



ระบบระบุตำแหน่งและควบคุมเสถียรภาพของหุ่นยนต์บังคับใต้น้ำแบบสี่งศาอิสระ

ปรเมศวร์ สุวรรณวงศ์ และ พุทธิกร สมิตไมตรี*

ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสงขลานครินทร์

* ผู้นิพนธ์ประสานงาน โทรศัพท์ 0 7428 7214 อีเมล: pruittikorn.s@psu.ac.th DOI: 10.14416/j.kmutnb.2019.09.005

รับเมื่อ 24 มีนาคม 2562 แก้ไขเมื่อ 21 มิถุนายน 2562 ตอรับเมื่อ 24 มิถุนายน 2562 เผยแพร่ออนไลน์ 9 กันยายน 2562

© 2019 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

บทคัดย่อ

ปัจจุบันหุ่นยนต์บังคับใต้น้ำ (ROUV) กลายเป็นอุปกรณ์สำคัญสำหรับภารกิจใต้น้ำหลายอย่าง แต่ด้วยคุณลักษณะทางพลศาสตร์ของการเคลื่อนที่ใต้น้ำของ ROUV เช่น การไหลหลังจากการเคลื่อนที่ พลศาสตร์ของใบพัด ความสมดุลระหว่างน้ำหนักและแรงลอยตัว แรงกระทำที่ไม่แน่นอนจากสายเคเบิล และมุมมองภาพที่จำกัดขณะใช้งานใต้น้ำ ทำให้การควบคุม ROUV เป็นสิ่งที่ยากสำหรับผู้ควบคุมแม้จะใช้งานใต้น้ำ งานวิจัยนี้จึงทำการออกแบบระบบควบคุมเสถียรภาพของการทรงตัวด้วยตัวควบคุม PI และพัฒนาการระบุตำแหน่งของ ROUV บนพื้นฐานของ Hector SLAM ซึ่งเป็นโปรแกรมที่ทำงานบน Robot Operating System (ROS) การควบคุมแบบ PI ถูกนำมาใช้เพื่อพัฒนาระบบควบคุมการทรงตัวโดยอาศัยสัญญาณป้อนกลับเป็นค่าความเร็วในแนวระนาบ ระดับความลึกและความเร็วเชิงมุมทิศ Yaw เพื่อควบคุมการเคลื่อนที่ของหุ่นยนต์ในทิศทางเหล่านั้น อุปกรณ์หลักของระบบประกอบด้วย คอมพิวเตอร์ บอร์ด Arduino และบอร์ด Raspberry Pi เป็นส่วนประมวลผลมอเตอร์ขับเคลื่อน (Thruster) บอร์ดขับเคลื่อน (Drive Board) เซนเซอร์วัดความดัน (Pressure Sensor) สำหรับวัดระดับความลึก เซนเซอร์วัดทิศทาง (Gyroscope) และเซนเซอร์วัดสนามแม่เหล็ก (Magnetometer) จาก IMU สำหรับวัดทิศ Yaw เซนเซอร์วัดระยะด้วยเลเซอร์ (Lidar) สำหรับวัดตำแหน่งในแนวระนาบเพื่อหาตำแหน่งของตัวหุ่นยนต์โดยอาศัยอัลกอริทึม Scan Matching การทดสอบระบบใช้หุ่นยนต์ดำน้ำที่สร้างขึ้นเองในการทดสอบ การทดสอบระบบใช้หุ่นยนต์ดำน้ำที่สร้างขึ้นเองในการทดสอบ โดยหุ่นยนต์ประกอบด้วยมอเตอร์ขับเคลื่อน 6 ตัว ที่มีจุดเด่นของระบบควบคุมอัตโนมัติสำหรับการเคลื่อนที่ได้ 4 องศาอิสระ ผลการวิจัยแสดงให้เห็นว่า ระบบระบุตำแหน่งของ ROUV สามารถระบุตำแหน่งและทิศ Yaw แบบเวลาจริงได้อย่างแม่นยำ ระบบควบคุมเสถียรภาพสามารถทำให้ ROUV รักษาตำแหน่งใน 3 มิติ และทิศ Yaw พร้อมกันเมื่ออยู่นิ่ง และรักษาเส้นทางการเคลื่อนที่ตามที่ต้องการได้

คำสำคัญ: หุ่นยนต์ใต้น้ำ ตัวควบคุม PI ระบบควบคุมเสถียรภาพ ระบบระบุตำแหน่ง



Localization and Stabilization Control System of the 4 Degree-of-freedom Remotely Operated Underwater Vehicle

Paramet Suwanwong and Pruittikorn Smithmaitrie*

Department of Mechanical Engineering, Faculty of Engineering, Prince of Songkla University, Songkhla, Thailand

* Corresponding Author, Tel. 0 7428 7214, E-mail: pruittikorn.s@psu.ac.th DOI: 10.14416/j.kmutnb.2019.09.005

Received 24 March 2019; Revised 21 June 2019; Accepted 24 June 2019; Published online: 9 September 2019

© 2019 King Mongkut's University of Technology North Bangkok. All Rights Reserved.

Abstract

Nowadays Remotely Operated Underwater Vehicles (ROUV) has become necessary equipment for many underwater tasks. However, a number of underwater dynamic-motion characteristics of ROUV, including motion drifting, dynamics of the ROUV thruster, balancing between ROUV weight and buoyancy force, uncertain force from cable wire and underwater field-of-view limitation cause difficulty for an operator to control a ROUV irrespective of the current flows. In this research, PID stabilization control system was applied to the ROUV whereas the localization system was developed based on Hector SLAM, which is a software package of Robot Operating System (ROS). The PID stabilization controller relies on the horizontal velocity, depth and yaw-angular velocity feedbacks to control the robot motion in the corresponding directions. The ROUV hardware consists of a computer laptop, an Arduino board and a Raspberry Pi board as a processor unit, thrusters, motor drive boards, a pressure sensor for depth measurement, a gyroscope and magnetometer of IMU for orientation measurement, Lidar sensor for measuring horizontal distances and determining robot position by using the Scan Matching algorithm. The experiments were performed on the developed underwater robot. The ROUV, consisting of 6 thrusters, has an automatic feedback-control system for the 4-degree-of-freedom motion, which is a main contribution of this research. The research results show that the localization system of the ROUV is able to precisely maintain real-time position and yaw orientation. The controlled system is able to maintain the ROUV at the 3D stationary target position and to maneuver along the desired path.

Keywords: ROUV, PI Controller, Stabilization Control System, Localization System

Please cite this article as: Paramet S. and P. Smithmaitrie, "Localization and stabilization control system of the 4 degree-of-freedom remotely operated underwater vehicle," *The Journal of KMUTNB*, vol. 29, no. 4, pp. 597-611, Oct.-Dec. 2019 (in Thai).

1. บทนำ

หุ่นยนต์ดำน้ำแบบบังคับ (Remotely Operated Underwater Vehicle; ROUV) เป็นเทคโนโลยีการสำรวจใต้น้ำ ซึ่งอำนวยความสะดวกได้อย่างมากเนื่องจากการสำรวจใต้น้ำเดิมอาศัยนักดำน้ำลงไปสำรวจ และจำเป็นต้องพึ่งอุปกรณ์ต่างๆ มากมาย ทำให้เกิดความยุ่งยาก สิ้นเปลือง อีกทั้งยังมีความเสี่ยงต่อสุขภาพนักดำน้ำอีกด้วย การใช้หุ่นยนต์ดำน้ำในการสำรวจสามารถทำได้ง่ายกว่า โดยไม่จำเป็นต้องมีนักดำน้ำทำงานได้นานและลึกกว่า ประหยัดกว่าในระยะยาว และลดความเสี่ยงต่อชีวิตของนักดำน้ำลงได้ แต่การควบคุมหรือใช้งานหุ่นยนต์ดำน้ำให้ทำงานได้อย่างเต็มที่ ต้องอาศัยผู้ใช้งานหรือผู้ควบคุมที่มีทักษะในการควบคุม อีกทั้งสภาพแวดล้อมหรือพื้นที่ทำการส่วนใหญ่หุ่นยนต์อยู่นอกสายตาผู้ใช้ภาพที่ผู้ใช้ได้รับจากกล้องในตัวหุ่นยนต์มีข้อจำกัดเรื่องมุมมอง ทำให้ผู้ใช้ไม่สามารถรับรู้ตำแหน่งและทิศทางของหุ่นยนต์ได้ ระบบระบุตำแหน่งจึงเป็นสิ่งสำคัญที่จะช่วยให้ผู้ใช้สามารถรับรู้ตำแหน่งและทิศทางของหุ่นยนต์ดำน้ำได้ และสามารถควบคุมหุ่นยนต์ไปในทิศทางและตำแหน่งที่ต้องการได้อย่างแม่นยำ ปัจจัยที่ส่งผลกระทบต่อการทำงานของหุ่นยนต์อันเนื่องมาจากการเคลื่อนที่ใต้น้ำของหุ่นยนต์ดำน้ำ มีทั้งการไหลหลังจากการเคลื่อนที่ ความไม่สมดุลระหว่างน้ำหนักกับแรงลอยตัว ที่ทำให้หุ่นยนต์ลอยขึ้นหรือจมลง และแรงกระทำจากความแข็งตึงของสายเคเบิลซึ่งกระทำต่อหุ่นยนต์ในรูปแบบที่ไม่สามารถคาดเดาได้ ทำให้การควบคุมหุ่นยนต์ให้อยู่กับที่โดยอาศัยเพียงการควบคุมโดยตรงจากผู้บังคับจึงเป็นงานที่ยาก ต้องใช้สมรรถนะสูงของระบบควบคุม ซึ่งทำให้เกิดความเหนียวล้า และประสิทธิภาพในการควบคุมลดลง ระบบรักษาตำแหน่งจึงเป็นส่วนสำคัญที่จะช่วยเพิ่มประสิทธิภาพการควบคุมหุ่นยนต์ และลดภาระของผู้บังคับลง

