

วิธีการระบุตำแหน่งและสร้างแผนที่พร้อมกันสำหรับหุ่นยนต์ที่ปฏิบัติงานร่วมกัน  
โดยอาศัยข้อมูลเส้นขอบแนวตั้งที่ได้จากกล้องที่มองเห็นได้รอบทิศทาง



นายมหิศร ว่องผาดิ

## สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

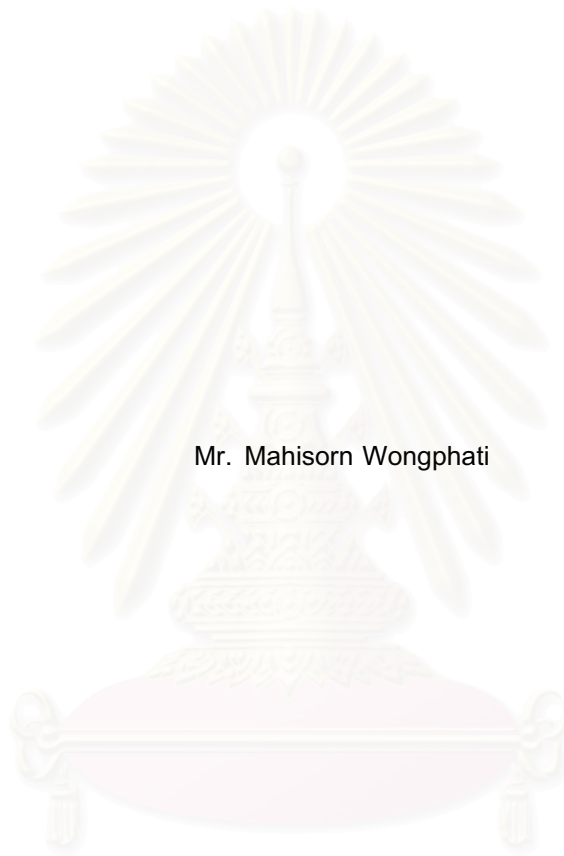
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2549

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

A SIMULTANEOUS LOCALIZATION AND MAPPING METHOD FOR COOPERATIVE MOBILE ROBOTS  
USING VERTICAL EDGES FROM AN OMNIDIRECTIONAL CAMERA



Mr. Mahisorn Wongphati


สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements  
for the Degree of Master of Engineering Program in Computer Engineering  
Department of Computer Engineering  
Faculty of Engineering  
Chulalongkorn University  
Academic Year 2006  
Copyright of Chulalongkorn University

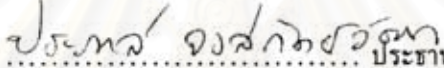
หัวข้อวิทยานิพนธ์   วิธีการระบุตำแหน่งและสร้างแผนที่พร้อมกันสำหรับหุ่นยนต์ที่ปฏิบัติงานร่วมกันโดย  
อาศัยข้อมูลเส้นขอบแนวตั้งที่ได้จากกล้องที่มองเห็นได้รอบทิศทาง  
โดย                   นายมหิศร ว่องผาติ  
สาขาวิชา           วิศวกรรมคอมพิวเตอร์  
อาจารย์ที่ปรึกษา   อาจารย์ ดร.อรรถวิทย์ สุดแสง

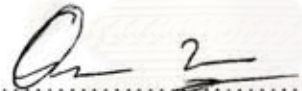
---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยาลัยนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

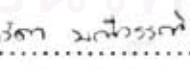
  
..... คณบดีคณะวิศวกรรมศาสตร์  
(ศาสตราจารย์ ดร.ดิเรก ลาวณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์

  
..... ประธานกรรมการ  
(รองศาสตราจารย์ ดร.ประภาส จงสัตยวัฒน์)

  
..... อาจารย์ที่ปรึกษา  
(อาจารย์ ดร.อรรถวิทย์ สุดแสง)

  
..... กรรมการ  
(ผู้ช่วยศาสตราจารย์ ดร.วิทยา วัฒนสุโกประสิทธิ์)

  
..... กรรมการ  
(อาจารย์ ดร.ธิดา มณีวรรณ)

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

มหิศร ว่องผาติ : วิธีการระบุตำแหน่งและสร้างแผนที่พร้อมกันสำหรับหุ่นยนต์ที่ปฏิบัติงานร่วมกัน โดยอาศัยข้อมูลเส้นขอบแนวตั้งที่ได้จากกล้องที่มองเห็นได้รอบทิศทาง (A SIMULTANEOUS LOCALIZATION AND MAPPING METHOD FOR COOPERATIVE MOBILE ROBOTS USING VERTICAL EDGES FROM AN OMNIDIRECTIONAL CAMERA). อาจารย์ที่ปรึกษา : อ. ดร. อรรถวิทย์ สุดแสง, 77 หน้า.

วิทยานิพนธ์ฉบับนี้นำเสนอวิธีการแก้ปัญหาในการทำงานของหุ่นยนต์เคลื่อนที่อันได้แก่ การระบุตำแหน่งและการสร้างแผนที่ โดยอาศัยการประสานงานของหุ่นยนต์สองตัว และการตรวจจับข้อมูลเส้นขอบแนวตั้ง ที่ได้จากกล้องที่สามารถมองภาพได้รอบทิศทางเพียงอย่างเดียว และผลการทดลองบนหุ่นยนต์จริง แสดงให้เห็นว่าวิธีการที่นำเสนอมีประสิทธิภาพในการระบุตำแหน่งและช่วยในการสร้างแผนที่



## สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา .....วิศวกรรมคอมพิวเตอร์ .....  
สาขาวิชา.....วิศวกรรมคอมพิวเตอร์ .....  
ปีการศึกษา .....2549.....

ลายมือชื่อนิสิต .....

ลายมือชื่ออาจารย์ที่ปรึกษา

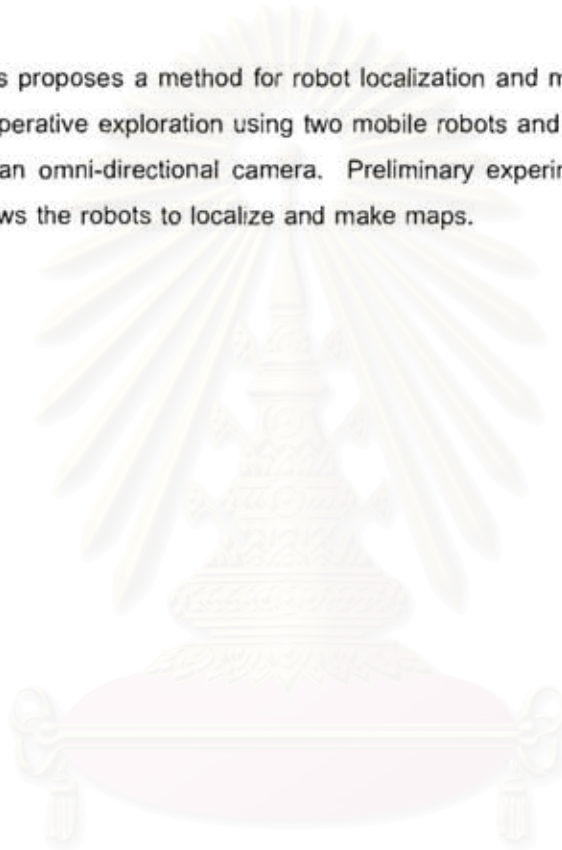


##4770410921 : MAJOR COMPUTER ENGINEERING

KEYWORDS : MOBILE ROBOT/ LOCALIZATION/ MAPPING/ OMNIDIRECTIONAL CAMERA/  
VERTICAL EDGE/ CALIBRATION/ SLAM

MAHISORN WONGPHATI : A SIMULTANEOUS LOCALIZATION AND MAPPING  
METHOD FOR COOPERATIVE MOBILE ROBOTS USING VERTICAL EDGES FROM  
AN OMNIDIRECTIONAL CAMERA. THESIS ADVISOR : ATTAWITH SUDSANG, Ph.D.,  
77 pp.

This thesis proposes a method for robot localization and mapping. The underlying idea is based on cooperative exploration using two mobile robots and ability to easily detect vertical segments from an omni-directional camera. Preliminary experiment shows that the method successfully allows the robots to localize and make maps.



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

Department ..... Computer Engineering .....

Field of study ..... Computer Engineering .....

Academic year ..... 2006 .....

Student's signature .....

Advisor's signature .....

*(Handwritten signatures)*

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยการสนับสนุนเป็นอย่างดีของอาจารย์ที่ปรึกษา ทั้งในส่วนของแนวคิดในการทำงานและข้อมูลที่เป็นหัวใจของงานวิจัย นอกจากนี้ท่านอาจารย์ ดร.อรรถวิทย์ สุดแสง ยังเป็นที่ปรึกษาที่ดีในเรื่องอื่นนอกเหนือจากงานวิจัย มีความเป็นห่วงและให้กำลังใจในการทำงานเมื่อพบอุปสรรคในการทำงาน ข้าพเจ้าจึงขอขอบคุณอาจารย์เป็นอย่างสูงมา ณ ที่นี้

นอกจากอาจารย์ที่ปรึกษาที่ดีแล้ว บุคคลที่มีส่วนสำคัญในการศึกษาตลอดชีวิตของข้าพเจ้า คือ มารดา ซึ่งเห็นความสำคัญของการศึกษาและเป็นโอกาสให้ศึกษาเล่าเรียนตามที่ข้าพเจ้าต้องการ ด้วยการสนับสนุนที่ไม่อาจประมาณค่าได้ทั้ง กำลังทรัพย์ กำลังกาย และกำลังใจ แม้แต่ในช่วงที่ยากลำบากของชีวิต มารดาของข้าพเจ้าก็ไม่เคยแสดงความย่อท้อให้เห็น แต่กลับให้กำลังใจและบอกให้ข้าพเจ้าตั้งใจศึกษาเล่าเรียนให้ได้ผลตามที่มุ่งหวัง ความสำเร็จของวิทยานิพนธ์ฉบับนี้จึงถือได้ว่าเป็นส่วนหนึ่งของความเข้าใจในความสำคัญของการศึกษาและพระคุณที่ไม่อาจตอบแทนได้หมดของมารดา

ในการทำงานงานวิจัยชั้นนี้ยังมีบุคคลที่เกี่ยวข้องอีกมาก ซึ่งประกอบไปด้วย กรรมการวิทยานิพนธ์ และอาจารย์หลายท่านในคณะที่ให้กำลังใจและคำปรึกษาต่างๆ พี่แต่ นัทที นิภาพันธ์ ซึ่งให้ความช่วยเหลือในเรื่องที่เกี่ยวข้องและไม่เกี่ยวข้องกับวิทยานิพนธ์รวมถึงความช่วยในการเรียบเรียงผลงานทางวิชาการชิ้นแรก น้องปัด นริศ บุญให้เจริญ ผู้ที่ออกแบบและสร้างอุปกรณ์รับตำแหน่งกล้องออมินิซึ่งเป็นส่วนสำคัญของงานวิจัยชั้นนี้ เพื่อนๆ พี่ๆ น้องๆ ทุกคนในแล็บวิจัยที่เป็นกำลังใจและให้คำปรึกษาในทุกเรื่อง และน้องอรที่เป็นกำลังใจ ช่วยอ่านและตรวจคำผิดในวิทยานิพนธ์ฉบับนี้



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย . . . . .	ง
บทคัดย่อภาษาอังกฤษ . . . . .	จ
กิตติกรรมประกาศ . . . . .	ฉ
สารบัญ . . . . .	ช
สารบัญตาราง . . . . .	ฅ
สารบัญรูปภาพ . . . . .	ญ
บทที่	
1 บทนำ . . . . .	1
1.1 ปัญหา . . . . .	1
1.2 งานวิจัยที่เกี่ยวข้อง . . . . .	2
1.3 การนำเสนอและลำดับเนื้อหาในวิทยานิพนธ์ . . . . .	3
2 กล้องออมนิจิ (Omnidirectional Camera) . . . . .	4
2.1 ความเป็นมา . . . . .	4
2.2 คุณสมบัติของภาพจากกล้องออมนิจิในงานวิจัย . . . . .	6
2.3 ปัญหาการใช้งานกล้องออมนิจิ . . . . .	8
2.4 การหาค่าพารามิเตอร์และแก้ปัญหาการเรียงตัวของชิ้นส่วนในกล้องออมนิจิ . . . . .	9
2.4.1 ขั้นตอนการหาพารามิเตอร์ของกล้อง . . . . .	11
2.5 ข้อมูลอ้างอิงเพิ่มเติม . . . . .	13
3 ปัญหาการระบุตำแหน่งและสร้างแผนที่ . . . . .	14
3.1 โครงสร้างและส่วนประกอบของปัญหา . . . . .	14
3.2 การระบุตำแหน่งและสร้างแผนที่ในเชิงความน่าจะเป็น . . . . .	14
3.3 Gaussian Filter . . . . .	16
3.3.1 Kalman Filter . . . . .	16
3.3.2 Extended Kalman Filter: EKF . . . . .	17
3.4 Particle Filter . . . . .	19
3.4.1 ส่วนประกอบขั้นตอนการทำงานของ Particle Filter . . . . .	20
3.5 FastSLAM . . . . .	20
3.5.1 ส่วนประกอบและขั้นตอนการทำงานของ FastSLAM . . . . .	21
4 การระบุตำแหน่งและสร้างแผนที่โดยอาศัยข้อมูลจากกล้องออมนิจิ . . . . .	24
4.1 การตรวจจับเส้นแหวดิ่งในภาพจากกล้องออมนิจิ . . . . .	24
4.2 การระบุตำแหน่งของเส้นแหวดิ่งโดยอาศัยข้อมูลจากการวัดมุม . . . . .	27
4.3 FastSLAM โดยใช้ข้อมูลเส้นแหวดิ่งในสภาพแวดล้อม . . . . .	29
4.3.1 โมเดลการเคลื่อนที่ของหุ่นยนต์ . . . . .	30

