarks (buoy balls) are located at the position:  $x=5\text{m}$ ,  $y=10\text{m}$  and  $x=10\text{m}$ ,  $y=10\text{m}$ . During the experiment, the AUV tracked the two landmarks by the vision system. Motor input data are recorded within entire trip. Beside PF, we also implemented extend Kalman filter (EKF) localization algorithm.

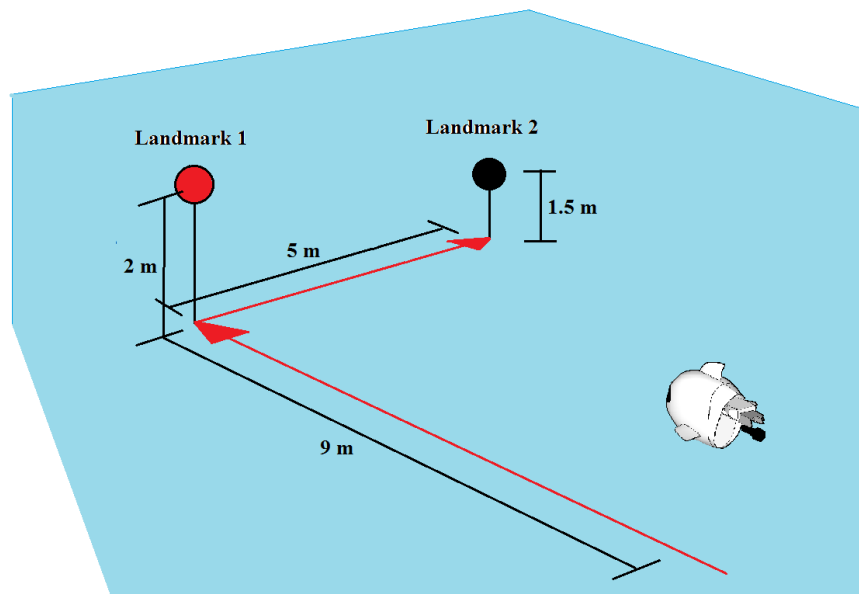


Figure 5.2 Stage arrangement of the experiment 1. While turning, two buoys are used for guiding the AUV.



### 5.1.1 Extended Kalman Filter

Table 5.1 illustrates the EKF localization algorithm [22]. Here we provide the initial guess for both algorithms. We show that the PF algorithm has the global localization ability, on the other hand, EKF algorithm cannot track without given initial guess. Another dissimilarity is that in order to provide the inverse measurement for EKF localization, we use stereo vision method to get the depth information, then estimate the relative pose between the landmark and AUV.

The EKF tracking result is shown by the red dot “x” trajectory in Fig.5.3. It shows that in the beginning of the tracking, the AUV is only running in dead-reckoning mode and kept diving until the target appeared in the view of camera. This is due to the field of view of camera was not able to cover the target. Once the target appears in the view of the camera, the measurement model started to update, and the outputted the best estimation of the vehicle pose. Additionally, in the Fig.5.4, black lines present Gaussian estimate of the AUV state at time  $t-1$ , with  $X_{t-1}$ . Blue lines present Gaussian estimate of the AUV state at time  $t-1$ , with x-position and y-position covariance  $\Sigma_{t-1}$ .



Table 5-1 Extended Kalman filter localization algorithm.

<p><b>Input</b> (<math>X_t, \Sigma_t, M_t</math>)</p> $J_{x_t} = \begin{bmatrix} 1 & 0 & (-\sin \theta V_z + \cos \theta V_x) \Delta t \\ 0 & 1 & (\cos \theta V_z + \sin \theta V_x) \Delta t \\ 0 & 0 & 1 \end{bmatrix}$ $J_{M_t} = \begin{bmatrix} \sin \theta \Delta t & \cos \theta \Delta t & 0 \\ -\cos \theta \Delta t & \sin \theta \Delta t & 0 \\ 0 & 0 & \Delta t \end{bmatrix}$ $\Sigma_{M_t} = \begin{bmatrix} (\alpha_1  V_x  + \alpha_2  V_z  + \alpha_3  \dot{\theta} )^2 & 0 & 0 \\ 0 & (\alpha_4  V_x  + \alpha_5  V_z  + \alpha_6  \dot{\theta} )^2 & 0 \\ 0 & 0 & (\alpha_7  V_x  + \alpha_8  V_z  + \alpha_9  \dot{\theta} )^2 \end{bmatrix}$ $\bar{X}_t = \begin{bmatrix} x_t + (\cos \theta V_z + \sin \theta V_x) \cdot \Delta t \\ z_t + (\sin \theta V_z - \cos \theta V_x) \cdot \Delta t \\ \theta + \dot{\theta} \Delta t \end{bmatrix}$ $\bar{\Sigma}_t = J_{x_t} \Sigma_t J_{x_t}^T + J_{M_t} \Sigma_{M_t} J_{M_t}^T$ $H_t = \frac{\partial h_t}{\partial X_t} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $\Sigma_{H_t} = H_t \bar{\Sigma}_t H_t^T$ $K_t = \bar{\Sigma}_t [H_t]^T [\Sigma_{H_t}]^{-1}$ $\bar{X}_t = \bar{X}_t + K_t e_t$ $\bar{\Sigma}_t = (I - K_t H_t) \bar{\Sigma}_t$ <p>Return, <math>\bar{X}_t, \bar{\Sigma}_t</math></p>
--