งานวิจัยที่ผ่านมามุ่งเน้นไปในด้านการพัฒนาขีดความสามารถของหุ่นยนต์ดำน้ำในหลายประเด็น ได้แก่ ด้านการพัฒนาาระบบรักษาตำแหน่ง ตัวอย่างเช่น ระบบรักษาตำแหน่งความลึก รักษาระยะห่างจากจุดอ้างอิง และรักษามุม Yaw โดยอาศัยข้อมูลจากกล้อง Gyroscope และ Pressure Sensor ด้วยระบบควบคุมแบบ PID [1] การพัฒนาาระบบรักษาตำแหน่ง

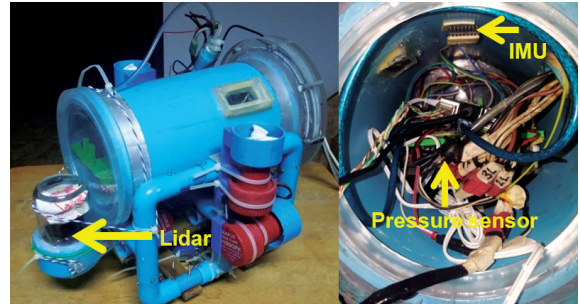
ความลึกอาศัยข้อมูลจาก Pressure Sensor โดยใช้ระบบควบคุม PID ที่ป้อนกลับค่าความลึก [2], [3] การพัฒนาาระบบรักษาตำแหน่งความลึก ระยะห่างจากจุดเป้าหมาย และมุม Yaw เทียบกับตำแหน่งอ้างอิงโดยใช้ข้อมูลจาก LVS ร่วมกับ IMU [4] การพัฒนาาระบบรักษาเส้นทางการเคลื่อนที่ในระนาบ 2 มิติ โดยอาศัยข้อมูลจากเซนเซอร์ Gyroscope, Accelerometer และ Ultrasound [5] อีกทั้งยังมีงานวิจัยเกี่ยวกับด้านการพัฒนาาระบบนำร่อง ตัวอย่างเช่น ระบบนำร่องสำหรับการสำรวจโครงสร้างใต้น้ำอัตโนมัติของ AUV [6] งานวิจัยที่เกี่ยวกับการพัฒนาาระบบสร้างแผนที่ใต้น้ำ ตัวอย่างเช่น ระบบการทำแผนที่ใต้น้ำโดยอาศัยข้อมูลจาก Scanning Sonar [7] ระบบการทำแผนที่ใต้น้ำโดยอาศัยข้อมูลจาก Scanning Sonar, Depth Sensor และ Compass [8] งานวิจัยด้านเซนเซอร์สำหรับใช้งานใต้น้ำ เช่น การเปรียบเทียบข้อมูลตำแหน่งของ ROUV ที่ได้จากการประมวลผลระหว่างเซนเซอร์ DVL และ GPS [9] ศึกษาการใช้ข้อมูลจาก Scanning Sonar และ Digital Compass ในการระบุตำแหน่งของ ROUV ใต้น้ำ [10] มีการศึกษาเซนเซอร์ต้นแบบที่พัฒนาขึ้นเพื่อใช้ในการทำแผนที่ใต้น้ำซึ่งตัวเซนเซอร์ประกอบด้วย Laser Line Generators 2 ตัว และ CCD Camera [11] รวมไปถึงงานวิจัยด้านระบบควบคุมการทำงานสำหรับหุ่นยนต์ใต้น้ำ เช่น การใช้ระบบควบคุมแบบ PID, Sliding-mode, Fuzzy และ Neural Network [12] การพัฒนาาระบบรักษาตำแหน่งความลึก โดยใช้ระบบควบคุมแบบ Adaptive Fuzzy Sliding Mode [13] การควบคุม ROUV ด้วยระบบควบคุมแบบ Adaptive และ Sliding-mode [14] การศึกษาเพื่อลด Overshoot ของระบบควบคุมแบบ PID ด้วย Continuous Input Smoother และ Discrete Fuzzy Smoother [15] การควบคุมตำแหน่งของ ROUV โดยการควบคุมความเร็วการหมุนของใบพัดด้วยระบบควบคุมแบบ Fuzzy Sliding-mode [16] ซึ่งพบว่าในประเด็นของระบบระบุตำแหน่งและควบคุมเสถียรภาพของหุ่นยนต์ดำน้ำขนาดเล็กนั้น งานวิจัยที่ผ่านมาส่วนมากศึกษาและวิจัยระบบควบคุมเสถียรภาพที่สามารถควบคุมได้พร้อมกันสูงสุดเพียง 3 องศาอิสระเท่านั้น ทำให้หุ่นยนต์รักษาตำแหน่งและทิศทางได้ไม่สมบูรณ์ หุ่นยนต์ไม่สามารถควบคุม

ความคลาดเคลื่อนในองศาอิสระอื่น เช่น ทิศทาง (Heading Orientation) ที่ไม่ได้ถูกกำหนดไว้ในระบบควบคุม ซึ่งเป็นผลมาจากการที่ระบบระบุตำแหน่งไม่สามารถระบุตำแหน่งได้มากกว่า 3 องศาอิสระ หรือหุ่นยนต์ดำน้ำประเภทที่ไม่มี Thruster ในบางองศาอิสระ ทำให้การพัฒนาความสามารถของหุ่นยนต์ในการเคลื่อนที่อัตโนมัติหรือรักษาเส้นทางเคลื่อนที่ทำได้ยากและมีความคลาดเคลื่อนสูง

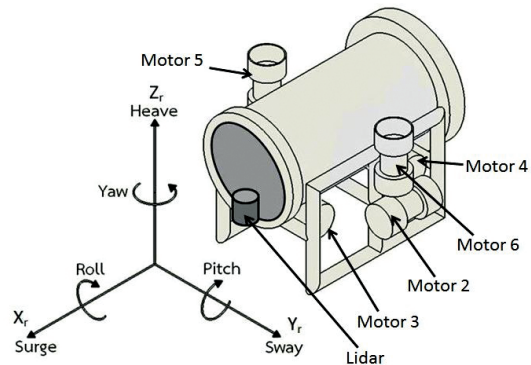
ด้วยเหตุนี้งานวิจัยนี้จึงเกิดขึ้นและมุ่งเน้นเพื่อพัฒนาระบบระบุตำแหน่งและควบคุมเสถียรภาพของ ROUV ให้มีความสามารถในการระบุตำแหน่งของ ROUV เทียบกับสิ่งแวดล้อมได้ใน 4 องศาอิสระ และควบคุมเสถียรภาพได้ใน 4 องศาอิสระ โดยทดลองกับหุ่นยนต์ดำน้ำที่สร้างขึ้นเองโดยอาศัยการควบคุมความเร็วด้วยระบบควบคุม PI ที่ปรับสถานะของหุ่นยนต์จากข้อมูลของ Pressure Sensor, Gyroscope, Magnetometer และ Lidar

2. โครงสร้างหุ่นยนต์

หุ่นยนต์ดำน้ำที่ใช้ในงานวิจัยนี้ถูกออกแบบและสร้างขึ้นเองเพื่อการทดสอบระบบควบคุม ลักษณะของหุ่นยนต์ดังแสดงในรูปที่ 1 โครงสร้างของหุ่นยนต์ทำจากท่อ PVC เป็นวัสดุหลัก เพราะมีความแข็งแรงทนทานต่อความดันใต้น้ำ ส่วนลำตัวทำจากท่อ PVC ขนาด 6 นิ้ว (Nominal Size) ภายในบรรจุอุปกรณ์อิเล็กทรอนิกส์ ส่วนโครงรอบนอกที่ยึดติดกับลำตัวใช้ท่อ PVC และข้อต่อขนาด 0.5 นิ้ว เพื่อเป็นแท่นติดตั้ง Thrusters และเป็นขาตั้งบนพื้น อีกทั้งยังใช้เป็นส่วนปรับตั้งน้ำหนักถ่วงสมดุล หุ่นยนต์นี้มีมอเตอร์ขับเคลื่อน หรือ Thruster ทั้งหมด 6 ตัว ทำหน้าที่ควบคุมการเคลื่อนที่ในทิศ Surge, Sway และ Heave ทิศละสองตัว โดยมอเตอร์ที่ควบคุมการเคลื่อนที่ในทิศทาง Surge และ Sway ทั้งสี่ตัวสามารถใช้ควบคุมการหมุนในทิศมุม Yaw ได้ด้วยดังแสดงในรูปที่ 2 ส่วนควบคุมภาคพื้นดิน (Ground Control Station; GCS) ประกอบด้วยหม้อแปลงแหล่งจ่ายพลังงาน คอมพิวเตอร์แล็ปท็อปเป็นส่วนประมวลผลและแสดงผลการทำงาน มีจอस्टикสำหรับบังคับหุ่นยนต์ หุ่นยนต์เชื่อมต่อกับ GCS ผ่านสายเคเบิลที่รวมสายไฟฟ้าและสาย Ethernet เข้าไว้ด้วยกัน หุ่นยนต์รับพลังงานจากแหล่งจ่าย



รูปที่ 1 ลักษณะโครงสร้างของหุ่นยนต์ดำน้ำภายนอก (ซ้าย) และภายใน (ขวา)

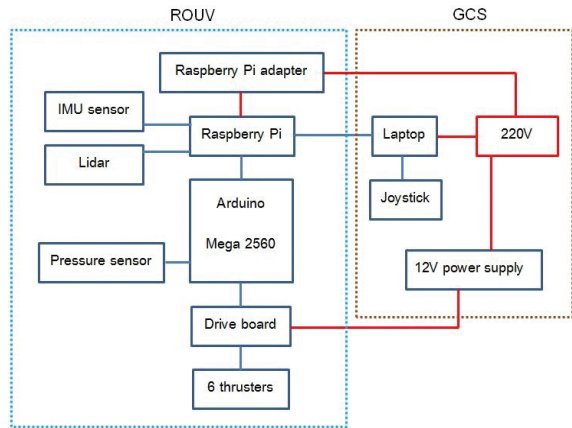


รูปที่ 2 กรอบอ้างอิงทิศทางของหุ่นยนต์ดำน้ำ

ไฟฟ้าผ่านหม้อแปลงของ GCS ไปทางสายไฟฟ้า มีการรับส่งข้อมูลต่างๆ ผ่านทางสาย Ethernet หุ่นยนต์ถูกควบคุมผ่านจอยสติคหรือสามารถป้อนคำสั่งได้โดยตรงผ่านคอมพิวเตอร์ รวมทั้งมีระบบควบคุมตำแหน่ง ที่ทำให้หุ่นยนต์สามารถรักษาตำแหน่งตัวเองได้อย่างอัตโนมัติ การอ้างอิงทิศทางของหุ่นยนต์ในงานวิจัยนี้ ใช้กรอบอ้างอิง (Frame) ตามรูปที่ 2 ซึ่งเป็นระบบการอ้างอิงโดยทั่วไปที่ใช้สำหรับเรือและเรือดำน้ำ ประกอบด้วย 6 องศาอิสระ ได้แก่ Surge, Sway, Heave, Roll, Pitch และ Yaw หุ่นยนต์สามารถควบคุมได้ 4 องศาอิสระ คือ Surge, Sway, Heave และ Yaw ส่วน Roll และ Pitch นั้นถูกควบคุมโดยการปรับตัวเองจากความสมดุลของโครงสร้าง