บทที่	หน้า
4.3.2 โมเดลการวัดตำแหน่งของเส้นแนวตั้ง . . . . .	31
4.3.3 ขั้นตอนการระบุตำแหน่งและสร้างแผนที่โดยอาศัยข้อมูลเส้นแนวตั้งด้วย วิธีการ FastSLAM . . . . .	33
4.3.3.1 ความหมายของตัวแปรของระเบียบวิธีการ FastSLAM ในงานวิจัย . . .	33
4.3.3.2 ปัญหาความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกต . . . . .	34
4.3.3.3 การกำจัดจุดตำแหน่งที่ผิดพลาด . . . . .	35
4.3.3.4 การปรับแก้ระเบียบวิธีการ FastSLAM 1.0 เพื่อใช้ในงานวิจัย . . . . .	35
4.4 การทำงานร่วมกันของหุ่นยนต์สองตัวในการระบุตำแหน่งและสร้างแผนที่ . . . . .	36
4.4.1 การตรวจหาและวัดตำแหน่งหุ่นยนต์ที่ปฏิบัติร่วมกัน . . . . .	36
4.4.2 วิธีการรวมข้อมูลแผนที่ของหุ่นยนต์ที่แยกกันปฏิบัติงาน . . . . .	37
<b>5 การทดลองและผลการทำงาน . . . . .</b>	<b>41</b>
5.1 อุปกรณ์สำหรับการทดลอง . . . . .	41
5.1.1 หุ่นยนต์ . . . . .	41
5.1.2 กล้องอ้อมนิ . . . . .	41
5.2 สถานที่ทดลอง . . . . .	42
5.3 การทดลองระบุตำแหน่งและสร้างแผนที่ด้วยหุ่นยนต์หนึ่งตัว . . . . .	43
5.4 การทดลองระบุตำแหน่งและสร้างแผนที่โดยการทำงานร่วมกันของหุ่นยนต์สองตัว . . . . .	54
<b>6 สรุปผลของงานวิจัยและสิ่งที่ได้รับ . . . . .</b>	<b>58</b>
6.1 สรุปผลการทดลอง . . . . .	58
6.2 แนวทางงานวิจัยในขั้นถัด . . . . .	59
6.3 สิ่งที่ได้รับจากงานวิจัย . . . . .	59
รายการอ้างอิง . . . . .	60
ประวัติผู้เขียนวิทยานิพนธ์ . . . . .	65

บทที่	หน้า
4.3.2 โมเดลการวัดตำแหน่งของเส้นแนวตั้ง . . . . .	31
4.3.3 ขั้นตอนการระบุตำแหน่งและสร้างแผนที่โดยอาศัยข้อมูลเส้นแนวตั้งด้วย วิธีการ FastSLAM . . . . .	33
4.3.3.1 ความหมายของตัวแปรของระเบียบวิธีการ FastSLAM ในงานวิจัย . . .	33
4.3.3.2 ปัญหาความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกต . . . . .	34
4.3.3.3 การกำจัดจุดตำแหน่งที่ผิดพลาด . . . . .	35
4.3.3.4 การปรับแก้ระเบียบวิธีการ FastSLAM 1.0 เพื่อใช้ในงานวิจัย . . . . .	35
4.4 การทำงานร่วมกันของหุ่นยนต์สองตัวในการระบุตำแหน่งและสร้างแผนที่ . . . . .	36
4.4.1 การตรวจหาและวัดตำแหน่งหุ่นยนต์ที่ปฏิบัติร่วมกัน . . . . .	36
4.4.2 วิธีการรวมข้อมูลแผนที่ของหุ่นยนต์ที่แยกกันปฏิบัติงาน . . . . .	37
<b>5 การทดลองและผลการทำงาน . . . . .</b>	<b>41</b>
5.1 อุปกรณ์สำหรับการทดลอง . . . . .	41
5.1.1 หุ่นยนต์ . . . . .	41
5.1.2 กล้องออมินิ . . . . .	41
5.2 สถานที่ทดลอง . . . . .	42
5.3 การทดลองระบุตำแหน่งและสร้างแผนที่ด้วยหุ่นยนต์หนึ่งตัว . . . . .	43
5.4 การทดลองระบุตำแหน่งและสร้างแผนที่โดยการทำงานร่วมกันของหุ่นยนต์สองตัว . . . .	54
<b>6 สรุปผลของงานวิจัยและสิ่งที่ได้รับ . . . . .</b>	<b>58</b>
6.1 สรุปผลการทดลอง . . . . .	58
6.2 แนวทางงานวิจัยในขั้นถัด . . . . .	59
6.3 สิ่งที่ได้รับจากงานวิจัย . . . . .	59
รายการอ้างอิง . . . . .	60
ประวัติผู้เขียนวิทยานิพนธ์ . . . . .	65

## สารบัญตาราง

ตารางที่	หน้า
5.1 พารามิเตอร์ของโปรแกรมหาและระบุตำแหน่งเส้นแนวตั้ง . . . . .	44
5.2 พารามิเตอร์ของโปรแกรมระบุตำแหน่งและสร้างแผนที่ . . . . .	45
5.3 พารามิเตอร์ของแผนที่อ้างอิงจากการทดลอง . . . . .	46



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย



## สารบัญญรูปภาพ

รูปที่	หน้า
2.1 ลักษณะการตัดกันของลำแสงในกล้องอ้อมนินที่สำคัญต่อการหา Mapping Function . . . . .	4
2.2 การรับภาพรอบทิศทางในสามมิติ (ภาพจากงานวิจัย [2]) . . . . .	5
2.3 ลักษณะและส่วนประกอบของเลนส์ตาปลา . . . . .	5
2.4 กลไกถ่ายภาพมุมกว้างโดยการหมุนกล้อง . . . . .	5
2.5 โครงสร้างการรับภาพสะท้อนจากกระจกทรงพาราโบล่าผ่าน Orthographic Lens . . . . .	6
2.6 ลักษณะและส่วนประกอบของกล้องอ้อมนินในงานวิจัย . . . . .	7
2.7 ภาพจากกล้องอ้อมนิน . . . . .	7
2.8 ระยะห่างของวัตถุจากกล้องอ้อมนินเทียบกับระยะห่างจากจุดศูนย์กลางภาพ . . . . .	8
2.9 ตัวอย่างภาพจากกล้องอ้อมนินเมื่อขึ้นส่วนเรียงตัวไม่สมบูรณ์ . . . . .	8
2.10 ความสัมพันธ์ระหว่างลำแสงในสามมิติกับตำแหน่งในภาพ . . . . .	10
2.11 ภาพจากงานวิจัย [46] แสดงการรับภาพรอบทิศทางในรูป (a) และภาพรอบทิศทางบนฉาก รับภาพหรือ เมื่อส่วนประกอบเรียงตัวอย่างสมบูรณ์ในรูป (b) โดยมีภาพ (c) แสดงลักษณะ ของภาพจากกล้องเมื่อการเรียงตัวของอุปกรณ์ไม่สมบูรณ์ ซึ่งสามารถอธิบายความสัมพันธ์ ระหว่าง (b) และ (c) ได้ด้วยสมการที่ 2.5 . . . . .	11
2.12 โปรแกรม Omnidirectional Camera Calibration Toolbox ในงานวิจัย [46] . . . . .	11
2.13 ภาพถ่ายดาวารวมการูกที่ทราบขนาดแน่นอนสำหรับใช้ในการหาค่าพารามิเตอร์กล้อง . . . . .	12
2.14 ตำแหน่งมุมของดาวารวมในภาพ . . . . .	12
2.15 ผลจากการหาค่าพารามิเตอร์สำหรับกล้องอ้อมนินในงานวิจัย . . . . .	12
3.1 ตัวอย่าง ตำแหน่ง ของ หุ่นยนต์ ที่ เกิด จาก การ ประมาณค่า เมื่อ สิ่ง หุ่นยนต์ เคลื่อนที่ จาก จุดเริ่มต้นไป 1 เมตร โดยมี $\sigma_{tran} = 5cm/m$ และ $\sigma_{drift} = 10^{\circ}/m$ . . . . .	22
4.1 รูปแบบการระบุตำแหน่งและสร้างแผนที่ . . . . .	24
4.2 ความแตกต่างของภาพก่อนและหลังทำ Gaussian Blur . . . . .	25
4.3 ภาพที่ผ่านการแปลงเป็นโทนสีเทา . . . . .	25
4.4 ภาพขอบแนวตั้งที่เกิดจากขอบซ้ายและขวาของเส้นแนวตั้งในภาพ . . . . .	26
4.5 ตำแหน่งเส้นแนวตั้งจากการใช้ Kernel สองแบบตามขั้นตอนที่ 3 . . . . .	26
4.6 เส้นแนวตั้งจากกระบวนการ Hough Line Transformation . . . . .	27
4.7 การระบุตำแหน่งของเส้นแนวตั้งในภาพ . . . . .	28
4.8 การหาตำแหน่งของเส้นแนวตั้งด้วยการ Triangulation . . . . .	29
4.9 ความเปลี่ยนแปลงของเส้นแนวตั้งในภาพจากกล้องอ้อมนินเมื่อหุ่นยนต์เคลื่อนที่ . . . . .	29
4.10 การเคลื่อนที่ของหุ่นยนต์ และลักษณะของความไม่แน่นอนที่เกิดขึ้น . . . . .	30
4.11 เส้นทางจากค่าซึ่งวัดโดย encoder และเส้นทางทั้งหมดที่สามารถประมาณได้ผ่านโมเดล การเคลื่อนที่ของหุ่นยนต์ตามสมการ 4.1 โดยกำหนดให้ $\sigma_{tran} = 1cm/m$ $\sigma_{drift} = 5^{\circ}/m$ และ $\sigma_{rot} = 5^{\circ}/\pi$ . . . . .	31
4.12 ภาพเปรียบเทียบการกระจายตัวของตำแหน่งเส้นแนวตั้งจากการวัดและการประมาณด้วย โมเดลการวัดตำแหน่ง . . . . .	32
4.13 ปัญหาความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกต เมื่อไม่ทราบตำแหน่งแน่นอนของ หุ่นยนต์ (ภาพจากงาน [44]) . . . . .	34