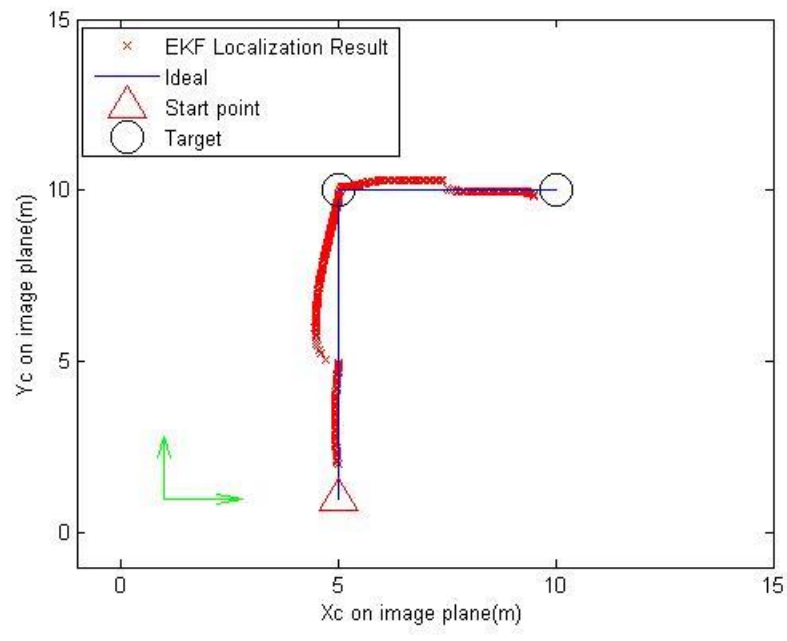


Figure 5.3 Result of EKF localization

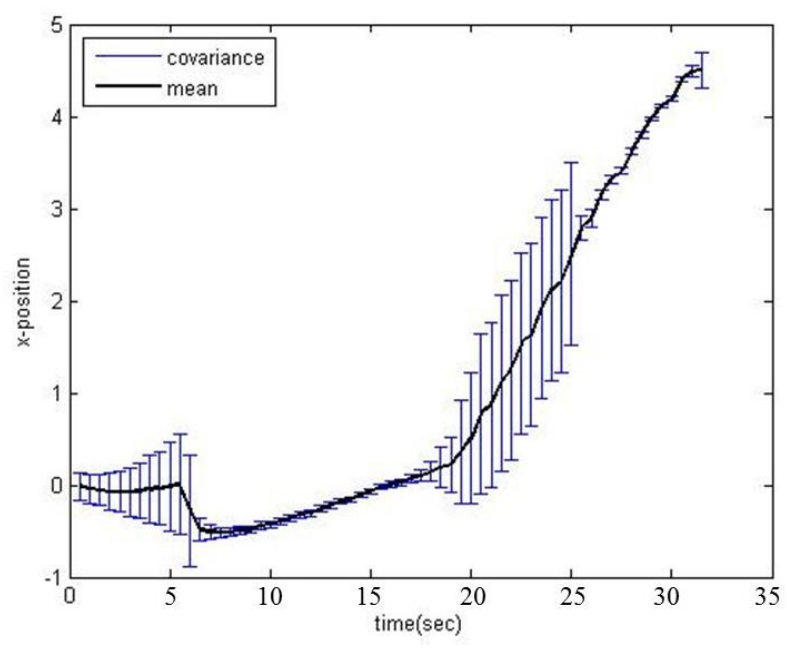


Figure 5.4 x-position change of covariance with time

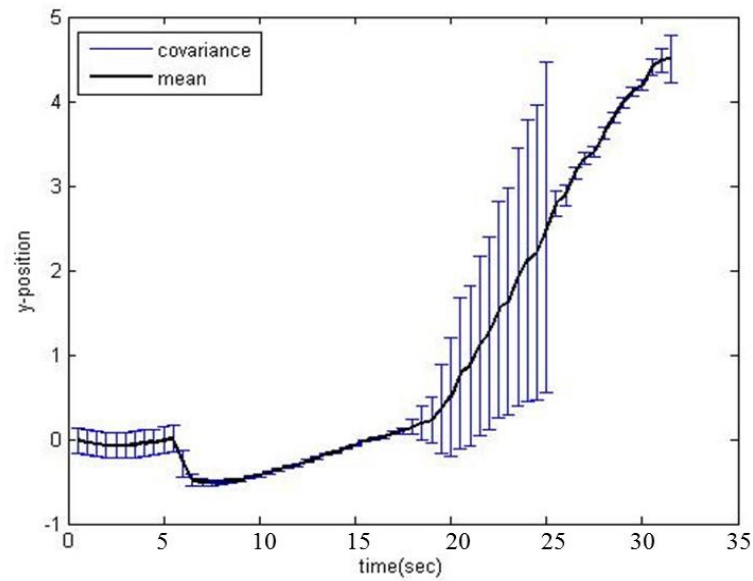


Figure 5.5 y-position change of covariance with time

## 5.1.2 Particle Filter Localization

PF localization method was tested using the same environment. Fig.5.6 shows in the beginning of the tracking, also, because of the camera of the AUV unable to observe the target, the AUV predicted its pose by motion model until the target appeared in the view of the camera. And once the target appears in the view of the camera, the update step will be involved. AUV kept the target in the field of view until it decided to execute next task (tracking another buoy). Fig.5.7 shows that maximum weight of all samples were changed. We can see the target was contacted at about  $t=5$  sec, and the huge vibration was due to the re-sampling procedure.

