國立臺灣大學工學院工程科學及海洋工程學研究所

碩士論文

Graduate Institute of Engineering Science and Ocean Engineering

College of Engineering

National Taiwan University

Master Thesis

以觀察氣泡運動輔助水下載具在水流中之姿態控制

Underwater Vehicle Gesture Control Aided by Air Bubble Motion Observation in a Water Flow

李志偉

Chih-Wei Lee

指導教授：郭振華 博士

Advisor: Jen-Hwa Guo, Ph.D.

中華民國 105 年 2 月

February, 2016

國立臺灣大學碩士學位論文
口試委員會審定書

以觀察氣泡運動輔助水下載具在水流中之姿態控制
Underwater Vehicle Gesture Control Aided by Air Bubble Motion Observation in a Water Flow

本論文係李志偉君（學號 R02525028）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 02 月 01 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：

______________________（指導教授）

______________ ______________

______________ ______________

______________ ______________

系　主　任 ______________

致謝

在這艱難的研究所生活中，要感謝的人非常多，首先最要感謝的莫過於辛苦指導我論文的郭振華老師，很謝謝老師能夠不停地指引我們方向,另外也要非常感謝江茂雄老師，因為當初老師的一席話，讓我能夠順利推上工科所的丙組，也很感謝李佳翰老師與邱逢琛老師在論文上的指導，在實驗室的生涯中，非常開心能夠遇到很好的四個夥伴王膺富、許哲維、任建柏、楊勝凱，在壓力很大的生活中，能夠一同打世紀和羽球，讓我的心情得以調適，另外很感謝我的女朋友王麗婷，總是鼓勵我語帶給我歡笑，讓我能夠充滿動力地面對挑戰，並且總是陪著我面對困難，也非常感謝邱亦倫學長在實驗上以及程式上還有 AOK 的不斷教導，感謝陳漢穎學長與黃聖偉學長在硬體上的幫忙，感謝吳柏威學長、陳建儒學長、張凱閔學長、郭倍禮學長在實驗室生涯的規劃上有很多提點，感謝蕭安庭學弟、廖君圃學弟、張佳恩學妹在實驗上幫忙出了很多力，最後很感謝這個實驗室讓我在求學路上能夠豐富我的知識，感謝台灣大學在這六年多的栽培。

摘要

本研究描述仿生機器魚利用雙魚眼攝影機、電子羅盤以及加速度計在一個有水流的環境中，藉由調整姿態以達到節能的控制演算法。在實驗場地中，仿生機器魚在向前運動的過程中會遭受到水流，而造成多餘的能量消耗及控制上的困難。本文根據氣泡與水流流動的相對關係，利用影像處理演算法描述氣泡特徵並萃取出其資訊，透過校正後的雙魚眼攝影機估測仿生機器魚與氣泡的相對位置關係，再利用盧卡斯-卡納德移動物體追蹤演算法及影像金字塔，估測氣泡隨時間變化的位置資訊。另外整合來自電子羅盤及加速度計的資料，根據氣泡流動的觀測資訊，調整仿生機器魚的運動路徑及魚身與水流之攻角，以達到在水流中穩定控制的目的。最後, 本論文展示了實驗數據，以驗證此視覺追蹤的回授控制演算法的可行性。

關鍵字:仿生機器魚、水流、氣泡、姿態控制、角點偵測、相機校正、雙眼視覺、盧卡斯-卡納德追蹤演算法、影像金字塔

Abstract

This thesis describes the gesture control of a Biomimetic Autonomous Underwater Vehicle (BAUV) in a water flow by utilizing information derived from on board sensors of stereo cameras, a compass, and an accelerometer. In an alternating water flow, the BAUV suffers from drag forces and consumes more energy when it advances. The relationship between air bubbles and water flow is first discussed. The air bubble is detected by Harris corner. The relative position between air bubble and BAUV is estimated based on the calibrated stereo cameras and the bubble is tracked by using Lucas-Kanade method and image pyramid algorithm. By integrating observation information from air bubbles, heading angles and 3-axis accelerations from compass and the accelerometer, the BAUV adjusts its heading angle to optimize the gesture of control in the water flow. Finally, the control algorithm based on the computer vision algorithm is verified by experimental data. The control power consumed in the driving motors are calculated to compare the energy used in a water flow with and without gesture control.

Keywords: BAUV, water current, air bubbles, harris corner, camera calibration, stereo vision, Lucas-Kanade tracking algorithm, image pyramid

List

Figures List

Tables List

Symbol List

v_b	the air bubble velocity
u_f	the fluid velocity
τ_b	air bubble response time
v_{bT}	air bubble rise velocity
D/Dt	Lagrangian derivatives
C	the air bubble concentration
$w(x,y)$	shifting window of harris corner detector
$I(x,y)$	image intensity in (x,y)
u	object displacement in x direction of image
v	object displacement in y direction of image
λ_1	eigenvalue
Dis	sum of the value of difference between center and its neighbor of image
λ	a scale factor
f	the focal length
$\mathbf{X}$	the coordinates of point P in the space
(o_x, o_y)	the principal point of image

(s_x, s_y)	the scale factors in the image x and y axis
$\mathbf{K}$	the calibration matrix
k_0'	the zoom factor
$(\hat{x}\ ,\hat{y}\ )$	the corrected pixel coordinates
$\mathbf{F}$	the fundamental matrix
q'	the projection point
$(\mathbf{R}_e, \mathbf{T}_e)$	the extrinsic parameters
$\overline{\mathbf{SS}'}$	the baseline of image pair
(n_x^L, n_y^L)	largest integer in image pyramid
L_m	the height of pyramid
L	the lift force
D	the drag force
C_D	the drag coefficient of BAUV
C_L	the lift coefficient of BAUV
S_{ref}	the reference area
V	the velocity of the BAUV
J_{bz}	Inertia moment about z-axis
$N_{\dot{r}}$	Effect of adding mass about z-axis
ρ	Mass density of water

α	Attack angle
A_r	Reference surface area of vehicle
r	Angular velocity of BAUV
C_{MX}	Coefficient of hydrodynamic restoring moment
C_{Mr}	Coefficient of hydrodynamic viscous moment
V	Velocity of BAUV
a	Lever arm of propeller
α_a	attack angle of BAUV (degrees)
$\theta_{air\ bubble}$	air bubble moving angle (degrees) relative to BAUV
$\alpha_{a,max}$	attack angle with maximum lift force

Chapter 1 Introduction

1.1 Motivation

The concern of a different problem has been investigated that AUVs always encounter great effect of current in the region with strong and variable current, such as surf zone [1] and river. Strong and variable current would lead to high energy cost of AUVs, make it difficult to control, and even be trapped by the vortex. Therefore, selecting an optimal gesture to decrease the energy consumption has become more important.

In the natural world, many fish swimming in the region with strong water flow would sometimes hide behind the rocks to decrease the energy cost of themselves by reducing muscle activity [2]. Liao et al. demonstrated that fish exploit vortices and therefore acquire enough thrust from vortex to balance drag just using their fins in [3]. Liao et al. suggested that fish control their gesture to comply with the current of water instead of generating propulsive movement in [4]. Therefore, it is crucial for optimizing the energy cost by adjusting gesture of body when confronting with water flow. It is difficult to sense the flow of water by sensor to adjust the gesture. However, it could be accomplished by observing air bubbles produced by the water flow crossing the obstacles such as rocks or due to hydraulic jump.

1.2 Literature Review

In previous work, several approaches to solve the path-finding problem for stable control and energy-saving could be found in [5]. In recent years, interests in oceanography have focused on littoral ocean. Due to the variable and strong current circumstance, planning a safety route with optimizing energy cost is highly important. Alvarez et al. proposed a genetic algorithm for path planning of autonomous underwater vehicles (AUVs) in an ocean with strong current to optimize the energy cost [6]. However, it is difficult to adjust the strategy of control by observing condition of current on time. Accordingly, selecting a method that AUVs could therefore get the information of current is desirable.

In the natural world, when fish swim in the water, they would sense the change of flow and then re-plan the route where they swim or adjust their gesture to take advantage of lift forces from current [3]. Although it is difficult to directly estimate the condition of flow, it is still possible to estimate the condition of flow indirectly by observing air bubbles in the water flow. Many researches about the correlation between air bubbles and flow were proposed. Air bubble motion equation was derive in [11]. An experiment of the correlation of motion between bubbles and flow in turbulent boundary layer was studied by Felton, Keith in [10]. Delnoij et al derived the force acting on bubbles and the dynamics of bubbles [9]. The formations of air bubbles such as due to wind or hydraulic jumps were widely studied in previous works. In the river, hydraulic jumps would result in entraining the air by water current and produces bubbles due to the different velocity between the current was described in [2]. In the ocean, breaking ocean waves entrain air bubbles [8]. The basic air entrainment and energy dissipation process was proposed in [1]. It is able to indirectly observe the water flow due to the correlation between air bubbles and water flow. Air bubbles observed by camera could be seen as many corners and be detected by Harris corner method. Lukas-Kanade optical method with image pyramid in [14] is applied to track air bubbles which are descripted as corners when they are moving with water flow.

1.3 Thesis Organization

This work aims to study the detection of air bubbles in a water flow, and use the bubble information to alter the BAUV’s gesture. The thesis is divided into five chapters. In Chapter 2, the system architecture of the BAUV is described. In Chapter 3, the theory of the relationship between air bubbles and water flow is described. Besides, the theory of camera calibration, stereo vision, air bubbles detection and tracking algorithm, the feature correspondence algorithm of stereo images, the performance of BAUV, and the control strategy are described. In Chapter 4, the experimental results are presented. Finally, the conclusions and the future works are given in Chapter 5.

Chapter 2 Hardware

In our research, we build a 90-cm long BAUV composed of a water-proof aluminum box. Inside the water-proof box, industrial computer, batteries, and circuits are mounted. In front of water-proof box, there are two cameras for stereo vision, an electric compass, and a pressure gauge. We mounted the wireless LAN card above the water-proof box. After the water-proof box, the propeller is mounted by an iron stand. In addition there are two rudders in its sides for adjusting depth of our BAUV.

Figure 2-1 Appearance of the BAUV

Table 2-1 Basic system of BAUV

Weight (kg)	19.9
Length (cm)	82.2
Width (cm)	23.1
Height (cm)	53

2.1 The Hardware

Inside the water-proof aluminum box, there are an industrial computer mother board, circuits, power supply, and batteries presented in Figure 2-2 and Figure 2-3.

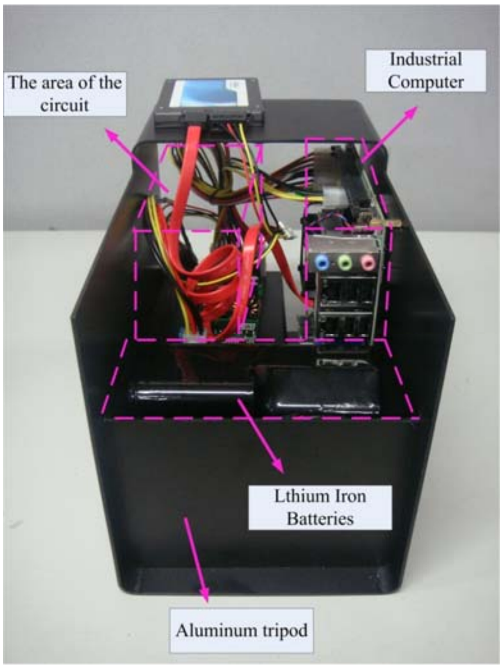

Figure 2-2 Internal of the BAUV

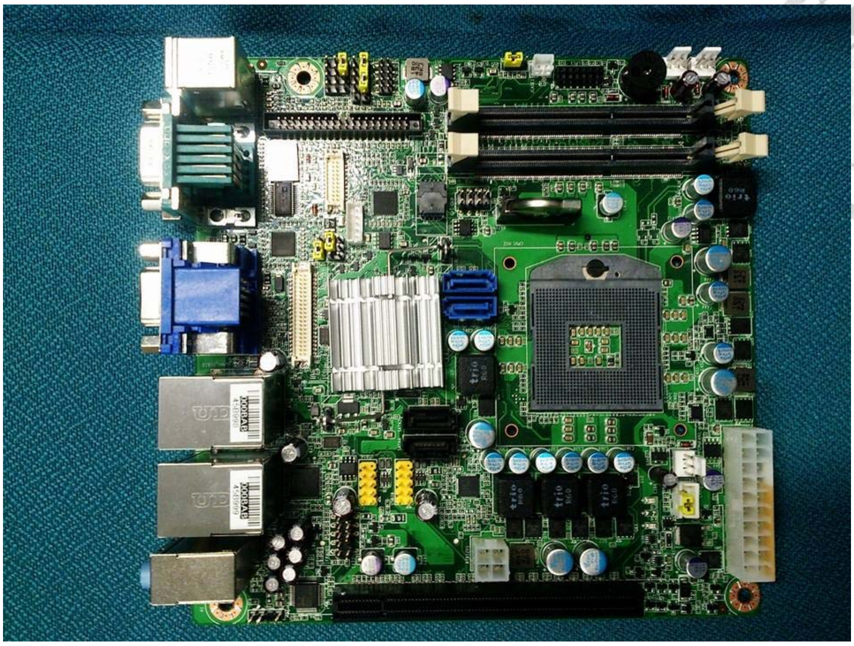

Figure 2-3 Industrial computer motherboard

Table 2-2 Specifications of the computer motherboard

Trade Name	Advantech
Item Mode	AIMB-272
CPU	Intel Core i7-2710QE
Memory Capacity	8GB DDR3
Interface	SATA 3.0 × 2, SATA 2.0 × 2, PS/2, RS-232×2, USB 2.0 ×4
Work Voltage (Volt)	DC +3.3~ + 12
Size(mm)	170 × 170 × 40

In front of water-proof aluminum box, there are (1) two cameras, (2) electronic compass, and (3) pressure gauge as sensors:

(1) Camera:

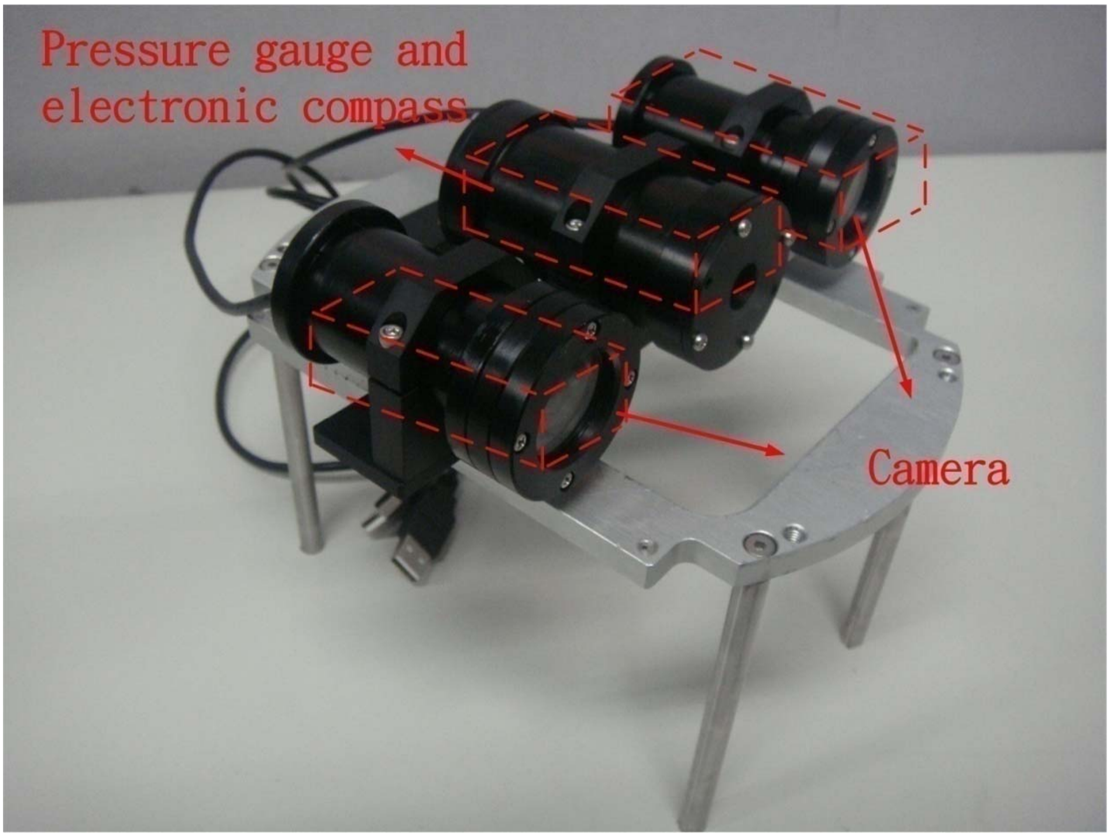

Figure 2-4 Camera and electronic compass

Figure 2-5 Camera

Table 2-3 Specifications of the camera

Trade name	ARTRAY
Item Model	ARTCAM-022MINI
Pixel size	$6.0(\mu m) \times 6.0(\mu m)$
Image size	752×480
Work Voltage	DC 5
Power(Volt)	≤ 2
Size(mm)	Long 43, Diameter 14
Weight(gram)	45

(2) Electronic compass:

Figure 2-6 Electronic compass

Table 2-4 Specifications of the electronic compass

Trade name	OceanServer
Item Mode	OS5000-USD
Work Voltage (Volt)	0.5~2.5
Measure Range	±80
Output frequency (Hz)	0.01~40
Size	1 × 1 × 0.3
Weight (grams)	2

(3) Pressure gauge:

Figure 2-7 Pressure gauge

Table 2-5 Specifications of the pressure gauge

Trade Name	Measurement
Item Mode	MSP-300-100-P-3-N-1
Pressure Range (psi)	0~300
Upper Limit (psi)	600
Sensor Safety Factor	5
Work Voltage (Volt)	0.5~4.5
Weight (gram)	82.5

Propeller is mounted by an iron stand after the water-proof box and in the other side of the stand is the motor for controlling the direction of the vehicle, and is connected to the computer through the water-proof wires.

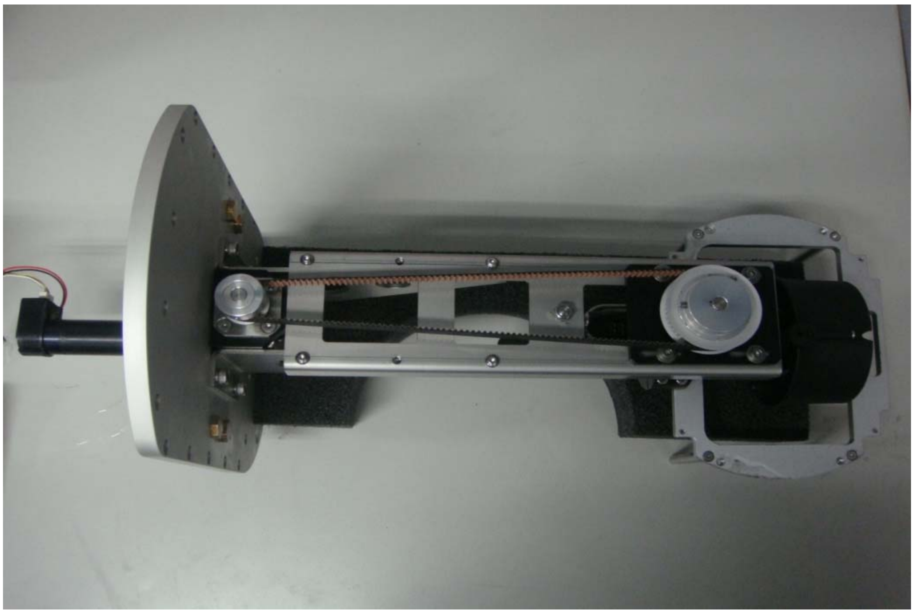

Figure 2-8 Propeller and rotating motor

Figure 2-9 Rotating motor

Table 2-6 Specifications of the motor

Trade Name	FAULHABER
Item Mode	Series 2657_024CR
Work Voltage (Volt)	24
Work Current (A)	1.54
Rotational Speed Limit (RPM)	6400
Maximum Torque (N-m)	1.324
Power (Watt)	47.9
Work Temperature (℃)	$-30\sim+125$
Weight (gram)	295

Figure 2-10 Propeller

Table 2-7 Specifications of the propeller

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

Beside the water-proof box, the motor for controlling rudders is mounted. The motor controls the angle of rudders and therefore controls the motion depth of the BAUV.

Figure 2-11 Rudder

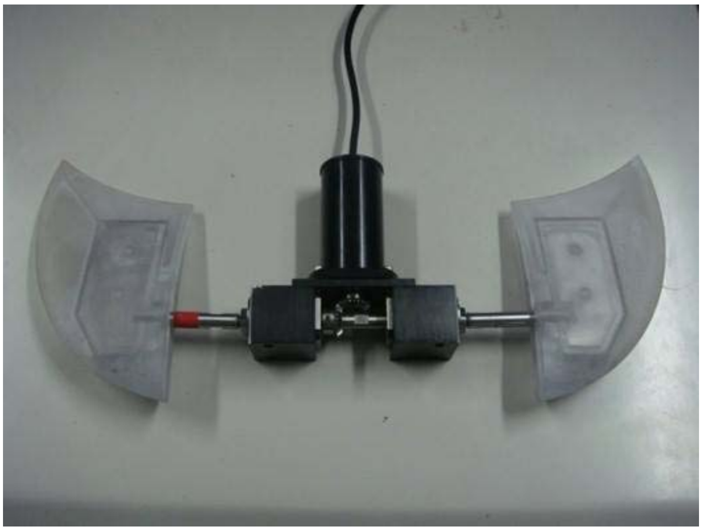

Figure 2-12 Motor of rudder plates

Table 2-8 Specifications of the motor of rudder plates

Trade Name	FAULHABER
Item Mode	Series 2224_012CR
Work Voltage (Volt)	12
Work Current (A)	0.57
Rotational Speed Limit (RPM)	8000
Maximum Torque (N-m)	5
Power (Watt)	4.05
Work Temperature (℃)	$-30\sim+85$
Weight (gram)	46

Above the water-proof box, the wifi board is mounted in the fin.

Figure 2-13 Fin and wifi-board

Figure 2-14 Arduino uno rev3

Table 2-9 Specifications of the Arduino Uno Rev3

Trade Name	Arduino Uno Rev3
Operating Voltage (Volt)	5
Input Voltage (Volt)	7 ~ 12
Digital I/O Pins	14
PWM Digital I/O Pins	6
Analog Input I/O Pins	6
DC Current per I/O Pin (Amps)	0.02
Clock Speed (MHz)	16

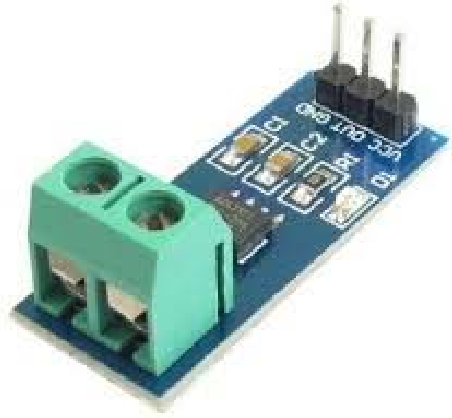

Figure 2-15 Current measurement module

Table 2-10 Specification of ACS 712

Trade Name	ACS 712
Operating Voltage (Volt)	5
Measurement Range (Amps)	−5 ~ 5
Voltage of 0A (Volt)	2.5
Scale Factor	185
Output Error	1.5%

Chapter 3 Guidance System

To achieve accurate control of BAUV in the water flow, it is important to know the correlation between air, air bubbles, and water flow. Air bubbles could be recognized by using harris corner and neighborhood process. Air bubbles then are tacked by using Lucas-Kanade optical method and image pyramid. Estimating the moving angle of air bubbles is performed by using stereo vision based on the two cameras. BAUV then could be controlled to an heading angle to decrease the effect of water flow and therefore utilize lift force contribution from water flow by observing the motion of air bubbles.

3.1 Air bubble dynamics

3.1.1 Air bubble dynamics

During the self-aeration, water flow entrains the air to the water and produce air bubbles. With the air, the water flow becomes gas-liquid two-phase flow. There are many forces acting on the air bubbles. Contribution from drag, virtual mass, liquid-phase pressure gradient, liquid-phase vorticity and gravity are considered [9]. By considering the air bubble is small enough, direct air bubble to air bubble interaction force is neglected in this thesis. The model of acting force on the air bubbles is based on Eulerian-Lagrange approach. Besides, the model developed applies to homogeneous, and this implies that the air bubbles can be seen as spherical, which simplifies the calculation forces on the air bubbles.

Understanding the dynamics of air bubbles motion in turbulent can lead to the derivation of the govern equation of air bubble motion. In this section, the govern equation adopts the Lagrange-Eulerian approach. We consider the diameter of air bubble is small enough that the air bubble to bubble interaction force can be neglected. The air bubble motion equation in [6] is modeled as:

$$\frac{dv_b(t)}{dt} = 3\frac{Du_f}{Dt} + \frac{1}{\tau_b}\left(u_f(t) - v_b(t)\right) - 2g - (v_b(t) - u_f(t)) \times \omega(t) \quad \text{(3-1)}$$

where v_b stands for the air bubble velocity, u_f is the fluid velocity, τ_b represents the air bubble response time that is connected with the air bubble rise velocity v_{bT} in still fluid by $\tau_b = \frac{v_{bT}}{2g}$, and the last term ω stands for the fluid vorticity. Note that Du_f/Dt is the Lagrangian derivatives $D/Dt = \partial/\partial t + u_f\,\partial$. Besides, we can find that the relative velocity between air bubbles and liquid is small enough in [10]. Here we assume that the range of air bubbles we observe is small enough that the acceleration of water flow and air bubbles and the vorticity of water could be neglected. Equation (3-9) then could be rewritten as:

$$u_f(t) \approx 2\tau_b g + v_b(t) \tag{3-2}$$

Therefore, we assume that the velocity of water flow is approximately equal to the velocity of air bubble based on the equation (3-2).

3.2 .Air bubble detection

To detect air bubbles by cameras, it is crucial to establish the air bubbles feature description algorithm. In this section, we introduce the theory of Harris corner and neighborhood process for air bubbles detection. Then, we establish the stereo camera system for air bubbles tracking introduced in the following section.

3.2.1 Corner detection

Air bubbles can be described as many points of corner due to the size reflected on the camera. Therefore, BAUV could detect air bubbles using cameras based on the Harris corner theory. Harris corner detector is an algorithm that we can find the corner point in a region. Three cases are described in the below: edge, planar, corner. In the first case, there is no change when putting a window in a flat region and shifting it to all directions. In the second case, there is no change when shifting the window along the edge direction. In the last case, there are significant changes when shifting the window in all directions.

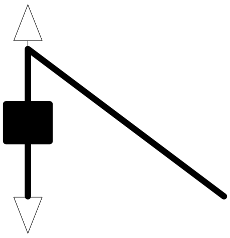

Figure 3-1 Edge

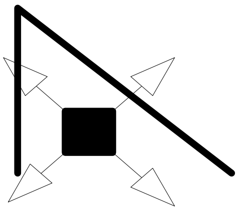

Figure 3-2 Planar

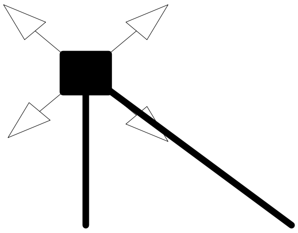

Figure 3-3 Corner

Due to the characteristic of corner, it is able to recognize the point of corner by checking the change of value in a region when shifting a small window.

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u,y+v) - I(x,y)]^2 \qquad (3\text{-}3)$$

Consider a grayscale image I and we are going to shifting a window $w(x, y)$ with displacement u in x direction and v in y directions. To find the corner, we have to maximize the term $[I(x+u, y+v) - I(x, y)]^2$ in the cost function. Using Taylor expansion and $I_x = \frac{\partial I}{\partial x}, I_y = \frac{\partial I}{\partial y}$, $E(u, v)$ is then represented as :

$$E(u,v) \approx \sum_{x,y} w(x,y)[I(x,y) + uI_x + vI_y + (high\ order\ term) - I(x,y)]^2 \quad (3\text{-}4)$$

Expansion the equation and cancelling high order term properly:

$$E(u,v) \approx \sum_{x,y} w(x,y)\left[u^2 {I_x}^2 + 2uvI_xI_y + v^2 {I_y}^2\right] \quad (3\text{-}5)$$

We then express the equation in a matrix form:

$$E(u,v) \approx [u \ \ v]\left(\sum_{x,y} w(x,y) \begin{bmatrix} {I_x}^2 & I_xI_y \\ I_xI_y & {I_y}^2 \end{bmatrix}\right)\begin{bmatrix} u \\ v \end{bmatrix} \quad (3\text{-}6)$$

and we denote:

$$\mathrm{M} = \sum_{x,y} w(x,y) \begin{bmatrix} {I_x}^2 & I_xI_y \\ I_xI_y & {I_y}^2 \end{bmatrix} \quad (3\text{-}7)$$

Consequently the equation becomes:

$$E(u,v) \approx [u \;\; v] M \begin{bmatrix} u \\ v \end{bmatrix} \tag{3-8}$$

A score is then calculated by R for each window:

$$R = \det(M) - k(trace(M))^2 \tag{3-9}$$

When λ_1 and λ_2 are large enough and $\lambda_1 \sim \lambda_2$ where λ_1 and λ_2 are eigenvalues of M, E increases in all directions that we can consequently determine where is the point of corner based on the score R.