รูปที่	หน้า	
4.14	ลักษณะของเครื่องหมายหุ่นยนต์และตำแหน่งจากการคำนวณ . . . . .	37
4.15	การคำนวณตำแหน่งของหุ่นยนต์โดยอาศัยมุมและระยะเคลื่อนที่ของหุ่นยนต์ . . . . .	37
4.16	ภาพแสดงการทำงานร่วมกันในการระบุตำแหน่งและสร้างแผนที่ ณ เวลา $k = 4$ หุ่นยนต์พบกันเป็นครั้งแรก หุ่นยนต์ตัวที่หนึ่งหยุดรอตัวที่สองเคลื่อนที่ต่อไปที่เวลา $k = 5$ เพื่อทำการวัดตำแหน่งแล้วจึงปรับปรุงข้อมูลแผนที่ร่วมกัน . . . . .	38
4.17	แผนที่จากการแยกกับทำงานของหุ่นยนต์สองตัว . . . . .	38
4.18	การรวมข้อมูลแผนที่ที่ขัดแย้งกันโดยอาศัยการวัดมุมของเครื่องหมายหุ่นยนต์ . . . . .	39
5.1	หุ่นยนต์ทั้งสองตัวที่ใช้ในการทดลอง . . . . .	42
5.2	แผนที่ยึดกล้องอ้อมนิปรับระดับได้ . . . . .	42
5.3	แผนผังสถานที่ทดลองบนห้องแลป ISL2 . . . . .	43
5.4	รายละเอียดของสถานที่ทดลอง . . . . .	43
5.5	พารามิเตอร์ที่มีผลต่อข้อมูลการวัดตำแหน่งเส้นแนวตั้ง . . . . .	44
5.6	ภาพแสดงตำแหน่งของหุ่นยนต์และแผนที่จากการทดลองเทียบกับแผนผังของสถานที่ทดลอง . . . . .	46
5.7	ภาพเปรียบเทียบเส้นทางการเคลื่อนที่ของหุ่นยนต์กับตำแหน่งจริงซึ่งประมาณจากตำแหน่งของหุ่นยนต์บนกระเบื้อง . . . . .	47
5.8	กำหนดให้โมเดลการวัดตำแหน่งมี $\sigma_r = 0.001m, \sigma_{theta} = 0.001^\circ$ . . . . .	47
5.9	กำหนดให้โมเดลการวัดตำแหน่งมี $\sigma_r = 1m, \sigma_{theta} = 1.5^\circ$ . . . . .	48
5.10	กำหนดให้โมเดลการวัดตำแหน่งมี $\sigma_r = 0.5m, \sigma_{theta} = 10^\circ$ . . . . .	48
5.11	กำหนดให้โมเดลการเคลื่อนที่มี $\sigma_{tran} = 0m, \sigma_{drift} = 0^\circ/m, \sigma_{rot} = 0^\circ/\pi$ . . . . .	49
5.12	กำหนดให้โมเดลการเคลื่อนที่มี $\sigma_{tran} = 0.5m, \sigma_{drift} = 5^\circ/m, \sigma_{rot} = 5^\circ/\pi$ . . . . .	50
5.13	กำหนดให้โมเดลการเคลื่อนที่มี $\sigma_{tran} = 1m, \sigma_{drift} = 20^\circ/m, \sigma_{rot} = 20^\circ/\pi$ . . . . .	50
5.14	กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น $I = 10$ . . . . .	51
5.15	กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น $I = 20$ . . . . .	51
5.16	กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น $I = 50$ . . . . .	52
5.17	กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น $I = 100$ . . . . .	52
5.18	กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น $I = 200$ . . . . .	53
5.19	กำหนดให้ระยะทำงานของเซนเซอร์เป็น $r_{max} = 2m$ . . . . .	54
5.20	กำหนดให้ระยะทำงานของเซนเซอร์เป็น $r_{max} = 3m$ . . . . .	54
5.21	กำหนดให้ระยะทำงานของเซนเซอร์เป็น $r_{max} = 4m$ . . . . .	55
5.22	กำหนดให้ ระยะทำงานของเซนเซอร์เป็น $r_{max} = 3m$ และ ระยะของเซนเซอร์ในระเบียบวิธีการ FastSLAM เป็น $Max_{range} = 0.5$ . . . . .	55
5.23	ลักษณะของแผนที่จากหุ่นยนต์ทั้งสองตัว . . . . .	56
5.24	ลักษณะของแผนที่ก่อนและหลังการปรับแก้ค่าพิกัดของหุ่นยนต์ . . . . .	57
5.25	แผนที่ รวม จาก หุ่นยนต์ ทั้ง สอง เมื่อ ผ่าน การ ปรับ แก้ ค่า และ กำจัด จุด สังกะสี ที่ มีความแปรปรวนสูง . . . . .	57

# บทที่ 1

## บทนำ

### 1.1 ปัญหา

การนำข้อมูลจากกล้องซึ่งสามารถรับภาพได้รอบทิศทางหรือกล้องอ้อมนิ (Omnidirectional Camera) มาใช้ในงานวิจัยได้รับความสนใจในกลุ่มนักวิจัยเป็นจำนวนมากในช่วง 15 ปีที่ผ่านมา [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] โดยเฉพาะการนำมาใช้งานเป็นเซนเซอร์ของหุ่นยนต์ [11, 12, 13, 14, 15, 16, 17, 18, 19] เนื่องจากคุณสมบัติโดดเด่นในการรับภาพ 360° รอบทิศทางและสามารถออกได้หลายลักษณะตามการใช้งาน [20]

ปัญหาสำคัญในงานวิจัยด้านหุ่นยนต์ คือ การระบุตำแหน่งและการสร้างแผนที่ ในสภาพแวดล้อมที่ปฏิบัติงาน ปัญหาดังกล่าวมีการนำเสนอวิธีการแก้ปัญหาในงานวิจัยเป็นจำนวนมาก [21, 22] ซึ่งสามารถแบ่งได้เป็นสองกลุ่มหลัก คือ การแก้ปัญหาการระบุตำแหน่งเมื่อทราบแผนที่ของสถานที่ที่ปฏิบัติงาน และปัญหาการสร้างแผนที่โดยรู้ตำแหน่งแน่นอนของหุ่นยนต์ ซึ่งแตกต่างจากการทำงานจริงของหุ่นยนต์ส่วนใหญ่ เนื่องจากต้องทำงานในสภาพแวดล้อมที่ไม่รู้จักหรือมีการเปลี่ยนแปลงตลอดเวลา โดยเฉพาะในงานที่เกี่ยวข้องกับหุ่นยนต์เคลื่อนที่ได้อิสระ (Mobile Robot) เนื่องจากข้อมูลที่จำเป็นในการระบุตำแหน่งของหุ่นยนต์ คือ ความรู้เกี่ยวกับสภาพแวดล้อมที่ปฏิบัติงานหรือแผนที่ แต่ในการสร้างแผนที่ต้องการทราบตำแหน่งแน่นอนของหุ่นยนต์ เพื่อนำข้อมูลจากเซนเซอร์มาสร้างเป็นแผนที่ต่อเนื่องกันตามเส้นทางการเคลื่อนที่ของหุ่นยนต์ ซึ่งเป็นความต้องการที่ขัดแย้งกันเองในรับรู้และนำข้อมูลจากเซนเซอร์มาใช้งาน

ปัญหาข้างต้นนำไปสู่การออกแบบวิธีการแก้ปัญหาการระบุตำแหน่งพร้อมกับการสร้างแผนที่ (Simultaneous Localization and Mapping: SLAM) [23] ซึ่งต้องการเซนเซอร์ที่สามารถให้ข้อมูลที่แม่นยำและครอบคลุมพื้นที่การทำงานของหุ่นยนต์ รวมถึงสามารถให้ข้อมูลในการแยกแยะจุดสังเกต (Feature) ซึ่งเป็นจุดเด่นของสิ่งแวดล้อมที่ได้ดีเพื่อใช้ในการสร้างแผนที่ และช่วยในการระบุตำแหน่งของหุ่นยนต์ เซนเซอร์ที่ให้ข้อมูลภาพจึงเป็นตัวเลือกที่ได้รับความนิยมในการเลือกมาใช้งานเป็นเซนเซอร์ของหุ่นยนต์ในงานระบุตำแหน่งและสร้างแผนที่ [24, 25, 26, 27, 28]

แต่ปัญหาสำคัญที่สุดของการใช้ข้อมูลจากกล้องทั่วไป คือ ข้อจำกัดของมุมมองและความละเอียดของภาพ งานวิจัยชิ้นนี้จึงเล็งเห็นคุณสมบัติของกล้องอ้อมนิซึ่งสามารถรับภาพได้รอบทิศทางและมีความละเอียดของข้อมูลเชิงมุมสูง เมื่อเทียบกับความละเอียดของ CCD เซนเซอร์ทั่วไป ซึ่งมีขนาดประมาณ  $640 \times 480$  พิกเซล และตรงกับความต้องการของการระบุตำแหน่งและสร้างแผนที่ โดยอาศัยข้อมูลจากการวัดมุมเพียงอย่างเดียวในการทำงาน (Bearing-only SLAM) [29]

นอกจากนั้นปัญหาสำคัญอีกปัญหาหนึ่งของการระบุตำแหน่งและสร้างแผนที่ของหุ่นยนต์ คือ ระยะเวลาการทำงานของเซนเซอร์ โดยเฉพาะอย่างยิ่งในกรณีที่ระยะของจุดสังเกตซึ่งจำเป็นต่อการทำงานของหุ่นยนต์ อยู่ห่างกันมากหรือไม่สามารถตรวจจับได้ในระยะทำงานของเซนเซอร์ เช่น การทำงานในที่โล่งหรือโรงอาคารขนาดใหญ่ ส่งผลให้ไม่สามารถระบุตำแหน่งของหุ่นยนต์และสร้างแผนที่ในสภาพแวดล้อมนั้นได้อย่างสมบูรณ์ จึงมีแนวคิดในการใช้งานหุ่นยนต์หลายตัวทำงานร่วมกัน [27, 30] เพื่อทำหน้าที่

เป็นจุดสังเกตเสมือน และวัดตำแหน่งของกันและกัน เพื่อเพิ่มระยะปฏิบัติงานและความรวดเร็วในการระบุตำแหน่งและสร้างแผนที่ โดยการแลกเปลี่ยนข้อมูลระหว่างหุ่นยนต์ที่ปฏิบัติงานร่วมกัน การทำงานดังกล่าวต้องการเซนเซอร์ในการตรวจหาหุ่นยนต์ที่ทำงานร่วมกันได้อย่างแม่นยำและครอบคลุมพื้นที่การทำงาน ด้วยการแยกเครื่องหมายพิเศษที่ติดตั้งบนหุ่นยนต์ออกจากสภาพแวดล้อม ซึ่งเป็นคุณสมบัติของข้อมูลภาพที่ได้จากกล้องอ้อมโดยไม่ต้องใช้ข้อมูลเสริมจากเซนเซอร์ชนิดอื่น

จากปัญหาการระบุตำแหน่งและสร้างแผนที่ คุณสมบัติในการวัดมุมของกล้องอ้อมนี้ และความสามารถในการให้ข้อมูลภาพสำหรับตรวจจับหุ่นยนต์ที่ทำงานร่วมกัน ทำให้งานวิจัยชิ้นนี้มีแนวความคิดในการหาวิธีการสำหรับ ระบุตำแหน่งและสร้างแผนที่ร่วมกันของหุ่นยนต์ในสภาพแวดล้อมภายในอาคาร ด้วยข้อมูลจากกล้องอ้อมนี้เพียงอย่างเดียว โดยมีความถูกต้องของแผนที่และเส้นทางการเคลื่อนที่ของหุ่นยนต์เป็นตัวชี้วัดความสำเร็จของงานวิจัย

## 1.2 งานวิจัยที่เกี่ยวข้อง

การใช้งานกล้องอ้อมในการแก้ปัญหาด้านหุ่นยนต์เป็นงานที่เริ่มได้รับความสนใจในช่วงปี 1990 โดยมีในงานวิจัย [1] โดย Y. Yagi ที่อ้างกล่าวได้ว่านักวิจัยกลุ่มแรกๆ ที่นำข้อมูลจากกล้องอ้อมมาประยุกต์ใช้งานด้านหุ่นยนต์ หลังจากนั้นผู้วิจัยกลุ่มเดิมได้นำเสนอการใช้งานกล้องอ้อมในงานด้านหุ่นยนต์มาอย่างต่อเนื่อง [7] รวมถึงสร้างแผนที่โดยใช้ข้อมูลจากหุ่นยนต์ที่แยกกันทำงานโดนอาศัยเส้นแนวตั้งในสภาพแวดล้อมในงาน [31] แต่ไม่มีการกล่าวถึงการแก้ปัญหาการระบุตำแหน่งพร้อมกับการสร้างแผนที่

หลังจากนั้นได้มีการนำกล้องอ้อมไปใช้งานอย่างแพร่หลายในงานด้านหุ่นยนต์รูปแบบต่างๆ เช่น การแข่งขันหุ่นยนต์เตะฟุตบอลอัตโนมัติขนาดกลาง (RoboCup Medium Size League) [32] ใช้กล้องอ้อมในการระบุตำแหน่งของหุ่นยนต์ในสนามแข่งขัน งานระบุตำแหน่งของหุ่นยนต์ในสภาพแวดล้อมทั้งภายในอาคาร [33, 13, 34, 14] และภายนอกอาคาร [17] งานสร้างแผนที่ของสภาพแวดล้อม [11, 35, 36, 37, 38] และงานระบุตำแหน่งพร้อมการสร้างแผนที่โดนใช้ข้อมูลจากกล้องอ้อมนี้ มีการนำเสนอวิธีการการปัญหาในหลายงานวิจัย เช่น งานวิจัย [39] และ [40] นำเสนอการแก้ปัญหาโดยการระบุตำแหน่งเส้นแนวตั้งด้วยกล้องอ้อมแบบสเตอริโอ รวมถึงงานวิจัยอื่นอีกเป็นจำนวนมาก [41] นำเสนอการแก้ปัญหาด้วยข้อมูลเส้นแนวตั้งจากกล้องอ้อม

เมื่อค้นคว้าในรายละเอียดของงานวิจัยที่น่าสนใจจาก [41] รวมถึงรายการอ้างอิงจากงานต่างๆ พบว่าการระบุตำแหน่งพร้อมการสร้างแผนที่โดยอาศัยข้อมูลจากการกล้องอ้อมนี้โดยมาก ใช้ภาพจากกล้องอ้อมนี้เป็นข้อมูลเสริม สำหรับการระบุตำแหน่งของหุ่นยนต์ในแผนที่ที่สร้างจากเลเซอร์วัดระยะทางมีความแม่นยำสูงกว่า [42, 43] และยังไม่พบรายงานการประยุกต์ใช้ข้อมูลเส้นแนวตั้งจากกล้องอ้อมนี้เข้ากับระเบียบวิธีการ FastSLAM [44] ซึ่งเป็นวิธีการที่อ้างกล่าวได้ว่ามีประสิทธิภาพในการทำงานและมีการใช้งานในงานวิจัยเป็นจำนวนมาก [23] ในการ ระบุตำแหน่งพร้อมการสร้างแผนที่โดยหุ่นยนต์ที่ปฏิบัติงานร่วมกัน ซึ่งเป็นหัวข้อของงานวิจัยชิ้นนี้