3. ระบบอิเล็กทรอนิกส์

ระบบอิเล็กทรอนิกส์ของหุ่นยนต์ ประกอบด้วย บอร์ด Arduino Mega 2560 และบอร์ด Raspberry Pi ทำหน้าที่



รูปที่ 3 แผนภาพการเชื่อมต่ออุปกรณ์ของหุ่นยนต์ดำน้ำและ GCS

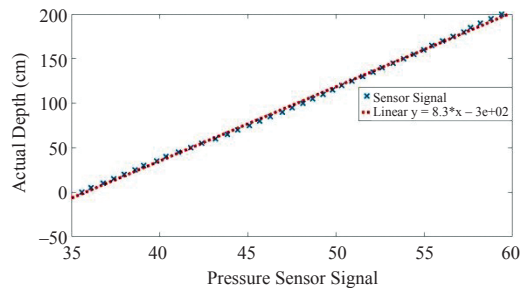
ประมวลผล เซนเซอร์วัดความดัน เซนเซอร์ IMU ที่มี Gyroscope และ Magnetometer สำหรับวัดมุม Yaw, Lidar 2 มิติสำหรับวัดระยะในระนาบ มอเตอร์ขับเคลื่อน (Thruster) และบอร์ดขับมอเตอร์ การเชื่อมต่อและการสื่อสารของอุปกรณ์อิเล็กทรอนิกส์ของระบบหุ่นยนต์ดำน้ำ และ GCS แสดงในรูปที่ 3

3.1 เซนเซอร์วัดความดัน

เซนเซอร์วัดความดัน NXP Semiconductor MPX5700GP ถูกนำมาใช้วัดความดันของน้ำที่เพิ่มขึ้นตามระดับความลึกของหุ่นยนต์ใต้น้ำ จากคุณสมบัติการทำงานที่แรงดัน 4.75–5.25 โวลต์ ดี.ซี. และมีแรงดัน Output 0.2–4.7 โวลต์ ดี.ซี. เป็นสัญญาณแอนะล็อกทำให้เซนเซอร์ชนิดนี้ ใช้งานร่วมกับไมโครคอนโทรลเลอร์ Arduino ได้ง่าย เซนเซอร์ถูกติดตั้งไว้ในส่วนลำตัวของหุ่นยนต์ แล้วต่อท่อสำหรับวัดความดันบนเซนเซอร์เข้ากับสายยางที่เจาะทะลุออกมาข้างภายนอกเพื่อวัดความดันของน้ำ ข้อมูลดิบจากเซนเซอร์ถูกนำมาผ่านตัวกรองเฉลี่ย (Average Filter) โดยกำหนดความถี่ของผลลัพธ์ไว้ที่ 20 เฮิรตซ์ แล้วนำมาผ่านตัวกรองความถี่ต่ำผ่าน (Low-pass Filter) จากนั้นจึงนำผลลัพธ์ที่ได้ไปใช้ในการคำนวณระดับความลึก ความสัมพันธ์ระหว่างข้อมูลความดันจากเซนเซอร์และความลึกหาได้จากการทดสอบเปรียบเทียบเซนเซอร์ โดยการต่อเซนเซอร์เข้ากับ



รูปที่ 4 การทดสอบเปรียบเทียบเซนเซอร์วัดความดัน



รูปที่ 5 ความสัมพันธ์ระหว่างข้อมูลความสูงของน้ำกับสัญญาณจากเซนเซอร์

สายยางใสที่วางตัวในแนวตั้งตามรูปที่ 4 เพิ่มระดับน้ำในสายยางครั้งละ 5 เซนติเมตร อ่านค่าที่วัดได้และบันทึกผลจนระดับน้ำสูงถึง 2 เมตร นำข้อมูลที่ได้มาเขียนกราฟความสัมพันธ์ระหว่างความดันที่วัดได้กับความสูงของน้ำ ผลที่ได้แสดงในรูปที่ 5 จะเห็นว่าความสัมพันธ์ระหว่างความดันและความสูงของน้ำมีแนวโน้มเป็นเชิงเส้น และมีสมการความสัมพันธ์ในการหาความลึกเป็น $D = 8.3 \times P - D_0$ เมื่อ D คือ ระดับความลึกจากระดับผิวน้ำ (เซนติเมตร) P คือ ค่าที่อ่านได้จากเซนเซอร์ความดัน และ D_0 คือค่าที่ได้จากการวัดเหนือผิวน้ำ (เซนติเมตร)

3.2 เซนเซอร์วัดทิศทาง

เซนเซอร์ IMU GY-87 ถูกนำมาใช้ในการวัดมุม Yaw โดยติดตั้งไว้ในส่วนลำตัวของหุ่นยนต์ เชื่อมต่อกับบอร์ด Raspberry Pi ค่าของมุม Yaw ได้จากการนำข้อมูลจากเซนเซอร์ Gyroscope และเซนเซอร์ Magnetometer ที่อยู่

ภายใน IMU มาประมวลผลโดยขั้นแรกทำการอ่านค่าข้อมูลดิบจากเซนเซอร์ทั้งสอง แล้วทำการปรับเทียบสัญญาณจากเซนเซอร์ Magnetometer โดยวิธี Soft Iron Calibration จากนั้นนำค่าจากเซนเซอร์ทั้งสองไปประมวลผลด้วยโปรแกรม imu_filter_madgwick โดยโปรแกรมนี้ใช้อัลกอริทึมรวมค่า (Fusion Algorithm) และอัลกอริทึมแบบคาลมาน (Kalman-based Algorithm) ในการประมวลผล แล้วได้ผลลัพธ์เป็นค่าทิศทาง เซนเซอร์ทั้งสองชนิดนี้มีการใช้งานกันอย่างแพร่หลาย ให้ข้อมูลการวัดมุมที่แม่นยำและน่าเชื่อถือเพียงพอต่อการใช้งาน [2] เพียงแค่ต้องทำการปรับเทียบค่าเริ่มต้นสำหรับการใช้งานเท่านั้น

3.3 มอเตอร์ขับเคลื่อน

มอเตอร์ขับเคลื่อนดัดแปลงมาจากปั้มน้ำขนาดเล็ก (Seaflo Bilge Pump 1100GPH 12V) โดยการตัดส่วนที่เป็นท่อและใบพัดสำหรับปั้มน้ำออก เหลือไว้เพียงมอเตอร์ที่มีซีลกันน้ำ แล้วติดตั้งใบพัดสำหรับขับเคลื่อนหุ่นยนต์แทนที่ใบพัดเดิม ใบพัดขับเคลื่อนขึ้นรูปจากเครื่องพิมพ์สามมิติ ออกแบบให้ใบพัดที่ทำงานคู่กันมีทิศทางตรงข้ามกัน เพื่อให้แรงบิดของมอเตอร์หักล้างซึ่งกันและกันเมื่อขับเคลื่อนหุ่นยนต์ในทิศ Heave, Surge และ Sway สำหรับการหมุนในทิศ Yaw อาศัยคู่มอเตอร์ในทิศ Surge และ Sway ที่ขับเคลื่อนในทิศตรงข้ามกันจึงทำให้แรงบิดลัพธ์รอบแกน Z, ที่ไม่เท่ากับศูนย์เป็นแรงบิดควบคุมการหมุนในทิศ Yaw

กรณีที่คู่มอเตอร์ในทิศ Surge ขับเคลื่อนในทิศเดียวกันแต่แรงบิดของมอเตอร์หักล้างกันไม่หมดเพราะความเร็วรอบไม่เท่ากันจะส่งผลกระทบต่ออาการหมุนในทิศ Roll ในทำนองเดียวกันแรงบิดของคู่มอเตอร์ในทิศ Sway ที่หักล้างไม่หมดจะส่งผลกระทบต่ออาการหมุนในทิศ Pitch แต่ก็ไม่ได้มีผลกระทบมากนัก เพราะการทรงตัวในทิศ Roll และ Pitch ถูกควบคุมด้วยสมดุลโครงสร้างอยู่แล้ว

3.4 เซนเซอร์วัดระยะด้วยเลเซอร์ (Lidar)

การหาตำแหน่งในระนาบ X และ Y ของหุ่นยนต์เทียบกับสิ่งแวดล้อมทำได้โดยใช้ Lidar (HOKUYO URG-

04LX-UG01) ควบคู่กับอัลกอริทึม hector_slam [17] ซึ่งเป็น ROS Package ที่จะนำข้อมูลการสแกนระยะจากเซนเซอร์มาประมวลผลสร้างแผนที่ของสิ่งแวดล้อม รวมทั้งระบุตำแหน่งหุ่นยนต์เทียบกับสิ่งแวดล้อมนั้น ทั้งนี้การใช้งาน Lidar ได้นี้ยังมีข้อจำกัด เนื่องจากคุณสมบัติของน้ำที่ดูดกลืนพลังงานแสงทำให้ระยะการทำงานของ Lidar ลดลงจากปกติ 4 เมตร เหลือประมาณ 40 เซนติเมตร ในกรณีที่น้ำใส และหากน้ำขุ่นหรือมีตะกอนอาจจะส่งผลให้เซนเซอร์ทำงานไม่ได้ จึงเป็นข้อจำกัดของเซนเซอร์ที่ต้องอยู่ในระยะที่สามารถตรวจจับโครงสร้างใต้น้ำได้ ในขณะเดียวกันผนังหรือสิ่งแวดล้อมก็จำเป็นต้องมีจุดสังเกต (Key Point) มากเพียงพอที่อัลกอริทึมจะสามารถนำข้อมูลไปประมวลผลเป็นจุดอ้างอิงในแผนที่ได้