3.2.2 Neighborhood process

To strengthen the detection system of air bubbles, we introduce the neighborhood process. Neighborhood process is widely used in digital image process. To focus on the region where the brightness is greatly different from its neighbor, we adopt the 8-neighborhood process. The 8-neighborhood range is from (col-1 and row-1) of the checked point pixel to the (col+1 and row+1).

$$\mathrm{Dis} = \sum_{8-neighbor}(I_o - I_{neighbor}) \geq Threshold1 \qquad (3\text{-}10)$$

We take the Dis as the sum of the value of difference between center and its neighbor value over the threshold1. It is essentially a voting process that each neighbor point votes to the sum. These votes are accumulated in the value of Dis and by comparing the sum to the threshold2 that can be seen as a point has similar character with the light-spot if the sum over the threshold2.

$$\mathrm{Dis} \geq Threshold2 \ \rightarrow \ \text{Candidate of air bubble} \qquad (3\text{-}11)$$

By the series process of Harris corner, neighborhood process, and image intensity comparison, it is possible to detect the air bubble on the image. However, it should be tracked for getting more information due to the movement of air bubble in the water flow. The tracking algorithm will be introduced in the following chapters.

3.2.3 Coordinate transformation

To introduce stereo cameras system, we firstly introduce the camera coordinate system. Coordinate transformation is an important concept in computer vision especially for the stereo vision. It describes the relationship among the image coordinate system, the camera coordinate system, and the world coordinate system. The standard matrix for ideal perspective camera coordinate system can be represented as:

$$\lambda\begin{bmatrix}x\\y\\1\end{bmatrix}=\begin{bmatrix}f&0&0\\0&f&0\\0&0&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\end{bmatrix}\begin{bmatrix}\mathbf{R}&\mathbf{T}\\\mathbf{0}&\mathbf{1}\end{bmatrix}\begin{bmatrix}X_0\\Y_0\\Z_0\\1\end{bmatrix} \tag{3-12}$$

where λ is a scale factor, x and y are the coordinate in the 2D image, f is the focal length, $[\mathrm{RT}]$ is rotation and translation matrix, X_0, Y_0, Z_0 are the coordinate of point in 3D space. In practice, when the image is formed from camera, the measurement information is typically with the origin of coordinate in the upper-left corner of image frame instead of the optical center of image frame.

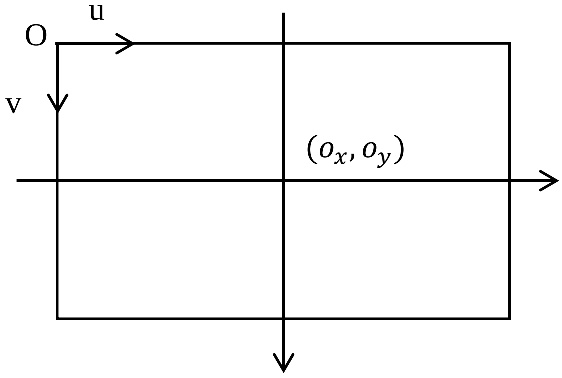

Figure 3-4 Relationship between two images

coordinate representations

The image frame is assumed to be rectangular such as 320×480. The relationship between the actual image coordinate and ideal image coordinate can be represented as:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \tag{3-13}$$

where s_x and s_y are the scale factors in the *x* and *y* axis of image respectively, (o_x, o_y) is the principal point of image, $[x\ y\ 1]^T$ and $[x_i\ y_i\ 1]^T$ are actual image coordinates and ideal image coordinates respectively. Combining the above two coordinate model with scaling and translation, it then becomes a more realistic model of a transformation between homogeneous coordinates of 3D points and homogeneous coordinates of image:

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{K} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3-14}$$

$$\mathbf{K} = \mathbf{K}_s \cdot \mathbf{K}_f = \begin{bmatrix} s_x & 0 & o_x \\ 0 & s_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} fs_x & fs_\theta & o_x \\ 0 & fs_y & o_y \\ 0 & 0 & 1 \end{bmatrix} \tag{3-15}$$

where **K** is combined by all intrinsic parameters of the particular camera and is called calibration matrix or intrinsic parameter matrix. This calibration matrix shows the geometry relationship between the coordinate of the point in real world and its corresponding image coordinate in image, which depends on the rigid body motion between the frame in real world and the frame of camera. Therefore, the model is then formed by:

$$\lambda\begin{bmatrix}x\\y\\1\end{bmatrix}=\begin{bmatrix}fs_x & fs_\theta & o_x\\0 & fs_y & o_y\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\end{bmatrix}\begin{bmatrix}\mathbf{R} & \mathbf{T}\\\mathbf{0} & \mathbf{1}\end{bmatrix}\begin{bmatrix}X_0\\Y_0\\Z_0\\1\end{bmatrix} \tag{3-16}$$

3.2.4 Radial distortion adjustment

Despite after the camera calibration by using the radial distortion parameter which are estimated in previous section, the distortion of image still exists especially near the edge of image. Therefore, it is necessary to optimize the parameters to achieve more accurate image coordinate.

Consider the coordinate origin $(x_o, y_o) = (0,0)$, zoom factor $k'_0 = 1$, and coefficients in odd order terms equal to zero, the radial distortion model can be represented as the following form:

$$\hat{x} = x + x[k_2'(x^2 + y^2) + k_4'(x^2 + y^2)^2]$$
$$\hat{y} = y + y[k_2'(x^2 + y^2) + k_4'(x^2 + y^2)^2] \tag{3-17}$$

where $(\hat{x},\ \hat{y})$ are the corrected point coordinates (undistorted), $(x,\ y)$ are the original point coordinates (distorted), and $k_2',\ k_4'$ are calibration coefficients. By transforming the coordinates from the principal point ($o_{x\,,}o_y$), the form then becomes:

$$\hat{x} = x + (x - o_x)[k_2'(x^2 + y^2) + k_4'(x^2 + y^2)^2]$$
$$\hat{y} = y + \left(y - o_y\right)[k_2'(x^2 + y^2) + k_4'(x^2 + y^2)^2] \tag{3-18}$$

Given *m* points in n images, there will exist 2*mn* equations which are similar to the form 3-18. Then, stacking all equations together and arranging into a matrix form as the following shows:

$$\begin{bmatrix} (x_{11} - o_x)(x_{i_{11}}^2 + y_{i_{11}}^2) & (x_{11} - o_x)(x_{i_{11}}^2 + y_{i_{11}}^2) \\ \left(y_{11} - o_y\right)(x_{i_{11}}^2 + y_{i_{11}}^2) & \left(y_{11} - o_y\right)(x_{i_{11}}^2 + y_{i_{11}}^2) \\ \vdots & \vdots \\ (x_{mn} - o_x)(x_{i_{mn}}^2 + y_{i_{mn}}^2) & (x_{mn} - o_x)(x_{i_{mn}}^2 + y_{i_{mn}}^2) \\ (y_{mn} - o_y)(x_{i_{mn}}^2 + y_{i_{mn}}^2) & (y_{mn} - o_y)(x_{i_{mn}}^2 + y_{i_{mn}}^2) \end{bmatrix}_{mn\times 2} \begin{bmatrix} k_2' \\ k_4' \end{bmatrix} = \begin{bmatrix} \hat{x} - x \\ \hat{y} - y \\ M \\ \hat{x}_{mn} - x_{mn} \\ \hat{y}_{mn} - y_{mn} \end{bmatrix} \tag{3-19}$$

it then can be simplified as $\mathbf{Dk}' = \mathbf{d}.$ The solution $\mathbf{k}'$could be obtained by using the least squares method $\mathbf{k}' = (\mathbf{D^T D})^{-1}\mathbf{D^T d}.$ Once $\mathbf{k}'$ is estimated, the intrinsic parameter matrix $\mathbf{K}$, the extrinsic parameters for each image $\mathbf{R}_n$ and $\mathbf{T}_n$ mentioned in previous section can be refined by a nonlinear optimizing process which is so-called Levenberg-Marquardt Algorithm. The objective function is shown as:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left\|\mathbf{m}_{ij}-\widehat{\mathbf{m}}(\mathbf{K},\mathbf{R}_i,\mathbf{T}_i,\mathbf{k}',\mathbf{M}_j)\right\|^2 \tag{3-28}$$

where $\widehat{\mathbf{m}}\left(\mathbf{K},\mathbf{R}_i,\mathbf{T}_i,\mathbf{k}',\mathbf{M}_j\right)=[\hat{x} \quad \hat{y} \quad 1]^{\mathrm{T}}$ is the projection of the distorted point $\mathbf{M}_j = [X_w \quad Y_w \quad 0 \quad 1]^{\mathrm{T}}$ in the image *i*, and $\mathbf{m}_{ij}$ is the undistorted point in the image *i*. The initial value of $\mathbf{k}'$ can be simply set to zero, and can be obtained by using the technique mentioned in previous section.

3.2.5 Epipolar geometry

In this thesis, stereo vision is used to accurately determine the angle of air bubbles and therefore get the information of water flow. Epipolar Geomery in an important part of stereo vision.

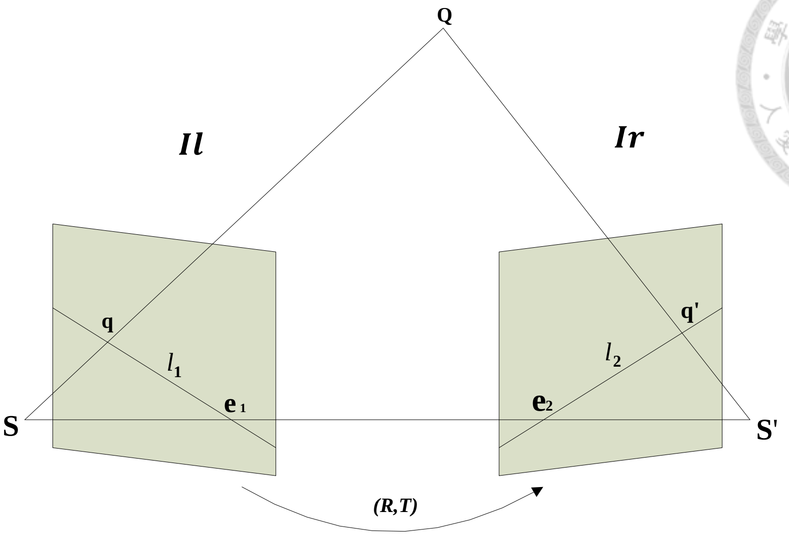

Figure 3-5 Epipolar geometry between stereo images

Consider $\boldsymbol{S}$ and $\boldsymbol{S'}$ are the centers of two camera lens in 3D space, and $\boldsymbol{q}$ **and** $\boldsymbol{q'}$ are the projection 2D points in images of $\boldsymbol{Q}$ by camera lens. The intersection of the line $\overline{\boldsymbol{SS'}}$ are points *e*$_1$ and *e*$_2$ in each images ($\boldsymbol{Il}, \boldsymbol{Ir}$), and the lines *l*$_1$, *l*$_2$ are so-called epipolar lines. Epipoles have the property that all epipolar lines in $\boldsymbol{Il}$ pass through $\mathbf{e_1}$**;** likewise, all epipolar lines in $\boldsymbol{Ir}$ pass through $\mathbf{e_2}$. We then define two vectors $\mathbf{P}_l = \overrightarrow{\mathbf{SQ}}$ and $\mathbf{P}_r = \overrightarrow{\boldsymbol{S'}\mathbf{Q}}$ with respect to the left and right camera coordinate system, the translation vector $\mathbf{T}_e$, and rotation matrix $\mathbf{R}_e$. The relation then be expressed as:

$$\mathbf{P}_r = \mathbf{R}_e(\mathbf{P}_l - \mathbf{T}_e) \tag{3-20}$$

Where $\mathbf{T}_e = \mathbf{S'} - \mathbf{S}$ and $\mathbf{R}_e$ is an orthogonal matrix. Because $\mathbf{T}_e$, $\mathbf{P}_r$, and $\mathbf{P}_l$ are coplanar, it satisfies the constraint:

$$(\mathbf{P}_l - \mathbf{T}_e)^{\mathrm{T}} \cdot (\mathbf{T}_e \times \mathbf{P}_l) = 0 \tag{3-21}$$

Then rearrange the formula (3-20) and substitute into (3-21), the formula can be derived as:

$$\left(\mathbf{R}_e{}^{T}\mathbf{P}_r\right)^{T} \cdot (\mathbf{T}_e \times \mathbf{P}_l) = 0 \tag{3-22}$$

If considering the stereo intrinsic parameter matrices $\mathbf{q} = \mathbf{A}_{c,l}\mathbf{P}_l$ and $\mathbf{q}' = \mathbf{A}_{c,r}\mathbf{P}_r$, epipolar is then constraint by $\mathbf{q}'^{\mathrm{T}}\hat{\mathbf{T}}\mathbf{R}\mathbf{q} = 0$ and $\mathbf{F} = \hat{\mathbf{T}}\mathbf{R} \ \in \mathbf{R}^{3\times 3}$, which $\mathbf{F}$ is the fundamental matrix:

$$\mathbf{F} = \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{bmatrix} \tag{3-23}$$

$\mathbf{F}$ maps points in $\boldsymbol{Il}$ to lines in $\boldsymbol{Ir}$:

$$\mathbf{Fq} = \boldsymbol{l_1}$$

$$\mathbf{F}^{\boldsymbol{T}}\mathbf{q}' = \boldsymbol{l}_{\mathbf{2}} \tag{3-24}$$

Moreover, the fundamental matrix also exists the constraints:

$$\mathbf{Fe}_{\mathbf{1}} = \mathbf{Fe}_{\mathbf{2}} = \mathbf{0} \tag{3-25}$$

Assume the homogeneous coordinate of projection points q and q' are respectively $[u \quad v \quad 1]^{\mathrm{T}}$ and $[u' \quad v' \quad 1]^{\mathrm{T}}$, the constraint equation $\mathbf{q'}^{\mathrm{T}}\mathbf{Fq} = 0$ is then expansion to:

$$uu'F_{11} + vu'F_{12} + u'F_{13} + uv'F_{21} + vv'F_{22} + v'F_{23} + uF_{31} + vF_{32} + F_{33} = 0 \tag{3-26}$$

Let

$$\mathbf{F}_{v} = [F_{11} \;\; F_{12} \;\; F_{13} \;\; F_{21} \;\; F_{22} \;\; F_{23} \;\; F_{31} \;\; F_{32} \;\; F_{33}] \tag{3-27}$$

the equation then becomes:

$$[uu' \;\; vu' \;\; u' \;\; uv' \;\; vv' \;\; v' \;\; u \;\; v \;\; 1]\mathbf{F}_v = 0 \tag{3-28}$$

If there are n corresponding points, it will be reasonable to obtain a set of linear equations:

$$\begin{bmatrix} u_1u_1' & u_1u_1' & u_1' & u_1v_1' & v_1v_1' & v_1' & u_1 & v_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_nu_n' & u_nu_n' & u_n' & u_nv_n' & v_nv_n' & v_n' & u_n & v_n & 1 \end{bmatrix} \mathbf{F}_v \equiv \mathbf{KF}_v = 0 \tag{3-29}$$

Additional constraint is added that $||\mathbf{F}_v|| = 1$ for avoiding the trivial solution, and there will be unique solution for $\mathbf{F}_v$ when n≥8. Minimizing $||\mathbf{KF}_v||$ under the constraint $||\mathbf{F}_v|| = \mathbf{F}_v^T\mathbf{F}_v = 1$ by using Lagrange multipliers, the $\mathbf{F}_v$ then can be obtained. Furthermore, the solution to this problem is the unit eigenvector corresponding to the smallest eigenvalue of $\mathbf{K}^{\mathrm{T}}\mathbf{K}$.

Table 3-1 The intrinsic parameters of the stereo cameras

	Left image	Right image
Focal Length (pixel)	[*fs*$_x$*fs*$_y$] = [660.06564 643.34422] with uncertainty	[*fs*$_x$*fs*$_y$] = [660.62277 643.53198] with uncertainty

	[7.20618 6.78063]		[7.80598 7.34994]	
Principal point (pixel)	$[o_x o_y]$ = [403.88702 280.29007] with uncertainty [2.98761 3.03845]		$[o_x o_y]$ = [401.29696 278.24653] with uncertainty [3.08398 3.20967]	
Distortion parameters for adjustment	k_o	0.01321	k_o	0.00485
	k_1	-0.00297	k_1	0.10126
	k_2	0.00368	k_2	0.00423
	k_3	-0.00028	k_3	0.00068
	k_4	0	k_4	0
Rotation vector(radians)	[-0.0033 0.00056 0.00024]			
Translation vector(mm)	[3.98202 -13.26767 -7.81723]			

3.2.6 Stereo vision

By using stereo vision, it is able to determine the moving angle of air bubble during the vortex. Therefore, the auv then could be controlled to an adaptive gesture to fit the current and decrease the locomotion cost. In this section, how to estimate a 3D point by using the stereo system is introduced and the mathematical model is derived. After rectifying the epipolar lines, it is assumed that the camera geometry with parallel planes is known, and the center of image coordinate is located at the principal point in each camera. Given a three-dimensional point **M** which represents the air bubble in three-dimensional space is projected onto the right image at $\mathbf{P}_{R\backslash}$ and the left image at $\mathbf{p}_L$. The coordinate system is represented as $O_{c,s}$, $X_{c,s}$, $Y_{c,s}$, $Z_{c,s}$; x_R, y_R and x_L, y_L are respectively the image coordinate system of the right camera and the left camera. By considering the similar triangulation, the perspective projection equations are found to be:

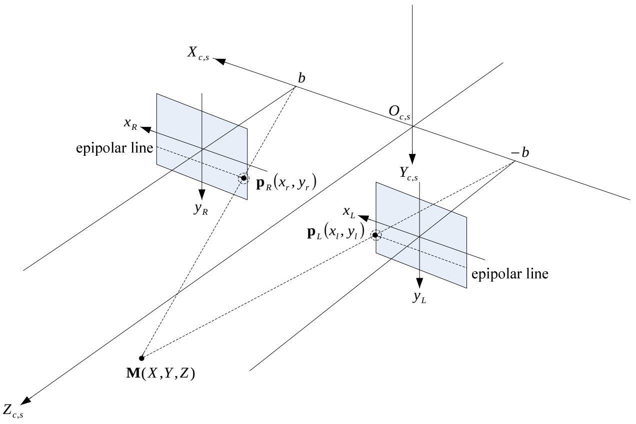

Figure 3-6 Stereo geometry

$$\begin{bmatrix} x_l \\ y_l \end{bmatrix} = {}^{f}\!/_{Z} \begin{bmatrix} X + b \\ Y \end{bmatrix}$$

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = {}^{f}/_{Z} \begin{bmatrix} X - b \\ Y \end{bmatrix} \tag{3-30}$$

where f is the focal length of the stereo rigs and $2b$ is the baseline of the stereo rigs. In practice, the force length f is usually set to the average of the focal length of the left camera f_L and the focal length of the right camera f_R; that is, $f = (f_L + f_R)/2$

The formula (3-30) can be inverted to solve for X, Y and Z. The inverse perspective projection equations could be written as:

$$X = \frac{x_l + x_r}{x_l - x_r} b$$

$$Y = \frac{y_l + y_r}{x_l - x_r} b \tag{3-31}$$

$$\mathrm{Z} = \frac{2bf}{x_l - x_r} b$$

Note that $x_l - x_r$ is so-called disparity.

3.2.7 Sum of absolute differences

Sum of absolute difference (SAD) is a common correspondence method in digital image process. SAD measures the similarity of two image block by calculating the sum of difference in two window block. We firstly make a window around the target object and make window around all of the detected points. We then accumulate the difference between two windows. If the sum of difference between them less than the threshold, the aim point then becomes the correspondence candidate point of the target object.

$$Sum\ of\ differences = \sum_{(i,j)\in W} |(I_1(x_1+i, y_1+j) - I_2(x_2+i, y_2+j)| \qquad (3\text{-}32)$$

We build a window around the target point:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \qquad (3\text{-}1)$$

We then build windows around each aim points, here we assume there are three aim points:

$$\begin{bmatrix} p_1 & p_2 \\ p_3 & p_4 \end{bmatrix} \quad \begin{bmatrix} q_1 & q_2 \\ q_3 & q_4 \end{bmatrix} \quad \begin{bmatrix} k_1 & k_2 \\ k_3 & k_4 \end{bmatrix} \qquad (3\text{-}33)$$

Sum of differences between target point and three aim points:

$$SAD1 = |(a - p_1)| + |(b - p_2)| + |(c - p_3)| + |(d - p_4)| \tag{3-34}$$

$$SAD2 = |(a - q_1)| + |(b - q_2)| + |(c - q_3)| + |(d - q_4)| \tag{3-35}$$

$$SAD3 = |(a - k_1)| + |(b - k_2)| + |(c - k_3)| + |(d - k_4)| \tag{3-36}$$

The correspondence point of target point is then be chosen in aim points with minimum SAD, that means there are the smallest difference between target point and aim point.

$$Correspondence\ point\ pair = \text{point in arg}[Min(SAD)] \tag{3-37}$$

After calculating the sum of difference between the target object and all of the detected points, the correspondence candidate point with minimum SAD then is seen as the correspondence point of the target object.

3.3 Air bubble tracking

3.3.1 Lucas-Kanade optical method

Corners that represent air bubbles are essentially moving objects, we tracked those moving air bubbles based on Lucas-Kanade optical method (LK optical method). The Lucas-Kanade optical method is widely used in computer vision and is an effective algorithm for tracking moving object.

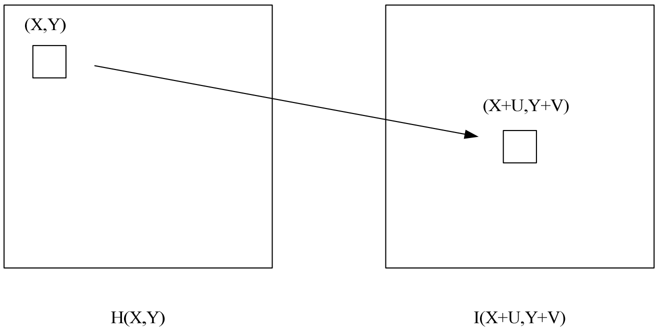

Figure 3-7 Relationship between same point in different time t and t+1

It assumes that there are small displacement between two consecutive images respectively in t and t+1. We denote u and v are small motion in x and y direction. Moreover, it assumes that the optical between neighbor pixels is a constant:

$$I(x+u, y+v) = H(x, y) \tag{3-38}$$

Use Taylor expansion:

$$0 \approx I(x, y) + uI_x + vI_y + (high\ order\ term) - H(x, y) \tag{3-39}$$

Cancel high order term properly:

$$0 \approx \big(I(x,y) - H(x,y)\big) + uI_x + vI_y \tag{3-40}$$

$I(x,y) - H(x,y)$ can be denote as I_t and we rewrite the optical flow equation:

$$0 \approx I_t + \nabla I \,\cdot\, [u \;\; v] \tag{3-41}$$

Then expanding the formula:

$$\begin{aligned} I_t = {} & \frac{1}{4}[I(x,y) + I(x+1,y) + I(x,y+1) + I(x+1,y+1)] \\ & -\frac{1}{4}[H(x,y) + H(x+1,y) + H(x,y+1) + H(x+1,y+1)] \end{aligned} \tag{3-42}$$

$$\begin{aligned} I_x = {} & \frac{1}{4}[I(x+1,y) + I(x+1,y+1) + H(x+1,y) + H(x+1,y+1)] \\ & -\frac{1}{4}[I(x,y) + I(x,y+1) + H(x,y) + H(x,y+1)] \end{aligned} \tag{3-43}$$

$$\begin{aligned} I_y = {} & \frac{1}{4}[I(x,y+1) + I(x+1,y+1) + H(x,y+1) + H(x+1,y+1)] \\ & -\frac{1}{4}[I(x,y) + I(x+1,y) + H(x,y) + H(x+1,y)] \end{aligned} \tag{3-44}$$

To avoid aperture problem, we have to impose additional constraints on equation.

Basically, we assume the flow field is smooth. In addition, it is assumed that the optical flow equation can hold for all pixels in a window, due to the small displacement between two consecutive images.

$$\begin{gathered} I_x(q_1)u + I_y(q_1)v = -I_t(q_1) \\ I_x(q_2)u + I_y(q_2)v = -I_t(q_2) \\ \vdots \\ I_x(q_n)u + I_y(q_n)v = -I_t(q_n) \end{gathered} \tag{3-45}$$

We then rewrite the equation as matrix:

$$AV = b \tag{3-46}$$

where $A = \begin{bmatrix} I_x(q_1) & I_y(q_1) \\ \vdots & \vdots \\ I_x(q_n) & I_y(q_n) \end{bmatrix}$, $V = \begin{bmatrix} u \\ v \end{bmatrix}$, $b = \begin{bmatrix} -I_t(q_1) \\ \vdots \\ -I_t(q_n) \end{bmatrix}$

It is seems that usually we have more equations than unknowns, therefore it is over-determined. Thus, we use the least squares by minimizing $\|Ad - b\|^2$ to solve this:

$$A^T A V = A^T b \tag{3-47}$$

Note that $A^T A$ should be invertible, V is then solved:

$$V = \begin{bmatrix} u \\ v \end{bmatrix} = (A^T A)^{-1} A^T b \tag{3-48}$$

3.3.2 Image pyramid

The Lucas-Kanade optical method is a robust algorithm for tracking. However, when the displacement of motion object is too large, LK optical method will be unreliable because it assumes that the displacement of moving object is small enough. If the displacement of moving object is large, the scale effect will affect the error of the LK optical method. Therefore, the image pyramid is applied on LK optical method to applied for the robustness [14] by decreasing the effect of scale. Consider an image I of size $n_x \times n_y$, the pyramid is built in many layers with different resolutions of images $I^0\ to\ I^L$ where $I^0$with the highest resolution and I^L is computed from I^{L-1}:

$$I^L(x, y) = \frac{1}{4} I^{L-1}(x, y)$$

$$+\frac{1}{8}\left(I^{L-1}(2x-1,2y) + I^{L-1}(2x+1,2y) + I^{L-1}(2x, 2y-1) + I^{L-1}(2x, 2y+1)\right)$$

$$+\frac{1}{16}(I^{L-1}(2x-1,2y-1) + I^{L-1}(2x+1,2y+1) + I^{L-1}(2x-1,2y+1)\ +$$

$$I^{L-1}(2x+1,2y+1)) \tag{3-49}$$

The equation above is then defined $0 \leq 2x \leq n_x^{L-1}-1$ and $0 \leq 2y \leq n_y^{L-1}-1$. The width n_x^L and height n_y^L of I^L satisfy:

$$n_x^L \leq \frac{n_x^{L-1}+1}{2}, n_x^L \leq \frac{n_x^{L-1}+1}{2} \tag{3-50}$$

Note that n_x^L and n_y^L are the largest integers. We can find that the width and height between two images in a pyramid becomes $\frac{1}{2}$. For example, if the size of I^2 is 320×480 then the size of I^3 will be 160×240. By the image pyramid mentioned above, if the displacement in the origin image I^0 is 8, and then the displacement in the image I^2 will become 2. Therefore, it will accord with the assumption of Lk optical method. Besides, due to the decrease of resolution in the pyramid, the image becomes blur; therefore, help for decreasing the effect of noise.

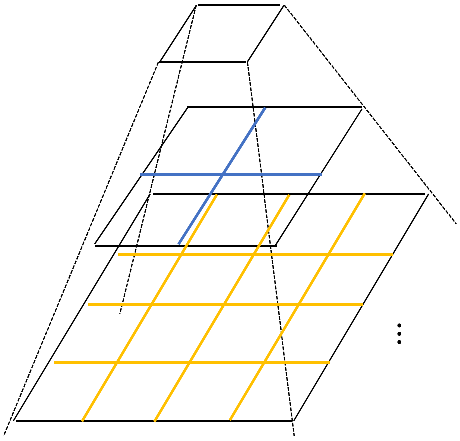

Figure 3-8 Image pyramid

The kernel for building the pyramid we use is:

$$\frac{1}{16}\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} \qquad (3\text{-}51)$$

3.4 BAUV motion control

3.4.1 Performance of BAUV

The water flow would influence the BAUV due to the drag force when the BAUV heads on the water flow direction. The BAUV would change its heading angle to reach the optimizing the energy cost, because the increase of lift force with the change of attack angle.