### 1.3 การนำเสนอและลำดับเนื้อหาในวิทยานิพนธ์

รูปแบบการนำเสนอในแต่ละบทแบ่งออกเป็นสองส่วน คือ เนื้อหาที่เกี่ยวข้องโดยตรงกับหัวข้อที่นำเสนอ และ *ข้อมูลอ้างอิงเพิ่มเติม* ที่นำเสนอเนื้อหาและงานวิจัยที่เกี่ยวข้องกับเนื้อหาทางตรงและทางอ้อม (ถ้ามี) เพื่อให้รูปของการนำเสนอ ข้อมูลส่วนเนื้อหาต่อเนื่อง และสะดวกต่อการอ่าน โดยลำดับของการนำเสนอสามารถแบ่งได้ตามลำดับดังนี้

- ความรู้และหลักการพื้นฐานของกล้องออมนิ
- ความรู้และหลักการพื้นฐานของการระบุตำแหน่งและสร้างแผนที่
- การระบุตำแหน่งและสร้างแผนที่โดยอาศัยข้อมูลจากกล้องออมนิ และการปฏิบัติงานร่วมกันของหุ่นยนต์สองตัวในการระบุตำแหน่งและสร้างแผนที่
- การทดลองเพื่อทดสอบการทำงานของหลักการและวิธีการที่นำเสนอ
- สรุปผลการทดลองและสิ่งที่ได้รับจากงานวิจัยชิ้นนี้



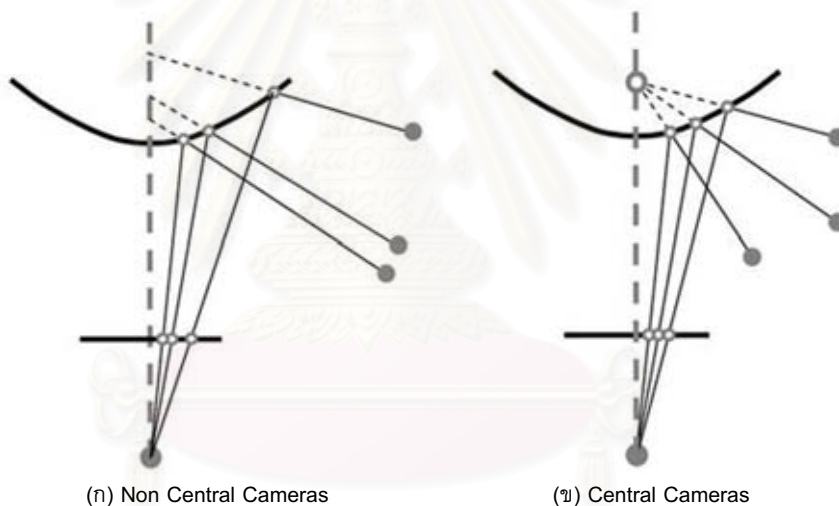
สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## บทที่ 2

### กล้องออมนิ (OMNIDIRECTIONAL CAMERA)

#### 2.1 ความเป็นมา

กล้องออมนิเป็นระบบรับภาพที่มีมุมมอง 360° รอบทิศทางจากการรับภาพในสองลักษณะ คือ การรับภาพจากการสะท้อนกระจก (Catadioptric) และรับภาพผ่านเลนส์ (Dioptric) คุณสมบัติสำคัญที่จำเป็นในการรับภาพของกล้องออมนิ คือ ลำแสงในสามมิติตัดกันที่จุดเดียวก่อนไปตกยังฉากรับภาพ (Single Effective Viewpoint) [45] ซึ่งเป็นคุณสมบัติของ Central Cameras ตามรูปที่ 2.1 ซึ่งมีผลโดยตรงต่อการหาโมเดลการรับภาพของกล้องออมนิที่มีมุมมองในการรับภาพเป็นทรงกลมตามรูปที่ 2.2 เพื่อใช้ในการแปลงรอบทิศทางไปเป็นภาพในมุมมองต่างๆ เช่น Panoramic และ Perspective ตามความต้องการในการใช้งานด้วย Mapping Function ซึ่งอธิบายความสัมพันธ์ระหว่างตำแหน่งสองมิติบนภาพจากกล้องถ่ายภาพ และตำแหน่งในสามมิติของจุดในสภาพแวดล้อมที่ปรากฏบนภาพ



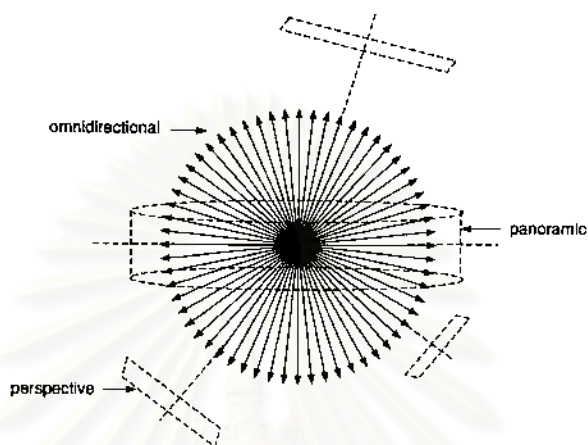
รูปที่ 2.1: ลักษณะการตัดกันของลำแสงในกล้องออมนิที่สำคัญต่อการหา Mapping Function

การรับภาพรอบทิศทางสามารถทำได้ด้วยอุปกรณ์หลายลักษณะ โดยแต่ละวิธีการมีข้อดีข้อเสียแตกต่างกันไปดังนี้

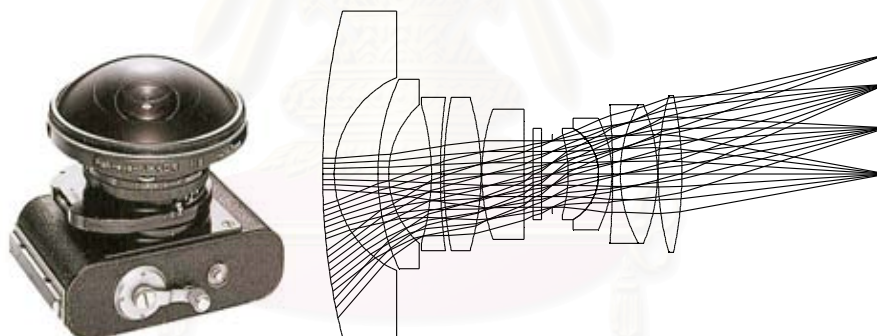
- **การรับภาพผ่านเลนส์มุมกว้าง หรือเลนส์ตาปลา (Fish-Eye Lens)** เป็นเทคโนโลยีที่เกิดขึ้นตั้งแต่ช่วงแรกของการประดิษฐ์กล้องถ่ายภาพ โดยการใช้เลนส์ความยาวโฟกัสสั้นมากในการถ่ายภาพ ที่มีมุมมองในการรับภาพขนาดครึ่งทรงกลมหรือมากกว่า ข้อดีซึ่งเห็นได้อย่างชัดเจนในการใช้งานเลนส์ตาปลา คือ ความมั่นคงของส่วนประกอบ เนื่องจากเป็นเลนส์สำเร็จรูปที่มีการใช้งานในเชิงพาณิชย์อย่างกว้างขวาง แต่มีราคาและข้อจำกัดของชนิดกล้องที่สามารถใช้งานเลนส์ดังกล่าวได้เป็นจุดอ่อนสำคัญ เลนส์ตาปลามีลักษณะและส่วนประกอบตามรูปที่ 2.3 โดยทั่วไปเลนส์ตาปลาคุณภาพสูงไม่มีคุณสมบัติของ Central Cameras แต่สามารถประมาณได้ว่าเป็น [46] ทำให้สามารถหาค่าพารามิเตอร์ได้ด้วยหลักการเดียวกับกล้องออมนิทั่วไป



- การรับภาพโดยการหมุนกล้อง เป็นวิธีการรับภาพรอบทิศทางที่ตรงตัวที่สุด ด้วยการหมุนกล้องถ่ายภาพไปรอบจุดโฟกัส ทำให้สามารถหา Mapping Function ที่แน่นอนได้โดยการหาพารามิเตอร์ของกล้องแบบปรกติ (Pinhole Camera Model) [47] แต่จุดอ่อนที่สำคัญคือ ใช้เวลานานในการถ่ายภาพเนื่องจากต้องหมุนตัวกล้องดังรูปที่ 2.4 ไปรอบจุดที่ทำการถ่ายภาพ ทำให้ไม่สามารถใช้ถ่ายภาพวัตถุที่มีการเคลื่อนไหวหรือเคลื่อนตัวกล้องไประหว่างการทำงานได้



รูปที่ 2.2: การรับภาพรอบทิศทางในสามมิติ (ภาพจากงานวิจัย [2])



(ก) เลนส์ ตาปลา สำหรับ งาน ถ่ายภาพนิ่ง

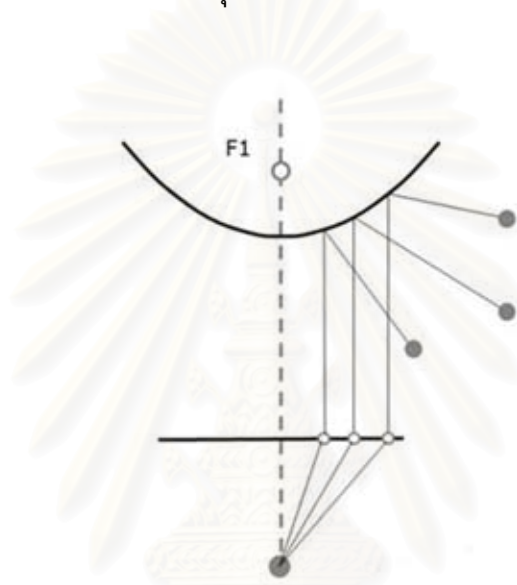
(ข) ส่วนประกอบภายในเลนส์ตาปลา

รูปที่ 2.3: ลักษณะและส่วนประกอบของเลนส์ตาปลา



รูปที่ 2.4: กลไกถ่ายภาพมุมกว้างโดยการหมุนกล้อง

- **การรับภาพผ่านการสะท้อนกระจก** เป็นวิธีการสร้างภาพรอบทิศทางที่มีการนำไปใช้อย่างกว้างขวาง ในงานวิจัยด้านหุ่นยนต์ กล้องลักษณะนี้สามารถออกแบบให้กล้องมีจุดตัดของลำแสงสะท้อนเพียงจุดเดียวได้หลายรูปแบบ เช่น รับภาพสะท้อนของกระจกทรงพาราโบล่าผ่าน Orthographic Lens ไปยังกล้องตามรูปที่ 2.5 และการรับภาพโดยอาศัยกล้องทั่วไปร่วมกับกระจกทรงไฮเพอร์โบล่าตามรูปที่ 2.1 (ข) นอกจากนี้ยังสามารถใช้กระจกสองชุดในการสะท้อนภาพเข้าสู่กล้องเพื่อช่วยลดความผิดพลาดของสีและเพิ่มความคมชัดของภาพ [7] จุดด้อยสำคัญของกล้องลักษณะนี้คือ ส่วนประกอบของอุปกรณ์จำเป็นต้องได้รับการประกอบอย่างถูกต้องและแม่นยำ เพื่อให้ลักษณะการตกกระทบและการเดินทางของลำแสงเป็นไปตามการออกแบบ เพื่อใช้ในการคำนวณ Mapping Function สำหรับการแปลงภาพไปในมุมมองตามการใช้งาน



รูปที่ 2.5: โครงสร้างการรับภาพสะท้อนจากกระจกทรงพาราโบล่าผ่าน Orthographic Lens

## 2.2 คุณสมบัติของภาพจากกล้องอ้อมในงานวิจัย

กล้องอ้อมในงานวิจัยมีลักษณะดังรูป 2.6 ตัวกล้องประกอบด้วยสองส่วนหลัก คือ กระจกสะท้อนภาพรอบทิศทางทรงพาราโบล่าจำนวนสองชิ้น และกล้องรับภาพ ซึ่งประกอบเข้าด้วยกันผ่านข้อต่อสามจุดตามรูปที่ 2.6 (ก) และเป็นจุดสำคัญที่ทำให้เกิดปัญหาการใช้งานที่เกิดจากเรียงตัวของอุปกรณ์ซึ่งกล่าวถึงในหัวข้อถัดไป

คุณสมบัติหนึ่งของกล้องอ้อมที่ต้องคำนึงถึงในการนำมาใช้เป็นเซนเซอร์ในงานวิจัยด้านหุ่นยนต์คือ ลักษณะของข้อมูลจากเซนเซอร์และคุณสมบัติเฉพาะของข้อมูลที่ได้อ ซึ่งสามารถอธิบายได้ดังนี้