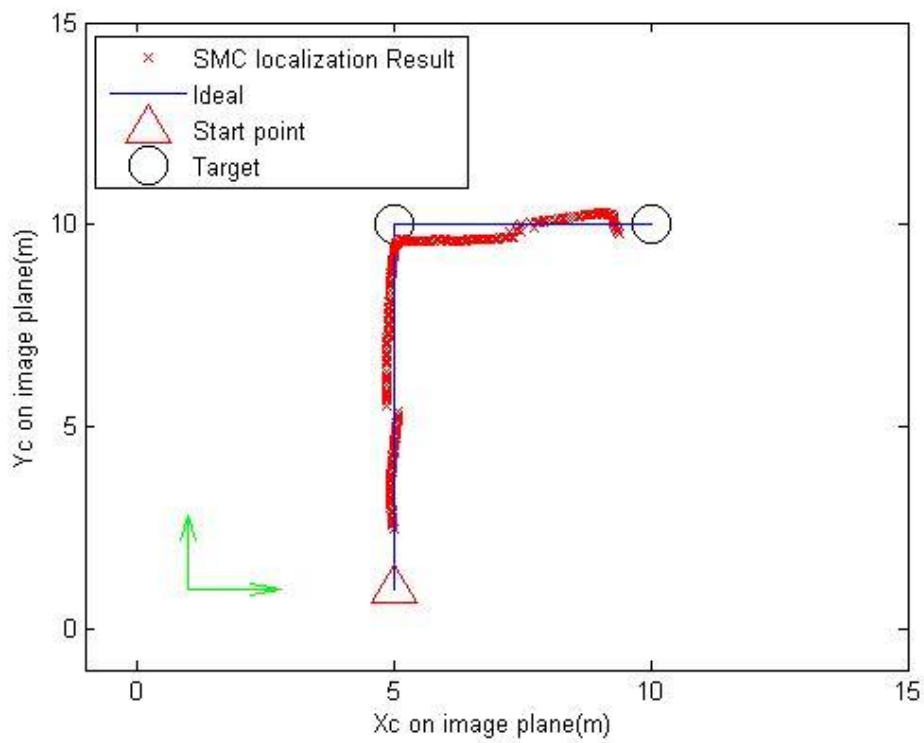


Figure 5.6 Result of SMC localization

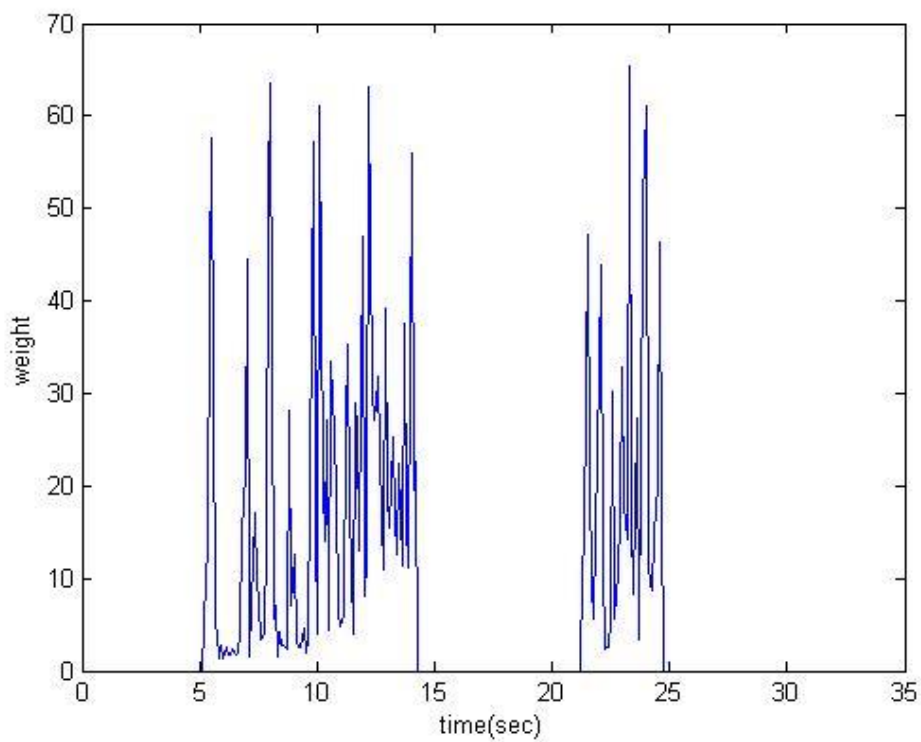


Figure 5.7 Changes of sample weight with time

## 5.2 Testing in NMMST's Tank



To test the tracking, the AUV was commanded to do numerous round trips to follow the reference path as shown in Fig. 5.9. The path is quite simple; however, at the end of the straight line path, the AUV needs to execute a 180 degrees turn. Motion model shows high speed turning motion is a challenge for tracking. The testing took 1.5 hours to verify the stability of the tracking system. We recorded the video and data while the AUV did regularly performance in the NMMST water tank for evaluating system performance. Shape of the tank is rectangular with 17.16 meters length, 6.67 meters width and 3.00 meters depth of water. Two green LED lights are mounted each side of the tank to be the landmark, the testing environment is shown in Fig.5.8. In this experiment, we used 200 particles for sampling. The start position of the path was given. The complete trajectories of the position included the X, Z, yaw angle are shown in Fig. 5.10.

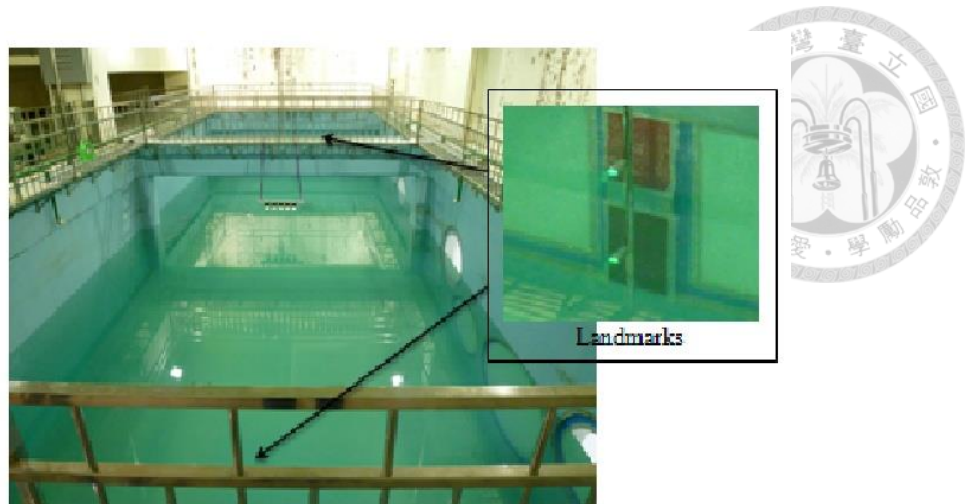


Figure 5.8 The environment of the testing water tank in National Museum of Marine Science & Technology, Keelung, Taiwan