4. ระบบควบคุม

4.1 ระบบระบุตำแหน่ง

ระบบระบุตำแหน่งอาศัยข้อมูลจากเซนเซอร์หลายชนิดร่วมกัน ได้แก่ IMU ใช้ในการวัดมุม Yaw เซนเซอร์วัดความดันในการวัดระดับความลึกจากผิวน้ำ และเซนเซอร์ Lidar ร่วมกับ hector_slam ในการหาตำแหน่งในแนวระนาบ โดยหลักการการทำงานของ hector_slam เป็นการหาตำแหน่งร่วมกันระหว่าง Package เปรียบเทียบข้อมูลสแกน (Scan Matching) กับ Package ทำแผนที่ (Mapping) โดย Scan Matching จะทำการเปรียบเทียบข้อมูลที่ได้จากการสแกนกับแผนที่ที่มีอยู่เพื่อประมวลผลหาตำแหน่งของเซนเซอร์บนหุ่นยนต์เทียบกับแผนที่ในขณะที่ Mapping จะทำการวาดแผนที่สิ่งแวดล้อมเพิ่มเติมด้วยข้อมูลที่ได้จากการสแกนประกอบกับตำแหน่งของเซนเซอร์ในขณะนั้น โดยในตอนเริ่มต้นที่ยังไม่มีแผนที่ โปรแกรมจะทำการวาดสิ่งแวดล้อม ณ จุดเริ่มต้นจากข้อมูลสแกนขึ้นมาโดยกำหนดให้ตำแหน่งเซนเซอร์ขณะนั้นเป็นจุดเริ่มต้นของกรอบอ้างอิงตรึง (Fixed Frame) ด้วยเหตุนี้ hector_slam จึงสามารถใช้ในการหาตำแหน่งเทียบกับสิ่งแวดล้อมที่ไม่รู้จักได้

4.2 ระบบรักษาตำแหน่ง

ในงานวิจัยนี้อาศัยการอ้างอิงตำแหน่งของหุ่นยนต์เทียบกับ

กรอบอ้างอิงตั้งฉากจาก hector_slam ซึ่งเกิดขึ้นโดยมีจุดเริ่มต้นอยู่ที่ตำแหน่งของหุ่นยนต์ขณะเริ่มทำงาน การควบคุมหุ่นยนต์เพื่อรักษาตำแหน่งต้องอาศัยข้อมูลตำแหน่งอ้างอิงที่ผ่านการแปลงจากกรอบอ้างอิงตั้งฉาก มาอยู่ในกรอบอ้างอิงของหุ่นยนต์ (Robot Frame) ซึ่งสามารถหาสมการการแปลงได้จากรูปที่ 6 ดังสมการที่ (1) ได้แก่

$$\tilde{V}_r = \begin{bmatrix} \cos \theta & \sin \theta & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tilde{V} \quad (1)$$

เมื่อ \tilde{V} คือ เวกเตอร์ความเร็ว $[V_x, V_y, V_z, \omega]^T$ เทียบกับ Fixed Frame, \tilde{V}_r คือเวกเตอร์ความเร็ว $[V_{x_r}, V_{y_r}, V_{z_r}, \omega_r]^T$ เทียบกับ Robot Frame, θ คือ มุมของ Robot Frame เทียบกับ Fixed Frame รอบแกน Z และ ω แทนความเร็วเชิงมุมรอบแกน Z

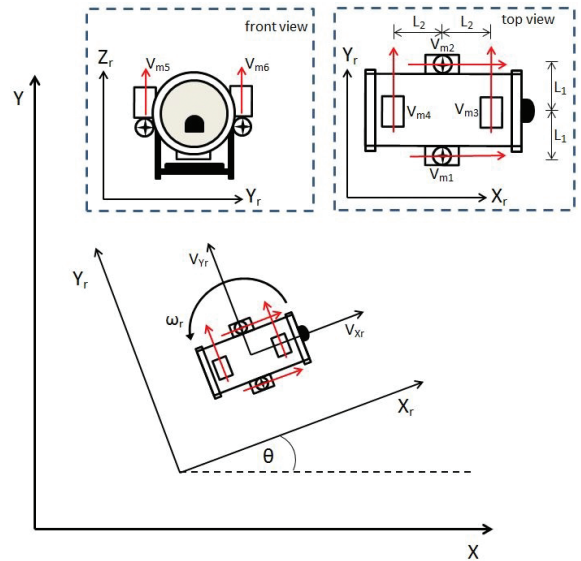
สมการความเร็วเชิงเส้นของมอเตอร์ทั้งหกตัวตามกรอบอ้างอิงรูปที่ 6 ดังสมการที่ (2) ได้แก่

$$\tilde{V}_m = \begin{bmatrix} 1 & 0 & 0 & L_1 \\ 1 & 0 & 0 & -L_1 \\ 0 & 1 & 0 & L_2 \\ 0 & 1 & 0 & -L_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tilde{V}_r \quad (2)$$

เมื่อ \tilde{V}_m คือเวกเตอร์ความเร็วของมอเตอร์ $[V_{m1}, V_{m2}, V_{m3}, V_{m4}, V_{m5}, V_{m6}]^T$, L_1 และ L_2 คือ ระยะห่างระหว่างจุดกึ่งกลางหุ่นยนต์กับตำแหน่งของมอเตอร์

สัญญาณควบคุมมอเตอร์ได้จากการประมวลผลความคลาดเคลื่อนของความเร็วด้วยระบบควบคุม PI โดยความเร็วที่ต้องการ (\tilde{V}_{set}) มีความสัมพันธ์กับสัญญาณตำแหน่งตามสมการที่ (3)

$$\tilde{V}_{set} = \tilde{K} \cdot (\tilde{X}_{set} - \tilde{X}_{current}) \quad (3)$$



รูปที่ 6 กรอบอ้างอิงตั้งฉาก $[X, Y, Z]$ เทียบกับกรอบอ้างอิงของหุ่นยนต์ $[X_r, Y_r, Z_r]$

เมื่อ \tilde{V}_{set} คือเวกเตอร์สถานะความเร็วที่ต้องการ $[V_{xset}, V_{yset}, V_{zset}, \omega_{set}]^T$, \tilde{X}_{set} คือเวกเตอร์สถานะตำแหน่งเป้าหมาย $[X_{set}, Y_{set}, Z_{set}, \theta_{set}]^T$, $X_{current}$ คือเวกเตอร์สถานะตำแหน่งปัจจุบัน $[X_{current}, Y_{current}, Z_{current}, \theta_{current}]^T$ และ \tilde{K} คืออัตราขยายสัญญาณ (Gain) $[K_x, K_y, K_z, K_\omega]^T$

ความเร็ว ณ ตำแหน่งปัจจุบันมีความสัมพันธ์กับสัญญาณตำแหน่งตามสมการที่ (4)

$$\tilde{V}_{current} = (\tilde{X}_{current} - \tilde{X}_{previous}) / \tilde{t} \quad (4)$$

เมื่อ $\tilde{V}_{current}$ คือเวกเตอร์สถานะความเร็วปัจจุบัน $[V_{xcurrent}, V_{ycurrent}, V_{zcurrent}, \omega_{current}]^T$, $\tilde{X}_{previous}$ คือเวกเตอร์สถานะตำแหน่งก่อนปัจจุบัน $[X_{previous}, Y_{previous}, Z_{previous}, \theta_{previous}]^T$, \tilde{t} คือความแตกต่างของเวลาระหว่างตำแหน่งปัจจุบันกับตำแหน่งก่อนปัจจุบัน $[t_x, t_y, t_z, t_\omega]^T$ โดยที่ t_x, t_y มาจากข้อมูล Real Time ของ hector_slam, t_z มาจากข้อมูลของเซนเซอร์วัดความดัน และ t_ω มาจากข้อมูลของ IMU ซึ่งค่า $\omega_{current}$ สามารถอ่านได้โดยตรงจากเซนเซอร์ IMU เช่นกัน

จากสมการที่ (3) และ (4) คำนวณหาความคลาดเคลื่อนของความเร็วได้เป็นดังสมการที่ (5)

$$\tilde{e} = \tilde{V}_{set} - \tilde{V}_{current} \quad (5)$$

เมื่อ \tilde{e} คือ เวกเตอร์สถานะความคลาดเคลื่อนของความเร็วอ้างอิงเทียบกับกรอบอ้างอิงตรง $[e_x, e_y, e_z, e_w]^T$

จากนั้นอาศัยสมการการแปลง (1) แปลงความคลาดเคลื่อนให้มาอยู่ในแกน Robot Frame ได้ดังสมการที่ (6)

$$\tilde{e}_r = \begin{bmatrix} \cos \theta & \sin \theta & 0 & 0 \\ -\sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tilde{e} \quad (6)$$

เมื่อ \tilde{e}_r คือ เวกเตอร์สถานะความคลาดเคลื่อนของความเร็วอ้างอิงเทียบกับกรอบอ้างอิงของหุ่นยนต์ $[e_{x_r}, e_{y_r}, e_{z_r}, e_{w_r}]^T$

ระบบรักษาตำแหน่งของหุ่นยนต์ใช้ตัวควบคุมแบบ PI ซึ่งระบบควบคุมแบบ PI สามารถทำความเข้าใจได้ง่ายประยุกต์ใช้งานได้หลากหลาย [12] ระบบควบคุมแบบป้อนกลับที่มีตัวควบคุมแบบ PI สามารถออกแบบได้ดังสมการที่ (7)