**ลักษณะของข้อมูล** เป็นข้อมูลภาพซึ่งมีลักษณะดังรูปที่ 2.7 โดยมีมุมมองรอบทิศทาง  $360^\circ$  และมุมก้มเงยประมาณ  $-5^\circ$  ถึง  $65^\circ$  บริเวณกลางภาพเป็นพื้นที่สีดำที่เกิดจากช่องว่างสำหรับสะท้อนภาพเข้าสู่ตัวกล้องผ่านกระจกทรงพาราโบล่าทั้งสองชิ้น คุณสมบัติของภาพที่ถ่ายโดยกล้องอ้อม เช่น ความละเอียด คุณภาพ และความเร็วในการทำงาน ขึ้นกับกล้องที่เลือกมาใช้เป็นอุปกรณ์รับภาพ

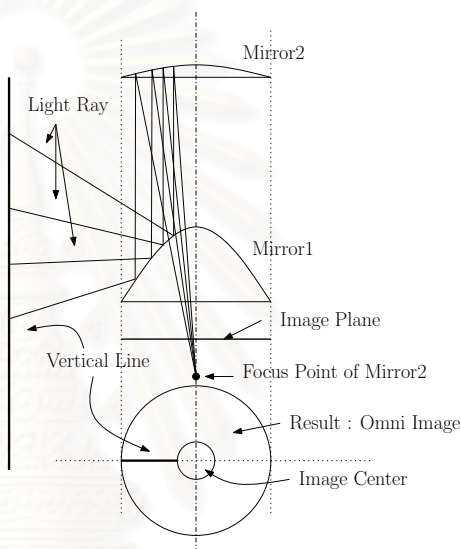
**คุณสมบัติเฉพาะ** ของกล้องอ้อมซึ่งสำคัญต่อการนำมาใช้เป็นเซนเซอร์ของหุ่นยนต์ประกอบด้วย



- อัตราส่วนระหว่างข้อมูลในภาพต่อจำนวนพิกเซล เป็นคุณสมบัติต้องคำนึงถึงในการนำข้อมูลภาพจากกล้องอ้อมมาใช้งาน โดยเฉพาะการระบุตำแหน่งของจุดสังเกตต่างๆ ในภาพ เนื่องจากเป็นภาพสะท้อนจากกระจกโค้ง ทำให้ข้อมูลภาพบริเวณขอบภาพมีความหนาแน่นของข้อมูลต่อจำนวนพิกเซลสูงกว่าบริเวณกลางภาพตามกราฟในรูปที่ 2.8 ซึ่งแสดงให้เห็นอย่างชัดเจนว่าพิกเซลบริเวณใกล้ศูนย์กลางภาพให้ข้อมูลชัดเจนกว่าบริเวณขอบภาพ
- ลักษณะของเส้นตรงที่ปรากฏในภาพ และสำคัญต่อการใช้งานกล้องอ้อมนิ คือ เส้นซึ่งขนานกับแนวแกนของกล้อง และปรากฏเป็นเส้นในแนวรัศมีบนภาพเสมอไม่ขึ้นกับทิศทางที่วัดได้ [7] ดังรูปที่ 2.6 (ข) ทำให้ข้อมูลดังกล่าวมีจุดเด่นในการนำมาใช้งานเป็นข้อมูลจุดสังเกต ตามคุณสมบัติของจุดสังเกตที่ดี ซึ่งต้องแยกจากสิ่งแวดล้อมได้ง่าย และการเปลี่ยนแปลงตำแหน่งมี

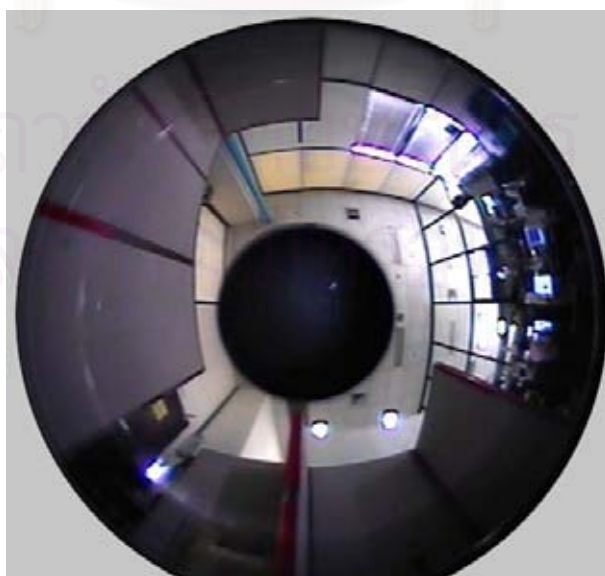


(ก) กล้องอ้อมนิในงานวิจัย



(ข) ส่วนประกอบภายในตัวกล้อง

รูปที่ 2.6: ลักษณะและส่วนประกอบของกล้องอ้อมนิในงานวิจัย



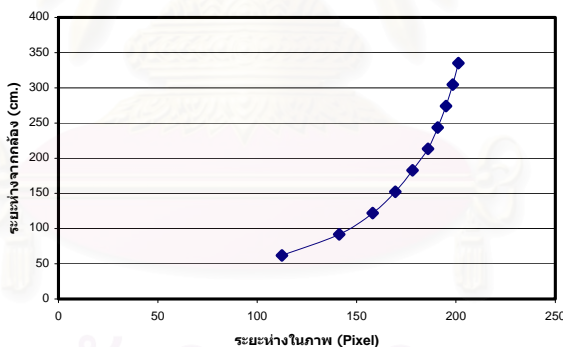
รูปที่ 2.7: ภาพจากกล้องอ้อมนิ

ความต่อเนื่องตามทิศทางของเซนเซอร์ โดยเฉพาะอย่างยิ่งเมื่อทำการพิจารณาสภาพแวดล้อมภายในอาคาร ซึ่งประกอบด้วยโครงสร้างที่ทำให้เกิดเส้นตั้งฉากกับระนาบการเคลื่อนที่ของหุ่นยนต์หรือเส้นแนวตั้งได้เป็นจำนวนมาก เช่น ฉากกั้นห้อง ขาโต๊ะ แนวกำแพง และขอบประตู ซึ่งเหมาะสมในการนำมาใช้เป็นจุดสังเกตสำหรับการสร้างแผนที่ เนื่องจากเป็นข้อมูลสำคัญที่บ่งบอกลักษณะของสิ่งแวดล้อมภายในอาคารได้ชัดเจน เช่น ลักษณะของห้อง ตำแหน่งที่ตั้งของโต๊ะหรือตู้ ทำให้ข้อมูลเส้นแนวตั้งมีการนำไปใช้อย่างกว้างในงานวิจัยของหุ่นยนต์ที่ทำงานภายในอาคาร [48, 12, 33, 13, 14, 37] เช่นเดียวกับงานวิจัยชิ้นนี้ ซึ่งต้องการข้อมูลที่เหมาะสมสำหรับการระบุตำแหน่งและสร้างแผนที่ภายในอาคาร เส้นแนวตั้งจึงเป็นข้อมูลหลักสำหรับใช้ในงานวิจัยชิ้นนี้

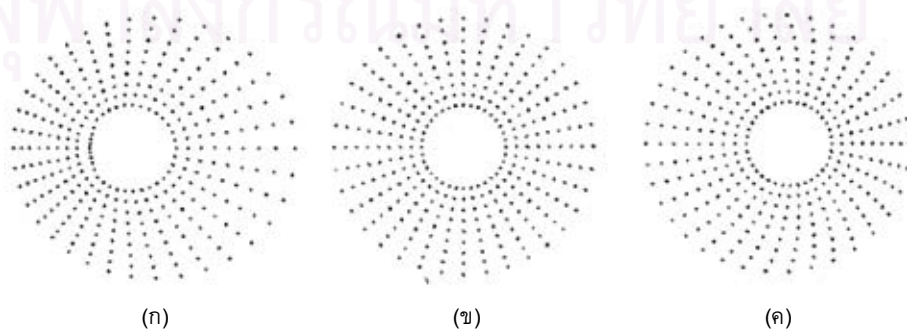
### 2.3 ปัญหาการใช้งานกล้องอ้อม

จากส่วนประกอบจำนวนมากของกล้องอ้อมต้องเรียงตัวอย่างถูกต้องเป็นแนวเดียวกันดังรูปที่ 2.6 (ข) และโครงสร้างการประกอบที่เกิดความคลาดเคลื่อนได้ง่ายทั้งจากผลผลิตและประกอบชิ้นส่วนตามรูป 2.6 (ก) ซึ่งแก้ไขได้ยากเมื่อนำกล้องดังกล่าวมาใช้ งาน ส่งผลให้ลักษณะของภาพจากกล้องอ้อมนี้เป็นไปตามรูป 2.9 และสามารถแก้ไขได้ด้วยการหา Mapping Function ที่เหมาะสมจากพารามิเตอร์ของตัวกล้องและลักษณะของความคลาดเคลื่อนที่เกิดขึ้น

ระยะห่างของวัตถุจากกล้องเทียบกับระยะห่างในภาพ



รูปที่ 2.8: ระยะห่างของวัตถุจากกล้องอ้อมเทียบกับระยะห่างจากจุดศูนย์กลางภาพ



รูปที่ 2.9: ตัวอย่างภาพจากกล้องอ้อมเมื่อชิ้นส่วนเรียงตัวไม่สมบูรณ์

แต่ปัญหาที่ตามมาในการใช้งานกล้องอ้อมนี้ทั่วไป คือ ไม่ทราบโมเดลและค่าพารามิเตอร์ของส่วนประกอบ เช่น ลักษณะของกระจก ระยะโฟกัส และมุมรับภาพในแนวดิ่ง เนื่องจากกล้องอ้อมนี้ที่จำหน่ายในเชิงพาณิชย์ส่วนใหญ่ [49] มีจุดประสงค์เพื่อใช้ในงานเฝ้าระวัง การถ่ายภาพในเชิงศิลปะ หรือตรวจสอบการเคลื่อนที่ จึงไม่มีการบอกพารามิเตอร์ที่สำคัญต่อการหา Mapping Function มาด้วย ทำให้ไม่สามารถใช้ข้อมูลจากกล้องอ้อมนี้ในงานด้านหุ่นยนต์ซึ่งต้องการ Mapping Function ที่มีความถูกต้องและแม่นยำ

## 2.4 การหาค่าพารามิเตอร์และแก้ปัญหาการเรียงตัวของชิ้นส่วนในกล้องอ้อมนี้

งานวิจัยเป็นจำนวนมากนำเสนอวิธีการหา Mapping Function ของกล้องอ้อมนี้โดยอาศัยโมเดลของกล้องที่ทราบแน่นอน ซึ่งไม่สามารถใช้กับปัญหาที่พบในทางปฏิบัติตามที่กล่าวถึงในหัวข้อก่อนหน้า โดยเฉพาะอย่างยิ่งในกรณีไม่ทราบพารามิเตอร์ของส่วนประกอบกล้อง เช่น ระยะโฟกัส รูปแบบของกระจก ทำให้ความถูกต้องในการใช้งานกล้องอ้อมนี้ในช่วงกว่าสิบปีที่ผ่านมา จำกัอยู่กับกล้องที่มีการออกแบบเพื่องานวิจัยซึ่งมีราคาสูงหรือกล้องที่ออกแบบและสร้างขึ้นโดยผู้วิจัยเองเป็นหลัก

อย่างไรก็ตามความสำคัญของพารามิเตอร์และ Mapping Function ขึ้นอยู่กับรูปแบบการนำข้อมูลไปใช้เป็นหลัก เช่น การแปลงภาพรอบทิศทางเป็นภาพในมุมกว้างเพื่อการตรวจจับความเคลื่อนไหวไม่จำเป็นต้องอาศัยค่าพารามิเตอร์ที่แน่นอน แต่ในงานที่ต้องการความแม่นยำสูง เช่น การสร้างพื้นผิวของโครงสร้างสามมิติด้วยข้อมูลภาพจากกล้องอ้อมนี้ในผลการทดลองของงานวิจัย [46] จำเป็นต้องทราบ Mapping Function ที่แน่นอนระหว่างเวกเตอร์ทิศทางในสามมิติกับตำแหน่งในภาพ ด้วยการหาค่าพารามิเตอร์และการแก้ปัญหาการเรียงตัวไม่ตรงกันของส่วนประกอบกล้อง

ด้วยความจำเป็นข้างต้นและปัญหาในการใช้งานกล้องอ้อมนี้ งานวิจัย [46] ได้นำเสนอหลักการที่สามารถใช้ในการประมาณค่าพารามิเตอร์ของกล้องอ้อมนี้ที่มีการตัดกันของลำแสงที่จุดเดียวได้หลายรูปแบบ รวมถึงสามารถใช้ได้กับเลนส์ตาปลาด้วยการหา Mapping Function ระหว่างเวกเตอร์ทิศทางในสามมิติกับตำแหน่งบนภาพจากกล้องอ้อมนี้ดังรูปที่ 2.10 ซึ่งกำหนดให้  $P(x, y, z)$  คือ เวกเตอร์ทิศทางของจุดในสามมิติ และ  $(u, v)$  เป็นตำแหน่งของจุด  $P$  ที่ปรากฏบนภาพ