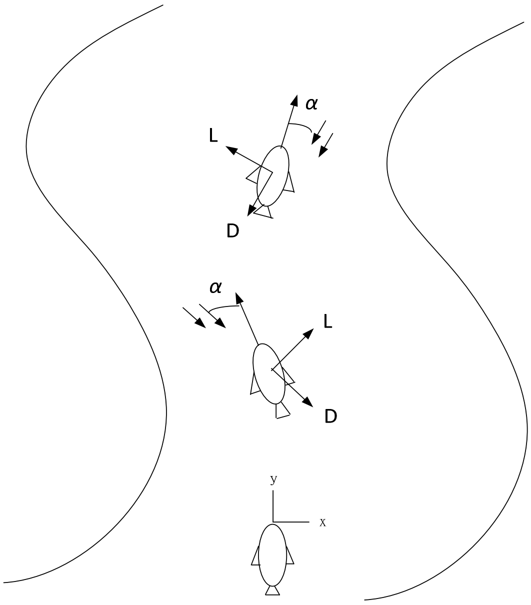

Figure 3-9 BAUV moving heads on the water flow direction

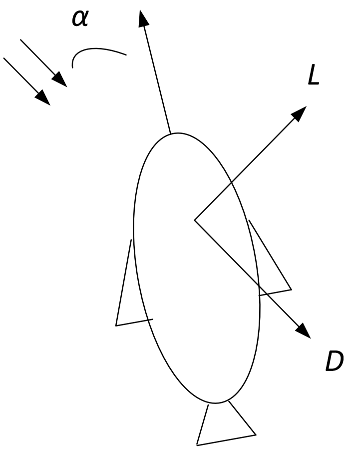

Figure 3-10 Relationship between water flow and BAUV

In Figure 3-10, α is the attack angle and L represent the lift force, while D is the drag force. The hydrodynamic forces which include drag force and lift force are modeled as:

$$D = \frac{1}{2}\rho \cdot S_{ref} \cdot (C_D) \cdot V^2 \tag{3-2}$$

$$C_D = C_{D0} + C_{D2}\alpha^2 \tag{3-3}$$

where C_D is the drag coefficient which depends on the shape of body, S_{ref} is the reference area for the calculation of the lift force, and V is the velocity of the BAUV.

$$L = \frac{1}{2}\rho \cdot S_{ref} \cdot (C_L) \cdot V^2 \tag{3-54}$$

$$C_L = C_{L1}\alpha_a \tag{3-55}$$

where C_L is the lift coefficient which depends on the shape of body. By the approximate approach, the shape of our BAUV can be approximated by the NACA trainer60-il airfoil. In table 3-3, we show the specifications of the BAUV including ρ, S_{ref} which is parallel to horizontal pane, and V.

Table 3-2 Specifications of the BAUV

BAUV	
$S_{ref}(m^2)$	0.665228
$\rho(\mathrm{Kgf} * \mathrm{sec}^2/\mathrm{m}^4)$	101.82
$V(cm/s)$	50

The relationship between the lift coefficient and attack angle of our BAUV is showed in Figure 3-11 and the relationship between the drag coefficient and attack angle of our BAUV is showed in Figure 3-12

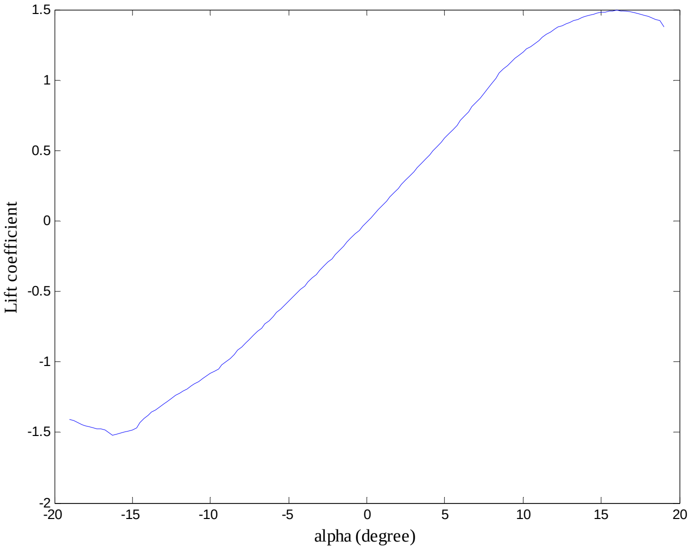

Figure 3-11 Relationship between lift coefficient and attack angle

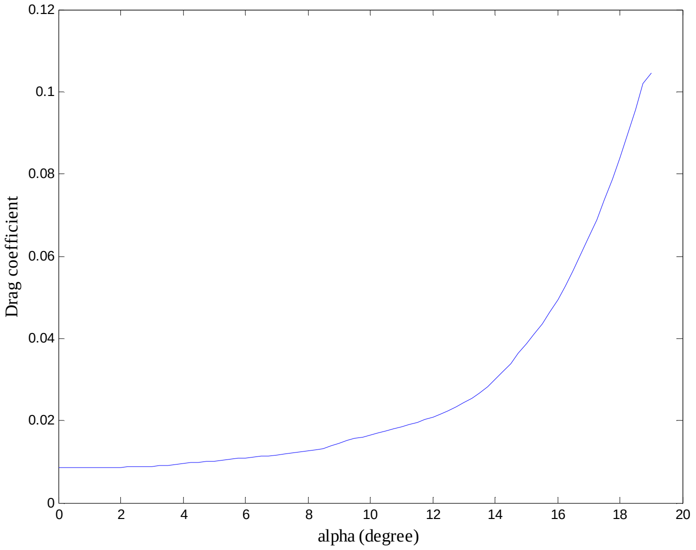

Figure 3-12 Relationship between drag coefficient and attack angle

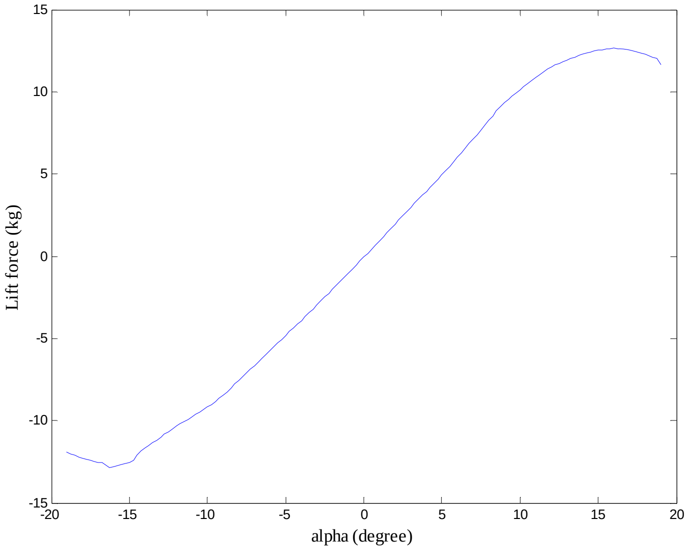

Figure 3-13 Relationship between lift force and attack angle

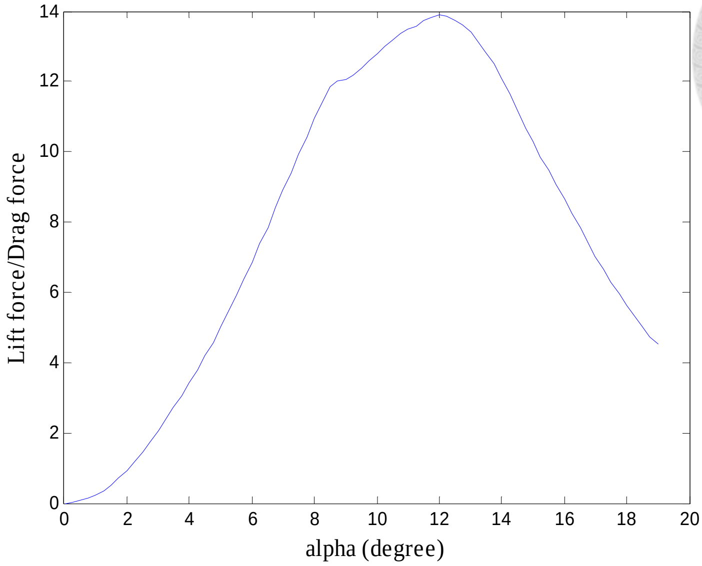

Figure 3-14 Relationship between drag force and attack angle

Figure 3-15 shows the relationship of lift force in different attack angle and Figure 3-16 shows the relationship of drag force in different attack angle.

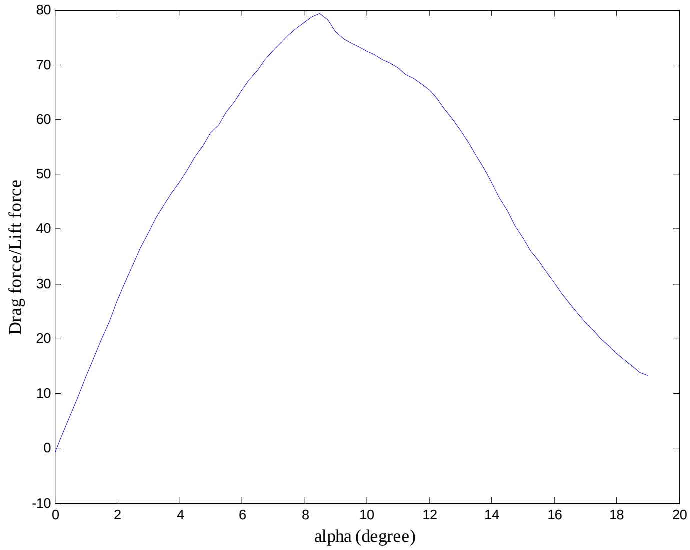

Figure 3-15 Relationship of lift force versus drag force in different attack angle

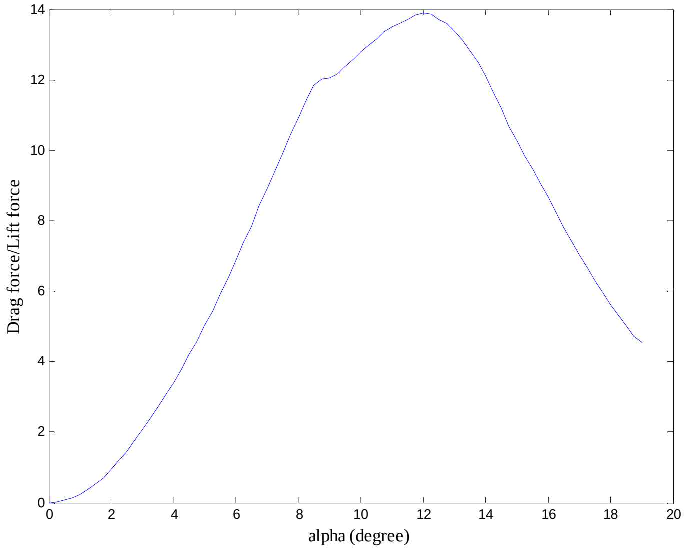

Figure 3-16 Relationship between lift force versus drag force

in y direction and attack angle

Figure 3-15 shows the relationship of lift force versus drag force in different attack angle, and Figure 3-16 shows the relationship between lift force versus drag force in y direction and attack angle. We then choose the attack angle of our BAUV as 13° to reach the most efficient performance with maximun lift force versus drag force in the y direction

3.4.2 Control System

To decide the attack angle of the BAUV when moving in water current, the BAUV firstly detects air bubbles, tracks air bubbles, and decides the adequate attack angle and position the BAUV has to go in front it in a preview distance. The preview distance is critical for the distance constraint in stereo vision and the response time for heading angle control shown in (3-56).

$$(J_{bz} - N_{\dot{r}})\dot{r} = \rho \mathrm{A}_r C_{MX} V(t)^2 \alpha_a(t) - 2\rho \mathrm{Ar} a^2 C_{Mr} V(t) r(t) - T(t) a \sin \emptyset(t) \quad \text{(3-56)}$$

where:

J_{bz}: Inertia moment about z-axis

$N_{\dot{r}}$: Effect of adding mass about z-axis

ρ: Mass density of water

α_a: Attack angle

A_r: Reference surface area of vehicle

r: Angular velocity of BAUV

C_{MX}: Coefficient of hydrodynamic restoring moment

C_{Mr}: Coefficient of hydrodynamic viscous moment

V: Velocity of BAUV

a: Lever arm of propeller

Our control coefficients are presented in appendix. Step responses of the heading angle based of the water flow of 30 degrees in relative to the BAUV as shown in Figure 3-17 and 45 degrees in Figure 3-18. We assume the velocity of water flow is and the velocity of our BAUV is about 20 (cm/s). The goal heading angle of BAUV are 17 degree and 28 degree respectively based on the adequate attack angle 13 degrees proposed in 3.4.1.

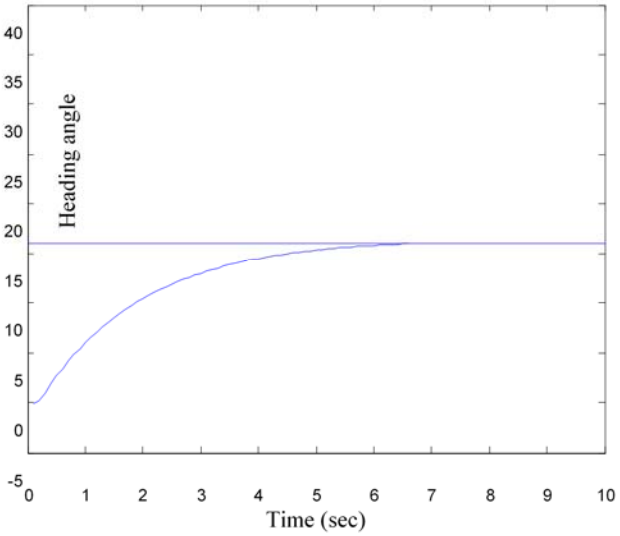

Figure 3-17 Heading angle step response to the 30-degree flow

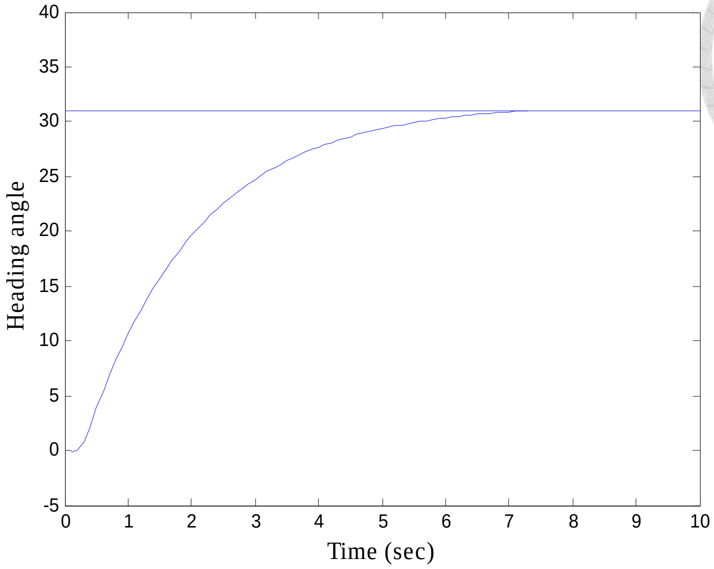

Figure 3-18 Heading angle step response to the 45-degree flow

The convergence time of heading angle control are about 6.6 sec and 7.3 sec. Therefore, it is important for BAUV control its heading angle before entering the water flow in a preview distance.

The relative location of an air bubbles group relative to the position of a BAUV should be calculated by stereo vision stated in previous section. The bubble concentration region is primarily divided into two sub-regions, right or left corresponds to the advancing direction of the BAUV. Then, each sub-region is again divided into four quadrants. We mainly focus on the third and the fourth quadrants. When the air bubbles in is in the head right direction corresponds to the BAUV, the water current direction in the third quadrant will be helpful for producing lift force. Alternately, when the air bubbles in is in the head left direction corresponds to the BAUV, the water current direction in fourth quadrant will be helpful for producing lift force.

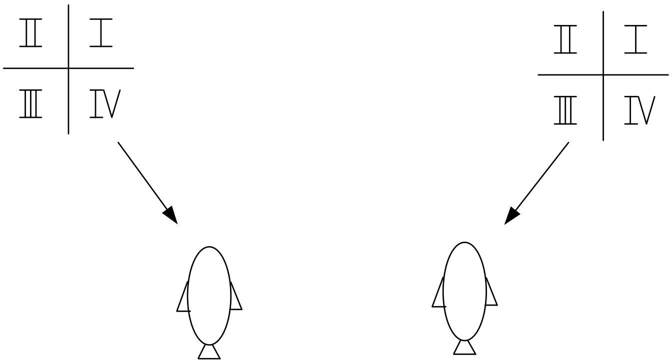

Figure 3-19 Quadrant of water flow relative to BAUV

$$\theta_{air\ bubble} = \mathrm{atan}(dx, dy) \tag{3-57}$$

$$\alpha_{head} = \begin{cases} \theta_{air\ bubble} - \alpha_{a,max}, & (if\ \theta_{air\ bubble} > 0) \\ \theta_{air\ bubble} + \alpha_{a,max}, & (if\ \theta_{air\ bubble} < 0) \end{cases} \tag{3-58}$$

where

α_{head}: heading angle of BAUV (degrees)

$\theta_{air\ bubble}$: air bubble moving angle (degrees)

$\alpha_{a,max}$: attack angle with maximum lift force

Specifically, the air bubbles moving toward the dx<0 and dy<0 direction observed by BAUV that means the water flow coms from the upper right of BAUV. We define the heading angle of BAUV is $\theta_{air\ bubble} - \alpha_{a,max}$ in such displacement of air bubbles. Likewise, the air bubbles moving toward the dx>0 and dy<0 direction observed by BAUV that means the water flow coms from the upper left of BAUV.. We define the heading angle of BAUV is $\theta_{air\ bubble} + \alpha_{a,max}$ in such displacement of air bubbles.

However, not every air bubble moving along the main water flow which can be seen in Figure 3-20 presented by long-arrow, some air bubbles drift out from side of main water flow which can be seen in Fig. 3-20 presented by two short-arrows. The air bubbles drift out from side of main water flow will affect the calculation of BAUV heading angle, because the difference between their direction and the direction of main water flow. Therefore, we analyze the distribution of the angle of air bubbles to find the main group of the air bubbles which can most likely represent the main direction of water flow. We count the number of air bubbles relative to different angles and consider those air bubbles in the range with most distributions as candidate ones. We then adopt the moving angle of those candidate air bubbles to calculate heading angle of BAUV.

Figure 3-20 Water flow and air bubble displacement

$\theta_{air\ bubble}$ then becomes:

$$\theta_{air\ bubble} = \theta \ \mathrm{argmax}(\mathrm{N\ of\ } displacement) \tag{3-59}$$

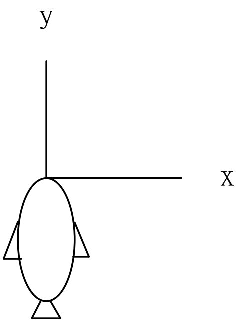

Figure 3-21 Reference frame of BAUV

The heading angle of BAUV and air bubbles correspondence X and Y relative to the BAUV become the input of our PID controller.

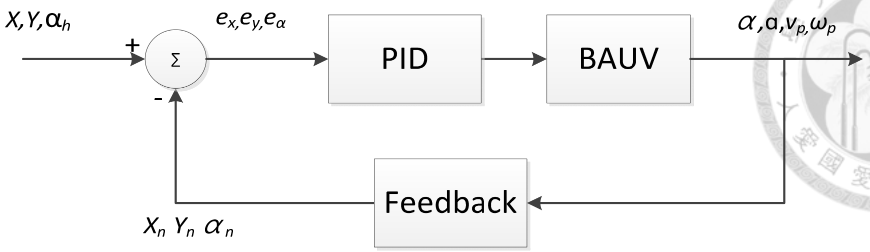

Figure 3-22 Block diagram of BAUV control system

The block diagram of the closed-loop control system is shown in the Figure 3-22, where X, Y, α_h are air bubbles relative X and Y coordinate to the BAUV and adequate heading angle of the BAUV.

Chapter 4 Experimental Results

To verify our air bubble detection system, air bubble tracking system, BAUV heading angle control system and energy saving. We separate our experiment to stand experiment and motion experiment and the experimental results are discussed in this chapter.

4.1.1 Air Bubble Detection Results

In this section 4.1.1 and 4.1.2, we use the water pipe to generate air bubbles in the water and set the light above the water flow to ensure that our camera could see the underwater view. Figure 4-1 shows the setup of the experiment.

Figure 4-1 Water pipe to generate air bubbles

We conduct experiment to verify the air bubble detection system. We use harris corner to find corner in each image from left camera and right camera, and describe air bubbles within these corners by using neighborhood process. In the experiment, we firstly detect air bubbles by using Harris corner only. We show the original image without using air bubbles detection system in Figure 4-2. We can observe that the result in Figure 4-3 is not accurate because one air bubble may be seen as many air bubbles. This error will affect the distribution of air bubbles velocities. After using image neighborhood process mentioned in 3-23 and 3-24 to improve the result of air bubble detection and as the image neighborhood process threshold is increasing, the error that air bubble which is seen as two or three air bubbles decreased and the result becomes more accurate in Figure 4-9 with the threshold≥ 6.

Figure 4-2 Original image from camera

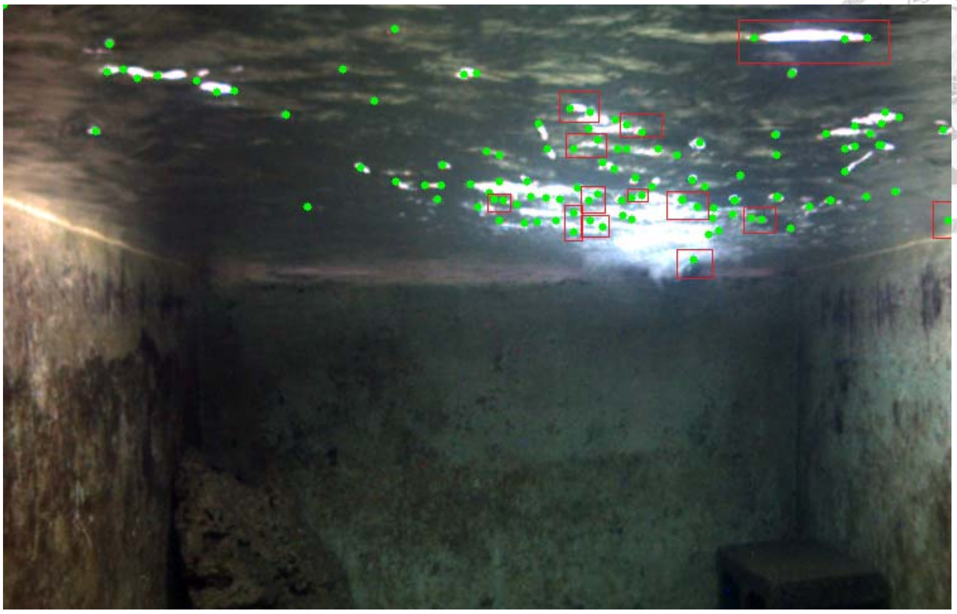

Figure 4-3 No neighborhood process. One air bubbles is detected as more than one point which are marked as squared region