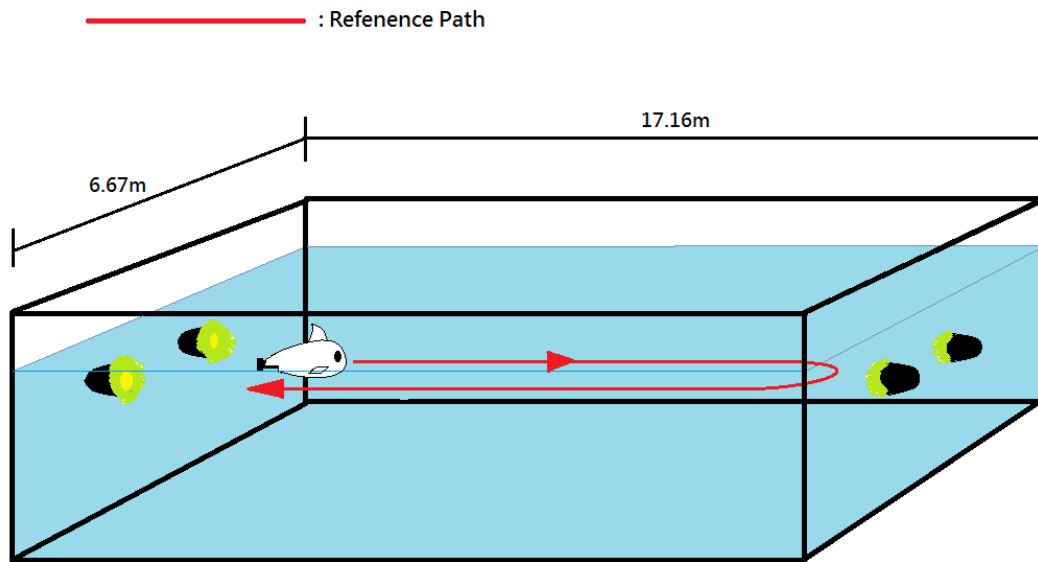


Figure 5.9 The long cruise testing environment, red line is the reference path for one trip, two landmarks are mounted each side. The initial position of AUV was located at the middle of two landmark and the AUV must keep 1m distance to the side walls.