การพิจารณา Mapping Function ในกรณีที่มีการเรียงตัวของอุปกรณ์เป็นไปอย่างถูกต้องสมบูรณ์สามารถกล่าวได้ว่า ตำแหน่ง  $x$  และ  $y$  ของ  $P$  จะแปรผันตามตำแหน่ง  $u$  และ  $v$  ในภาพ ตามสมการที่ 2.1 ซึ่งมี  $\alpha$  เป็นปัจจัยการปรับมาตรา (Scaling Factor)

$$\begin{bmatrix} x \\ y \end{bmatrix} = \alpha \cdot \begin{bmatrix} u \\ v \end{bmatrix}; \alpha > 0 \quad (2.1)$$

แต่ในการใช้งานจริงเราต้องการ Mapping Function จากพิกัดของภาพไปยังพิกัดในสามมิติ ซึ่งสามารถเขียนได้ตามสมการ 2.2 ที่รวม  $\alpha$  เข้าไปใน  $f'(u, v)$  เนื่องจากสิ่งที่ต้องการจากฟังก์ชันดังกล่าว

คือ ทิศทางในสามมิติ ซึ่งต่างจาก ตำแหน่งในสามมิติทำให้สามารถดรูปในลักษณะดังกล่าวได้ [46]

$$P = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \alpha \cdot u \\ \alpha \cdot v \\ f(u, v) \end{bmatrix} \Rightarrow \begin{bmatrix} u \\ v \\ f'(u, v) \end{bmatrix} \quad (2.2)$$

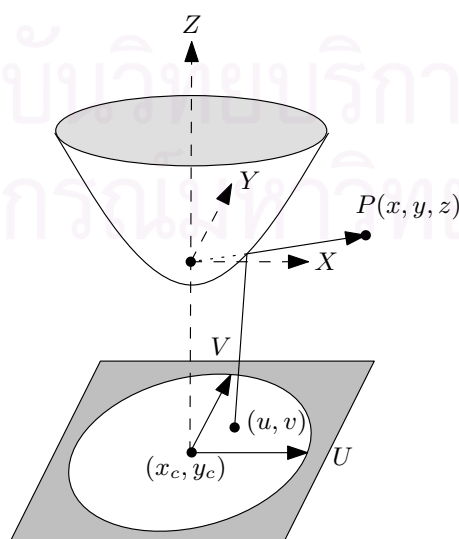
จากเงื่อนไขการเรียงตัวอย่างสมบูรณ์ของอุปกรณ ทำให้การให้เกิด Rotationally Symmetric ส่งผลให้  $f'(u, v)$  ขึ้นอยู่กับระยะห่างจากจุดศูนย์กลางของภาพรอบทิศทาง  $\rho = \sqrt{u^2 + v^2}$  เพียงอย่างเดียว และ  $P$  เป็นเพียงเวกเตอร์ทิศทางในสามมิติทำให้สามารถดรูปของสมการ 2.2 ให้อยู่ในรูปสมการ 2.3

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} u \\ v \\ f(\rho) \end{bmatrix}; \rho = \sqrt{u^2 + v^2} \quad (2.3)$$

เป็นผลให้การหาค่าพารามิเตอร์ของกล้องออปติคัลในงานวิจัย [46] คือ การประมาณฟังก์ชัน  $f(\rho)$  ด้วยสมการพหุนาม 2.4 และจากผลการทดลองในงานวิจัยข้างต้นปรากฏว่า สมการพหุนามกำลังสี่สามารถประมาณค่า  $f(\rho)$  ได้ดีที่สุด

$$f(\rho) = a_0 + a_1\rho + a_2\rho^2 + a_3\rho^3 + a_4\rho^4 + \dots \quad (2.4)$$

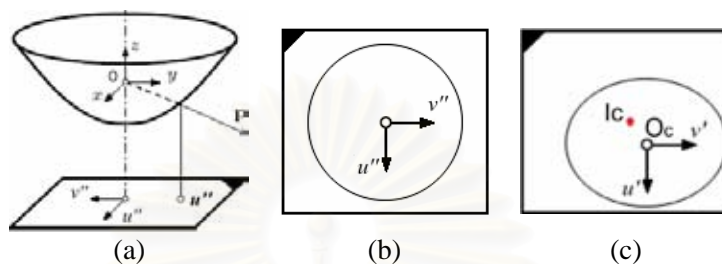
วิธีการข้างต้นยังไม่รวมการแก้ปัญหาการเรียงตัวไม่ตรงกันของอุปกรณ ซึ่งส่งผลให้ภาพไม่มีคุณสมบัติ Rotationally Symmetric จึงต้องประมาณความคลาดเคลื่อนที่เกิดขึ้นโดยอาศัย Affine Transformation ระหว่างจุดบนภาพที่มีการบิดเบี้ยวกับตำแหน่งสมมาตรในอุดมคติตามรูปที่ 2.11 โดยอาศัยสมการที่ 2.5 ซึ่งกำหนดให้  $A \in \mathbb{R}^{2 \times 2}$  และ  $[x_c, y_c]^T$  เป็นเมตริกซ์การแปลงและจุดศูนย์กลางภาพ ซึ่งได้จากข้อมูล



รูปที่ 2.10: ความสัมพันธ์ระหว่างลำแสงในสามมิติกับตำแหน่งในภาพ

ที่ใช้ในการหาค่าพารามิเตอร์ตามขั้นตอนในหัวข้อถัดไป ส่งผลให้การประมาณค่าพารามิเตอร์ด้วยวิธีการนี้สามารถใช้กับกล้องออปติคัลที่ใช้ในงานวิจัยได้โดยตรง

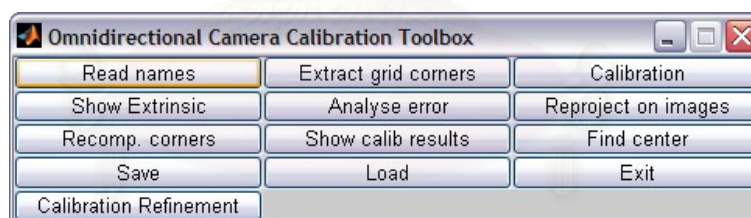
$$\begin{bmatrix} u'' \\ v'' \end{bmatrix} = A \begin{bmatrix} u' \\ v' \end{bmatrix} + \begin{bmatrix} x_c \\ y_c \end{bmatrix} \quad (2.5)$$



รูปที่ 2.11: ภาพจากงานวิจัย [46] แสดงการรับภาพรอบทิศทางในรูป (a) และภาพรอบทิศทางบนฉากรับภาพหรือ เมื่อส่วนประกอบเรียงตัวอย่างสมบูรณ์ในรูป (b) โดยมีภาพ (c) แสดงลักษณะของภาพจากกล้องเมื่อการเรียงตัวของอุปกรณ์ไม่สมบูรณ์ ซึ่งสามารถอธิบายความสัมพันธ์ระหว่าง (b) และ (c) ได้ด้วยสมการที่ 2.5

#### 2.4.1 ขั้นตอนการหาพารามิเตอร์ของกล้อง

เนื้อหาในส่วนนี้เป็นขั้นตอนการหาพารามิเตอร์กล้องออปติคัลด้วยซอฟต์แวร์ใน [46] ซึ่งทำงานบนโปรแกรม Matlab [50] ที่มีลักษณะดังรูปที่ 2.12 และมีขั้นตอนการใช้งานดังนี้

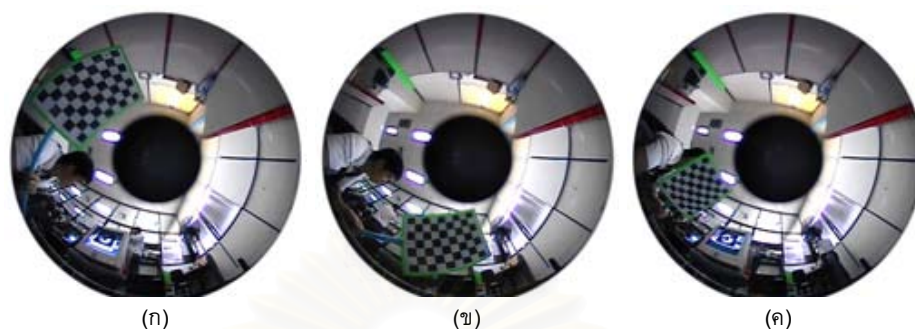


รูปที่ 2.12: โปรแกรม Omnidirectional Camera Calibration Toolbox ในงานวิจัย [46]

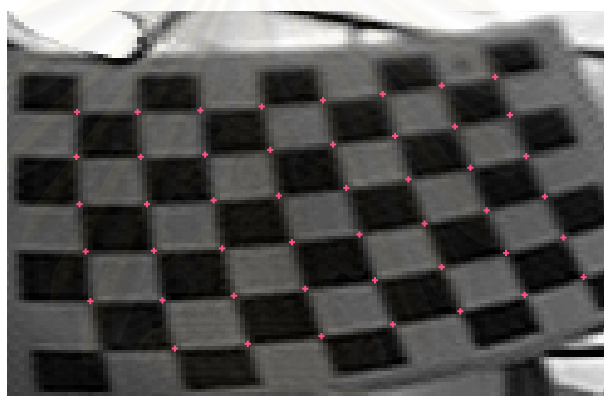
1. ถ่ายภาพสำหรับการหาค่าพารามิเตอร์ โดยใช้ตารางหมากรุกซึ่งทราบขนาดในมุมมองต่างๆ ดังรูปที่ 2.13 จำนวนสามภาพขึ้นไป
2. ระบจุดมุมของแต่ละตาราง เพื่อใช้เป็นจุดอ้างอิงในการหาค่าพารามิเตอร์ตามขั้นตอนในโปรแกรม ทำให้ได้ภาพตารางพร้อมจุดมุมตามรูปที่ 2.14
3. ให้โปรแกรมทำการหาค่าพารามิเตอร์ จากข้อมูลภาพทั้งหมดเพื่อให้ได้พารามิเตอร์ของกล้อง ซึ่งจำเป็นต่อการคำนวณ Mapping Function ที่ประกอบไปด้วย ตำแหน่งศูนย์กลางจริงของภาพ  $[x_c, y_c]^T$  ตามรูปที่ 2.15 (ก) และสัมประสิทธิ์ของ  $f(\rho)$  ในสมการที่ 2.4 รวมถึง  $A$  ในสมการที่ 2.5 ซึ่งสามารถใช้ในการหาตำแหน่งในสามมิติของตารางหมากรุกแต่ละภาพจากขั้นตอนที่ 1 ด้วยการคำนวณค่า Extrinsic Parameters ที่ให้ผลตามรูปที่ 2.15 (ข)



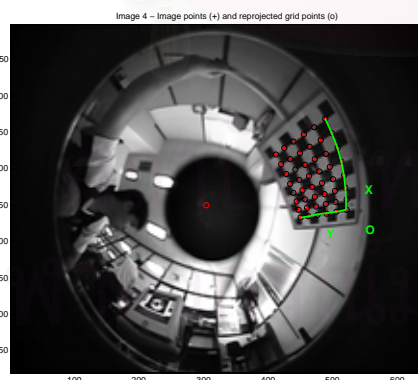
ด้วยขั้นตอนข้างต้นเราสามารถหา Mapping Function ที่มีความแม่นยำเพื่อนำไปใช้แปลงภาพรอบทิศทางไปเป็นภาพในมุมกว้าง เพื่อใช้ในการตรวจจับและระบุตำแหน่งของเส้นแหว่ง ซึ่งเป็นหัวใจหลักของงานวิจัยชิ้นนี้



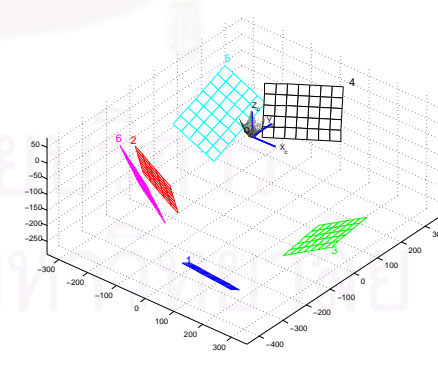
รูปที่ 2.13: ภาพถ่ายดาวทรงกลมที่ทราบขนาดแน่นอนสำหรับการหาค่าพารามิเตอร์กล้อง



รูปที่ 2.14: ตำแหน่งมุมของดาวทรงกลมในภาพ



(ก) จุดศูนย์กลางจริงของภาพจากกล้องอ้อมนิ



(ข) ตำแหน่งในสามมิติของดาวทรงกลมตามค่า Extrinsic Parameters

รูปที่ 2.15: ผลจากการหาค่าพารามิเตอร์สำหรับกล้องอ้อมนิในงานวิจัย

## 2.5 ข้อมูลอ้างอิงเพิ่มเติม

การใช้งานกล้องออบนิในงานวิจัยด้านหุ่นยนต์ได้เริ่มมีการรายงานในผลการทดลองในช่วง 1990 และมีการรวบรวมไว้ในงานวิจัย [7] ในช่วงแรกของการใช้งานกล้องออบนิเป็นการสะท้อนภาพจากกรวยโลหะ [1, 12] เนื่องจากสามารถสร้างและออกแบบได้ง่าย แต่อย่างไรก็ตามงานวิจัยดังกล่าวจำกัดอยู่ในวงแคบ เนื่องจากปัญหาการประมวลผลข้อมูลภาพและราคาของอุปกรณ์ที่จำเป็นการใช้งาน