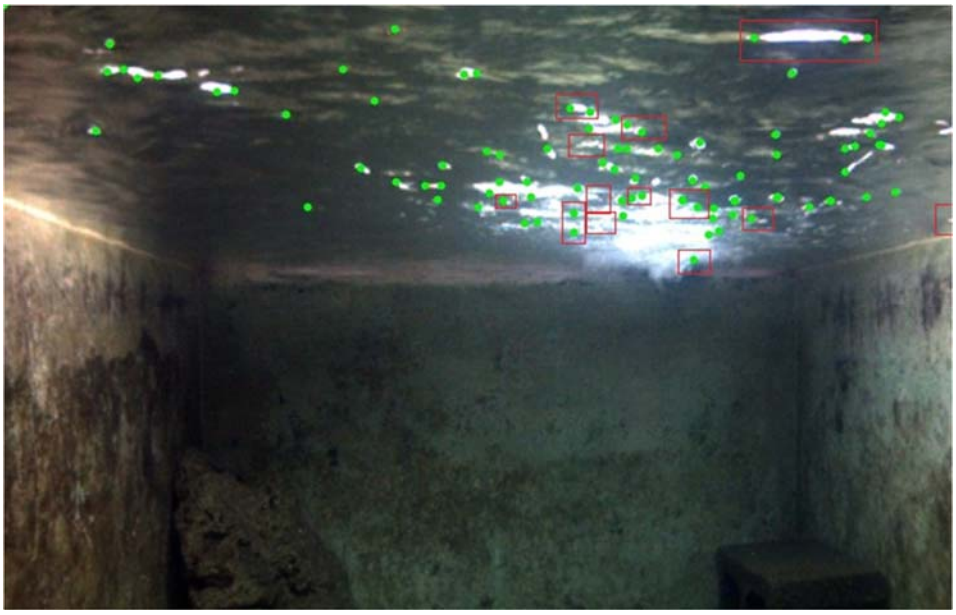

Figure 4-4 Dis ≥ 1 , candidates of air bubble

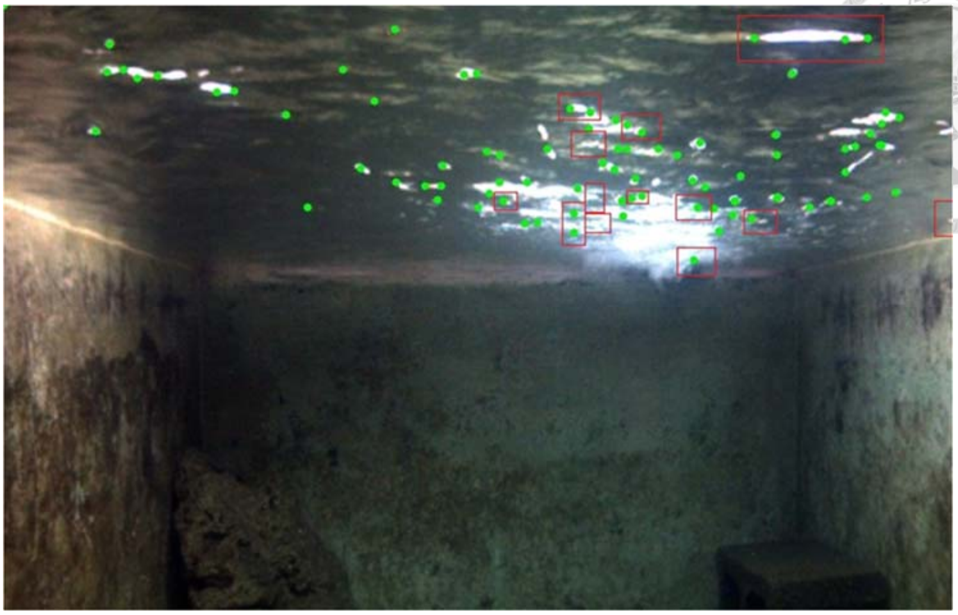

Figure 4-5 Dis ≥ 2 , candidates of air bubble

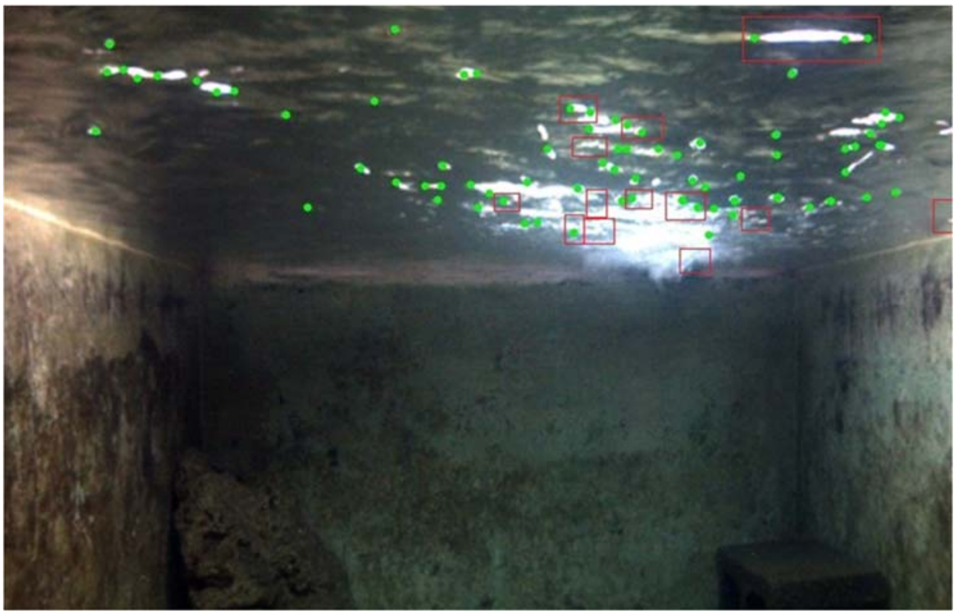

Figure 4-6 Dis ≥ 3, candidates of air bubble

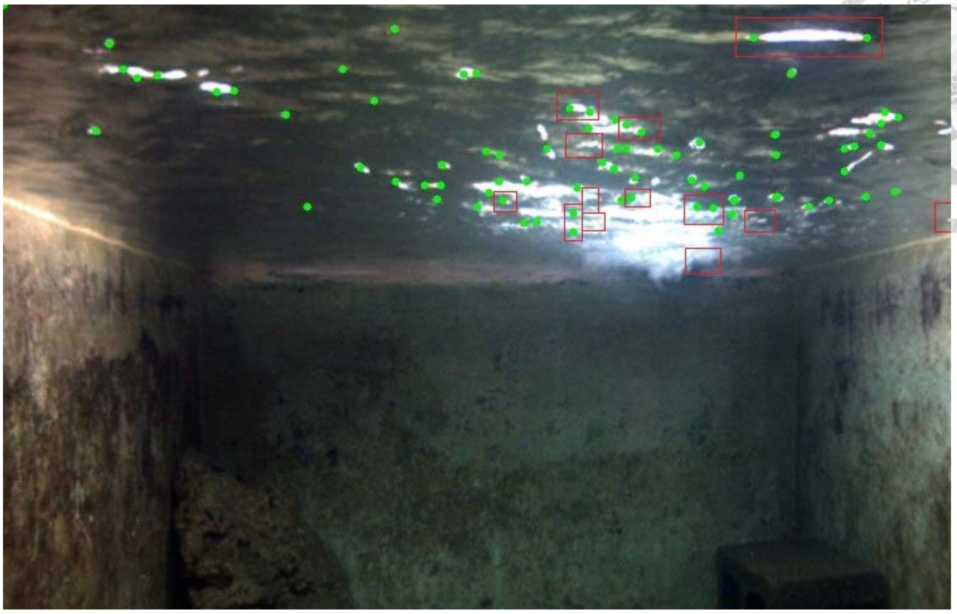

Figure 4-7 Dis ≥ 4， candidates of air bubble

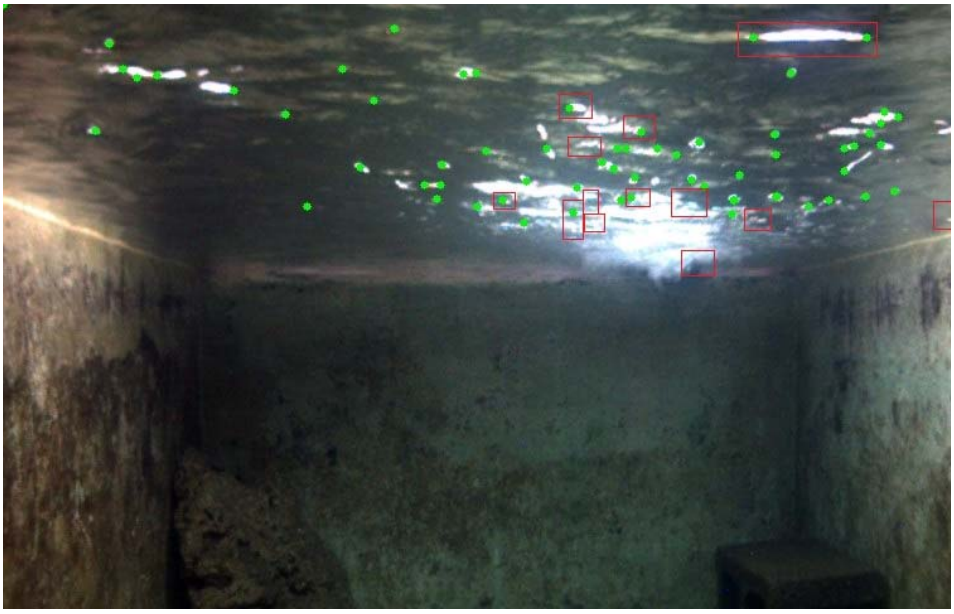

Figure 4-8 Dis ≥ 5， candidates of air bubble

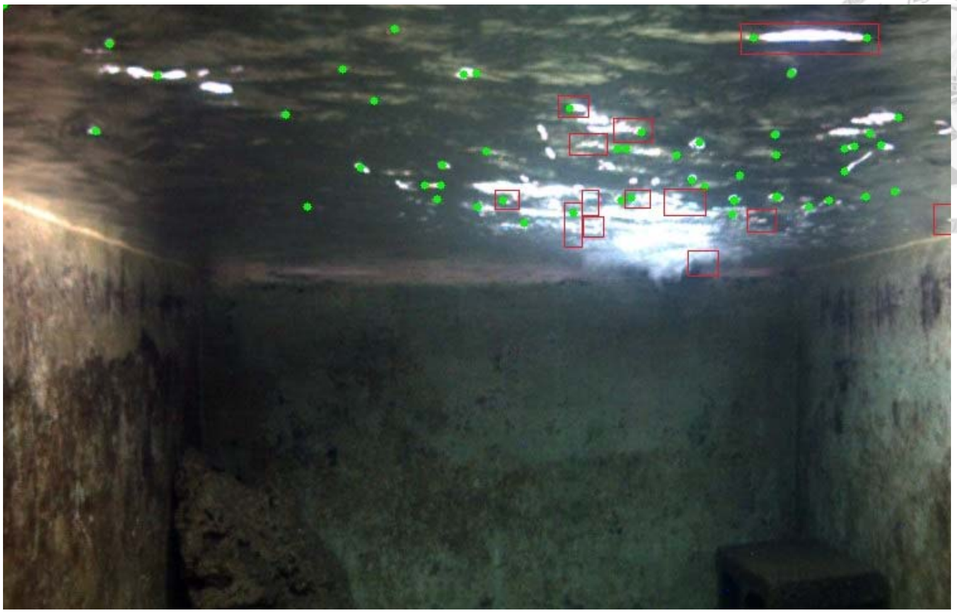

Figure 4-9 Dis ≥ 6 , candidates of air bubble

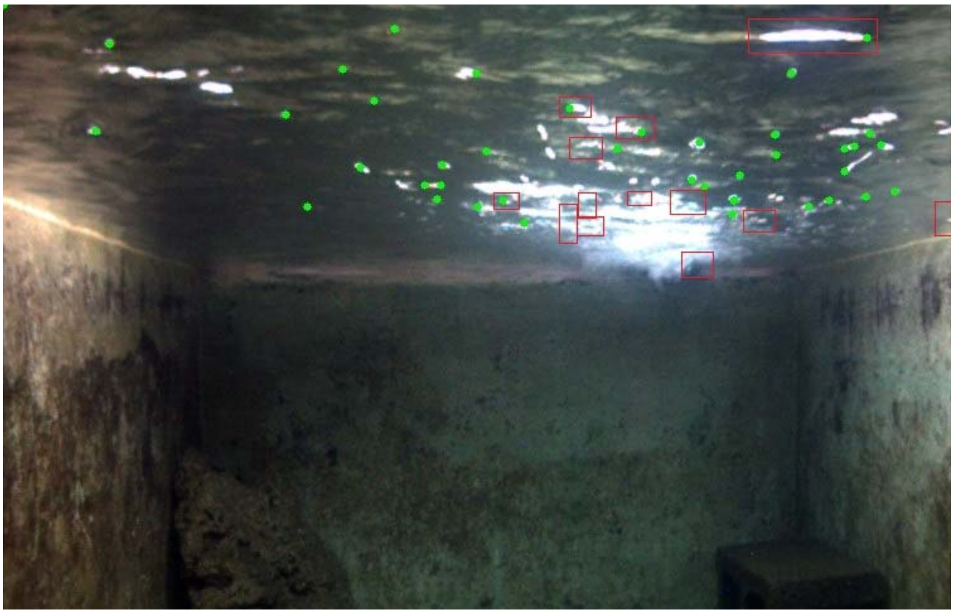

Figure 4-10 Dis ≥ 7 , candidates of air bubble

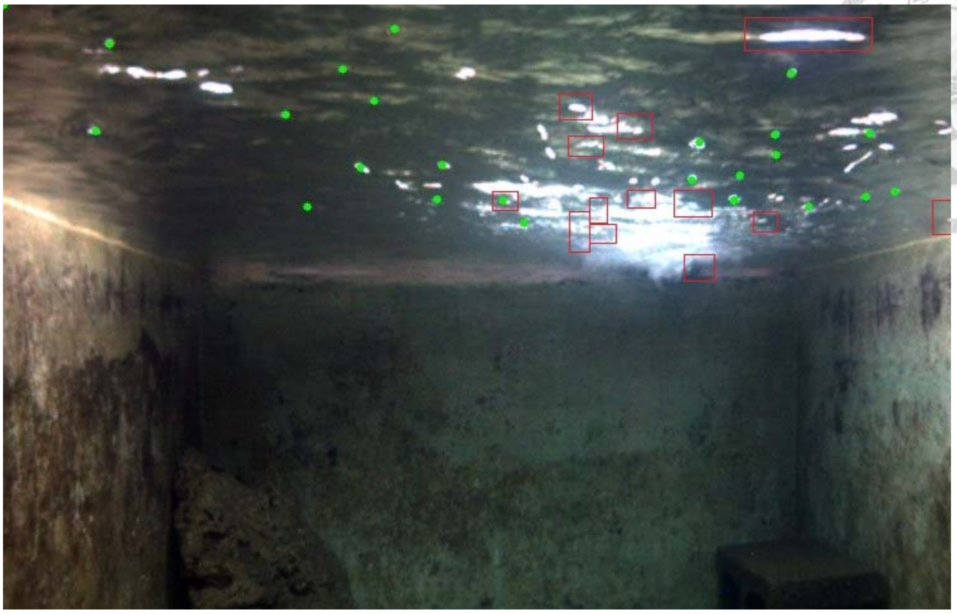

Figure 4-11 Dis ≥ 8， candidates of air bubble

However, we can clearly observe that many of detected air bubbles in relaxed threshold loss in threshold2≥ 8. The error of the neighborhood process with threshold2≥ 8 in Figure 4-11 increased again because many air bubbles cannot be detected as air bubbles. Therefore, the threshold cannot be set too strict. In Figure 4-12, we present the result that the number of detected corners and the number of undetected air bubbles in squared region. We can find that when the threshold become strict, the number of corners will decrease, and the number of undetected air bubbles increases correspondingly.

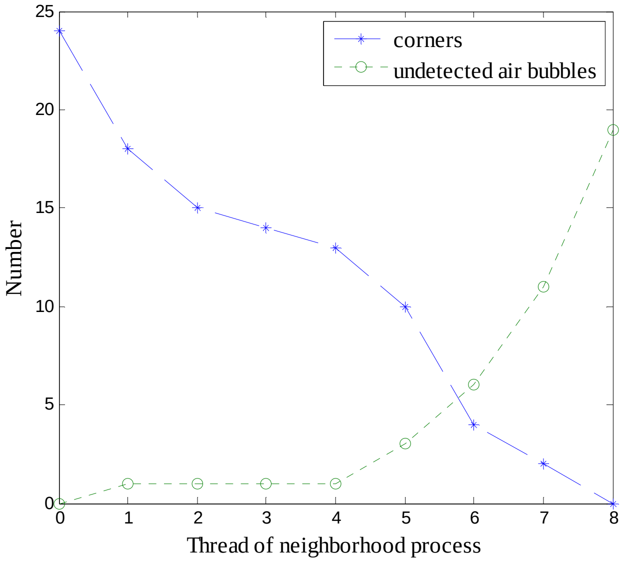

Figure 4-12 Number of corners and undetected air bubbles

In Table 4-1, we average the error of air bubble detection system by calculating number of points rightly represents air bubble versus points failed to represent air bubble within 200 images. The error is about 10.67% in left camera and about 9.34% in right camera.

Table 4-1 Error of detected point in air bubbles

	Error
Left Camera	10.67%
Right Camera	9.34%

4.1.2 Air Bubble Tracking Results

To calculate the moving angle of air bubbles, 3 dimensional coordinate of air bubbles have been known through formula: $\theta_{air\ bubble} = \mathrm{atan}(dx, dy)$ which has been mentioned in 3.4.2. Firstly, air bubbles are respectively detected in left and right cameras. Based on Talukder and Goldberg in [16], we adapt SAD method mentioned in 3.2.7 that we build 8X8 window in every air bubble points in both images from left and right cameras. If the sum of absolute difference between selected air bubble point in left camera and every air bubble points in right camera is lower than the threshold, the air bubble point in right cameras then becomes candidate point of the correspondence point to air bubble point in left camera. Until the sum of absolute difference of every air bubble points in right camera are calculated, the candidate point which is less than the threshold and with the smallest value of sum of absolute difference then becomes correspondence point. Besides, due to the camera calibration, the same point in both two cameras will have the same y-coordinate but smaller x-coordinate in right camera because the constraint mentioned in 3.2.5. If the candidate point does not fit in this constraint both in x-coordinate and y-coordinate, that candidate point will be eliminated before becoming correspondence point. The 3 dimensional coordinate can be calculated based on the inverse perspective projection equation mentioned in 3.2.6. The flow diagram of air bubbles 3-dimension coordinate calculation is shown in Figure. 4-13.

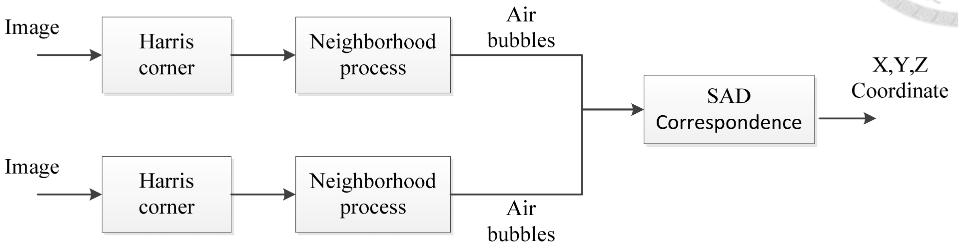

Figure 4-13 Block diagram of air bubbles detection system

Figure 4-14 is the result of air bubbles detection in the left camera, and Figure 4-15 is the result of air bubbles detected in the right camera. Because the difference in view, illumination, and contrast of cameras, we can find that the numbers of detected air bubbles in left camera and right camera are different.

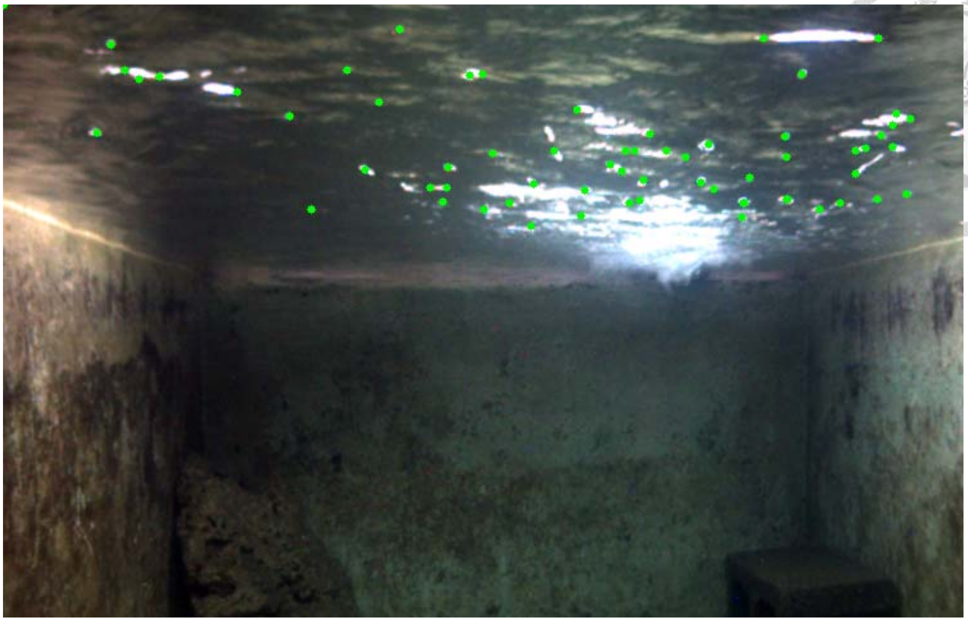

Figure 4-14 Left camera image with neighborhood process threshold2 ＝6

Figure 4-15 Right camera image with neighborhood process threshold2 ＝6

We can obviously observe that in the inception of air entrainment, the air bubbles are difficult to be described by harris corner because the density of air bubbles is too high to precisely distinguish air bubbles. The neighborhood process also has this issue because the difference between center of corner and its neighbor is lower than the threshold due to excessive concentration.

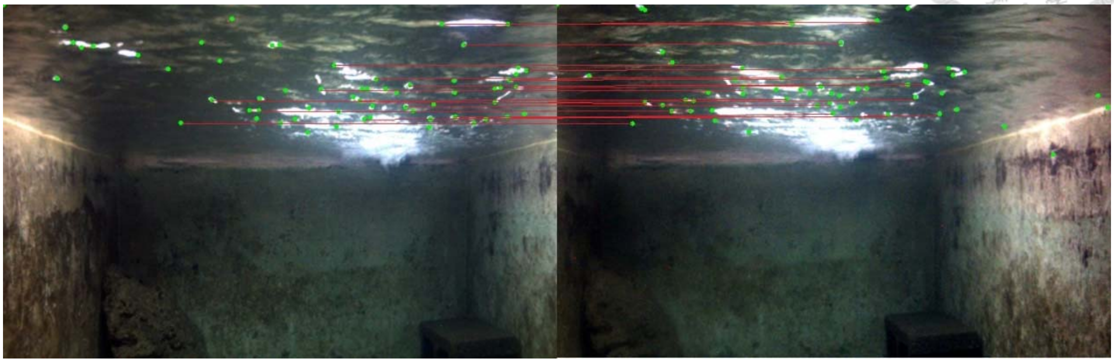

Figure 4-16 SAD feature correspondence between left camera and right camera

Figure 4-16 shows the result of feature correspondence by SAD. In real situation, due to the error of camera calibration, the y-coordinate of an air bubble in left camera image does not totally equal to the y-coordinate of the same air bubble in right camera image. Therefore, the restriction in y-coordinate should be relaxed to a small range. However, this tactic may sometimes result in mismatch of the point in image pair.

Table 4-2 Result of air bubble detection and feature correspondence by SAD

Camera	Detected bubbles	Correspondence
Left Camera	70	48
Reft Camera	72	

In Table 4-2, we can observe that the number of feature correspondence is less than the number of air bubbles detected by both cameras. The reason for that is the air bubbles detected by left camera may be different to the air bubbles detected by right camera due to the difference in viewpoint, luminance, and error of calibration. This reason not only affects the air bubbles detection but also affects the result of SAD correspondence.

Figure 4-17 Block diagram of air bubbles tracking process

The air bubbles with information of three dimensional coordinate are then be tracked by the air bubble tracking system (image pyramid and LK optical method) for motion information to calculate the moving angle of air bubbles. Therefore, consecutive position of air bubbles can be successfully recorded to calculate the moving angle of air bubbles. The block diagram of air bubbles tracking process is showed in Figure 4-17. The setup of this experiment is shown in Figure 4-18

Figure 4-18 Illustration of the position of water flow and the BAUV in the water tank

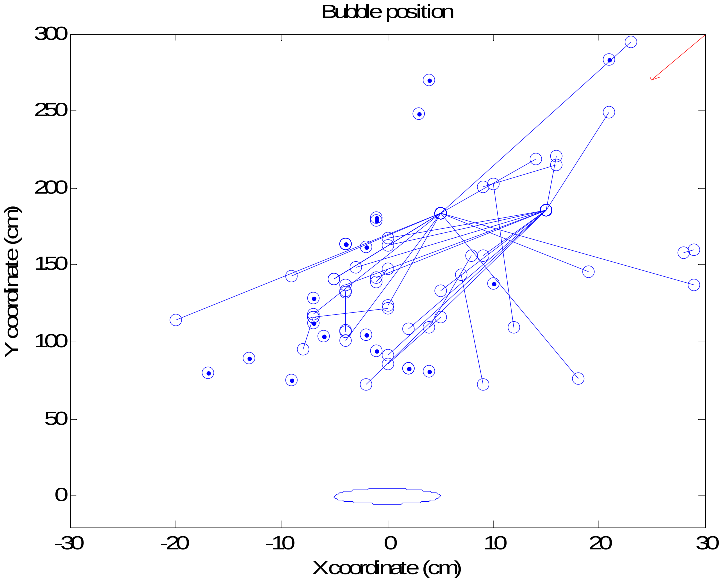

Figure 4-19 Trajectory diagram of air bubbles with oblique direction of water flow from (30,310) represented by red arrow

Figure 4-19 shows the air bubbles tracking result whithin 5s. Circle in Figure 4-19 represents the position of air bubbles and ellipse in the origin represents the position of the BAUV. The direction of water flow is about 30 degrees from the y-axis and is shown in Figure 4-19 represented by arrow. The trajectory of air bubbles is represented by lines in Figure 4-19. We can observe that the moving direction of air bubbles is similar to the water flow. However, there are errors in some trajectory of air bubbles. The reason might be the error in air bubbles detection or feature correspondence in stereo vision. If the corner incorrectly represents air bubbles, the trajectory might incorrectly represent the air bubbles. If the feature point incorrectly corresponds, three dimensional coordinate of air bubbles in each time may be wrong since it was affected by the calculated coordinate of coner in right or left camera and therefore shows incorrect trajectory. Besides, we can observe that there are some air bubbles represented by circle only without showing their trajectory. The reason might be that air bubble disappears in the second time of air bubble coordnate calculation. It means that we only have the coordinate of air bubble in the first time but without the coordinate of air bubble in the second time. It has no enough time to calculate their trajectory. The other reason might be the error in the air bubble trcking system that the algorithm fail to find the position of the air bubble in the following time. Because the Lucas-Kanade optical method mentioned in previous chapter has constraint that if the movement is too large, the moving point might be failed to track. Thus, the air bubble will be shown as one point without trajectory or incorrect trajectory.

4.1.3 Air Bubbles Observation

In this section, we set two water flow by water pipe in front of the BAUV to generate air bubbles which is shown in Figure 4-20. We will change the angle of water pipe to generate different angle set of water pipe which is shown in Figure 4-21 and we also set light above the air bubbles which is shown in Figure 4-22.

Figure 4-20 Setup of water pipes in tank experiment

Figure 4-21 Angle of water pipe in tank experiment

Figure 4-22 Lighting in tank experiment

The channel in this moving experiment is width: 119 cm, length: 550 cm. We will set two water flow and the distance between two water flow is 2.5m. We proceed with three set experiment with different angle of water pipe. Here, the water flow set is that first water flow with 30 degrees and the second one is that -30. The BAUV records the air bubbles position, moving angle and displacement relative to BAUV when moving.

Figure 4-23 shows the results of air bubbles position relative to BAUV when BAUV is moving. We could observe that there are no air bubbles position information recorded in t=11 to t=21. The reason is that when BAUV moving toward the water flow, the camera on BAUV is unable to see the air bubbles due to the short distance from first water flow but the large distance from second water flow.

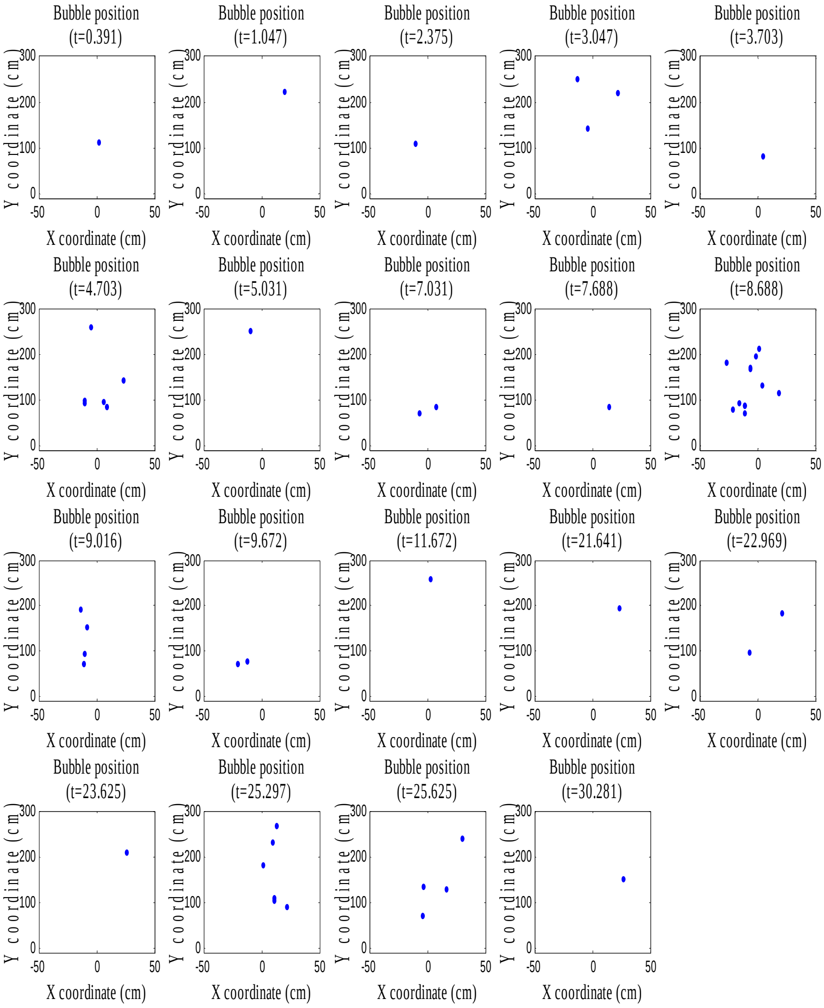

Figure 4-23 Air bubbles position relative to the BAUV

Figure 4-24 shows the results of air bubbles moving displacement relative to BAUV when BAUV is moving. The attack angle of BAUV is calculated based on the information in t =4.375 and t=24.969. The air bubbles with displacement in range 70 cm to 100cm are brought to be calculated for their moving angle based on the method mentioned in chapter 3.4.4.

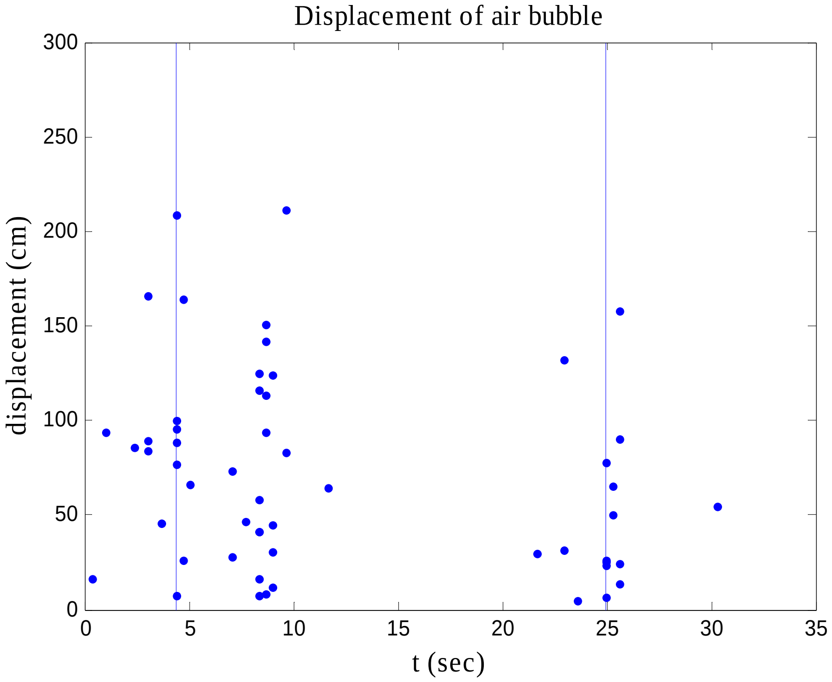

Figure 4-24 Air bubbles moving displacement relative to the BAUV

Figure 4-25 shows the results of air bubbles moving angle relative to BAUV when BAUV is moving. The attack angle of BAUV is calculated from air bubbles with moving angle

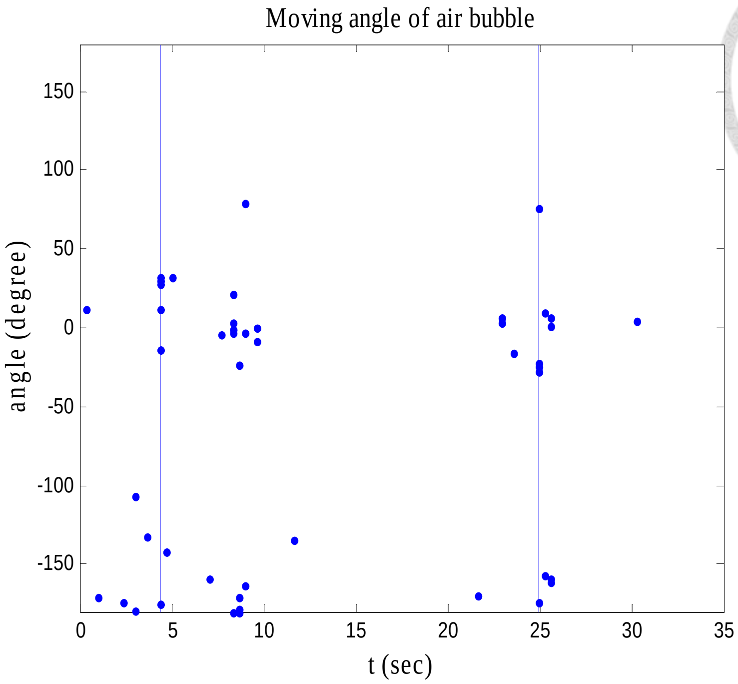

Figure 4-25 Air bubbles moving angle relative to the BAUV

According to the method mentioned in section 3.4.2, the goal attack angle of BAUV is calculated based on the moving angle of air bubbles in main displacement range of air bubbles. Table 4-3 shows the BAUV attack angle result calculated time and mainly based on the air bubbles displacement range of 0.7m and 0.2m.