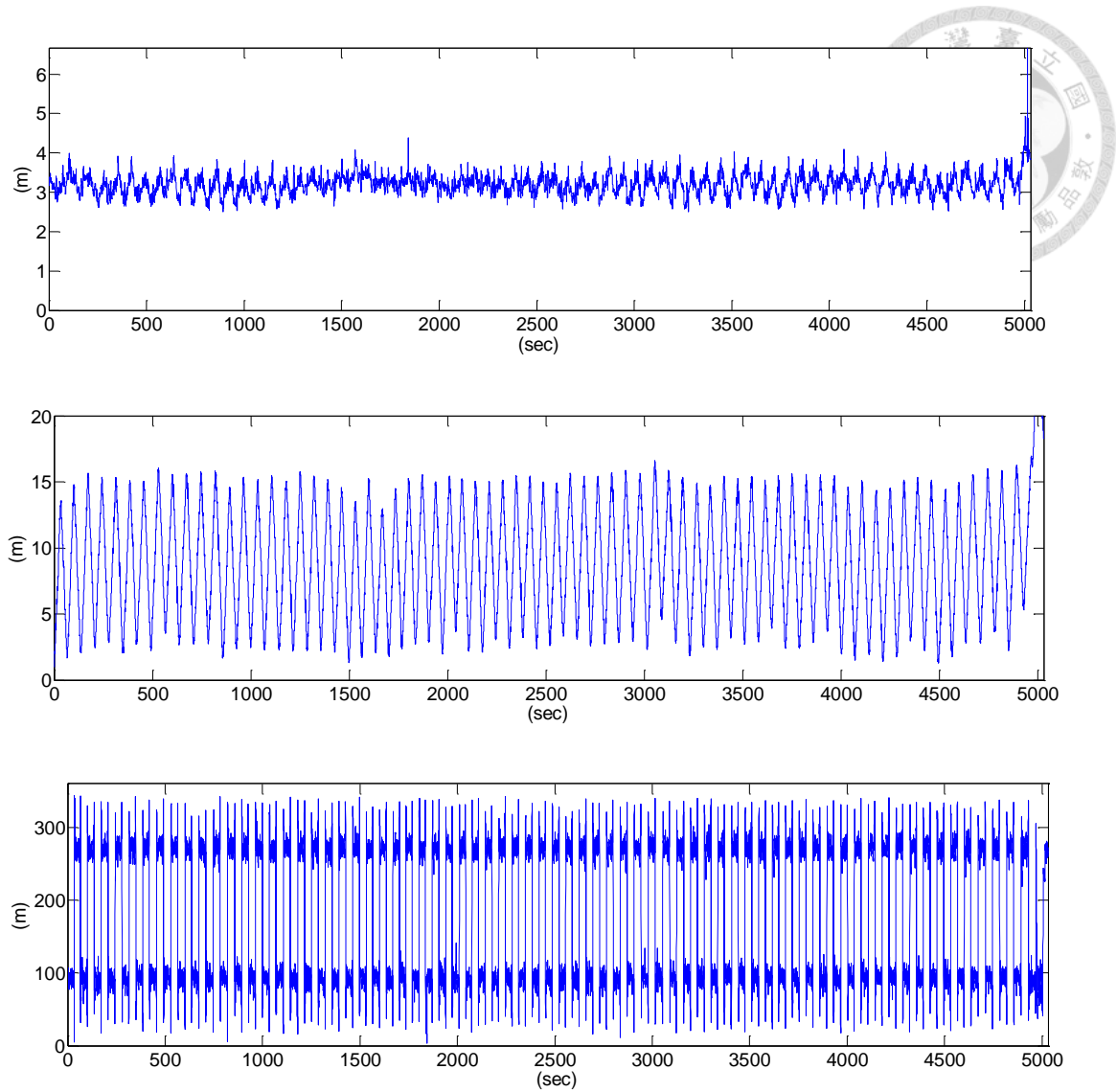


Figure 5.10 The tracking result of the cruise testing. (a) X-psition tracking result.

(b) Y-position tracking result. (c) Yaw angle tracking result.

### 5.3 The Kidnapped Problem

In addition to the tracking capability, we also tested the global localization capability. The robot was suffering an abrupt movement to another position; this

situation is called the *kidnapped robot problem*. In order to let our AUV “notice” that it is getting lost, we need a magnitude value to indicate uncertainty of position during the state update process. Here we introduce the information entropy for determining the situation that AUV needs to be “woke-up” or not. Entropy was devised in the late 1940s by Claude Shannon [31] when he invented information theory (then known as communication theory).

$$H_{(X)} = - \sum_{i=1}^n P(m_i) \log_2 P(m_i) \quad (5.2)$$