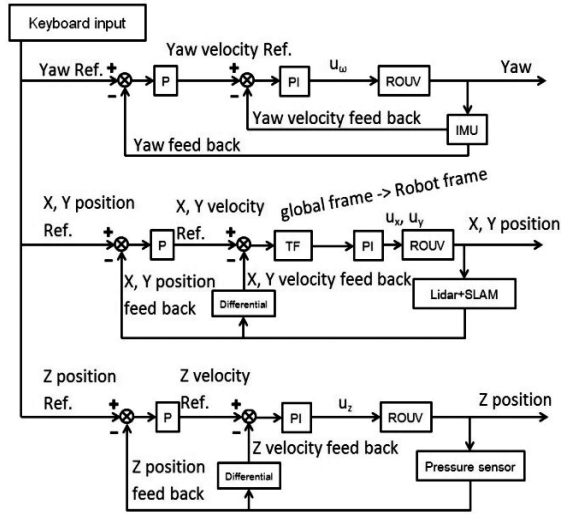
$$\ddot{u}(t) = \tilde{K}_p \cdot \tilde{e}_r(t) + \tilde{K}_i \cdot \sum_{i=t_0}^t \tilde{e}_r(i) \cdot T \quad (7)$$

เมื่อ $\ddot{u}(t)$ คือ เวกเตอร์สัญญาณควบคุม $[u_x, u_y, u_z, u_w]^T$, \tilde{K}_p คือ Proportional Gain $[K_{px}, K_{py}, K_{pz}, K_{pw}]^T$, \tilde{K}_i คือ Integral Gain $[K_{ix}, K_{iy}, K_{iz}, K_{iw}]^T$, และ T คือ คาบเวลาของข้อมูล

สัญญาณควบคุมที่ได้ถูกนำไปสั่งการมอเตอร์ทั้งหกด้วยความสัมพันธ์ตามสมการที่ (8)

$$\ddot{u}_m(t) = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \ddot{u}(t) \quad (8)$$

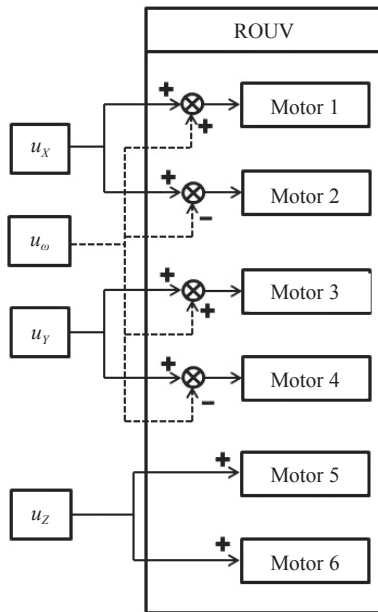
เมื่อ $\ddot{u}_m(t)$ คือสัญญาณควบคุมสำหรับมอเตอร์ตัวที่ 1 ถึงตัวที่ 6 $[u_{m1}, u_{m2}, u_{m3}, u_{m4}, u_{m5}, u_{m6}]^T$ ตามลำดับ



รูปที่ 7 แผนภาพการทำงานของระบบควบคุมตำแหน่งหุ่นยนต์ใต้น้ำแบบสี่ล้ออิสระ

การจูนหาพารามิเตอร์ K_p และ K_i ที่เหมาะสมสำหรับระบบควบคุมนี้ใช้วิธีการจูนโดยสังเกตจากผลการตอบสนองของระบบ โดยทดสอบหาค่าแยกกันในแต่ละแนวแกนการเคลื่อนที่ ในงานวิจัยนี้ ผลตอบสนองของความเร็วเป็นข้อมูลที่มีการเปลี่ยนแปลงความถี่สูงทำให้ยากต่อการสังเกต จึงต้องอาศัยการสังเกตผลตอบสนองของตำแหน่งควบคุมไปด้วยการจูนเริ่มต้นด้วยการปรับตั้งค่า K_i เป็น 0 จากนั้นเพิ่มค่า K_p จนกระทั่งผลตอบสนองของระบบขณะ Steady State เกิดการแกว่ง (Oscillation) จากนั้นทำการลดค่า K_p ที่ได้ลงครึ่งหนึ่งแล้วเพิ่มค่า K_i จนผลตอบสนองตรงกับค่าเป้าหมายโดยค่า K_i จะต้องไม่มากเกินไปจนทำให้ระบบเกิดการแกว่งหรือทำให้เกิดการ Overshoot ที่มากเกินไปในกรณีที่มีการเคลื่อนที่

ระบบระบุตำแหน่งทำการคำนวณค่าตำแหน่งของหุ่นยนต์จากข้อมูลเซนเซอร์ โดยตำแหน่งของหุ่นยนต์จะถูกนำมาแสดงผลเป็นสถานะของหุ่นยนต์และประมวลผลเป็นสัญญาณควบคุมการทรงตัว (Stabilization Control) ตามสมการที่ (3) ถึงสมการที่ (7) แผนภาพการทำงานของระบบควบคุมการทรงตัวแสดงดังรูปที่ 7 และสัญญาณการสั่งงานมอเตอร์เป็นไปตามความสัมพันธ์ของสมการที่ (8) แสดงดังรูปที่ 8

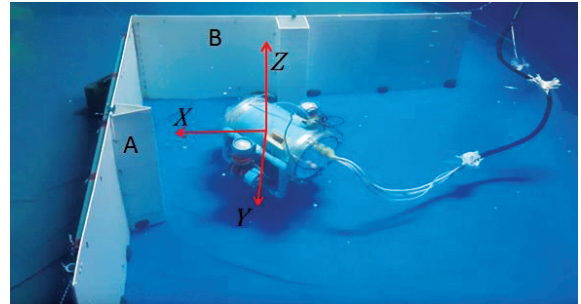


รูปที่ 8 สัญญาณควบคุมมอเตอร์ขับเคลื่อนหุ่นยนต์ใต้น้ำ

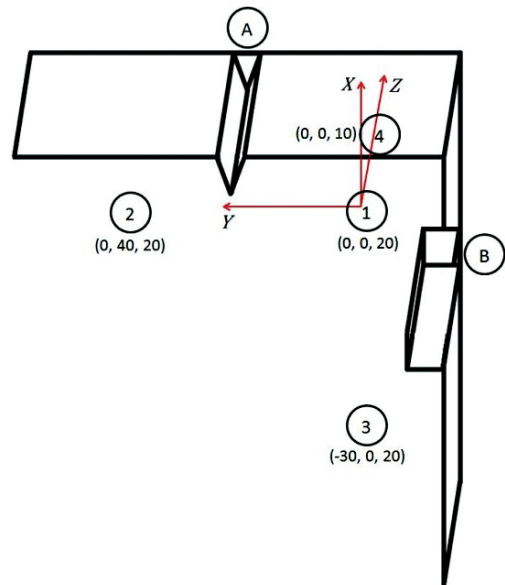
5. การทดลอง

สรณะน้ำฟ้าใบทรงกระบอก ลึก 70 เซนติเมตร ถูกใช้เป็นสถานที่ทดสอบงานวิจัยนี้ ภายในสระมีผนังจำลองลักษณะแสดงดังรูปที่ 9 หุ่นยนต์ขณะทดสอบระบบระบุตำแหน่งและทรงตัวอยู่ระหว่างมุมผนังจำลองนั้น ลักษณะการทดลองเป็นการจำลองการใช้หุ่นยนต์สำรวจโครงสร้างใต้น้ำ วัตถุประสงค์ของการทดลองคือ เพื่อทดสอบความสามารถของระบบระบุตำแหน่งและระบบรักษาตำแหน่งของหุ่นยนต์ขณะทำงานใต้น้ำ

แผนผังสถานที่ทดสอบในมุมมองด้านบนแสดงดังรูปที่ 10 จุดเริ่มต้นของหุ่นยนต์อยู่ในบริเวณที่ 1 เริ่มต้นโดยการควบคุมหุ่นยนต์ให้รักษาทิศทางตั้งฉากกับผนัง A มีค่ามุม Yaw เป็น 0 และรักษาระดับความลึกจากผิวน้ำ (Depth) อยู่ที่ 20 เซนติเมตร จากนั้นควบคุมให้หุ่นยนต์มาใกล้ตำแหน่งหมายเลข 1 แล้วเริ่มการทำงานของ hector_slam ตำแหน่งของหุ่นยนต์ขณะเริ่มต้นนี้เป็นจุดเริ่มต้นของกรอบอ้างอิง Fixed Frame ตามรูปที่ 10 ความถี่ของข้อมูลตำแหน่งในแนวแกน X และ Y มีค่าประมาณ 10 เฮิร์ตซ์ ความถี่ของข้อมูลความลึกและทิศทางมุม Yaw มีค่าประมาณ 20 เฮิร์ตซ์ ขณะทดสอบความเร็วเป้าหมายในแนวแกน X และ Y ถูกจำกัดอยู่ที่ไม่เกิน



รูปที่ 9 การรักษาตำแหน่งของหุ่นยนต์ขณะทดสอบกับผนังใต้น้ำ

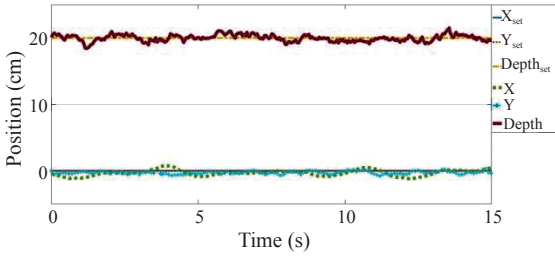


รูปที่ 10 ตำแหน่งในการทดสอบหุ่นยนต์และกรอบอ้างอิงเทียบกับผนัง

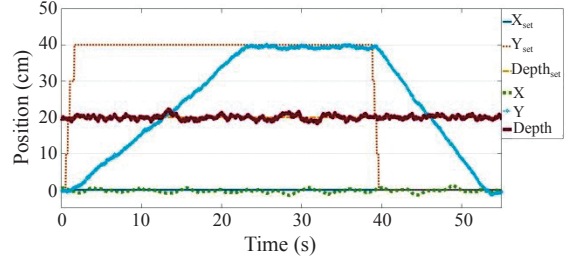
± 5 เซนติเมตรต่อวินาที ในแนวแกน Z ไม่เกิน ± 4.5 เซนติเมตรต่อวินาที และความเร็วเชิงมุมในทิศทางมุม Yaw ไม่เกิน ± 12 องศาต่อวินาที