Table 4-3 Observed information of air bubbles

	Displacement (m)	Calculated time (sec)	Attack angle (degree)
First	0.7	4.375	-10
Second	0.2	24.969	10.127

4.1.4 BAUV heading angle control

In this experiement, we set up two water flow by pump and water pipe to demonstrate our air bubbles tracking system and fish control system, we set up three kinds of water current set with different flow angle. First water flow set is that first water flow with 30 degrees and the second one is -45 degrees. Second water flow set is that first water flow with 30 degrees and the second one is that -30 degrees. The last water flow set is first water flow with 45 degrees and the second one is -30 degrees. Figure 4-26 shows the illustration of two water flows and the BAUV position in the water channel.

Figure 4-26 Illustration of the position of two water flows and the BAUV in the water channel

Firstly, we present our trajectory based on the water flow with 30 degrees in Figure 4-27 and the second one is -45 degrees in Figure 4-28 before BAUV moving angle is calculated.

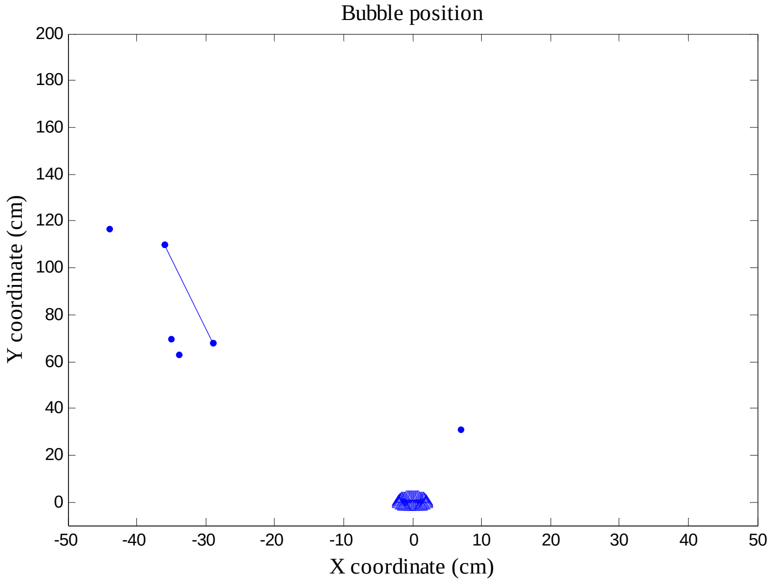

Figure 4-27 Trajectory of air bubbles of 30 degrees water current

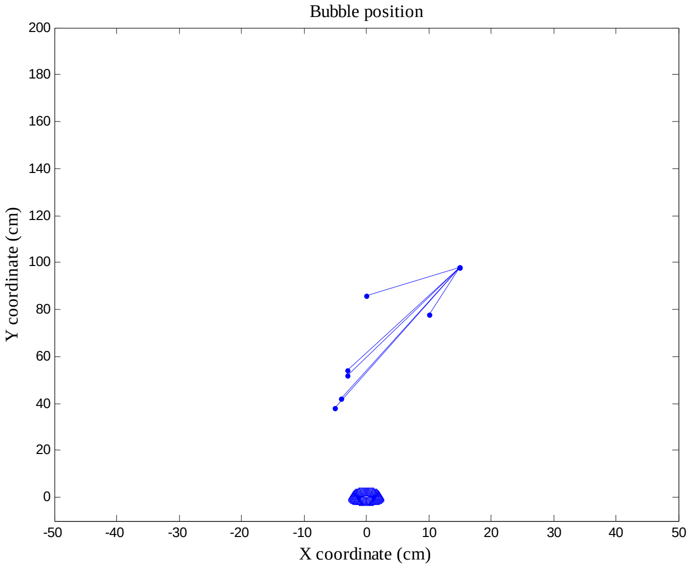

Figure 4-28 Trajectory of air bubbles of -45 degrees water current

In Figure 4-27 and Figure 4-28, the point represents the position of air bubbles and the line represents the trajectory of air bubbles. The block in the origin represents the position of our BAUV.

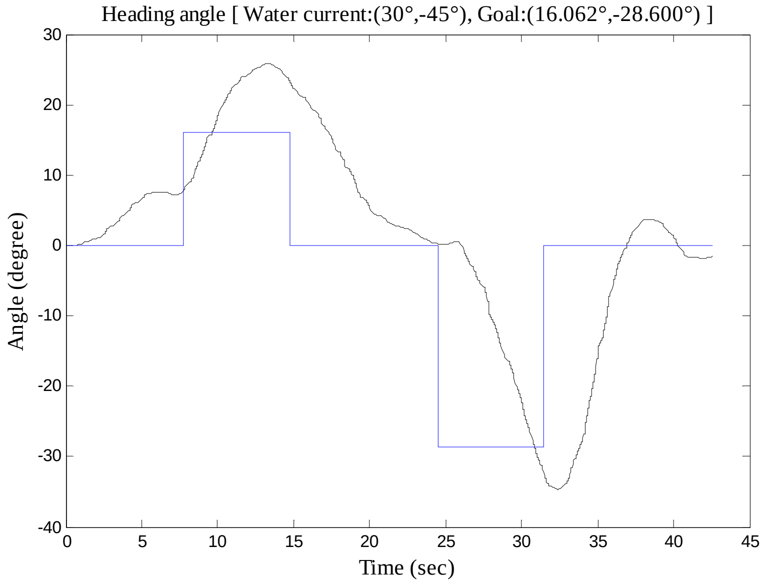

Figure 4-29 Relationship between heading angle versus time

and water current with 30 degrees and -45 degrees,

goal angles are 16.062 degrees and -28.600 degrees

The BAUV firstly detects air bubbles by cameras, tracks their trajectory to determine the angle of water current, and decides the heading angle of BAUV based on the information from air bubbles shown in Figure. 4-27 and 4-28. Figure 4-29 shows the result of decided heading angle to control based on the chapter 3.4.1 and the control result of BAUV’s heading angle versus time. The angle of water current are 30 degrees and -45 degrees. From t=0s to t=6.764s, BAUV is affected by water current, but BAUV tried to control its heading angle to 0 degree. From t=6.764s to t=13.764s, BAUV control its heading angle to 16.062 degree based on the observation information from first water current with 30 degrees to y axis which can be seen in Figure 4-27. From t=13.764s to t=19.875s, BAUV controls its heading to 0 degree when BAUV observes the information from the next water current. During t=23.325s to t=31.458s, BAUV controls its heading angle to -28.6 degree which is decided based on the observation information from second water current with -45 degrees to the y axis which can be seen in Figure 4-28. During t=31.458s to t=42.577s, BAUV controls its heading angle to 0 degree.

Seccondly, we present our result based on the water flow with 30 degrees in Figure 4-30 and the second one is -30 degrees in Figure 4-31 before BAUV heading angle is calculated.

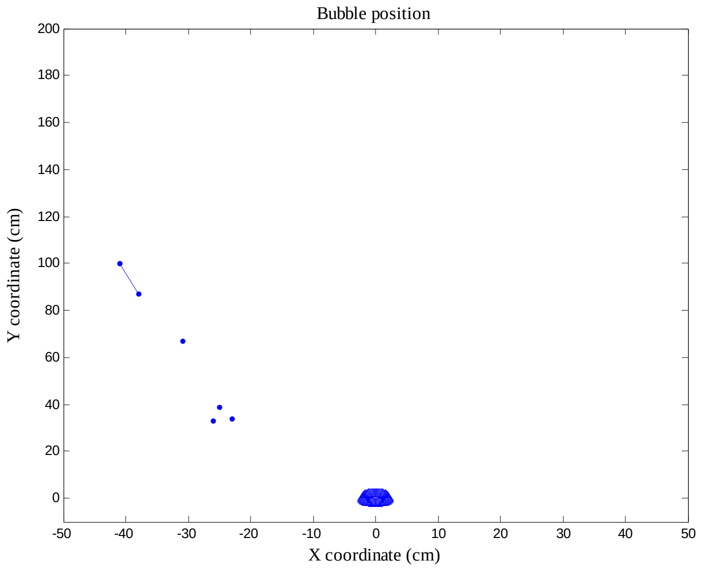

Figure 4-30 Trajectory of air bubbles of 30 degrees water current

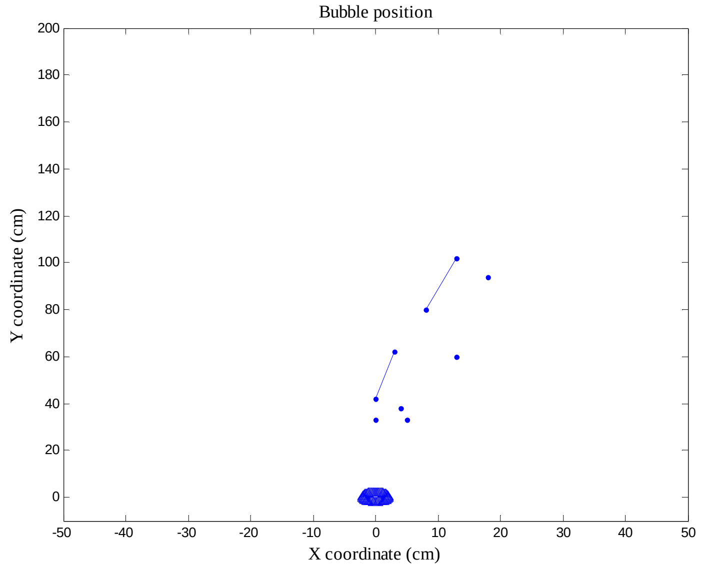

Figure 4-31 Trajectory of air bubbles of -30 degrees water current

In Figure 4-30 and Figure 4-31 the point represents the position of air bubbles and the line represents the trajectory of air bubbles. The block in the origin represents the position of our BAUV.

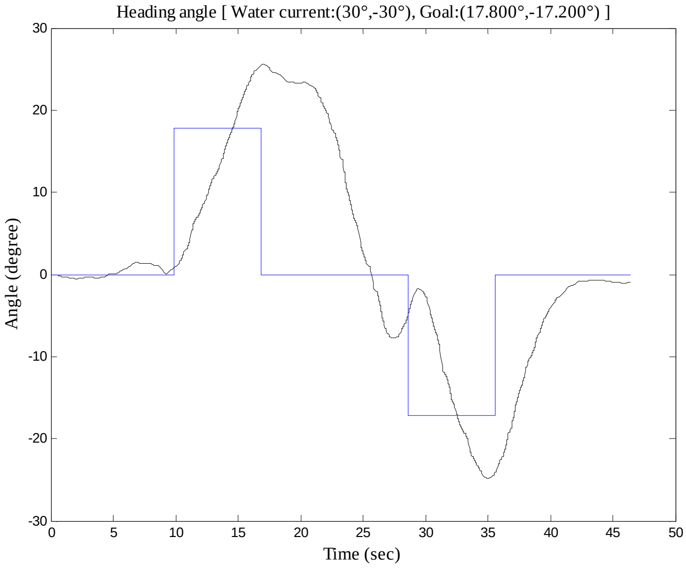

Figure 4-32 Relationship between heading angle versus time and water current with 30 degrees and -30 degrees, goal angles are 17.800 degrees and -17.200 degrees

The BAUV firstly detects air bubbles by cameras, tracks their trajectory to determine the angle of water current, and decides the heading angle of BAUV based on the information from air bubbles shown in Figure 4-30 and Figure 4-31. Figure 4-32 shows the result of decided heading angle to control based on the chapter 3.4.1 and the virtual control result of BAUV heading angle versus time. The angle of water current are 30 degrees and -30 degrees. From t=0s to t=8.879s, BAUV is affected by water current, but BAUV tried to control its heading angle to 0 degree. From t=6.764s to t=15.833s, BAUV control its heading angle to 17.8 degrees based on the observation information from first water current with 30 degrees which can be seen in Figure 4-30. From t=15.833s to t=24.427s, BAUV control its heading to 0 degree when BAUV observes the information from next one water current. From t=28.160s to t=35.331s, BAUV control its heading angle to -17.2 degrees based on the observation information from second water current with -30 degrees which can be seen in Figure 4-31. From t=35.331s to t=46.442s, BAUV control its heading angle to 0 degree.

Finally, we present our result based on the water flow with 30 degrees in Figure 4-33 and the second one is -30 degrees in Figure 4-34.

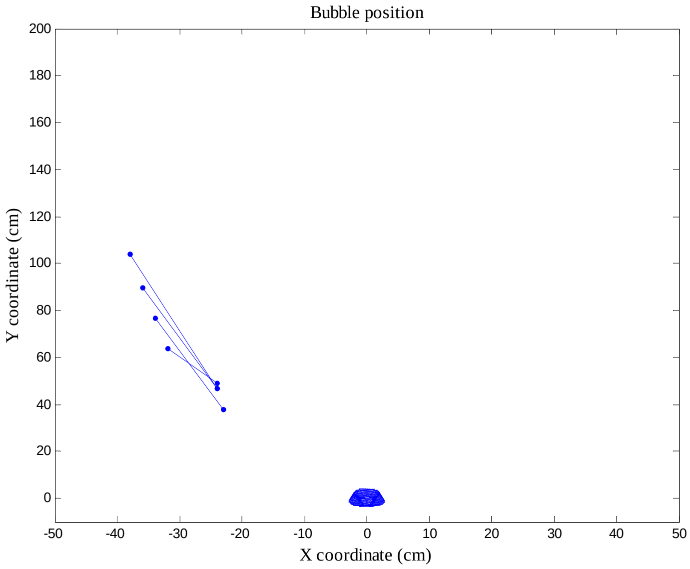

Figure 4-33 Trajectory of air bubbles of 45 degrees water current

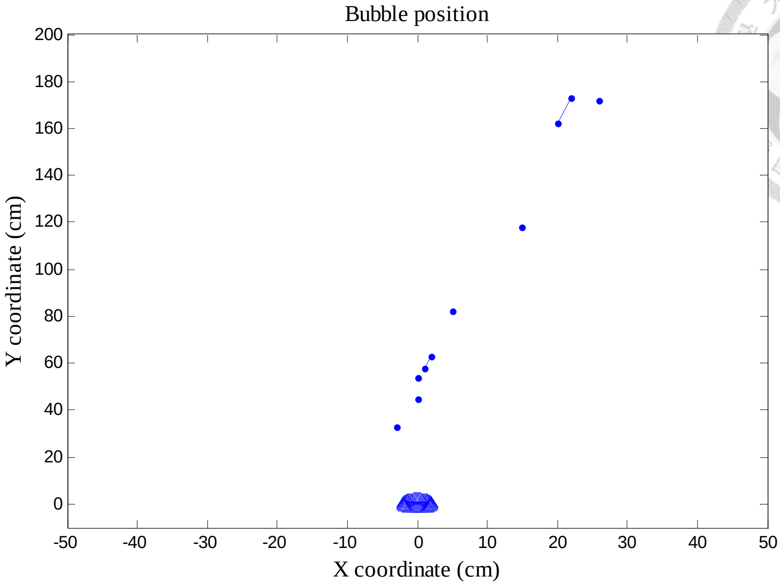

Figure 4-34 Trajectory of air bubbles of -30 degrees water current

In Figure 4-33 and Figure 4-34, the point represents the position of air bubbles and the line represents the trajectory of air bubbles. The block in the origin represents the position of our BAUV.

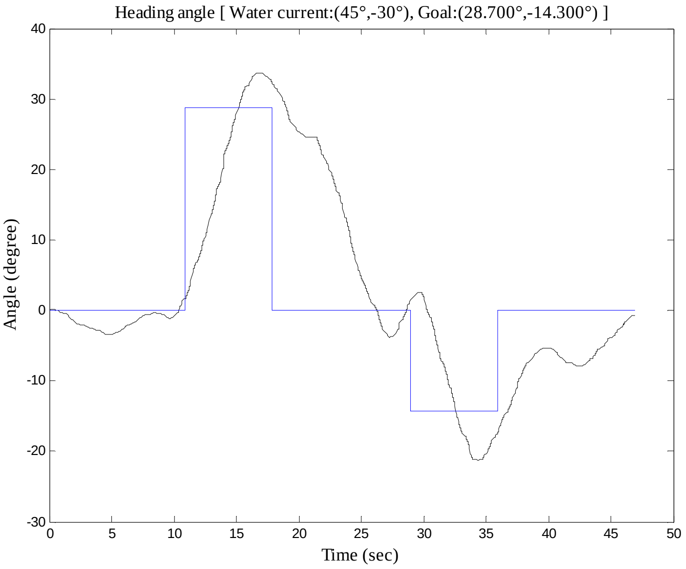

Figure 4-35 Relationship between heading angle versus time

in water currents of 45 degrees and -30 degrees, goal angle goal angle

is 28.7 degrees and -14.3 degrees

The BAUV firstly detects air bubbles by cameras, tracks their trajectory to determine the angle of water current, and decides the heading angle of BAUV based on the information from air bubbles shown in Figure 4-33 and Figure 4-34. Figure 4-35 shows the result of decided heading angle to control based on the chapter 3.4.1 and the virtual control result of BAUV heading angle versus time. The angle of water current are 45 degrees and -30 degrees. From t=0s to t=9.853s, BAUV is affected by water current, but BAUV tried to control its heading angle to 0 degree. From t=9.853s to t=17.234s, BAUV control its heading angle to 28.7 degrees based on the observation information from first water current. From t=17.234s to t=24.532s, BAUV control its heading to 0 degree when BAUV observes the information from next one water current. From t=27.486s to t=36.522s, BAUV control its heading angle to -14.3 degrees based on the observation information from second water current. From t=36.522s to t=46.923s, BAUV control its heading angle to 0 degree.

Obviously, we can observe that in Fgure 4-28, Figure 4-31, Figure 4-33, and Figure 4-34, the detected moving angle of air bubbles are not only one. Therefore, we have to analyze main moving angles of air bubbles based on chapter 3.4.2 and the BAUV moving angle is then calculated.

4.1.5 Energy consumption

In this experiment, we use Arduino Uno board and ACS712 to record power consumption of BAUV. Figure 2-14 shows the Arduino Uno board, and Figure 2-15 shows the Arduino module ACS712 which is based on Hall effect to measure the current of propeller. The output error of ACS712 is about 1.5% and the signal of ATC712 will return to computer through the Arduino uno board. We conduct experiment based on the water flow set shown in section 4.1.3. We compare the energy consumption of propeller and turning motor in two motion mode that one is motion with attack angle control and the other is without attack angle control. The same goal line and same angular velocity of propeller is set to both experiment mode

First water flow set is that first water flow with 30 degrees and the second one is -45 degrees. The propeller energy consumption result are shown in Figure 4-36. We could find that the average of propeller energy consumption of motion mode without attack angle control are approximately more than the mode with attack angle control. We could find that the time in two mode are different, because the path in the mode with attack angle is longer than the path in the mode without attack angle.

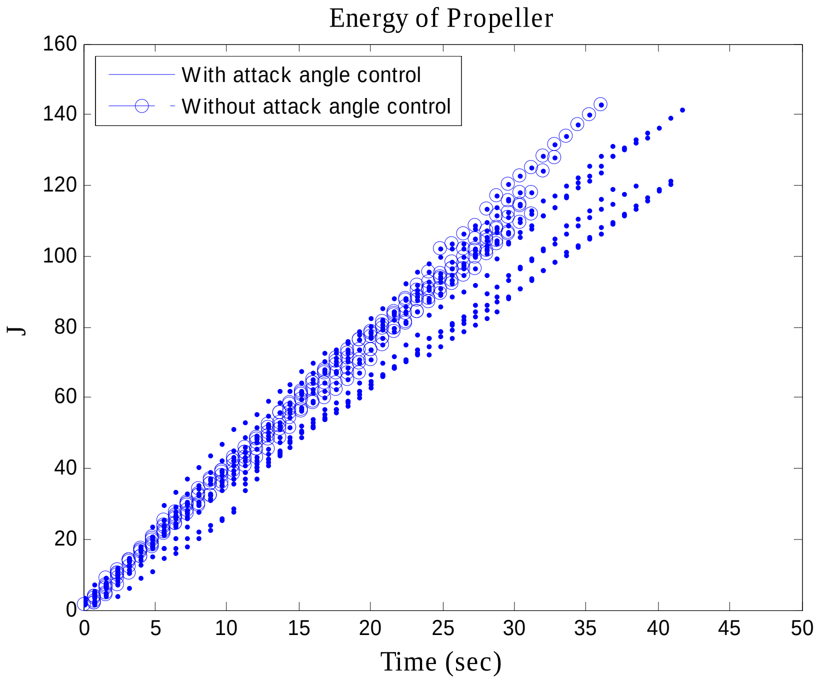

Figure 4-36 Comparison of propeller energy consumptions in two different modes

In the table 4-4 shows the average energy consumption in two mode. The percentage of energy saving is about 5.72%.

Table 4-4 Average propeller energy consumption

30,-45 degrees	Energy consumption (J)	Energy saving
Without attack angle	127.1228	5.72%
With attack angle	119.8511	

The turning motor energy consumption result are shown in Figure 4-37. We could find that the average of turning motor energy consumption of motion mode with attack angle control are slightly more than the mode without attack angle control.

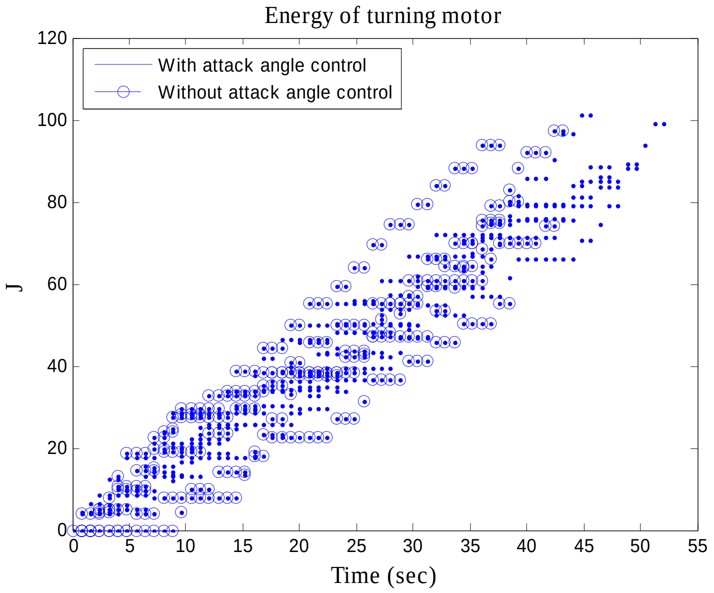

Figure 4-37 Comparison of turning motor energy consumption in two different modes

In the table 4-5 shows the average energy consumption of turning motor in two mode. However, turning motor energy consumption of mode with attack angle control is more than without attack angle control, the reason is that the turning motor consume more energy when turning more heading angle.

Table 4-5 Average turning motor energy consumption

30,-45 degrees	Energy consumption (J)	Energy saving
Without attack angle	76.4423	-13.4%
With attack angle	88.3225	

Table 4-6 shows total energy consumption of turning motor and propeller. Despite propeller energy consumption of attack angle control mode is less than without attack angle control mode, the turning motor energy consumption of attack angle control mode is more than the mode with no attack angle control. Therefore, the total energy consumption between two control modes is approximately equal to each other.

Table 4-6 Average energy consumption of turning motor and propeller

30,-45 degrees	Energy consumption (J)	Energy saving
Without attack angle	203.5651	-2.26%
With attack angle	208.1736	

The second experiment setup is that the first water flow has a 30-degrees input and the second one is -30 degrees. The energy consumption results are shown in Fig, 4-38.

We could find that the average of energy consumption without attack angle control is also approximately more than the mode with attack angle control. Most travel times of the attack angle mode are also longer than the ones in the mode without attack angle.

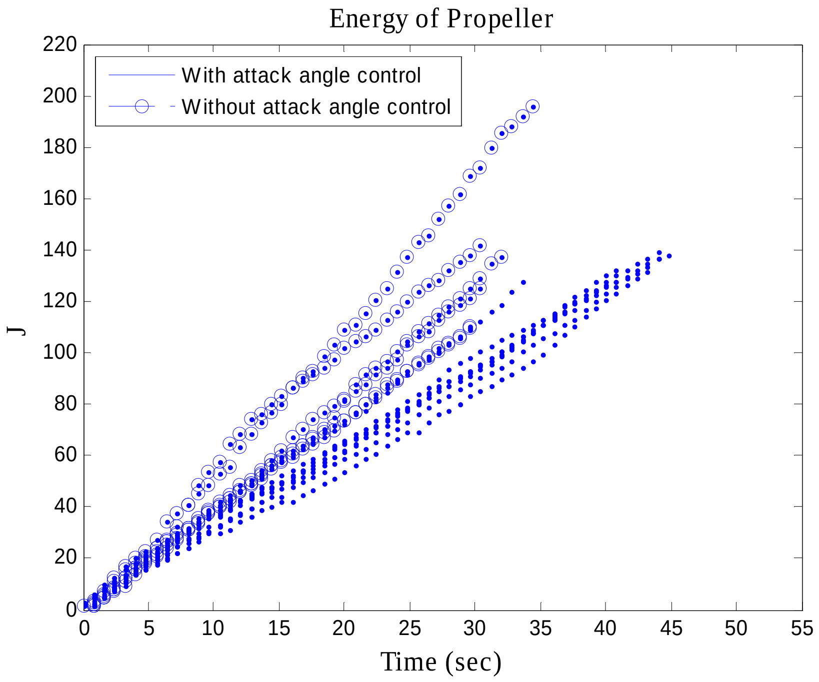

Figure 4-38 Comparison of propeller energy consumption in two different mode

Table 4-7 shows the average energy consumption in two modes. The percentage of energy saving is about 2.43%.

Table 4-7 Average propeller energy consumption

30,-30 degrees	Energy consumption (J)	Energy saving
Without attack angle	135.3385	2.43%
With attack angle	132.0459	

The turning motor energy consumption results are shown in Fig. 4-39. It is found that the average of turning motor energy consumption of motion mode with attack angle control are slightly more than the mode without attack angle control.