The Eq. 5.2 is same as the entropy in thermodynamics and that why it called entropy. From the equation, we know once the probability of events is closest to each other, the information entropy will be the maximum. It means the system has highest uncertainty now. Related to particles filter, the probability of event equals to the normalized weight of each particle, so we could use entropy for measuring the uncertainty of system. If the AUV was conscious that it needs a re-localization, the number of particles will re-distribute, but this time they will uniformly distribute on all map, no longer depend on the last state. Fig.5.11 shows the re-localization result and Fig.5.12 shows the uncertainty during the re-localization experiment.

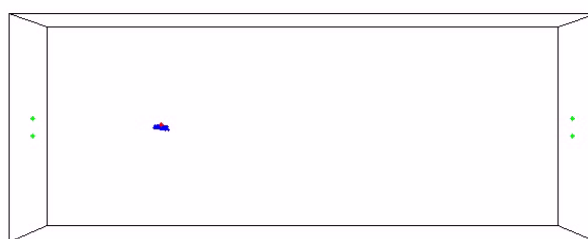


Figure 5.11 Re-localization for the kidnap problem.

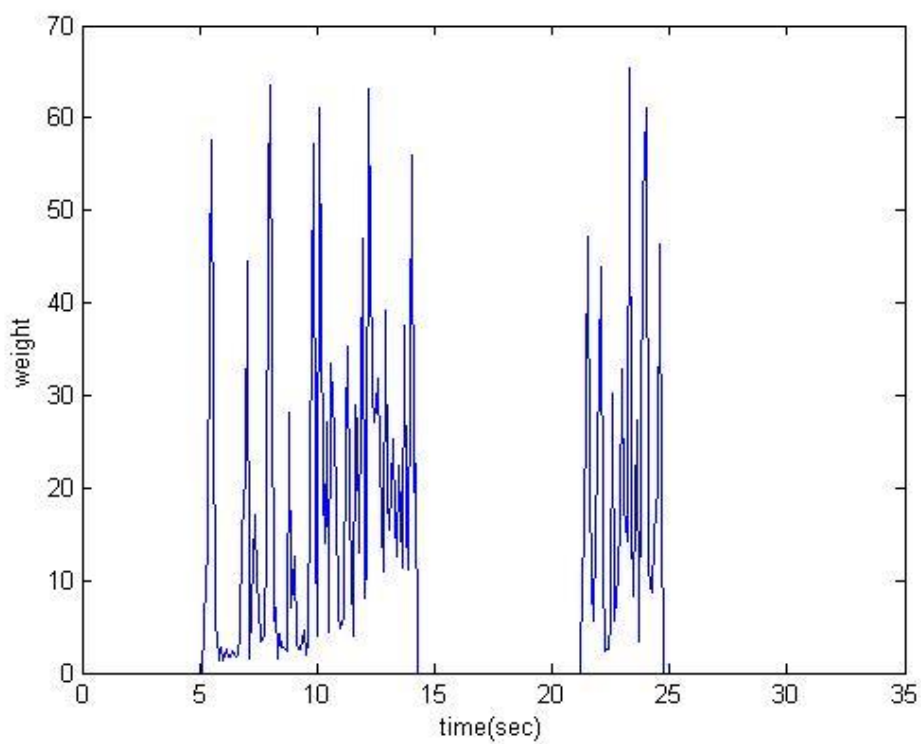
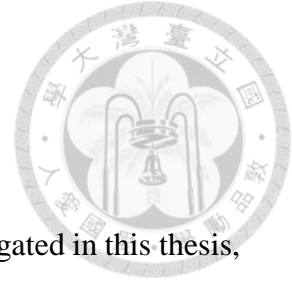