เมื่อราคาของอุปกรณ์ต่างๆ เช่น คอมพิวเตอร์ และกล้องรับภาพ ต่ำลงเนื่องจากเทคโนโลยีที่ก้าวหน้าขึ้น กล้องออบนิจึงเป็นอุปกรณ์ที่ได้รับความนิยมในการนำมาเป็นเซนเซอร์ของหุ่นยนต์ตามที่ได้กล่าวไปในตอนต้น ทำให้มีงานวิจัยต่างๆ ที่เกี่ยวข้องออกมาเป็นจำนวนมาก และด้วยปัญหาเดียวกับการใช้งานกล้องชนิดอื่นซึ่งต้องการทราบความสัมพันธ์ระหว่างตำแหน่งในสามมิติของสิ่งแวดล้อมและตำแหน่งในข้อมูลภาพ จึงมีงานวิจัย [51, 2, 3, 20] นำเสนอวิธีการหาคุณสมบัติและออกแบบกล้องออบนิที่สามารถหาความสัมพันธ์ดังกล่าวได้ด้วยรูปแบบการหาค่าพารามิเตอร์ในหลายลักษณะ โดยงานวิจัยส่วนใหญ่ [4, 52] ออกแบบให้ใช้ได้กับกล้องออบนิเฉพาะแบบและต้องทราบลักษณะของกล้อง (ระยะโฟกัส รูปทรงกระจก) ซึ่งเป็นปัญหาเชิงปฏิบัติตามที่ได้กล่าวในหัวข้อ 2.3 ที่นำไปสู่การออกแบบวิธีการที่ไม่ต้องการข้อมูลลักษณะในการหาค่าพารามิเตอร์ของกล้อง แต่ใช้การประมาณค่าจากข้อมูลที่ทราบลักษณะหรือคุณสมบัติชัดเจน เช่น ภาพตารางหมากรุก หรือเครื่องหมายพิเศษ ในงานวิจัย [6] ซึ่งจำกัดอยู่กับกล้องออบนิเพียงบางรูปแบบและยากต่อการนำมาใช้งานเนื่องจากความซับซ้อนในการออกแบบ ทำให้การใช้งานกล้องออบนิในงานวิจัยด้านหุ่นยนต์ที่ได้รับการยอมรับ จำกัดอยู่ในกลุ่มนักวิจัยเพียงบางกลุ่มหรือกล้องเพียงบางลักษณะ เนื่องจากไม่สามารถหา Mapping Function ที่แม่นยำสำหรับกล้องที่ใช้งานได้

ปัญหาความยุ่งยากในการใช้งานกล้องออบนิข้างต้น เป็นส่วนหนึ่งที่ทำให้งานวิจัยด้านหุ่นยนต์ซึ่งต้องการความถูกต้องของข้อมูลเซนเซอร์สูงในอดีต เช่น การระบุตำแหน่งพร้อมการสร้างแผนที่ที่ไม่สามารถใช้อข้อมูลจากกล้องออบนิเพียงอย่างเดียวในการทำงานได้ แต่สามารถทำได้ในวิทยานิพนธ์ฉบับนี้ เนื่องจากงานวิจัย [46] และ [10] ได้ออกแบบวิธีการสำหรับหาพารามิเตอร์ของกล้องออบนิที่ง่ายต่อการใช้งาน สำหรับกล้องออบนิทุกชนิดที่มีคุณสมบัติเหมือนหรือใกล้เคียงกับนิยามในงานวิจัย [3] ด้วยออกแบบโปรแกรมสำเร็จรูปเพื่อการใช้งานโดยทั่วไป ที่ทำให้อุปสรรคใหญ่ของการใช้งานกล้องออบนิหมดไป เช่นเดียวกับปัญหาการใช้งานกล้องรับภาพทั่วไปในงานด้านหุ่นยนต์ในอดีต ซึ่งแก้ปัญหานั้นได้ด้วยผลงานในงานวิจัย [47]



## บทที่ 3

### ปัญหาการระบุตำแหน่งและสร้างแผนที่

#### 3.1 โครงสร้างและส่วนประกอบของปัญหา

การระบุตำแหน่งของหุ่นยนต์ไปพร้อมกับการสร้างแผนที่ (Simultaneous Localization and Mapping: SLAM) ที่กล่าวถึงในบทที่ 1 คือ การสร้างแผนที่ไปในระหว่างที่หุ่นยนต์ปฏิบัติงาน แล้วนำข้อมูลที่ได้มาใช้ในการระบุตำแหน่งของตัวหุ่นยนต์ โดยไม่จำเป็นต้องมีความรู้เกี่ยวกับตำแหน่งและลักษณะของสถานที่ปฏิบัติงาน ด้วยการใช้เซนเซอร์บนตัวหุ่นยนต์เป็นเครื่องมือในการรวบรวมข้อมูลสำหรับการปฏิบัติงาน ที่สามารถอธิบายในรูปตัวแปรทางคณิตศาสตร์ได้ดังนี้

- $x_k$  : สถานะของหุ่นยนต์ที่ประกอบไปด้วยข้อมูลของตำแหน่งและทิศทางของหุ่นยนต์ ณ เวลา  $k$
- $u_k$  : คำสั่ง ณ เวลา  $k - 1$  ที่ทำให้หุ่นยนต์เคลื่อนที่จากตำแหน่ง  $x_{k-1}$  ไปยังตำแหน่ง  $x_k$
- $m_i$  : ตำแหน่งของจุดสังเกต (Landmark)  $i$  ที่ไม่มีการเคลื่อนที่หรือย้ายตำแหน่งเมื่อเวลาผ่านไป
- $z_{ik}$  : ข้อมูลการวัดตำแหน่งของจุดสังเกต  $i$  ณ เวลา  $k$  โดยอาศัยเซนเซอร์บนตัวหุ่นยนต์ และแทนการวัดตำแหน่งของจุดสังเกตหลายๆ จุดในคราวเดียวกันด้วย  $z_k$

นอกจากนี้ยังมีข้อมูลทั้งหมดที่หุ่นยนต์รวบรวมได้ขณะปฏิบัติงานจนถึงเวลา  $k$  ในรูปเซตของข้อมูลต่อไปนี้

- $X_{0:k} = \{x_0, x_1, x_2, \dots, x_k\} = \{X_{0:k-1}, x_k\}$  : เส้นทางเคลื่อนที่ของหุ่นยนต์
- $U_{0:k} = \{u_1, u_2, u_3, \dots, u_k\} = \{U_{0:k-1}, u_k\}$  : คำสั่งทั้งหมดที่สั่งงานหุ่นยนต์
- $M = \{m_1, m_2, \dots, m_n\}$  : ตำแหน่งของจุดสังเกตทั้งหมดในสถานที่ปฏิบัติหรือแผนที่
- $Z_{0:k} = \{z_1, z_2, z_3, \dots, z_k\} = \{Z_{0:k-1}, z_k\}$  : เซตของการวัดตำแหน่งจุดสังเกตทั้งหมดที่เกิดขึ้นระหว่างการปฏิบัติงาน

ข้อมูลทั้งหมดที่กล่าวมาเป็นส่วนสำคัญในการออกแบบวิธีการระบุตำแหน่งและสร้างแผนที่สำหรับหุ่นยนต์ที่มีหลากหลายวิธี โดยรูปแบบหนึ่งที่มีการใช้งานอย่างแพร่หลาย มีผลการทำงานที่ชัดเจน และมีการทดลองกับสภาพการทำงานที่หลากหลาย คือ การระบุตำแหน่งและสร้างแผนที่ในเชิงความน่าจะเป็น ที่เป็นเนื้อหาหลักของบทนี้

#### 3.2 การระบุตำแหน่งและสร้างแผนที่ในเชิงความน่าจะเป็น

จากข้อมูลทั้งหมดในการระบุตำแหน่งและสร้างแผนที่ สามารถแบ่งข้อมูลออกได้เป็นสองส่วน คือ ข้อมูลที่ทราบค่าแน่นอน และข้อมูลจากเซนเซอร์ทั้งทางตรงและทางอ้อม ตัวอย่างเช่น การสั่งให้หุ่นยนต์

เคลื่อนที่ไป 1 เมตร สามารถแยกข้อมูลจากการทำงานดังกล่าวได้เป็นสองส่วน คือ คำสั่งให้หุ่นยนต์เคลื่อนที่ 1 เมตรซึ่งเป็นข้อมูลที่ทราบค่าแน่นอน และระยะที่หุ่นยนต์เคลื่อนที่ไปหลังจากได้รับคำสั่งที่วัดได้โดยการแปลงข้อมูลจาก encoder เป็นระยะทาง

ถ้าระบบข้างต้นไม่มีความไม่แน่นอนในการสั่งงาน (Control) ที่เกิดได้จากหลายปัจจัย เช่น ขนาดของล้อไม่เท่ากัน แบตเตอรี่อ่อน หรือสภาพพื้นผิวไม่สม่ำเสมอ ข้อมูลการสั่งงานเพียงอย่างเดียวก็เพียงพอสำหรับการระบุตำแหน่งของหุ่นยนต์ที่เปลี่ยนไปได้อย่างถูกต้อง หรือในกรณีที่มีความไม่แน่นอนในการสั่งงานหุ่นยนต์ แต่การวัดระยะที่หุ่นยนต์เคลื่อนที่ไป (Measurement) สามารถทำได้โดยไม่มีความผิดพลาดเกิดขึ้น ย้อนนำไปสู่การระบุตำแหน่งที่สมบูรณ์เช่นเดียวกัน

แต่ในทางปฏิบัติความไม่แน่นอนในระบบเกิดขึ้นกับทั้งสองส่วนพร้อมกันอย่างหลีกเลี่ยงไม่ได้ [53] ทำให้การบอกค่าสถานะระบบ (System State) เช่น ตำแหน่งของหุ่นยนต์ที่เปลี่ยนไปเมื่อได้รับคำสั่งให้เคลื่อนที่ ไม่สามารถอธิบายได้ในรูปแบบสมการที่แน่นอน จึงมีการคิดหาวิธีการอธิบายสถานะระบบที่ขึ้นอยู่กับคำสั่งที่เข้ามาในระบบและค่าที่วัดได้ในเชิงความน่าจะเป็นในรูปแบบของ Bayes' Filter เพื่อให้สามารถรวมความไม่แน่นอนที่เกิดขึ้น เข้าไปในระบบสมการสำหรับประมาณค่าสถานะระบบ โดยอาศัยวิธีการต่างๆ เช่น Kalman Filter และ Particle Filter

สำหรับปัญหาการระบุตำแหน่งและสร้างแผนที่ของหุ่นยนต์ ที่มีข้อมูลสถานะระบบเป็นตำแหน่งของหุ่นยนต์และแผนที่ โดยมีข้อมูลสั่งงานหุ่นยนต์ และข้อมูลที่เกิดจากวัดตำแหน่งของจุดสังเกต เป็นตัวแปรที่ใช้ในการประมาณค่าสถานะระบบ เราสามารถอธิบายความสัมพันธ์ระหว่างข้อมูลและสถานะระบบได้ในรูปกระจายของความน่าจะเป็นของ (Probability Distribution)

$$P(x_k, M \mid Z_{0:k}, U_{0:k}, x_0) \quad (3.1)$$

ที่แสดงความสัมพันธ์ระหว่างตำแหน่งของหุ่นยนต์และแผนที่ ณ เวลา  $k$  จากข้อมูลการสั่งงานหุ่นยนต์ ( $U_{0:k}$ ) และการวัดตำแหน่ง ( $Z_{0:k}$ ) ทั้งหมดที่ผ่านมา ร่วมกับตำแหน่งเริ่มต้นของหุ่นยนต์ ( $x_0$ ) โดยอาศัยข้อมูลทั้งหมดในอดีต ( $Z_{0:k-1}, U_{0:k-1}$ ) ร่วมกับข้อมูลปัจจุบัน ( $u_k, z_k$ ) ทำให้ต้องการหน่วยความจำและเวลามากในการประมวลผล ที่สามารถแก้ไขได้ด้วยการประยุกต์ใช้กฎของเบย์ (Bayes' Rules) เพื่อแยกสถานะของระบบในอดีต  $P(x_{k-1}, M \mid Z_{0:k-1}, U_{0:k-1})$  ออกจากสถานะของระบบในปัจจุบัน โดยมีคำสั่ง ( $u_k$ ) และการวัดตำแหน่ง ( $z_k$ ) เป็นตัวแปรหลักในการประมาณค่าสถานะปัจจุบันของระบบ ผ่านโมเดลการเคลื่อนที่ของหุ่นยนต์ที่ตั้งอยู่บนสมมุติฐานว่า ตำแหน่งถัดไป  $x_k$  ขึ้นอยู่กับตำแหน่งก่อนหน้า  $x_{k-1}$  และคำสั่งล่าสุด  $u_k$  เท่านั้น

$$P(x_k \mid x_{k-1}, u_k) \quad (3.2)$$

และโมเดลการวัดตำแหน่ง ที่มีสมมุติฐานว่าค่าที่วัดได้ขึ้นอยู่กับตำแหน่งปัจจุบันของหุ่นยนต์และแผนที่ ณ เวลาปัจจุบันเท่านั้น

$$P(z_k \mid x_k, M) \quad (3.3)$$

ด้วยเงื่อนไขข้างต้น สามารถอธิบาย ปัญหา การ ระบุ ตำแหน่ง และ สร้าง แผนที่ ใน รูป ขั้นตอน การ