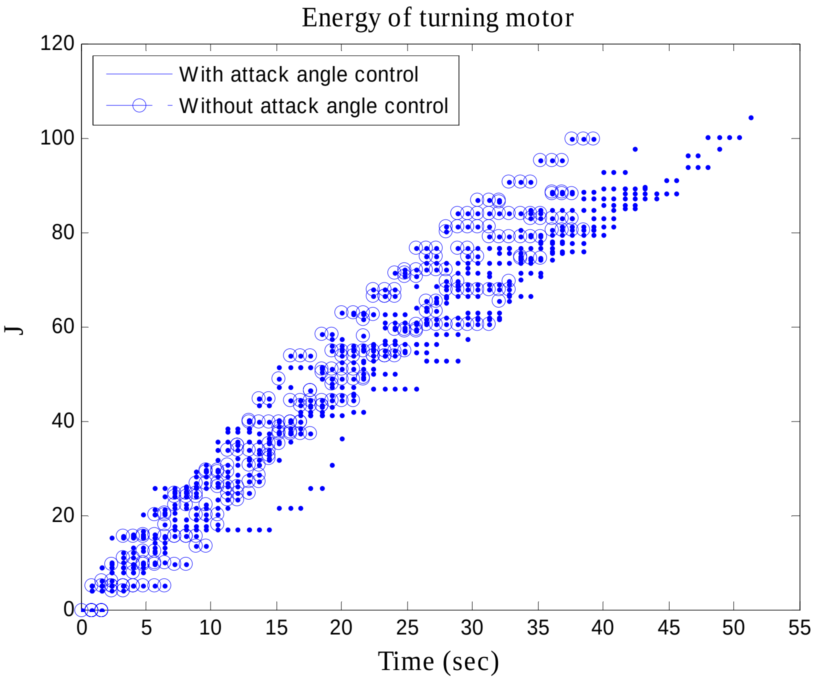

Figure 4-39 Comparison of turning motor energy consumption in two different mode

Table 4-8 shows the average energy consumption of turning motor in two modes. However, turning motor energy consumption of mode with attack angle control is more than without attack angle control, the reason is that the turning motor consumes more energy when turning large heading angles.

Table 4-8 Average turning motor energy consumption

30,-30 degrees	Energy consumption (J)	Energy saving
Without attack angle	85.8752	-8.01%
With attack angle	93.3570	

Table 4-9 shows total energy consumption of turning motor and propeller. Despite propeller energy consumption of attack angle control mode is less than without attack angle control mode, the turning motor energy consumption of attack angle control mode is more than the mode of no attack angle control. Therefore, total energy consumption between two control modes is approximately equal to each other.

Table 4-9 Average energy consumption of turning motor and propeller

30,-30 degrees	Energy consumption (J)	Energy saving
Without attack angle	221.2137	-1.85%
With attack angle	225.4029	

The third experimental setup is that first water flow is 45 degrees and the second one is -30 degrees. The propeller energy consumption results are shown in Fig. 4-40. We could find that the average of energy consumption of motion mode without attack angle control is also approximately more than the mode with attack angle control. Some travel times in with attack angle are also longer than the mode without attack angle.

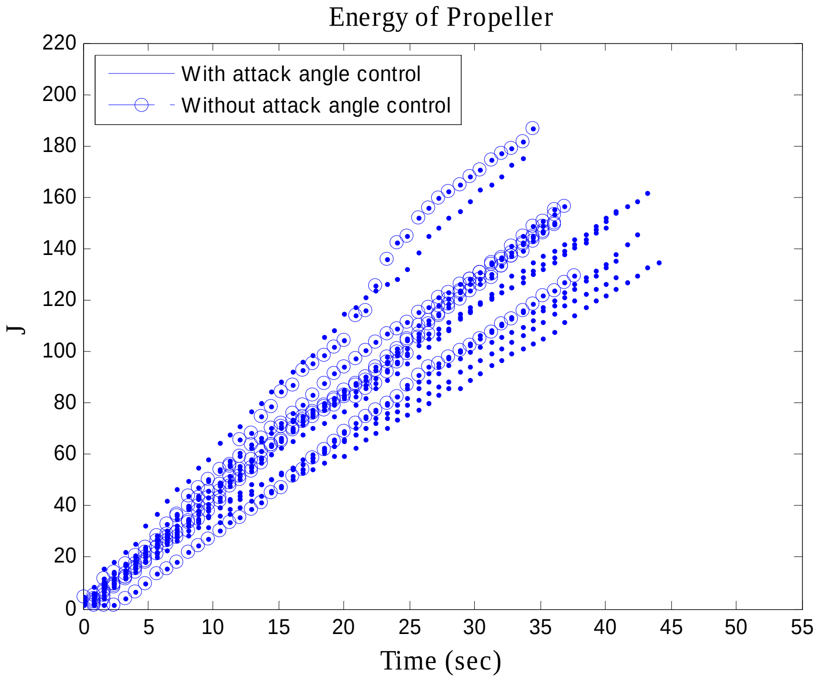

Figure 4-40 Comparison of energy consumption in two different mode

In the table 4-10 shows the average energy consumption of propeller in two modes. The percentage of energy saving is about 8.53%.

Table 4-10 Average energy consumption of propeller

45,-30 degrees	Energy consumption (J)	Energy saving
Without attack angle	157.5412	8.53%
With attack angle	144.1046	

The turning motor energy consumption results are shown in Fig. 4-41. We could find that the average of turning motor energy consumption of motion mode with attack angle control is slightly more than the mode without attack angle control.

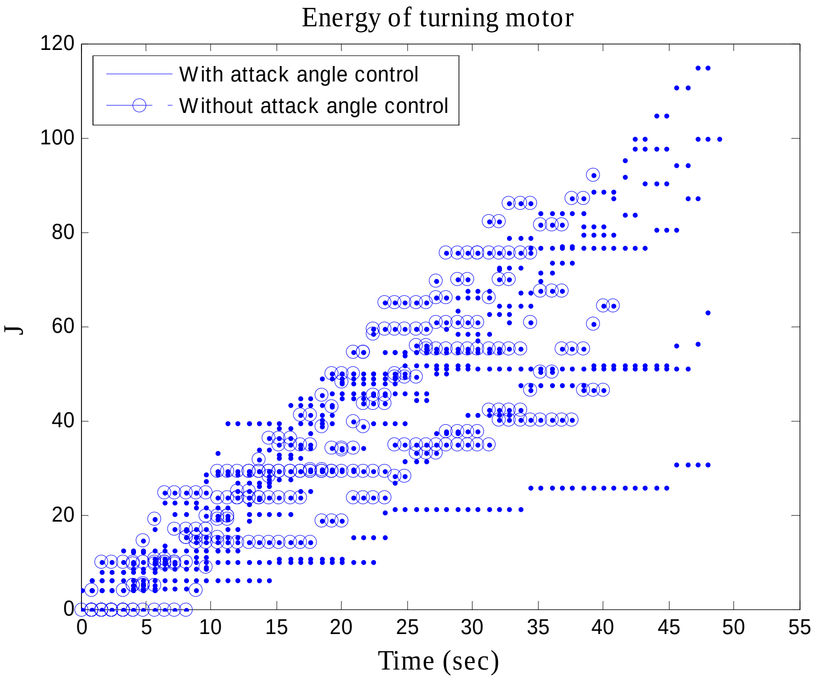

Figure 4-41 Comparison of turning motor energy consumption in two different mode

Table 4-11 shows the average energy consumption of turning motor in two modes.

However, turning motor energy consumption of mode with attack angle control is more than without attack angle control, the reason is that the turning motor consumes more energy when turning more heading angle.

Table 4-11 Average turning motor energy consumption

45,-30 degrees	Energy consumption (J)	Energy saving
Without attack angle	69.8995	-10.9%
With attack angle	78.4680	

Table 4-12 shows total energy consumption of turning motor and propeller. Despite propeller energy consumption of attack angle control mode is less than without attack angle control mode, the turning motor energy consumption of attack angle control mode is more than without attack angle control mode. Therefore, total energy consumption between two control modes is approximately equal to each other.

Table 4-12 Average energy consumption of turning motor and propeller

45,-30 degrees	Energy consumption (J)	Energy saving
Without attack angle	227.4407	-2.14%
With attack angle	222.5726	

Chapter 5 Conclusion

This chapter is a summary for the contribution on the development of air bubbles detection and tracking method on our BAUV control system. Firstly, the air bubbles entrainment, dynamics and motion have been introduced. Then, air bubbles detection method has been proposed based on feature description method: the Harris corner and neighborhood process. The Lucas-Kanade optical method strengthened by image pyramid was developed for air bubbles tracking. Sum of absolute differences was utilized on stereo vision correspondence. Air bubbles observation system was built combining air bubbles detection and tracking method by stereo vision technology. Lift force and drag force in different attack angle was analyzed. Convergence time of heading angle control was simulated for the determination of proper preview distances. Finally, experimental results have been demonstrated to verify our air bubbles observation method on BAUV control system. Besides, the energy consumption of BAUV has been demonstrated and the results show that our BAUV could not save energy by control the attack angles in an alternating flow. The reason might be that our BAUV uses two motors to control the forward speed and the turning rate, respectively. It is not totally bionic-like. We suggest in the future to check the BAUV design with only a single motor/actuator for the motion control of the forward speed and the turning.

References

[1] F. Resch and e. H. Leutheusser, "Le ressaut hydraulique: mesures de turbulence dans la region diphasique," *La Houille Blanche*, pp. 279-293, 1972.

[2] T. E. McMahon and G. F. Hartman, "Influence of cover complexity and current velocity on winter habitat use by juvenile coho salmon (Oncorhynchus kisutch)," *Canadian Journal of Fisheries and Aquatic Sciences,* vol. 46, pp. 1551-1557, 1989.

[3] J. C. Liao, D. N. Beal, G. V. Lauder, and M. S. Triantafyllou, "Fish exploiting vortices decrease muscle activity," *Science,* vol. 302, pp. 1566-1569, 2003.

[4] J. C. Liao, "Neuromuscular control of trout swimming in a vortex street: implications for energy economy during the Karman gait," *Journal of Experimental Biology,* vol. 207, pp. 3495-3506, 2004.

[5] J. Bellingham and J. S. Willcox, "Optimizing AUV oceanographic surveys," in *Autonomous Underwater Vehicle Technology, 1996. AUV'96., Proceedings of the 1996 Symposium on*, 1996, pp. 391-398.

[6] A. Alvarez, A. Caiti, and R. Onken, "Evolutionary path planning for autonomous underwater vehicles in a variable ocean," *Oceanic*

Engineering, IEEE Journal of, vol. 29, pp. 418-429, 2004.

[7] H. Chanson, "Air-water gas transfer at hydraulic jump with partially developed inflow," *Water Research,* vol. 29, pp. 2247-2254, 1995.

[8] G. B. Deane and M. D. Stokes, "Scale dependence of bubble creation mechanisms in breaking waves," *Nature,* vol. 418, pp. 839-844, 2002.

[9] E. Delnoij, F. Lammers, J. Kuipers, and W. Van Swaaij, "Dynamic simulation of dispersed gas-liquid two-phase flow using a discrete bubble model," *Chemical Engineering Science,* vol. 52, pp. 1429-1458, 1997.

[10] K. Felton and E. Loth, "Spherical bubble motion in a turbulent boundary layer," *Physics of Fluids (1994-present),* vol. 13, pp. 2564-2577, 2001.

[11] M. Maxey, B. Patel, E. Chang, and L.-P. Wang, "Simulations of dispersed turbulent multiphase flow," Fluid Dynamics Research, vol. 20, pp. 143-156, 1997.

[12] O. Druzhinin and S. Elghobashi, "Direct numerical simulations of bubble-laden turbulent flows using the two-fluid formulation," *Physics of Fluids (1994-present),* vol. 10, pp. 685-697, 1998.

[13] D. Peregrine, "Surf zone currents," *Theoretical and computational fluid dynamics,* vol. 10, pp. 295-309, 1998.

[14] J.-Y. Bouguet, "Pyramidal implementation of the affine lucas kanade feature

tracker description of the algorithm," *Intel Corporation,* vol. 5, pp. 1-10, 2001.

[15] I. M. Mazzitelli and D. Lohse, "Lagrangian statistics for fluid particles and bubbles in turbulence," *New Journal of Physics,* vol. 6, p. 203, 2004.

[16] A. Talukder, S. Goldberg, L. Matthies, and A. Ansar, "Real-time detection of moving objects in a dynamic scene from moving robotic vehicles," in *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on,* 2003, pp. 1308-1313.

[17] H. Chanson, "Turbulent air–water flows in hydraulic structures: dynamic similarity and scale effects," *Environmental Fluid Mechanics,* vol. 9, pp. 125-142, 2009.

Appendix

Table 0-1 Coefficient of simulation of heading angle control

$J_{bz} - N_{\dot{r}}$	2
$\rho A_r \alpha C_{MX}$	-0.5
$2\rho A r a^2 C_{Mr}$	8.5
T	27.7805N

Table 0-2 C_L and C_D versus alpha

alpha	C_L	C_D
0.000	-0.0069	0.00859
0.250	0.0226	0.00857
0.750	0.0811	0.00858
1.000	0.1109	0.00857
1.250	0.1409	0.00856
1.500	0.1705	0.00858
1.750	0.2005	0.00860

2.000	0.2308	0.00862
2.250	0.2603	0.00869
2.500	0.2907	0.00872
2.750	0.3203	0.00881
3.000	0.3499	0.00891
3.250	0.3793	0.00904
3.500	0.4087	0.00919
3.750	0.4376	0.00938
4.000	0.4662	0.00958
4.250	0.4959	0.00975
4.500	0.5262	0.00990
4.750	0.5561	0.01008
5.000	0.5864	0.01021
5.250	0.6159	0.01042
5.500	0.6479	0.01056
5.750	0.6801	0.01074
6.000	0.7122	0.01091
6.250	0.7439	0.01105

6.500	0.7767	0.01126
6.750	0.8098	0.01141
7.000	0.8418	0.01160
7.250	0.8743	0.01183
7.500	0.9072	0.01203
7.750	0.9400	0.01227
8.000	0.9741	0.01252
8.250	1.0092	0.01280
8.500	1.0473	0.01321
8.750	1.0776	0.01380
9.000	1.1033	0.01451
9.250	1.1282	0.01511
9.500	1.1526	0.01559
9.750	1.1749	0.01605
10.000	1.1962	0.01650
10.250	1.2184	0.01697
10.500	1.2391	0.01747
10.750	1.2614	0.01794

11.000	1.2823	0.01846
11.250	1.3020	0.01906
11.500	1.3223	0.01960
11.750	1.3422	0.02019
12.000	1.3609	0.02083
12.250	1.3767	0.02156
12.500	1.3851	0.02238
12.750	1.3968	0.02325
13.000	1.4081	0.02428
13.250	1.4186	0.02545
13.500	1.4291	0.02677
13.750	1.4408	0.02822
14.000	1.4505	0.02988
14.250	1.4586	0.03177
14.500	1.4657	0.03390
14.750	1.4720	0.03629
15.000	1.4791	0.03857
15.250	1.4840	0.04106

15.500	1.4880	0.04362
15.750	1.4923	0.04641
16.000	1.4934	0.04950
16.250	1.4924	0.05282
16.500	1.4895	0.05642
17.000	1.4811	0.06437
17.250	1.4762	0.06872
17.500	1.4700	0.07355
17.750	1.4621	0.07864
18.000	1.4530	0.08393
18.500	1.4311	0.09551
18.750	1.4190	0.10195
19.000	1.3771	0.10456

Results of propeelr in 30 degrees and -45 degrees are shown in Figure 0-1 to Figure 0-14:

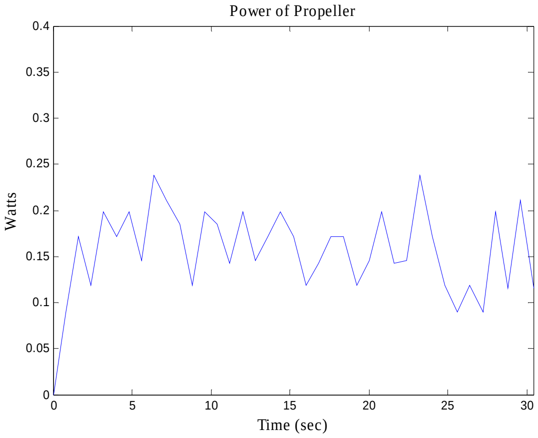

Figure 0-1 Power of propeller without attack angle control (1)

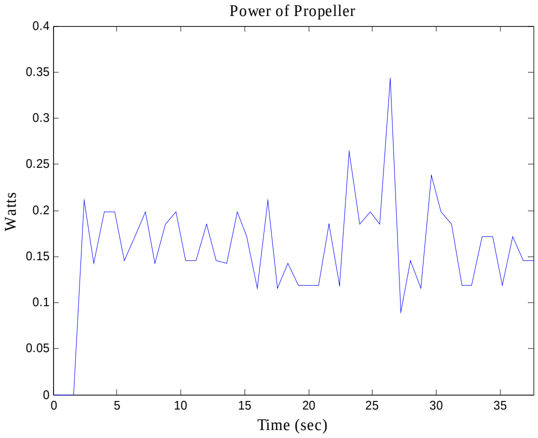

Figure 0-2 Power of propeller without attack angle control (2)

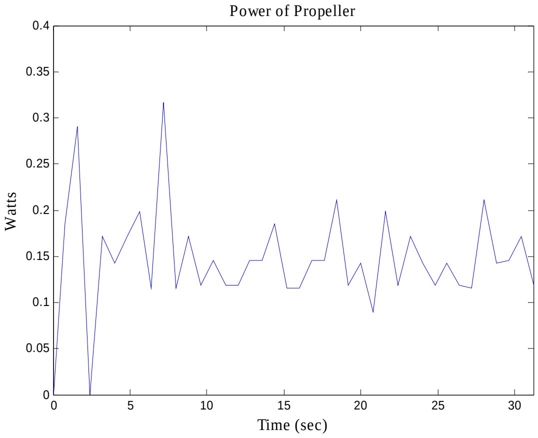

Figure 0-3 Power of propeller without attack angle control (3)

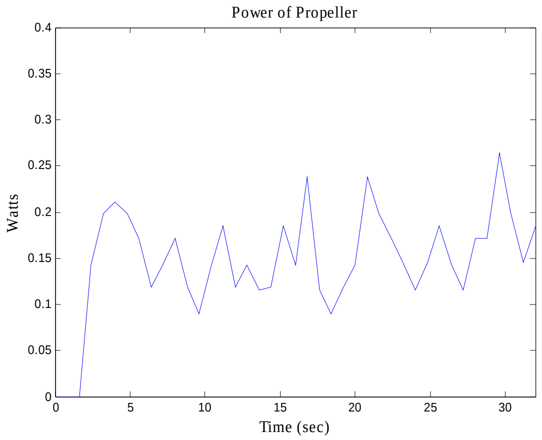

Figure 0-4 Power of propeller without attack angle control (4)

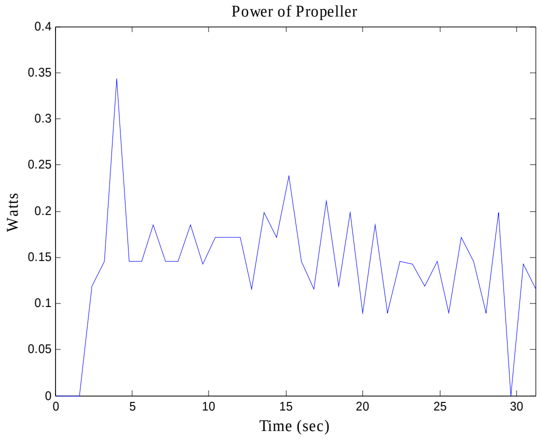

Figure 0-5 Power of propeller without attack angle control (5)

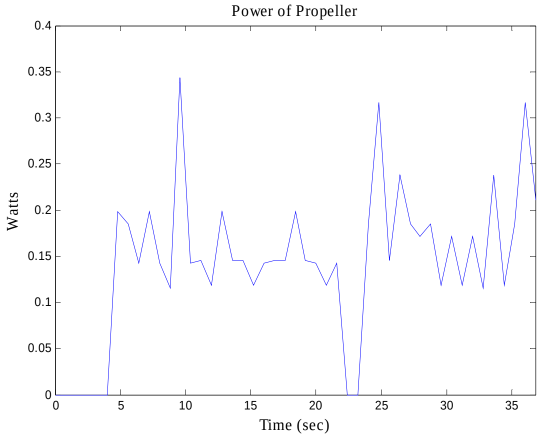

Figure 0-6 Power of propeller without attack angle control (6)

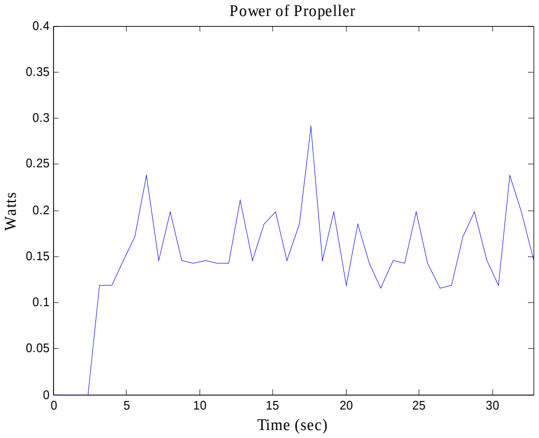

Figure 0-7 Power of propeller without attack angle control (7)

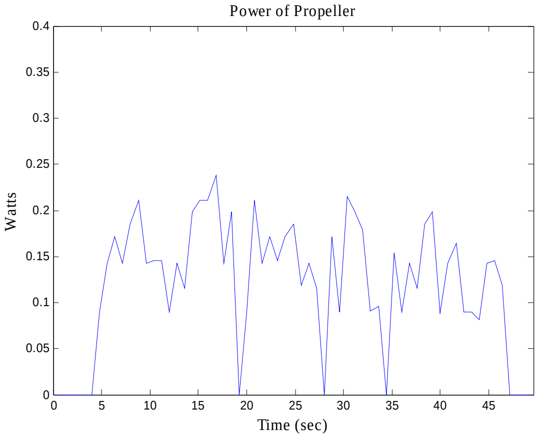

Figure 0-8 Power of propeller with attack angle control (1)

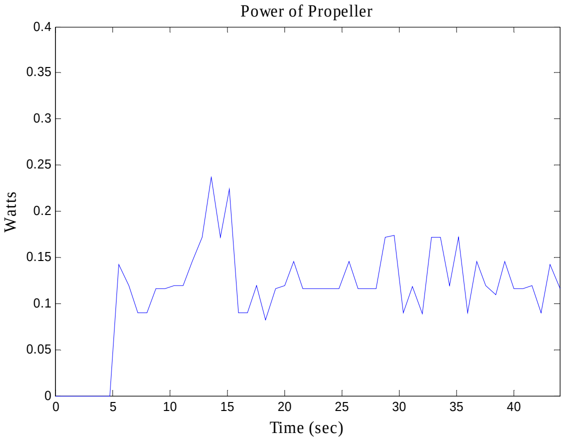

Figure 0-9 Power of propeller with attack angle control (2)

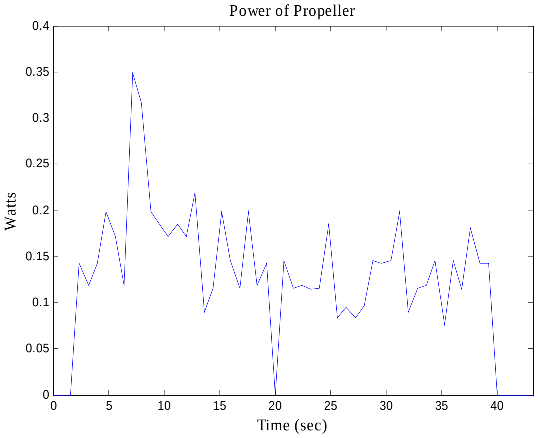

Figure 0-10 Power of propeller with attack angle control (3)

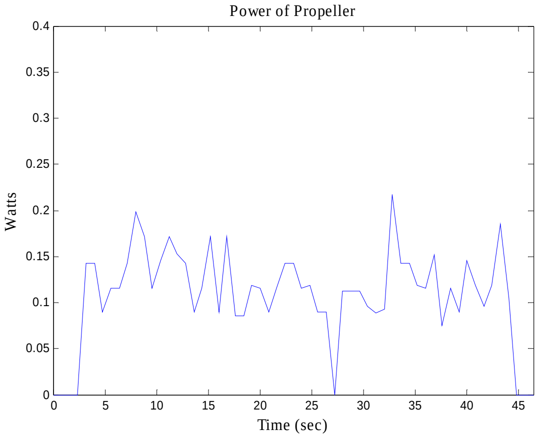

Figure 0-11 Power of propeller with attack angle control (4)

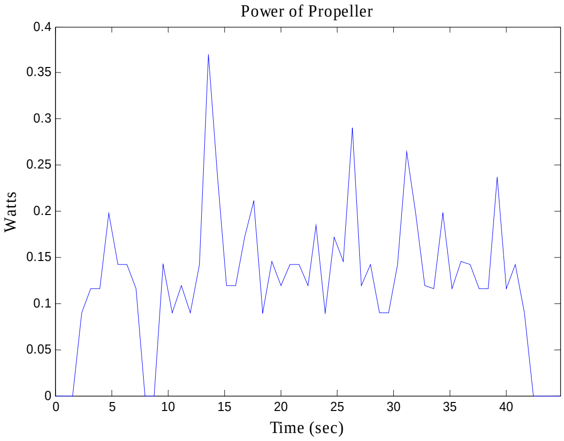

Figure 0-12 Power of propeller with attack angle control (5)

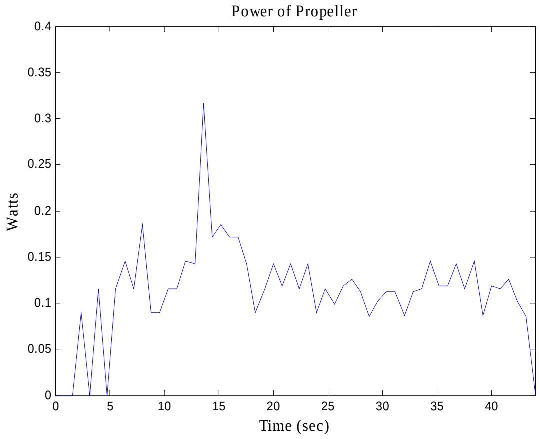

Figure 0-13 Power of propeller with attack angle control (6)

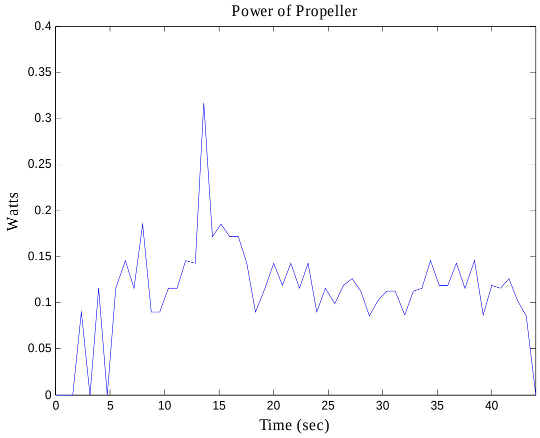

Figure 0-14 Power of propeller with attack angle control (7)

Results of propellr in 30 degrees and -30 degrees are shown in Figure 0-15 to Figure 0-28:

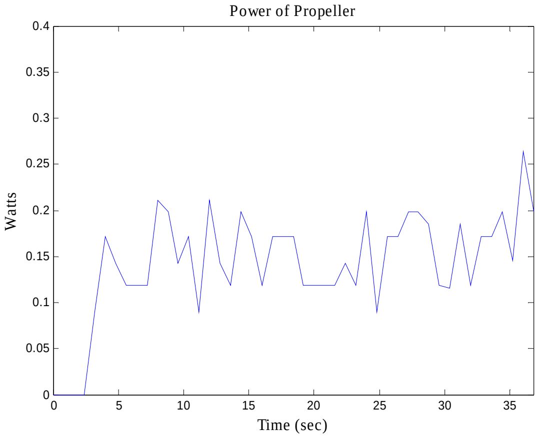

Figure 0-15 Power of propeller without attack angle control (1)

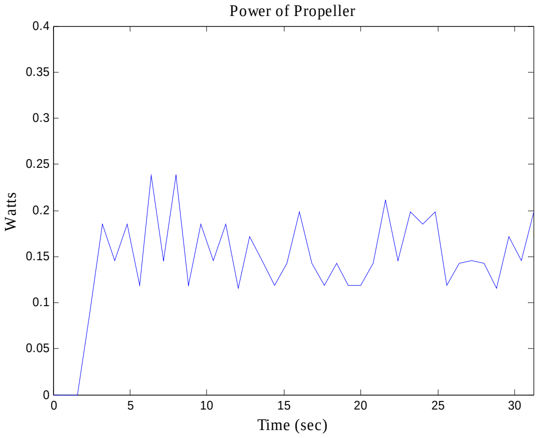

Figure 0-16 Power of propeller without attack angle control (2)

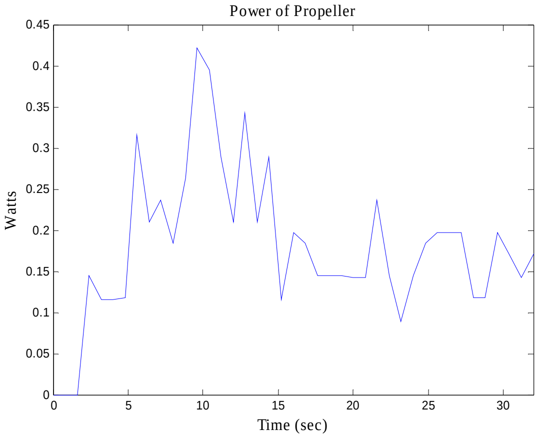

Figure 0-17 Power of propeller without attack angle control (3)