Figure 5.12 Changes of information entropy.

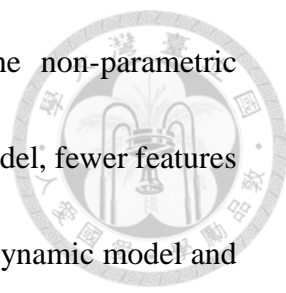
## Chapter 6 Conclusion



In conclusion, two fundamental problems of robotic are investigated in this thesis, which are the modeling the dynamic behavior of an AUV and the technique to robustly track the pose of robot body. We constructed an AUV that has a rotatable stern propeller for horizontal turning. High turning-speed bring serious drift motion of vehicle. In Chapter 3, the dynamic model of the AUV is derived. An effective and low cost visual-based method for recording the bodily motion data is presented in this work. The motion data are used to identify the parameters of model by a nonlinear grey-box method. Simulation results show this dynamic model is reliable enough for predicting vehicle's motion.

In addition, we present a novel monocular image-based tracking method for AUV in Chapter 4. By re-projecting the invariant features into current view of robot, we could get the similarity between predicted and real view of robot. In our system, we do not need to treat feature matching problem, and do not associate measurement to a map. Our method results in lower cost of computing load, and it is sufficient for real-time tracking of the robot in a water tank.

Fusion of the dynamic model and the re-projecting measurement model by particles filter are demonstrated to achieve a long-time navigation for the AUV in a



water tank. Particles filter was employed for carrying out the non-parametric appearance measurement. Due to the correctness of the dynamic model, fewer features (landmarks) number is needed in the tracking. The cooperation of dynamic model and visual-base measurement track the robot and recover its localization error, but also globally estimate its positions even the robot was "kidnapped". From the experiment, we show that with a limited amount of model errors, the time-independent error characteristics can be achieved.

The contribution of this paper is that. First, we addressed a compact method to estimate the parameters of dynamic model. Second, a robust and efficiency tracking method was presented. Further work of our research is stated as following:

(1) More formality of features instead of LED lights could be applied in the tracking algorithms, they not limit to be the artificial landmarks. In fact, color information is enough to provide measurement for tracking.

(2) Using fisheye lens is a feasible solution for increasing the field of view. This will bring less chance of vision loss of the robot. Fortunately, the fisheye projection model is easy to be introduced in our system.

(3) Model-based optimal control will be investigated for instantaneous reacting moves when the robot suddenly encounters obstacles. The reactive navigation of an underwater robot is still a challenging problem.

# Appendix



## Appendix A


Table 6-1 Specifications of the propeller:

Trade Name	SEABOTIX
Item Mode	BTD150 SPECIFICATIONS
Work Voltage (Volt)	19.1V DC $\pm$ 10%
Work Current (A)	4.25
Power(Watt)	110
Weight (grams)	705(in air) 350(in fresh water)

## Reference



- [1] J.-Y. Bouguet, "Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm," *Intel Corporation*, vol. 5, pp. 1-10, 2001.
- [2] C. Connolly and T. Fliess, "A study of efficiency and accuracy in the transformation from RGB to CIELAB color space," *IEEE Transactions on Image Processing*, vol. 6, pp. 1046-1048, Jul 1997.
- [3] H. Shin, et al. "Motion Analysis by Free-Running Model Test," *The Twelfth International Offshore and Polar Engineering Conference*, International Society of Offshore and Polar Engineers, 2002.
- [4] V. Kopman, J.Laut, F. Acquaviva, A. Rizzo, M. Porfiri, "Dynamic modeling of a robotic fish propelled by a compliant tail," *Oceanic Engineering, IEEE Journal of*, 40(1), 209-221, 2015.
- [5] T. I. Fossen, *Nonlinear modelling and control of underwater vehicles*, 1991.
- [6] 郭振華，王傑智，"自主式水下載具流體動力模式與運動控制，" *國立臺灣大學造船及海洋工程研究所碩士論文*， 1996.
- [7] A. I. Korotkin, *Added masses of ship structures* vol. 88: Springer Science & Business Media, 2008.
- [8] 施生達，"潛艇操縱性，" *國防工業出版社*， 1992