5.1 การรักษาตำแหน่ง

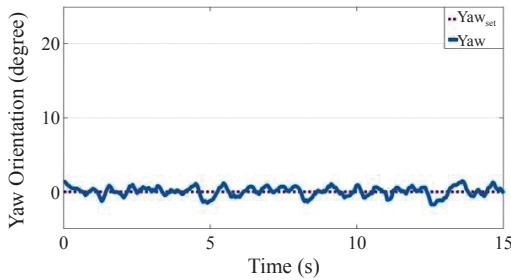
เริ่มการทดสอบรักษาตำแหน่งที่ตำแหน่งหมายเลข 1 พิกัดเป้าหมาย $[X, Y, \text{Depth}, \text{Yaw}]$ เท่ากับ $[0, 0, 20, 0]$ ตามลำดับ ผลการทดสอบแสดงดังรูปที่ 11 และ 12 ผลการทดสอบแสดงให้เห็นว่าหุ่นยนต์สามารถรักษาตำแหน่งและทิศทางเป้าหมายได้พร้อมกันทั้ง 4 องศาอิสระ



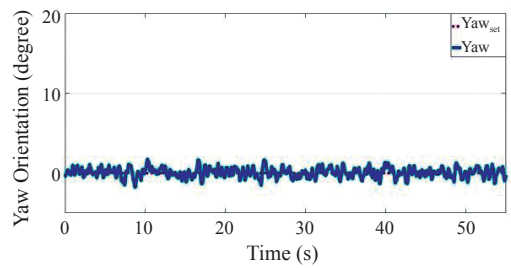
รูปที่ 11 สถานะตำแหน่งของหุ่นยนต์ขณะทดสอบการรักษาตำแหน่งที่ตำแหน่งหมายเลข 1



รูปที่ 13 สถานะตำแหน่งของหุ่นยนต์ขณะรักษาเส้นทางเคลื่อนที่ตามแนวแกน Y



รูปที่ 12 สถานะทิศ Yaw ของหุ่นยนต์ขณะทดสอบการรักษาตำแหน่งที่ตำแหน่งหมายเลข 1

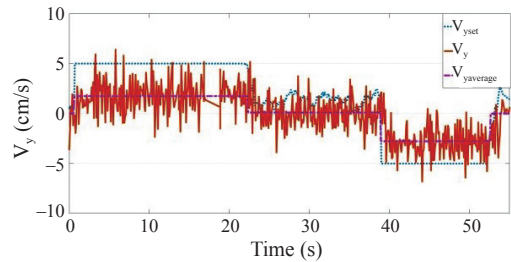


รูปที่ 14 สถานะทิศ Yaw ของหุ่นยนต์ขณะรักษาเส้นทางเคลื่อนที่ตามแนวแกน Y

5.2 การรักษาเส้นทางเคลื่อนที่ในระบบ X-Y

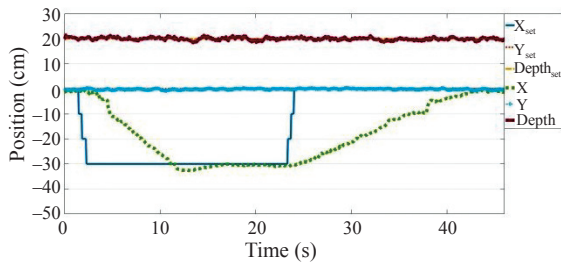
ทำการทดสอบการรักษาตำแหน่งหุ่นยนต์ในเส้นทางขณะเคลื่อนที่เป็นเส้นตรง โดยเปลี่ยนพิกัดเป้าหมายจากตำแหน่งหมายเลข 1 เดิมไปยังตำแหน่งหมายเลข 2 พิกัด $[0, 40, 20, 0]$ รอกจนกระทั่งหุ่นยนต์เคลื่อนถึงตำแหน่งเป้าหมายและรักษาตำแหน่งเป็นเวลา 15 วินาที จากนั้นเปลี่ยนพิกัดเป้าหมายกลับมายังจุดที่ 1 และรอกจนกระทั่งหุ่นยนต์เคลื่อนที่กลับมา ผลการทดสอบแสดงในรูปที่ 13-15

ผลการทดสอบแสดงให้เห็นว่าหุ่นยนต์มีการเปลี่ยนตำแหน่งมุ่งไปยังเป้าหมายโดยเคลื่อนที่เป็นเส้นตรงตามแนวแกน Y และมีอัตราการเปลี่ยนแปลงตำแหน่งตามแนวแกน Y เทียบเวลา (ความชัน) ค่อนข้างคงที่ดังแสดงในรูปที่ 13 ส่วนการขับเคลื่อนหุ่นยนต์ไปมีการรักษาเส้นทางเคลื่อนที่ตามแนวที่กำหนดไว้และยังคงรักษาองศาอิสระอื่น ได้แก่ ตำแหน่ง X, Depth (รูปที่ 13) และทิศ Yaw (รูปที่ 14) ให้คงเดิมไว้ได้ ระบบไม่เกิดการ Overshoot หลังการเคลื่อนที่ถึงเป้าหมาย

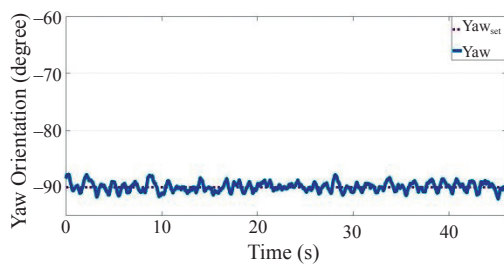


รูปที่ 15 สถานะความเร็วแนวแกน Y ของหุ่นยนต์ขณะรักษาเส้นทางเคลื่อนที่ตามแนวแกน Y

เมื่อพิจารณาความเร็วชั่วขณะของหุ่นยนต์ตามแนวแกน Y ที่เวลาต่างๆ (รูปที่ 15) จะสังเกตเห็นว่าความเร็วที่คำนวณได้ในขณะเวลาใดๆ จะมีการเปลี่ยนแปลงมาก เนื่องจากช่วงเวลาที่นำมาคำนวณหาความเร็วเป็นช่วงเวลาที่สั้นมากตามสมการที่ (4) ทำให้ค่าการเปลี่ยนตำแหน่งต่อเวลามีความผันผวนสูง แต่หากรวมการเคลื่อนที่ของหุ่นยนต์ตลอดช่วงการเคลื่อนที่ที่มีการอัตราการเปลี่ยนตำแหน่งที่ค่อนข้างสม่ำเสมอ นั่นคือความชันของข้อมูลตำแหน่ง Y ในรูป 13 ใกล้เคียงกับ $V_{y_{average}}$ ในรูปที่ 15



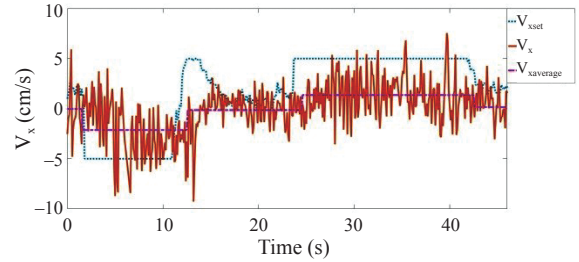
รูปที่ 16 สถานะตำแหน่งของหุ่นยนต์ขณะรักษาเส้นทางการเคลื่อนที่ตามแนวแกน X



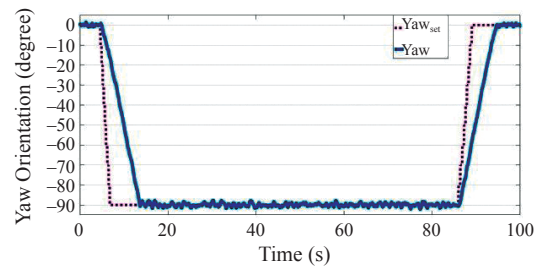
รูปที่ 17 สถานะทิศ Yaw ของหุ่นยนต์ขณะรักษาเส้นทางการเคลื่อนที่ตามแนวแกน X

ต่อมาเมื่อหุ่นยนต์กลับมาถึงจุดที่ 1 หลังจากการทดสอบก่อนหน้า ทำการทดสอบการเคลื่อนที่ไปยังตำแหน่งหมายเลข 3 โดยเปลี่ยนพิกัดเป้าหมายของมุม Yaw ได้พิกัดเป็น $[0, 0, 20, -90]$ เพื่อให้หุ่นยนต์หันหน้าตั้งฉากกับผนัง B ก่อน จากนั้นสั่งการเคลื่อนที่ของหุ่นยนต์ตามแนวแกน X ไปยังตำแหน่งหมายเลข 3 พิกัด $[-30, 0, 20, -90]$ รอจนหุ่นยนต์เคลื่อนที่ถึงเป้าหมายแล้วเปลี่ยนพิกัดกลับมาที่ตำแหน่งหมายเลข 1 พิกัด $[0, 0, 20, -90]$ จากนั้นเปลี่ยนพิกัดกลับมาเป็น $[0, 0, 20, 0]$ เพื่อให้หุ่นยนต์หันหน้าตั้งฉากกับผนัง A เหมือนตอนเริ่มต้น เหตุผลที่กำหนดให้หุ่นยนต์เคลื่อนที่โดยหันหน้าตั้งฉากกับผนังเพราะเป็นจำลองการเคลื่อนที่ในลักษณะการสำรวจสิ่งก่อสร้างหรือโครงสร้างใต้น้ำ

ผลการทดลองแสดงให้เห็นว่าหุ่นยนต์สามารถรักษาเส้นทางการเคลื่อนที่ตามแนวแกน X และ Y ได้ (รูปที่ 16) อีกทั้งยังสามารถปรับและรักษาทิศ Yaw ไว้ได้ (รูปที่ 17 และ 19) ในทำนองเดียวกันกับการทดลองก่อนหน้า ความเร็วเชิงเส้นชั่วขณะ V_x (รูปที่ 18) มีความผันผวนสูงด้วยเหตุผลของการ