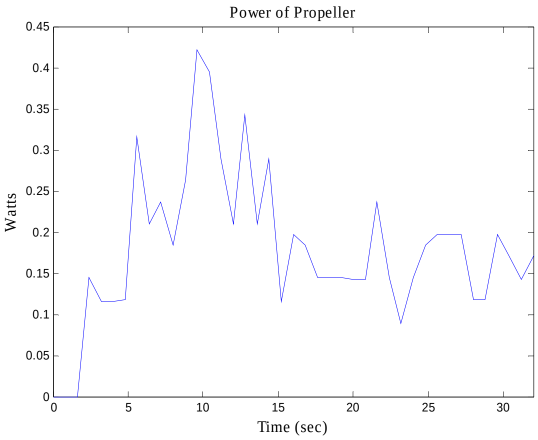

Figure 0-18 Power of propeller without attack angle control (4)

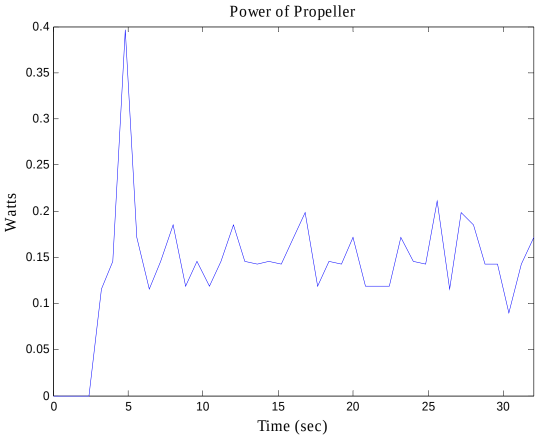

Figure 0-19 Power of propeller without attack angle control (5)

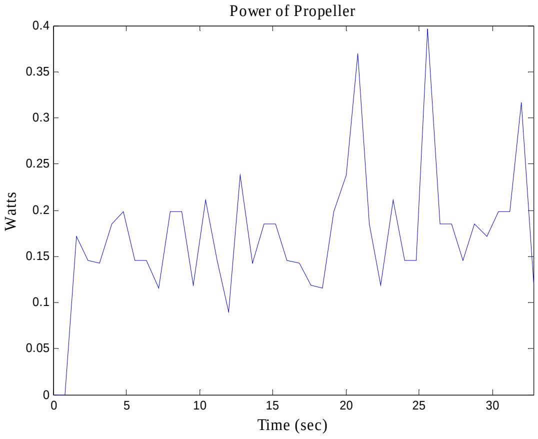

Figure 0-20 Power of propeller without attack angle control (6)

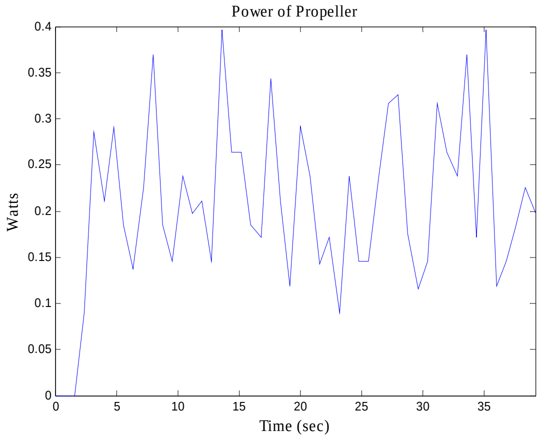

Figure 0-21 Power of propeller without attack angle control (7)

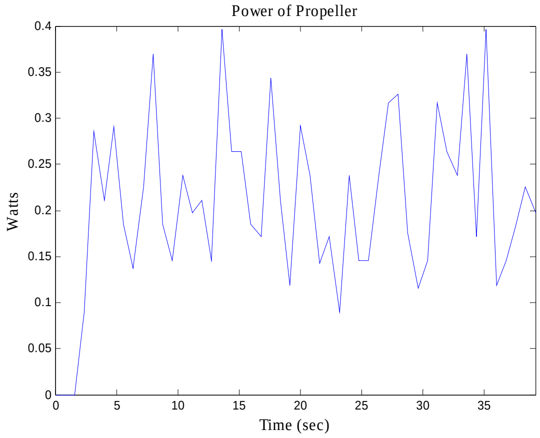

Figure 0-22 Power of propeller with attack angle control (1)

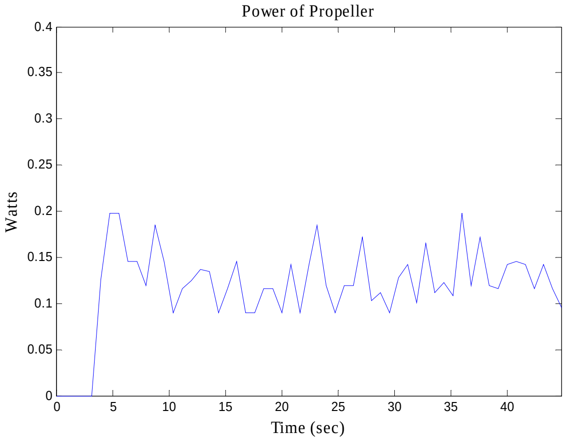

Figure 0-23 Power of propeller with attack angle control (2)

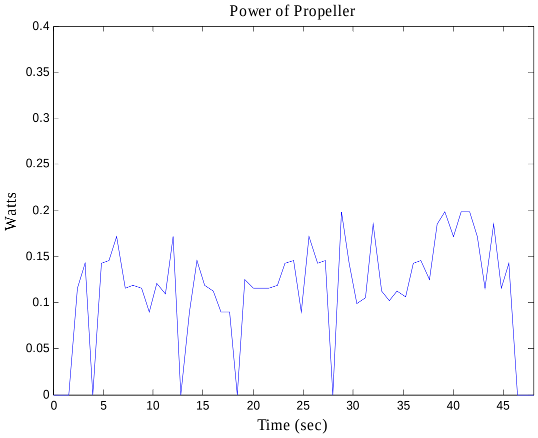

Figure 0-24 Power of propeller with attack angle control (3)

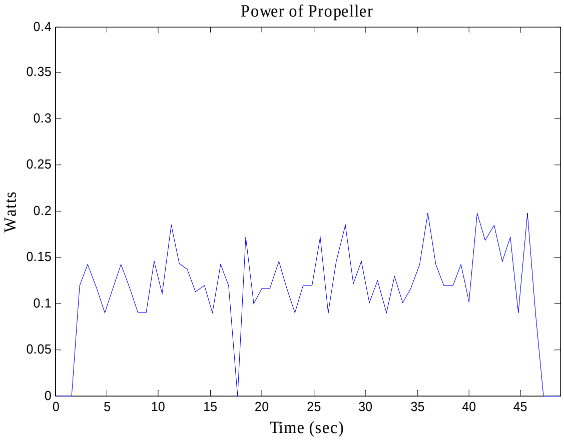

Figure 0-25 Power of propeller with attack angle control (4)

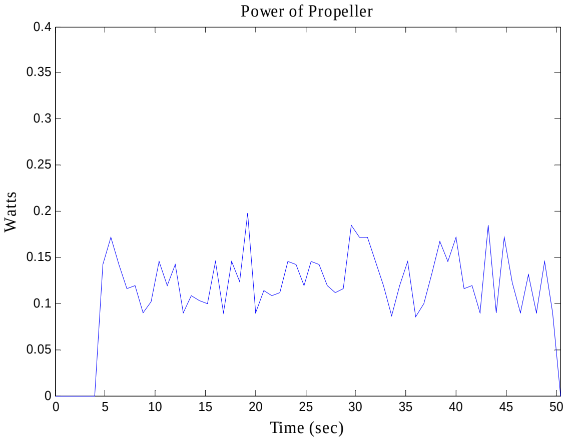

Figure 0-26 Power of propeller with attack angle control (5)

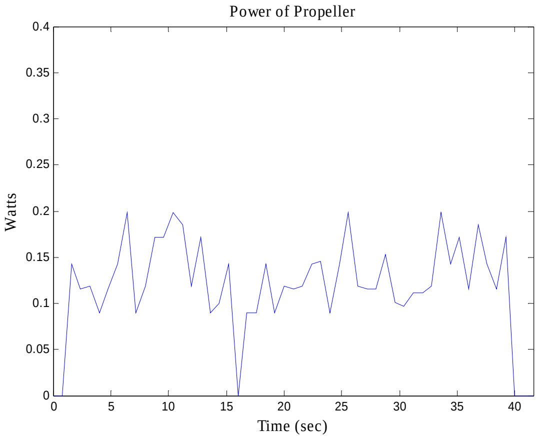

Figure 0-27 Power of propeller with attack angle control (6)

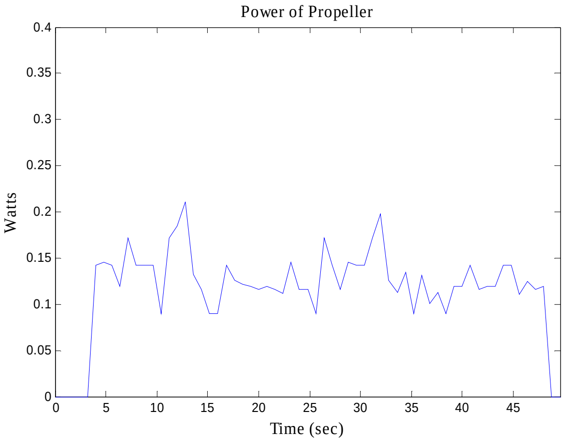

Figure 0-28 Power of propeller with attack angle control (7)

Results of propeller in 45 degrees and -30 degrees are shown in Figure 0-29 to Figure 0-42:

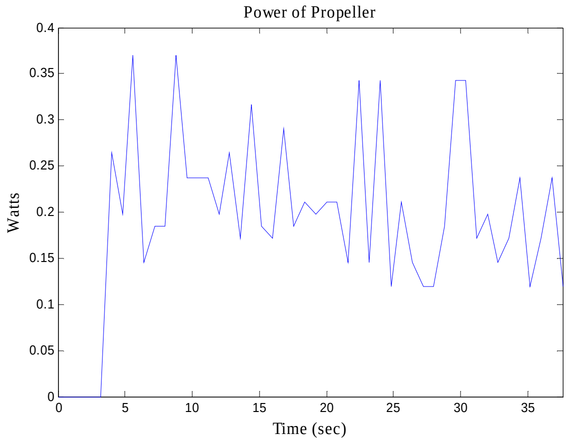

Figure 0-29 Power of propeller without attack angle control (1)

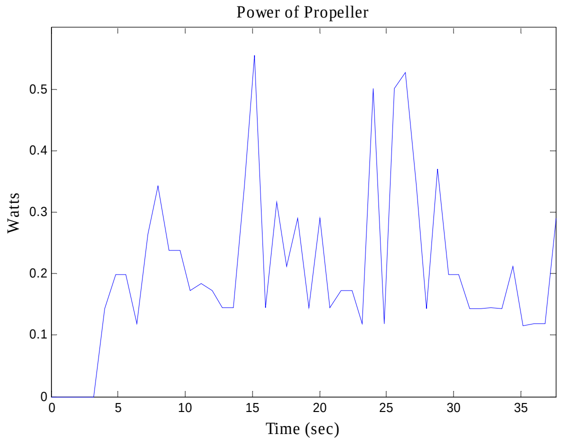

Figure 0-30 Power of propeller without attack angle control (2)

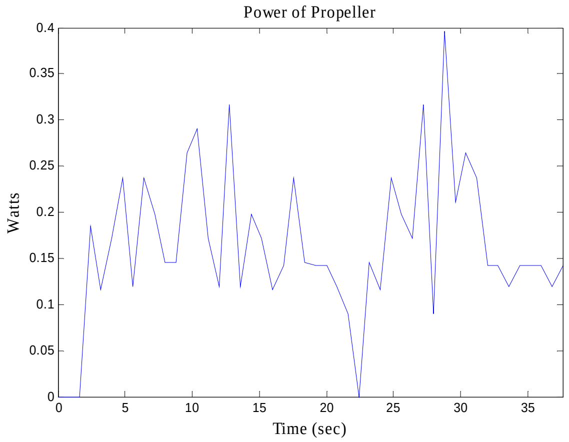

Figure 0-31 Power of propeller without attack angle control (3)

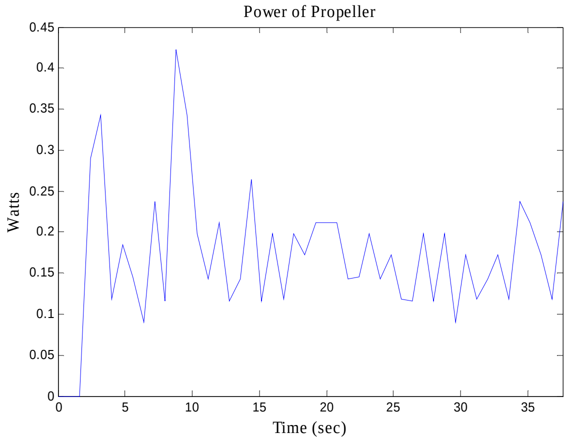

Figure 0-32 Power of propeller without attack angle control (4)

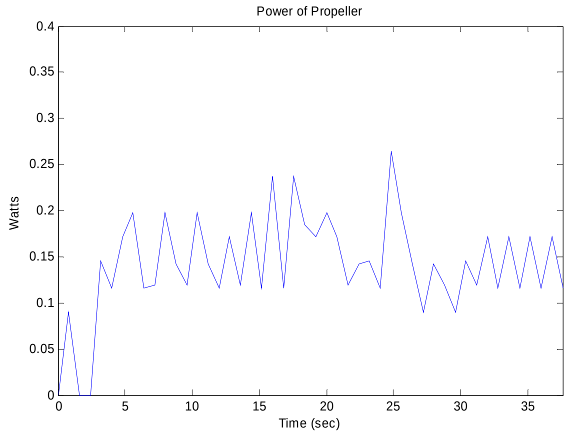

Figure 0-33 Power of propeller without attack angle control (5)

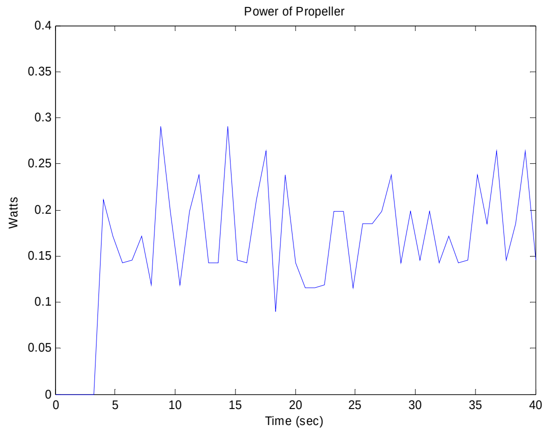

Figure 0-34 Power of propeller without attack angle control (6)

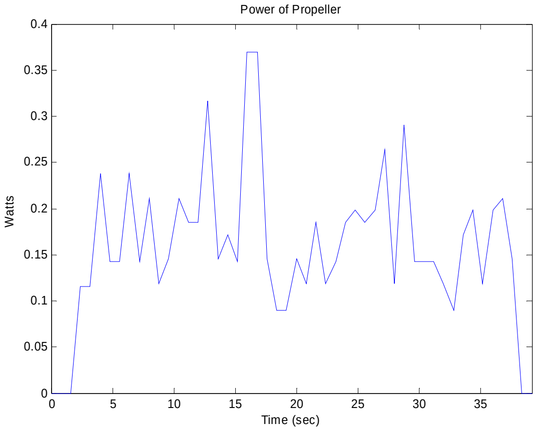

Figure 0-35 Power of propeller without attack angle control (7)

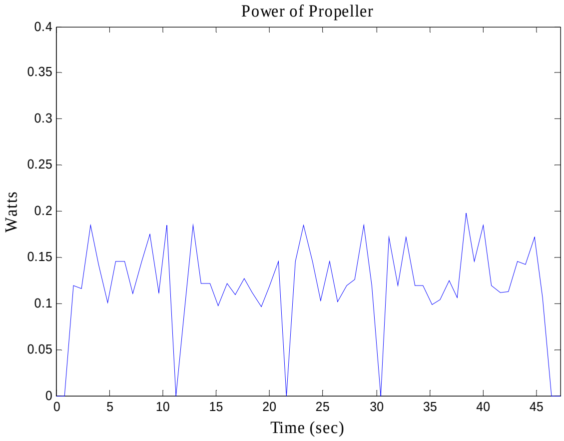

Figure 0-36 Power of propeller with attack angle control (1)

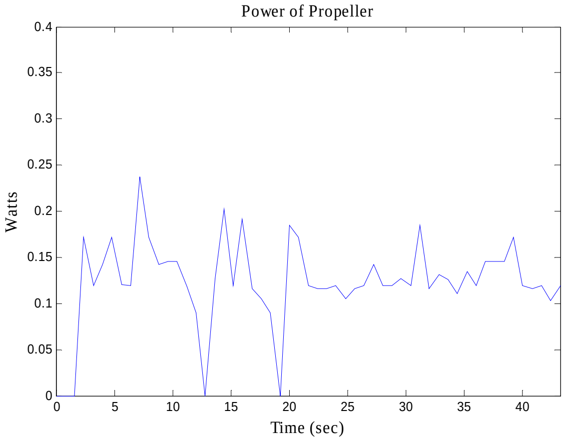

Figure 0-37 Power of propeller with attack angle control (2)

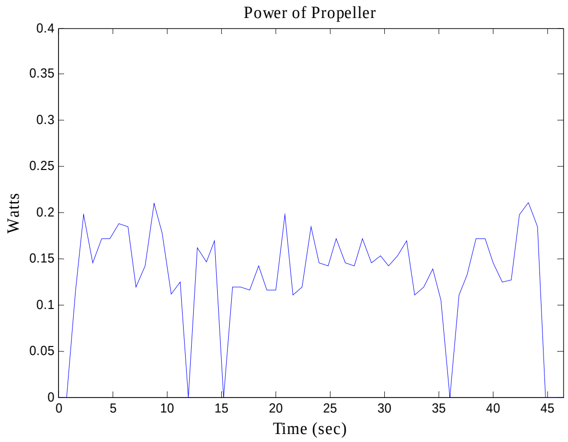

Figure 0-38 Power of propeller with attack angle control (3)

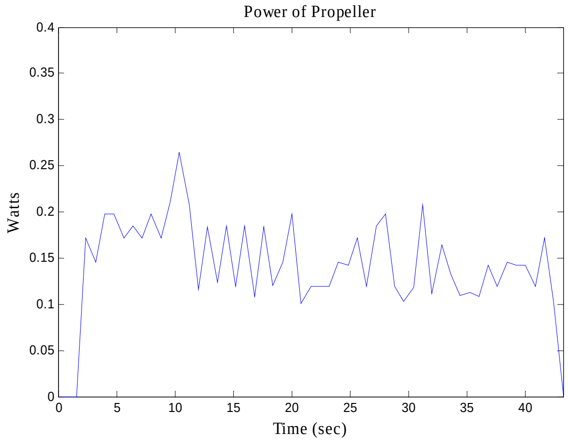

Figure 0-39 Power of propeller with attack angle control (4)

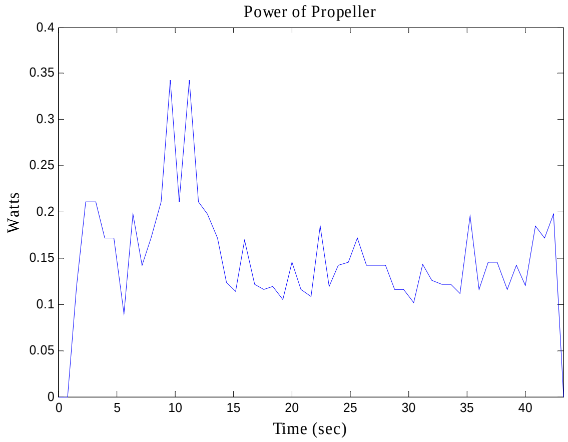

Figure 0-40 Power of propeller with attack angle control (5)

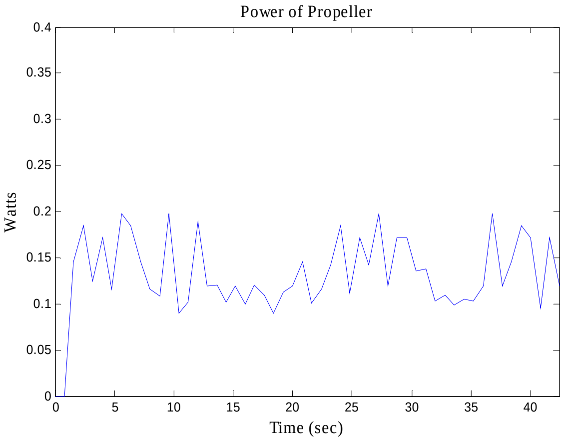

Figure 0-41 Power of propeller with attack angle control (6)

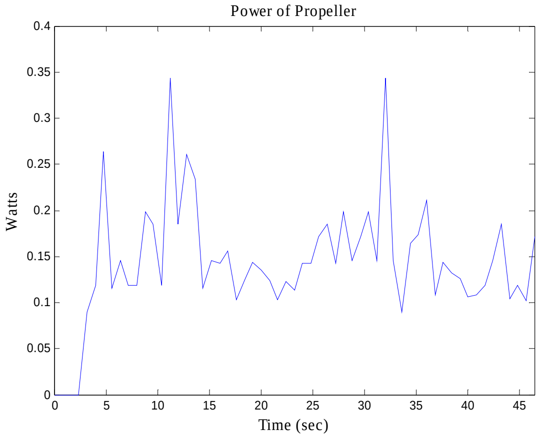

Figure 0-42 Power of propeller with attack angle control (7)

Results of turning motor in 30 degrees and -45 degrees are shown in Figure 0-43 to Figure 0-56:

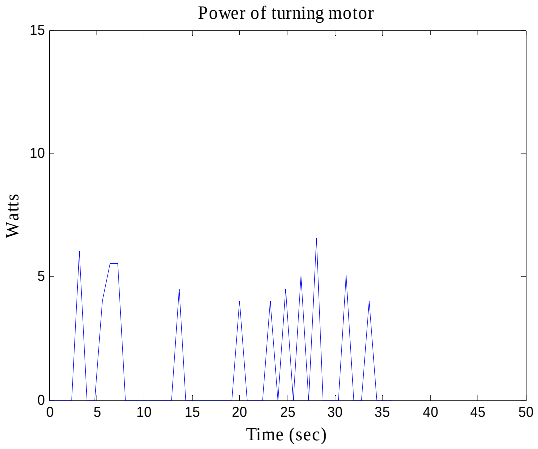

Figure 0-43 Power of turning motor without attack angle control (1)

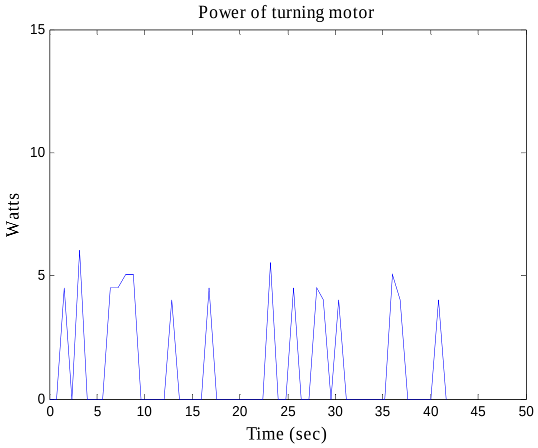

Figure 0-44 Power of turning motor without attack angle control (2)

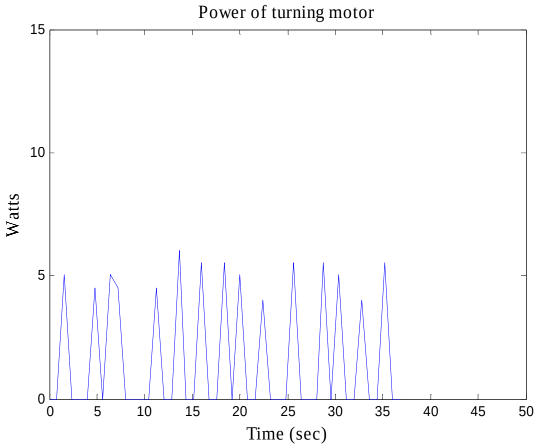

Figure 0-45 Power of turning motor without attack angle control (3)

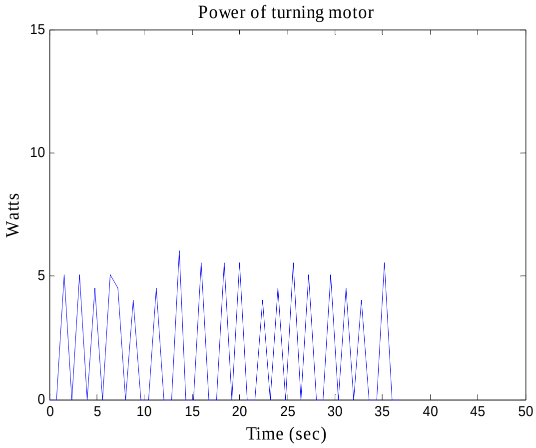

Figure 0-46 Power of turning motor without attack angle control (4)

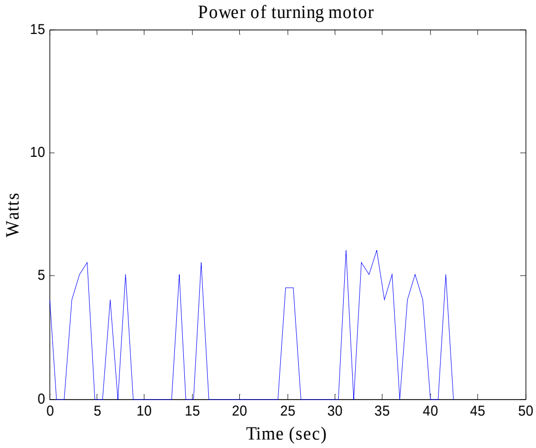

Figure 0-47 Power of turning motor without attack angle control (5)

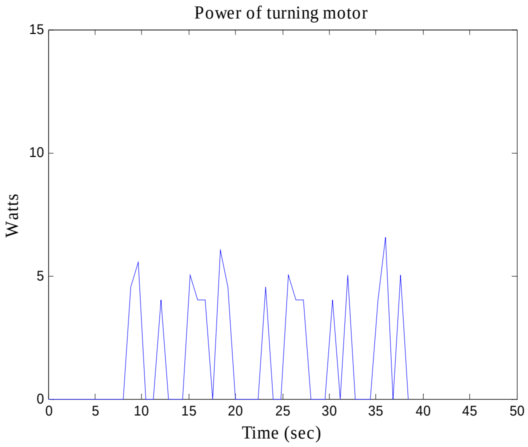

Figure 0-48 Power of turning motor without attack angle control (6)

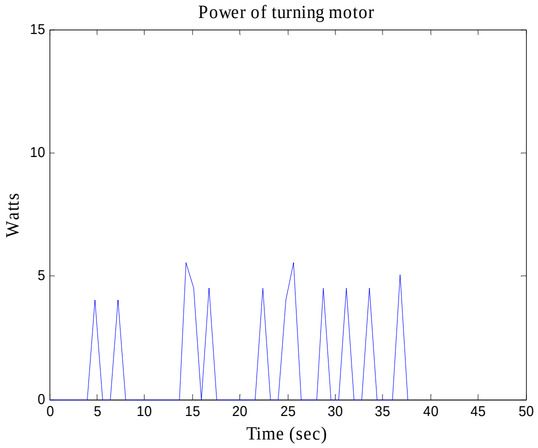

Figure 0-49 Power of turning motor without attack angle control (7)

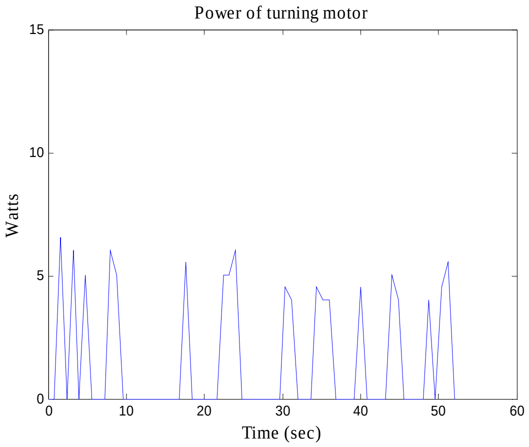

Figure 0-50 Power of turning motor with attack angle control (1)