- 
- [9] 郭振華，邱柏昇，"仿生型自主式水下載具利用雙魚眼攝影機在已知環境中之導航," 國立臺灣大學工程科學及海洋工程研究所碩士論文, 2012.
- [10] N. Gracias and J. Santos-Victor, "Underwater Video Mosaics as Visual Navigation Maps," *Computer Vision and Image Understanding*, vol. 79, no. 1, pp. 66-91, 2000.
- [11] R. HARTLEY, A. Zisserman, "*Multiple view geometry in computer vision*," Cambridge university press, 2003.
- [12] S. Shah, J. K. Aggarwal, "Depth Estimation Using Stereo Fisheye Lenses," *IEEE International Conference on Image Processing*, Vol. 2, pp 740-744, 1994.
- [13] R. M. Eustice, O. M. Pizarro, H. Singh, "Visually augmented navigation for autonomous underwater vehicles," *Oceanic Engineering, IEEE Journal of*, 33(2), 103-122, 2008.
- [14] W. Narzt, et al. "Augmented reality navigation systems," *Universal Access in the Information Society* 4.3 (2006): 177-187.
- [15] R. T. Azuma, "A survey of augmented reality," *Presence: Teleoperators and virtual environments*, 6(4), 355-385, 1997.
- [16] D. W. F. Van Krevelen, R. Poelman, "A survey of augmented reality technologies, applications and limitations," *International Journal of Virtual Reality*, 9(2), 1, 2010.
- [17] J. Wolf, W. Burgard, H. Burkhardt, "Robust vision-based localization by





- combining an image-retrieval system with Monte Carlo localization,” *Robotics, IEEE Transactions on*, 21(2), 208-216, 2005.
- [18] C. Roehrig and C. Kirsch, “Particle filter based sensor fusion of range measurements from wireless sensor network and laser range finder,” in *Proc. 41st Int. Symp. Robot. and 6th German Conf. Robot.*, Jun. 2010, pp. 1–8.
- [19] R. Karlsson and F. Gustafsson, “Particle filtering for underwater terrain navigation,” in *IEEE Statistical Signal Process, Workshop*, St. Louis, MO, pp. 526-529, Oct.2003.
- [20] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. “Robust Monte Carlo Localization for mobile robots,” *Artificial Intelligence Journal*, 128(1-2):99-141, 2001.
- [21] Z. Zhang, “Flexible Camera Calibration By Viewing a Plane From Unknown Orientations,” *International Conference on Computer Vision*, 1999.
- [22] S. Thrun, W. Burgard, D. Fox, “Probabilistic Robotics,” The MIT Press, London, England, 2005.
- [23] 郭振華，王偉翰，"使用側掃聲納掃描線輔助無人水下載具建立導航地圖，"  
*國立臺灣大學工程科學及海洋工程研究所碩士論文*， 2009.
- [24] F. Bonin-Font, A. Ortiz, G. Oliver, “Visual navigation for mobile robots: A survey,” *Journal of intelligent and robotic systems*, 53(3), 263-296, 2008.
- [25] D. Nistér, O. Naroditsky, J. Bergen, “Visual odometry for ground vehicle

- applications,” *Journal of Field Robotics*, 23(1), 3-20, 2006.
- [26] R. Haywood, “Acquisition of a micro scale photographic survey using an autonomous submersible,” in *Proc. of the OCEANS Conf.*, vol. 5, pp. 1423–1426, 1986
- [27] R.L. Marks, S.M. Rock, and M.J. Lee, “Real-time video mosaicking of the Ocean floor,” *IEEE Journal of Oceanic Engineering*, vol. 20, no.3, pp. 229–241, 1995.
- [28] T. Schöps, J. Engel, D. Cremers, “Semi-dense visual odometry for AR on a smartphone,” In *Mixed and Augmented Reality (ISMAR), 2014 IEEE International Symposium on*, pp. 145-150, 2014.
- [29] S. B. Han, J. H. Kim, H. Myung, “Landmark-based particle localization algorithm for mobile robots with a fish-eye vision system,” *Mechatronics, IEEE/ASME Transactions on*, 18(6), 1745-1756, 2013.
- [30] W. Yuan, J. Katupitiya, “A time-domain grey-box system identification procedure for scale model helicopters,” In *Proceedings of the 2011 Australasian Conference on Robotics and Automation*, Dec. 2011.
- [31] C. E. Shannon, “A mathematical theory of communication,” *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1), 3-55, 2001