รูปที่ 18 สถานะความเร็วแนวแกน X ของหุ่นยนต์ขณะรักษาเส้นทางการเคลื่อนที่ตามแนวแกน X



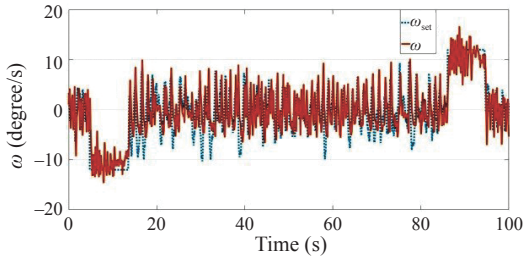
รูปที่ 19 สถานะทิศ Yaw ของหุ่นยนต์ขณะรักษาทิศทางการเคลื่อนที่ตามแนวแกน X

คำนวณการเปลี่ยนตำแหน่งในช่วงเวลาสั้นๆ แต่ความเร็วเฉลี่ยตลอดการเคลื่อนที่ $V_{xaverage}$ ยังสอดคล้องกับความชันของการเปลี่ยนตำแหน่งในรูปที่ 16

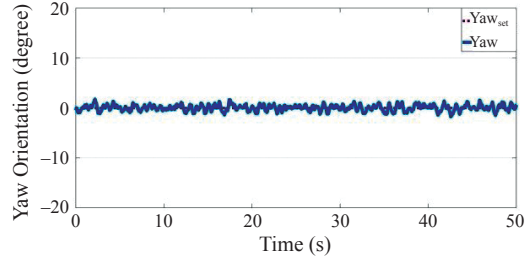
เมื่อพิจารณาความสามารถของระบบควบคุมของการรักษาทิศทาง Yaw จะเห็นได้ว่า หุ่นยนต์ตอบสนองต่อการหมุนเมื่อมีการเปลี่ยนแปลงค่าเชิงมุม Yaw_{set} ดังรูปที่ 19 ดีกว่าการตอบสนองต่อการเคลื่อนที่แบบเชิงเส้น เพราะการเคลื่อนที่เชิงมุมเกิดจากแรงขับของ Thruster 4 ตัว ในขณะที่การเคลื่อนที่เชิงเส้นแต่ละทิศทางเกิดจากแรงขับของ Thruster เพียง 2 ตัวเท่านั้น ทำให้ระบบสามารถควบคุมการเคลื่อนที่เชิงมุมได้ง่ายกว่าการเคลื่อนที่เชิงเส้น ส่วนสถานะความเร็วเชิงมุม ω ก็ตอบสนองทันกับการเปลี่ยนแปลงของ ω_{set} และติดตามการเปลี่ยนแปลงได้ตลอดการเคลื่อนที่ดังแสดงในรูปที่ 20

5.3 การควบคุมระดับความลึก

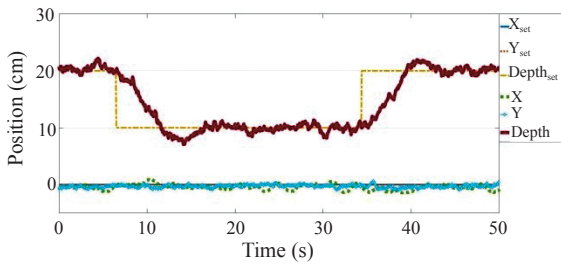
ต่อมาทำการทดสอบการเคลื่อนที่ไปยังตำแหน่งหมายเลข 4



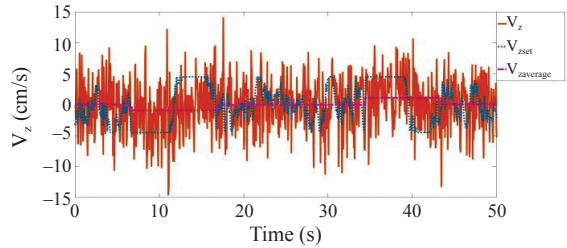
รูปที่ 20 สถานะความเร็วเชิงมุมทิศ Yaw ของหุ่นยนต์ขณะรักษาทิศทางในการเคลื่อนที่ตามแนวแกน X



รูปที่ 22 สถานะทิศ yaw ของหุ่นยนต์ขณะรักษาเส้นทางเคลื่อนที่ตามแนวแกน Z



รูปที่ 21 สถานะตำแหน่งของหุ่นยนต์ขณะรักษาเส้นทางเคลื่อนที่ตามแนวแกน Z



รูปที่ 23 สถานะความเร็วแนวแกน Z ของหุ่นยนต์ขณะรักษาเส้นทางเคลื่อนที่ตามแนวแกน Z

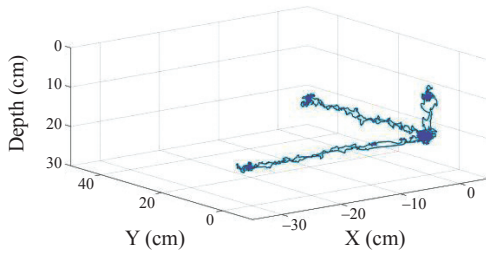
โดยทำการเปลี่ยนพิกัดความลึกเป้าหมายตามแนวแกน Z ที่พิกัด [0, 0, 10, 0] รวจนหุ่นยนต์เคลื่อนที่ถึงเป้าหมายแล้วเปลี่ยนพิกัดกลับมายังตำแหน่งเริ่มต้น [0, 0, 20, 0] ผลการทดสอบแสดงดังรูปที่ 21-23

จากผลการทดลอง การเปลี่ยนแปลงตำแหน่งเป้าหมายในแนวแกน Z หรือการเปลี่ยนแปลงระดับความลึก พบว่าหุ่นยนต์สามารถปรับระดับความลึกของตัวเองไปที่ตำแหน่งหมายเลข 4 แล้วเคลื่อนที่กลับมายังตำแหน่งหมายเลข 1 ได้ ซึ่งขณะเคลื่อนที่ตามแนวแกน Z หุ่นยนต์สามารถรักษาตำแหน่งในแนวแกน X, Y ดังแสดงในรูปที่ 21 และทิศ Yaw ให้คงที่ได้ดังแสดงในรูปที่ 22 ข้อมูลความเร็ว V_z ช่วงขณะ (รูปที่ 23) มีความผันผวนสูงมากรวมทั้งการเปลี่ยนแปลงตำแหน่งตามแนวแกน Z ไม่สม่ำเสมอและเส้นทางเคลื่อนที่ที่เกิด Overshoot มากที่สุดเมื่อเทียบกับการเคลื่อนที่ในแนวแกนอื่น แต่ความเร็วเฉลี่ย $V_{Zaverage}$ (รูปที่ 23) ก็ยังแสดงความสอดคล้องกับอัตราการเปลี่ยนตำแหน่งตามแนวแกน Z (รูปที่ 21) การควบคุมการเคลื่อนที่เชิงเส้นในแนวตั้งอาศัยแรงขับจาก Thruster 2 ตัว

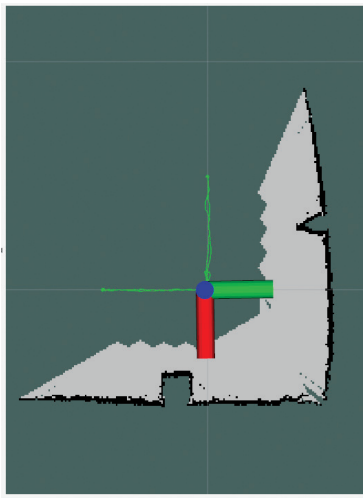
เช่นเดียวกับการเคลื่อนเชิงเส้นในแนวระดับ แต่สาเหตุหลักของการเปลี่ยนตำแหน่งตามแนวแกน Z ที่ไม่สม่ำเสมอเนื่องมาจากเซนเซอร์วัดความดันที่ระบุความลึกมีความละเอียดและความแม่นยำน้อยกว่าเซนเซอร์ Lidar และ IMU ที่วัดการเคลื่อนที่ในแนวระดับ ถึงกระนั้นก็ตาม ระบบควบคุมนี้ก็ยังสามารถที่จะประมวลผลและสั่งการหุ่นยนต์ให้ทำงานรักษาระดับความลึกได้ตามเป้าหมายที่เปลี่ยนไป เส้นทางเคลื่อนที่ของหุ่นยนต์ใน 3 มิติ จากทุกการทดลองดังแสดงในรูปที่ 24

5.4 การสร้างแผนที่ใต้น้ำ

กระบวนการสร้างแผนที่ใต้น้ำเริ่มต้นเมื่อเริ่มการทำงานของโปรแกรม hector_slam ในช่วงต้นของการทดลองและดำเนินการไปตลอดการทดลอง โดย hector_slam จะนำข้อมูลการสแกนระยะจาก Lidar มาประมวลผลหาตำแหน่งของหุ่นยนต์พร้อมทั้งวาดแผนที่ของสิ่งแวดล้อมในเวลาเดียวกัน แผนที่สามมิติสามารถถูกเปิดอ่านระหว่างการทดลองได้



รูปที่ 24 เส้นทางการเคลื่อนที่ของหุ่นยนต์ใน 3 มิติ จากการทดลอง



รูปที่ 25 แผนที่สิ่งแวดล้อมและเส้นทางการเคลื่อนที่ของหุ่นยนต์ที่ได้จากการทดลอง

ด้วยโปรแกรม rviz แผนที่ที่สร้างได้จากการทดลองนี้แสดงดังรูปที่ 25

การสร้างแผนที่ที่เกิดจากข้อมูลของ Lidar เพียงอย่างเดียวไม่ได้ใช้ข้อมูลจาก IMU ซึ่งสังเกตได้ว่าแผนที่ที่สร้างขึ้นมีความคลาดเคลื่อนของแนวผนังที่ได้มาเป็นเส้นโค้ง เพราะในบางขณะที่หุ่นยนต์เคลื่อนที่ เซนเซอร์ Lidar ที่ติดตั้งอยู่บนตัวหุ่นยนต์อยู่ห่างจากผนังเกินช่วงความสามารถที่จะทำงานได้น้ำได้อย่างถูกต้อง ทำให้เกิดความคลาดเคลื่อนในการวัดค่าของเซนเซอร์ รวมไปถึงผลจากอัลกอริทึม hector_slam ที่ใช้วิธีการประมาณตำแหน่งและทิศทางด้วยวิธี Scan Matching ซึ่งวิธีการนี้เหมาะสำหรับระบบที่ไม่มีข้อมูลตำแหน่งสัมบูรณ์ (Absolute Position) แต่ก็มีข้อจำกัดคือ จะเกิดการสะสม