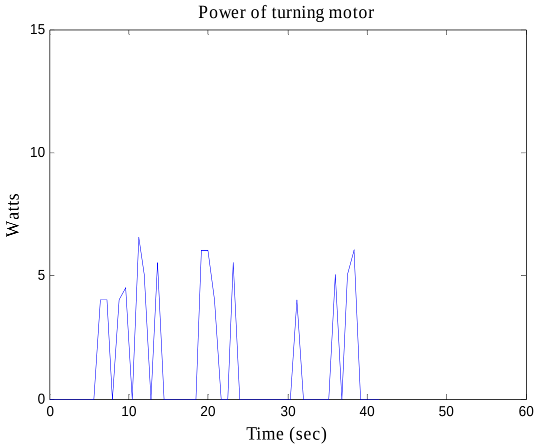

Figure 0-51 Power of turning motor with attack angle control (2)

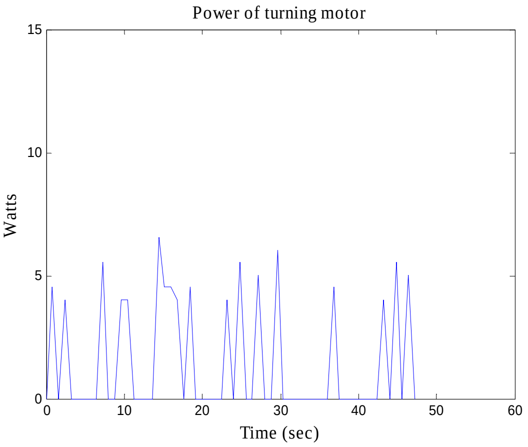

Figure 0-52 Power of turning motor with attack angle control (3)

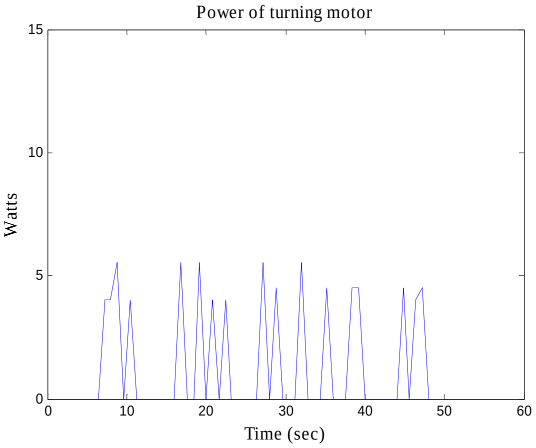

Figure 0-53 Power of turning motor with attack angle control (4)

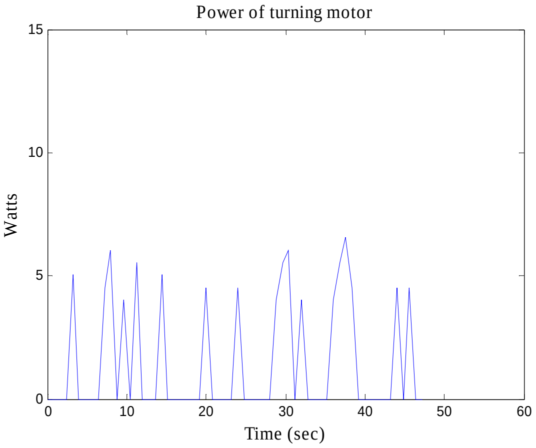

Figure 0-54 Power of turning motor with attack angle control (5)

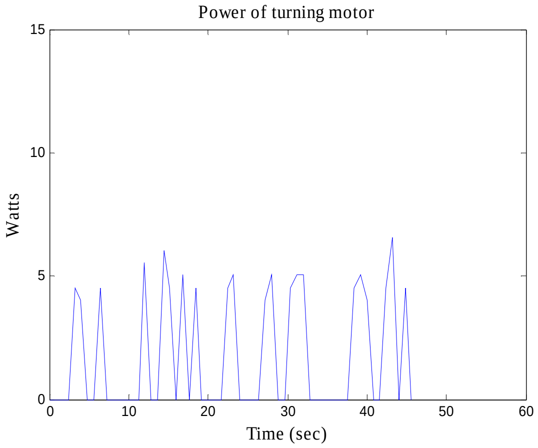

Figure 0-55 Power of turning motor with attack angle control (6)

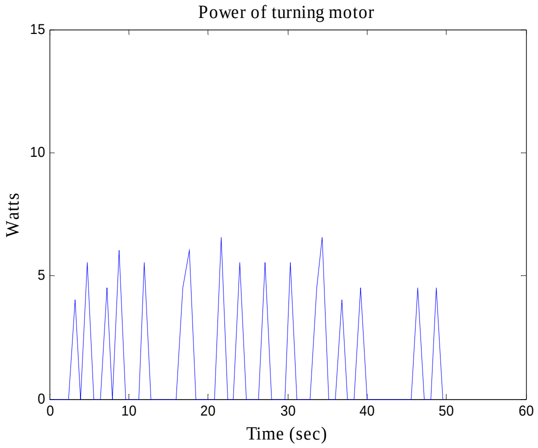

Figure 0-56 Power of turning motor with attack angle control (7)

Results of turning motor in 30 degrees and -30degrees are shown in Figure 0-57 to Figure 0-70:

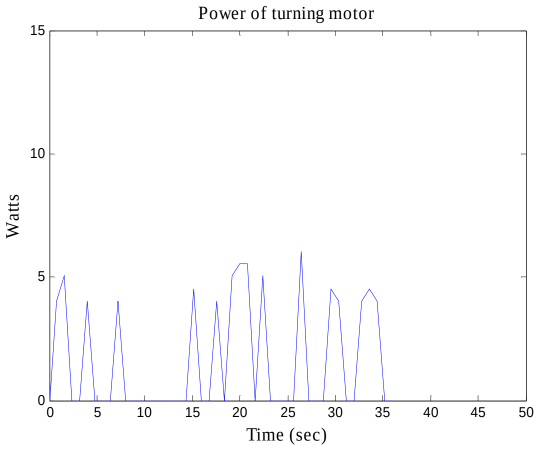

Figure 0-57 Power of turning motor without attack angle control (1)

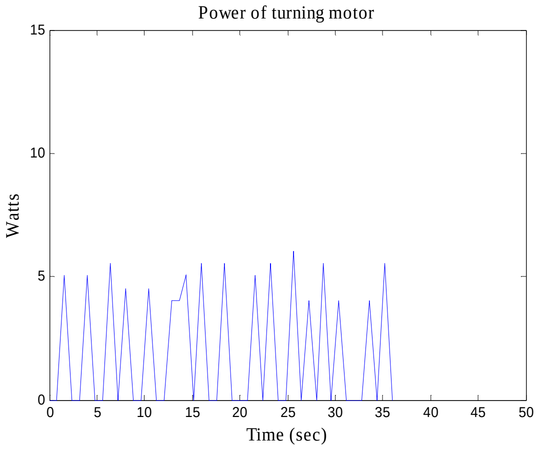

Figure 0-58 Power of turning motor without attack angle control (2)

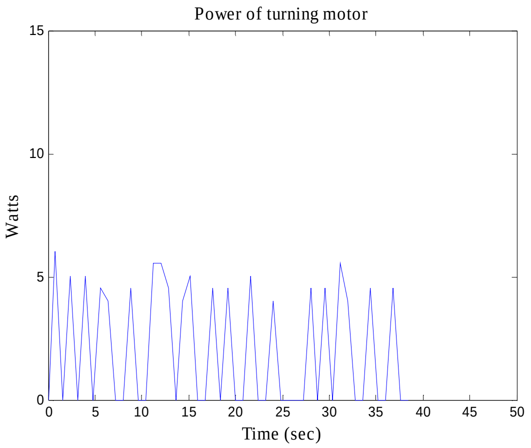

Figure 0-59 Power of turning motor without attack angle control (3)

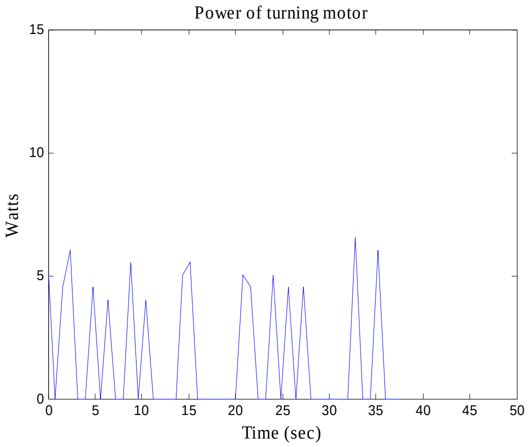

Figure 0-60 Power of turning motor without attack angle control (4)

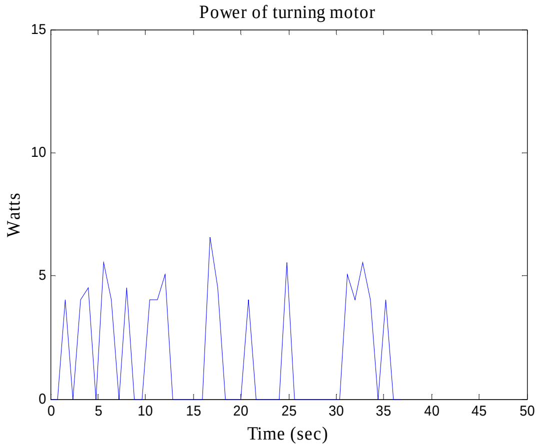

Figure 0-61 Power of turning motor without attack angle control (5)

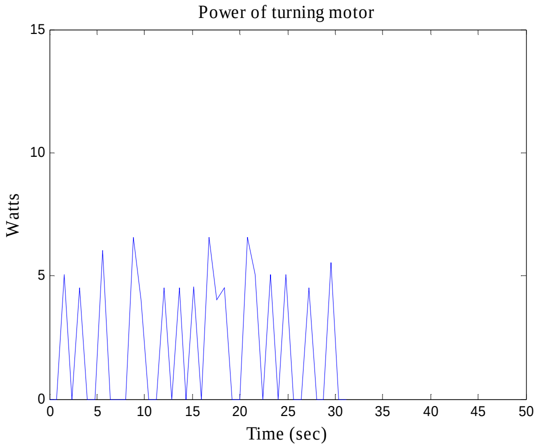

Figure 0-62 Power of turning motor without attack angle control (6)

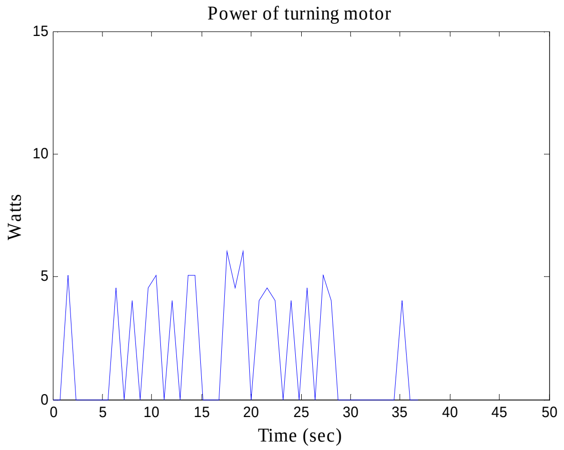

Figure 0-63 Power of turning motor with attack angle control (7)

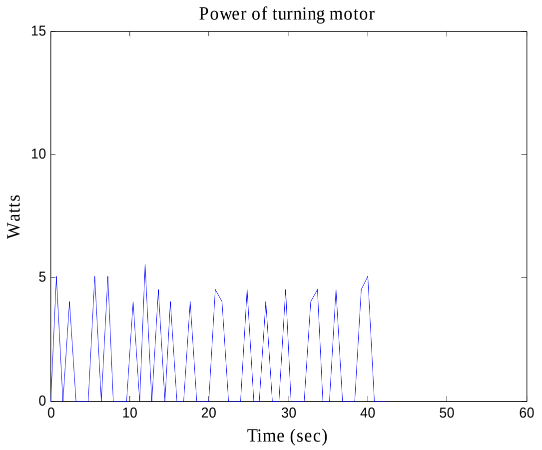

Figure 0-64 Power of turning motor with attack angle control (1)

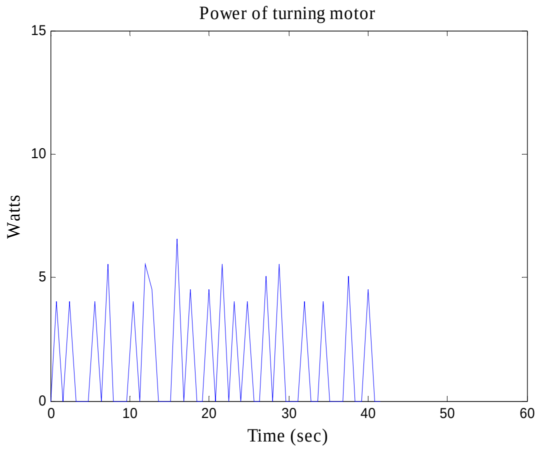

Figure 0-65 Power of turning motor with attack angle control (2)

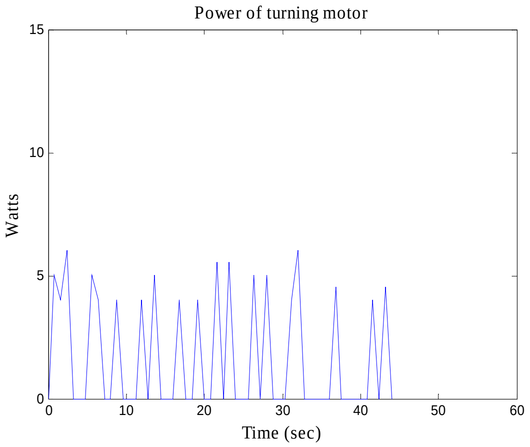

Figure 0-66 Power of turning motor with attack angle control (3)

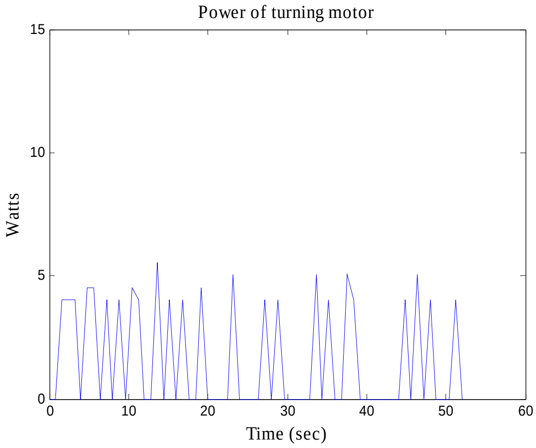

Figure 0-67 Power of turning motor with attack angle control (4)

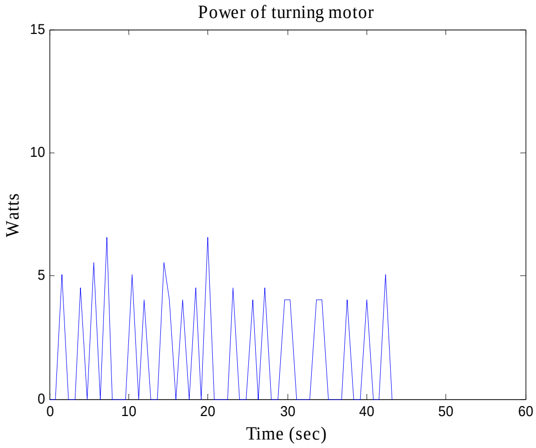

Figure 0-68 Power of turning motor with attack angle control (5)

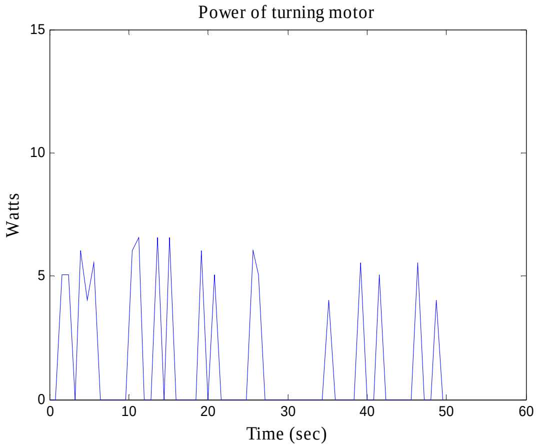

Figure 0-69 Power of turning motor with attack angle control (6)

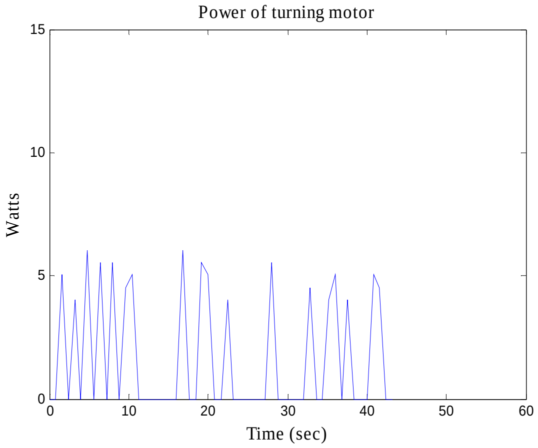

Figure 0-70 Power of turning motor with attack angle control (7)

Results of turning motor in 45 degrees and -30degrees are shown in Figure 0-71 to Figure 0-84:

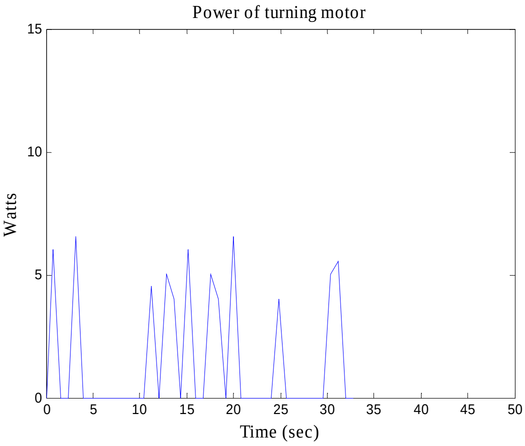

Figure 0-71 Power of turning motor without attack angle control (1)

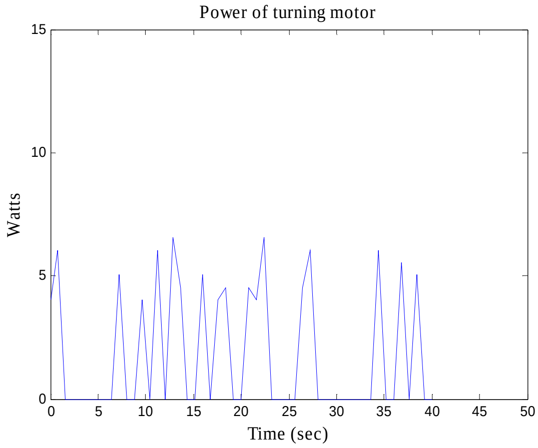

Figure 0-72 Power of turning motor without attack angle control (2)

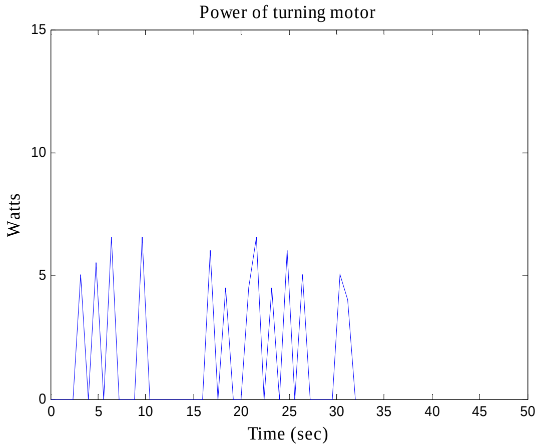

Figure 0-73 Power of turning motor without attack angle control (3)

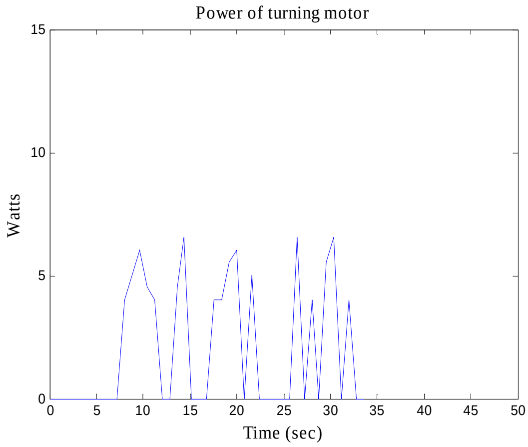

Figure 0-74 Power of turning motor without attack angle control (4)

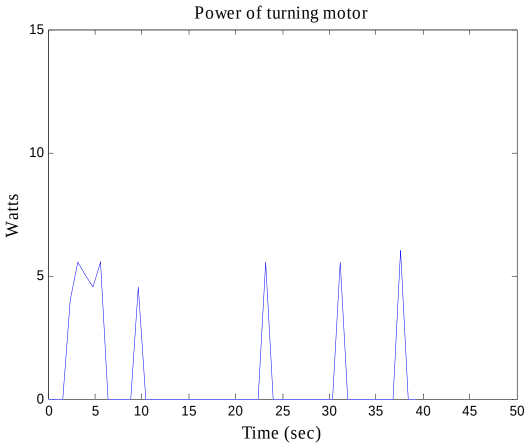

Figure 0-75 Power of turning motor without attack angle control (5)

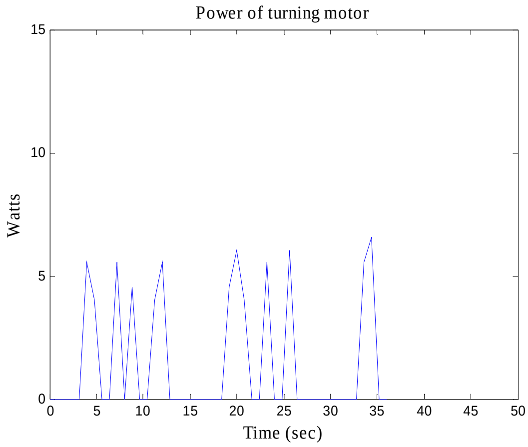

Figure 0-76 Power of turning motor without attack angle control (6)

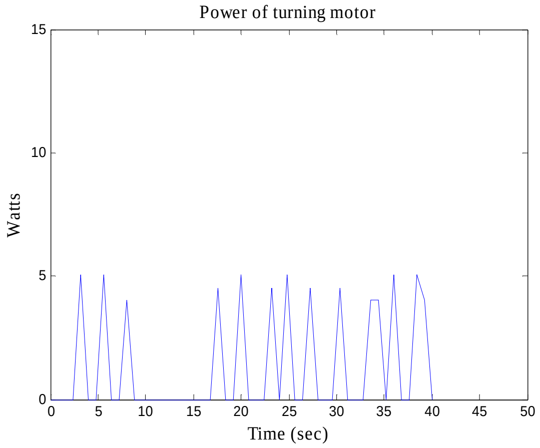

Figure 0-77 Power of turning motor without attack angle control (7)

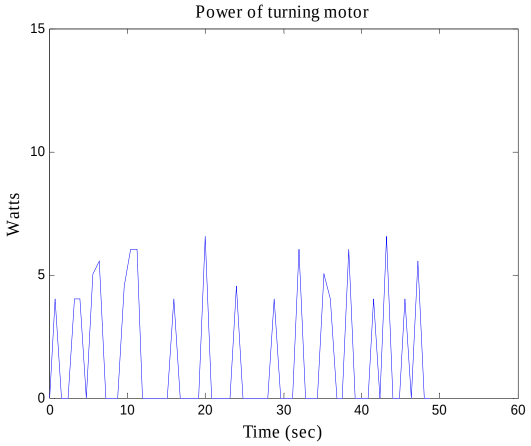

Figure 0-78 Power of turning motor with attack angle control (1)

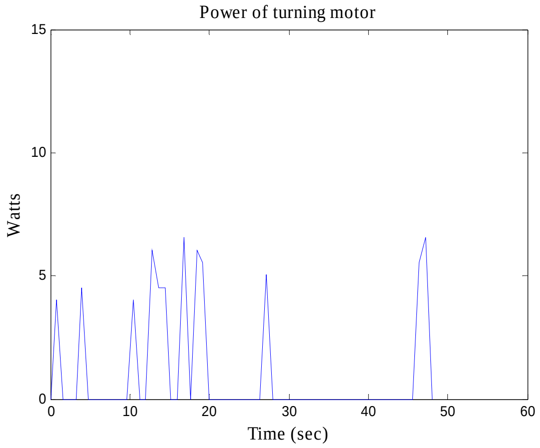

Figure 0-79 Power of turning motor with attack angle control (2)

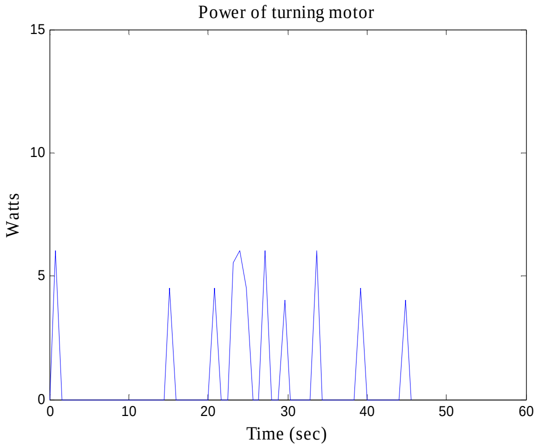

Figure 0-80 Power of turning motor with attack angle control (3)

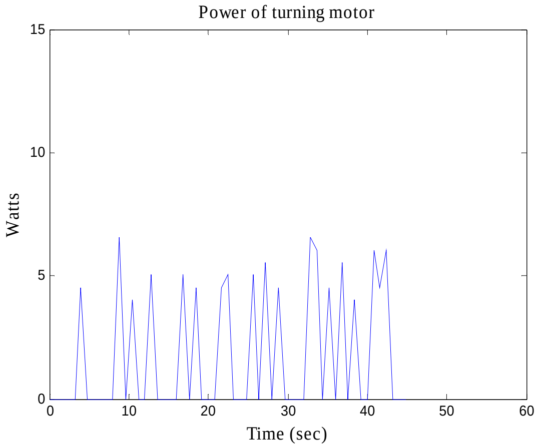

Figure 0-81 Power of turning motor with attack angle control (4)

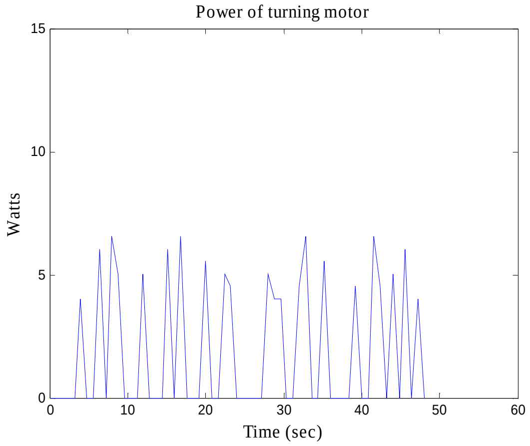

Figure 0-82 Power of turning motor with attack angle control (5)

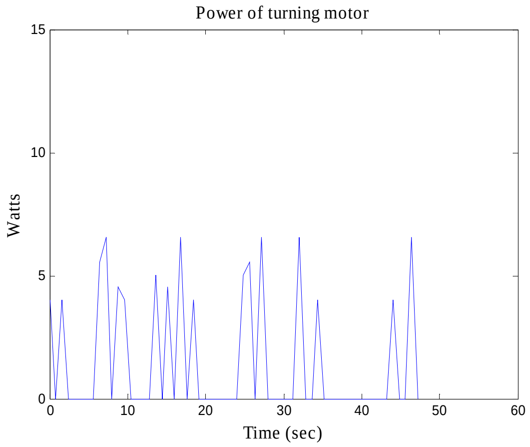

Figure 0-83 Power of turning motor with attack angle control (6)

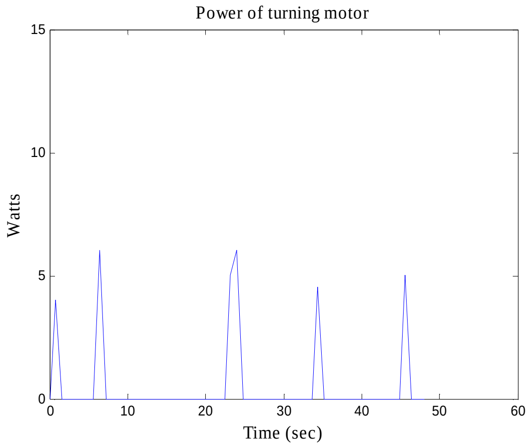

Figure 0-84 Power of turning motor with attack angle control (7)