ของความคลาดเคลื่อนจากการประมาณค่าตำแหน่งและทิศทาง โดยจะปรากฏชัดยิ่งขึ้นตามลำดับการสร้างนับตั้งแต่เริ่มต้นการทำงาน [18] ส่วนเส้นทางการเคลื่อนที่ (สีเขียว) ในรูปที่ 25 ยังคงมีลักษณะเป็นเส้นตรงเนื่องจากการทดลองเป็นการเคลื่อนที่ตามกรอบอ้างอิงจริง ผลที่ได้จึงเป็นเส้นตรงเทียบกับกรอบอ้างอิงจริง

ทั้งนี้โปรแกรมที่ใช้ในการสร้างแผนที่ (hector_slam) มีความสามารถในการสร้างแผนที่ได้เพียง 2 มิติ เท่านั้น ซึ่งการสร้างแผนที่แบบ 3 มิติ สามารถทำได้ในอนาคตโดยการพัฒนาโปรแกรมสร้างแผนที่ซึ่งรับและประมวลผลข้อมูลจากการตรวจวัดสภาพสิ่งแวดล้อมแบบสามมิติเพิ่มเข้าไป

6. สรุป

ระบบควบคุมเสถียรภาพของหุ่นยนต์ดำน้ำที่พัฒนาขึ้นสามารถรักษาการทรงตัวของหุ่นยนต์ได้ ระบบระบุตำแหน่งสามารถทำงานได้ต่อเนื่อง สามารถระบุตำแหน่งหุ่นยนต์แบบเวลาจริงได้ โดยมีความคลาดเคลื่อนของข้อมูลน้อยกว่า ± 1 เซนติเมตร ในแนวแกน X, Y, Z และน้อยกว่า ± 1 องศาในทิศ Yaw ซึ่งเพียงพอสำหรับการนำไปประมวลผลในระบบรักษาตำแหน่งหรือเส้นทางการเคลื่อนที่ ผลการทดสอบระบบรักษาตำแหน่ง พบว่า หุ่นยนต์ให้ผลตอบสนองต่อการเปลี่ยนแปลงตำแหน่งที่สถานะ Steady State ด้วยความคลาดเคลื่อนน้อยกว่า ± 3 เซนติเมตร และความคลาดเคลื่อนของตำแหน่งเชิงมุมน้อยกว่า ± 3 องศา ขณะหุ่นยนต์เคลื่อนที่ ระบบควบคุมสามารถรักษาเส้นทางการเคลื่อนที่ในแนวเส้นตรงได้ทั้งในแนวระนาบและแนวตั้ง โดยชนิดของความคลาดเคลื่อนของตำแหน่งที่วัดได้ในการทดลองขึ้นกับจุดอ้างอิงดังนี้ ความคลาดเคลื่อนของตำแหน่งตามแนวความลึกเป็นแบบสมบรูณ์เทียบกับระดับผิวน้ำ ความคลาดเคลื่อนเชิงมุม Yaw เป็นแบบสมบรูณ์เทียบกับสนามแม่เหล็กโลก ความคลาดเคลื่อนของตำแหน่งตามแกน X และแกน Y เป็นแบบสัมพัทธ์เทียบกับตำแหน่งของหุ่นยนต์เมื่อระบบควบคุมเริ่มทำงาน

อย่างไรก็ตาม รายละเอียดของสิ่งแวดล้อมต้องเพียงพอที่จะนำไปใช้ประมวลผลในระบบระบุตำแหน่งได้ รวมไปถึงความเร็วในการเคลื่อนที่ก็เป็นปัจจัยสำคัญที่มีผลต่อความ



แม่นยำในการระบุตำแหน่ง เพราะการประมวลผลของระบบระบุตำแหน่งอาจทำงานไม่ทันกับตำแหน่งที่เปลี่ยนไป ทำให้การระบุตำแหน่งเกิดความคลาดเคลื่อนขึ้นได้

ระบบควบคุมตำแหน่งที่พัฒนาขึ้นถึงแม้จะมีข้อจำกัดเรื่องความแม่นยำหากหุ่นยนต์เคลื่อนเร็ว แต่ก็ยังสามารถนำไปใช้เสริมการทำงานของหุ่นยนต์ที่ไม่มีระบบระบุตำแหน่งและระบบทรงตัวได้ ด้วยการเปิดใช้ระบบควบคุมการรักษาตำแหน่งเมื่อหุ่นยนต์อยู่ในตำแหน่งที่ต้องการหรือเคลื่อนที่สำรวจช้า แล้วเปลี่ยนเป็นการบังคับจากผู้ควบคุมหากต้องการเคลื่อนที่เร็ว รวมถึงไปถึงข้อมูลที่ได้จากการวิจัยสามารถนำไปประยุกต์ใช้ในการเพิ่มประสิทธิภาพการทำงานของระบบระบุตำแหน่งหรือระบบทรงตัวในหุ่นยนต์ดำน้ำแบบสามารถอาศัยสระให้เป็นระบบควบคุมแบบส่องศาสสระโดยเพิ่มเซนเซอร์รองรับตามความเหมาะสม รวมทั้งสามารถนำไปต่อยอดเป็นระบบควบคุมสำหรับหุ่นยนต์ใต้น้ำอัตโนมัติได้

เอกสารอ้างอิง

- [1] V. N. Kuhn, P. L. J. Drews, S. C. P. Gomes, M. A. B. Cunha, and S. S. da C. Botelho, "Automatic control of a ROV for inspection of underwater structures using a low-cost sensing," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 37, no. 1, pp. 361–374, 2015.
- [2] A. Z. Abidin, R. Mardiyanto, and D. Purwanto, "Implementation of PID controller for hold altitude control in underwater remotely operated vehicle," in *Proceedings 2016 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, Lombok, Indonesia, 2016, pp. 665–670.
- [3] S. Timpitak and P. Prempraneerach, "Remotely operated vehicle with depth control," presented at the 3rd TSME International Conference on Mechanical Engineering, Chiang Rai, Thailand, 2012.
- [4] G. C. Karras and K. J. Kyriakopoulos, "Localization of an underwater vehicle using an IMU and a laser-based vision system," in *Proceedings 2007 Mediterranean Conference on Control Automation*, Athens, Greece, 2007, pp. 1–6.
- [5] H.-W. Hsieh, C.-L. Lee, and C.-L. Kuo, "Localization of an underwater robot with inertial sensor fusion models," in *Proceedings 2010 5th IEEE Conference on Industrial Electronics and Applications*, Taichung, Taiwan, 2010, pp. 1562–1567.
- [6] H. Kondo and T. Ura, "Navigation of an AUV for investigation of underwater structures," *Control Engineering Practice*, vol. 12, no. 12, pp. 1551–1559, 2004.
- [7] D. Scaradozzi, L. Gambella, S. M. Zanoli, and G. Conte, "Acoustic mapping and localization of an rov," in *Proceedings 14th Mediterranean Conference on Control and Automation*, Ancona, Italy, 2006, pp. 1–6.
- [8] C. M. Clark, C. S. Olstad, K. Buhagiar, and T. Gambin, "Archaeology via underwater robots: Mapping and localization within maltese cistern systems," in *Proceedings Robotics and Vision 2008 10th International Conference on Control, Automation*, Hanoi, Vietnam, 2008, pp. 662–667.
- [9] J. Snyder, "Doppler Velocity Log (DVL) navigation for observation-class ROVs," in *Proceedings OCEANS 2010 MTS/IEEE SEATTLE*, Seattle, WA, USA, 2010, pp. 1–9.
- [10] Z. Eskinja, Z. Fabekovic, and Z. Vukic, "Localization of autonomous underwater vehicles by sonar image processing," in *Proceedings 49th International Symposium ELMAR-2007*, Zadar, Croatia, 2007,



- pp. 103–106.
- [11] C. Cain and A. Leonessa, “Validation of underwater sensor package using feature based slam,” *Sensors*, vol. 16, no. 3, pp. 380–408, 2016.
- [12] F. A. Azis, M. S. M. Aras, M. Z. A. Rashid, M. N. Othman, and S. S. Abdullah, “Problem identification for underwater remotely operated vehicle (rov): A case study,” *Procedia Engineering*, vol. 41, pp. 554–560, 2012.
- [13] W. M. Bessa, M. S. Dutra, and E. Kreuzer, “Depth control of remotely operated underwater vehicles using an adaptive fuzzy sliding mode controller,” *Robotics and Autonomous Systems*, vol. 56, no. 8, pp. 670–677, 2008.
- [14] T. I. Fossen and S. I. Sagatun, “Adaptive control of nonlinear underwater robotic systems,” in *Proceedings IEEE International Conference on Robotics and Automation Proceedings*, Sacramento, CA, USA, 1991, pp. 1687–1694.
- [15] S. M. Zanolini and G. Conte, “Remotely operated vehicle depth control,” *Control Engineering Practice*, vol. 11, no. 4, pp. 453–459, 2003.
- [16] W. M. Bessa, M. S. Dutra, and E. Kreuzer, “Dynamic positioning of underwater robotic vehicles with thruster dynamics compensation,” *International Journal of Advanced Robotic Systems*, vol. 10, no. 9, pp. 325–332, 2013.
- [17] S. Kohlbrecher, O. von Stryk, J. Meyer, and U. Klingauf, “A flexible and scalable SLAM system with full 3D motion estimation,” in *Proceedings IEEE International Symposium on Safety, Security, and Rescue Robotics*, Kyoto, Japan, 2011, pp. 155–160.
- [18] A. Diosi and L. Kleeman, “Laser scan matching in polar coordinates with application to SLAM,” in *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Alta, Canada, 2005, pp. 3317–3322.