ประมาณค่า (Prediction - time update) และปรับแก้ค่าที่เกิดจากการประมาณค่า (Correction - measurement update) โดยอาศัยข้อมูลจากการวัดตำแหน่ง แบบเวียนบังเกิด (Recursive) ได้ตามสมการที่ 3.4 และ 3.5

$$P(x_k, M | Z_{0:k-1}, U_{0:k}, x_0) = \int P(x_k | x_{k-1}, u_k) P(x_{k-1}, M | Z_{0:k-1}, U_{0:k-1}, x_0) dx_{k-1} \quad (3.4)$$

$$P(x_k, M | Z_{0:k}, U_{0:k}, x_0) = \frac{P(z_k | x_k, M) P(x_{k-1}, M | Z_{0:k-1}, U_{0:k}, x_0)}{P(z_k | Z_{0:k-1}, U_{0:k})} \quad (3.5)$$

ในทางปฏิบัติ การประมาณค่าฟังก์ชันการกระจายของความน่าจะเป็น (Probabilistic Distribution Function: PDF) ของสมการข้างต้นสามารถทำได้หลายรูปแบบที่มีข้อดีข้อเสียแตกต่างกันออกไป งานวิจัยชิ้นนี้ใช้วิธีการ FastSLAM [44] ใช้หลักการ Particle Filter ร่วมกับ Extended Kalman Filter (EKF) ในการประมาณค่า PDF ของ การระบุตำแหน่งและสร้างแผนที่จากข้อมูลเส้นแนวตั้ง เนื้อหาในส่วนถัดไปเป็นหลักการพื้นฐานของการประมาณค่า PDF ที่ประกอบไปด้วย Kalman Filter และ Particle Filter เพื่อนำไปสู่วิธีการพื้นฐานของ FastSLAM

### 3.3 Gaussian Filter

Gaussian Filter เป็นรูปแบบหนึ่งในการประมาณค่า PDF ของ Bayes Filter ในรูปฟังก์ชันการกระจายของความน่าจะเป็นแบบปรกติ ที่สถานะระบบอยู่ในรูปของค่าเฉลี่ย (Mean)  $\mu$  และความแปรปรวนร่วม (Covariance)  $\Sigma$

$$P(x) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right\} \quad (3.6)$$

เนื้อหาในส่วนนี้ดัดแปลงจากบทความเรื่อง An Introduction to the Kalman Filter โดย Gerg Welch และ Gary Bishop [54]

#### 3.3.1 Kalman Filter

Kalman Filter เป็นรูปแบบหนึ่งของ Gaussian Filter ที่ใช้ในการประมาณค่าสถานะระบบ ที่มีการเปลี่ยนแปลงเชิงเส้น และทราบสถานะเริ่มต้นของระบบ ที่ดีที่สุดวิธีการหนึ่ง [53] โดยมีจุดมุ่งหมายในการลดค่าความแปรปรวนร่วมของความผิดพลาดในการประมาณค่า (Error Covariance) ให้ได้มากที่สุด การประมาณค่าสถานะระบบของ Kalman Filter เป็นตามสมการ 3.7 ที่มีการวัดค่าสถานะระบบตามสมการ 3.8

$$x_k = Ax_{k-1} + Bu_k + w_{k-1} \quad (3.7)$$

$$z_k = Hx_k + v_k \quad (3.8)$$

โดยมี  $w_k \sim N(0, Q)$  และ  $v_k \sim N(0, R)$  เป็นการรบกวนที่มีการกระจายแบบปรกติที่มีค่าความแปรปรวนร่วมเป็น  $Q$  และ  $R$  และมีเมตริกซ์  $A$  แสดงความสัมพันธ์ระหว่างสถานะในอดีต  $x_{k-1}$  กับสถานะ

ปัจจุบัน  $x_k$  ของระบบ ร่วมกับเมตริกซ์  $B$  ที่แสดงความสัมพันธ์ระหว่าง  $x_k$  และ  $u_k$  ซึ่งเป็นการควบคุมที่ให้กับระบบ โดยมี  $H$  แสดงความสัมพันธ์ระหว่าง  $z_{k-1}$  กับ  $z_k$  จากสถานะปัจจุบันของระบบ  $x_k$

ขั้นตอนการทำงานของ Kalman Filter มีสองขั้นตอน คือ การประมาณค่าสถานะระบบ (Time update) ตามสมการที่ 3.9 และ 3.10 ด้วยสถานะในอดีต  $x_{k-1}$  ร่วมกับการควบคุมที่ให้กับระบบ  $u_k$

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \quad (3.9)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (3.10)$$

หลังจากนั้นเป็นขั้นตอนปรับแก้ค่าที่ประมาณไปก่อนหน้านี้ (Measurement update) ด้วยสมการที่ 3.12 และ 3.13 โดยอาศัยข้อมูลจากการวัดค่าสถานะระบบ  $z_k$

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (3.11)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (3.12)$$

$$P_k = (I - K_k H)P_k^- \quad (3.13)$$

ข้อมูลจากสองขั้นตอนข้างต้น คือ ความเชื่อมั่นในสถานะระบบที่มีการกระจายแบบปรกติ มีค่าเฉลี่ยเป็น  $\hat{x}$  และค่าความแปรปรวนร่วมเป็น  $P$  ที่ตั้งอยู่บนสมมุติฐานว่าทราบค่าสถานะเริ่มต้นของระบบ  $x_0$  และ  $P_0$

ค่าพารามิเตอร์ที่สำคัญที่สุดของ Kalman Filter คือ Kalman Gain ( $K$ ) เป็นพารามิเตอร์ซึ่งกำหนดการเปลี่ยนแปลงในการประมาณค่าสถานะระบบ ว่าเกิดจากการควบคุมหรือเกิดจากการวัดตำแหน่งมากกว่ากัน หรืออีกนัยหนึ่งถ้าความแปรปรวนร่วมของการรบกวนที่เกิดกับการควบคุมสูงกว่าการวัดค่าสถานะระบบมีแนวโน้มที่จะเปลี่ยนแปลงตามค่าที่วัดได้มากกว่า และให้ผลเช่นเดียวกัน ในทางตรงกันข้ามถ้าความแปรปรวนร่วมของการรบกวนที่เกิดกับการวัดค่าสูงกว่า

### 3.3.2 Extended Kalman Filter: EKF

ในงานด้านหุ่นยนต์การใช้ Kalman Filter เพื่อประมาณค่าของสถานะระบบสามารถทำได้จำกัดเนื่องจาก Kalman Filter ถูกออกแบบให้ใช้งานกับระบบที่มีการเปลี่ยนแปลงเชิงเส้นเท่านั้น แต่งานส่วนใหญ่ในหุ่นยนต์ โดยเฉพาะการระบุตำแหน่งและสร้างแผนที่จะต้องอาศัยการวัดตำแหน่งและทิศทางเป็นหลัก การเปลี่ยนแปลงสถานะระบบจึงไม่อยู่ในลักษณะเชิงเส้น

ตัวอย่างเช่น การวัดระยะและมุมของจุดสังเกตในแผนที่โดยอาศัยเลเซอร์วัดตำแหน่งจะได้ข้อมูลสองค่า คือ ระยะทาง ( $r$ ) และมุม ( $\theta$ ) ที่สามารถแปลงเป็นตำแหน่งเทียบกับตำแหน่งของหุ่นยนต์หรือตัวเซนเซอร์ได้โดยอาศัยสมการ

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} r \cos(\theta) \\ r \sin(\theta) \end{bmatrix} \quad (3.14)$$

ทำให้ไม่สามารถนำ Kalman Filter มาใช้งานได้โดยตรง แต่เนื่องจากคุณสมบัติและความสามารถในการทำงานของของ Kalman Filter จึงมีการเพิ่มเติมการประมาณค่าเชิงเส้น (Linearize) ของค่าที่เกิดจากการควบคุมและค่าที่วัดได้ ณ เวลาที่ทำการประมาณค่าของระบบ โดยอาศัยการแปลง Jacobian เมตริกซ์อนุพันธ์ย่อย (Partial Derivative) ของสมการประมาณค่าสถานะระบบ 3.15 และสมการการวัดค่าสถานะระบบ 3.16 ที่เป็นระบบสมการไม่เชิงเส้น

$$x_k = f(x_{k-1}, u_k, w_{k-1}) \quad (3.15)$$

$$z_k = h(x_k, v_{k-1}) \quad (3.16)$$

จากสมการ 3.15 และ 3.16 เห็นได้ว่าปัญหาการประมาณค่าเชิงเส้นของสมการดังกล่าว คือ ค่าของ  $w_k$  และ  $v_k$  ณ ช่วงเวลาที่ทำการประมาณค่าที่ไม่สามารถบอกค่าที่แน่นอนได้ ทำให้ต้องประมาณค่าเชิงเส้นโดยไม่คำนึงถึงการรบกวนที่เกิดขึ้นในรูป

$$\tilde{x}_k = f(\hat{x}_{k-1}, u_k, 0) \quad (3.17)$$

$$\tilde{z}_k = h(\tilde{x}_k, 0) \quad (3.18)$$

ทำให้ EKF เป็นวิธีการประมาณค่าสถานะระบบแบบเฉพาะกาล (ad hoc) ที่ PDF ของระบบสมการไม่อยู่ในรูปการกระจายแบบปกติ เนื่องจากการแปลงแบบไม่เป็นเชิงเส้น (Nonlinear Transformation) แต่สามารถประมาณได้ด้วยการประมาณค่าเชิงเส้นตามสมการ 3.17 และ 3.18 ที่กำหนดให้ตัวแปรแต่ละตัวมีความหมายตามรายการด้านล่าง

$$x_k \approx \tilde{x}_k + A(x_{k-1} - \hat{x}_{k-1}) + Ww_{k-1} \quad (3.19)$$

$$z_k \approx \tilde{z}_k + H(x_k - \tilde{x}_k) + Vv_k \quad (3.20)$$

- $x_k$  และ  $z_k$  คือ สถานะระบบจริง และค่าที่วัดได้จริง
- $\tilde{x}_k$  และ  $\tilde{z}_k$  คือ สถานะระบบจริงและค่าที่วัดได้จริง ที่เกิดจากการประมาณโดยสมการ 3.17 และ 3.18
- $\hat{x}_k$  ค่าประมาณความเชื่อมั่นในสถานะระบบ
- $w_k \sim N(0, Q)$  และ  $v_k \sim N(0, R)$  เป็นการรบกวนที่เกิดกับระบบในรูปการกระจายแบบปกติ
- $A$  คือ Jacobian เมตริกซ์อนุพันธ์ย่อยของฟังก์ชัน  $f$  เทียบกับการเปลี่ยนแปลงของ  $x$

$$A_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_{k-1}, u_k, 0)$$

- $W$  คือ Jacobian เมตริกซ์อนุพันธ์ย่อยของฟังก์ชัน  $f$  เทียบกับการเปลี่ยนแปลงของ  $w$

$$W_{[i,j]} = \frac{\partial f_{[i]}}{\partial w_{[j]}}(\hat{x}_{k-1}, u_k, 0)$$



- H คือ Jacobian เมตริกซ์อนุพันธ์ย่อยของฟังก์ชัน  $h$  เทียบกับการเปลี่ยนแปลงของ  $x$

$$H_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\tilde{x}_k, 0)$$

- V คือ Jacobian เมตริกซ์อนุพันธ์ย่อยของฟังก์ชัน  $h$  เทียบกับการเปลี่ยนแปลงของ  $v$

$$V_{[i,j]} = \frac{\partial f_{[i]}}{\partial v_{[j]}}(\tilde{x}_k, 0)$$

EKF มีขั้นตอนการทำงานเช่นเดียวกับ Kalman Filter ที่แบ่งออกเป็นสองขั้นตอน คือ ขั้นตอนการประมาณค่าตามสมการที่ 3.21 และ 3.22

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0) \quad (3.21)$$

$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T \quad (3.22)$$

และขั้นตอนการปรับแก้ค่าจากการประมาณโดยอาศัยข้อมูลจากการวัดสถานะระบบตามสมการ 3.24 และ 3.25

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1} \quad (3.23)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - h(\hat{x}_k^-, 0)) \quad (3.24)$$

$$P_k = (I - K_k H_k) P_k^- \quad (3.25)$$

โดยข้อมูลจากสองขั้นตอนของ EKF คือ ค่าเฉลี่ย  $\hat{x}$  และความแปรปรวนร่วมของสถานะระบบ  $P_k$  ที่กำหนดให้ค่าเริ่มต้นของสถานะระบบเป็น  $x_0$  และ  $P_0$  เช่นเดียวกับ Kalman Filter

### 3.4 Particle Filter

วิธีการหนึ่งที่แตกต่างจาก Gaussain Filter ในการประมาณ PDF คือ Particle Filter ที่แสดงค่าความเชื่อมั่นของสถานะระบบในรูปของ เซตตัวอย่างสถานะระบบ (Particles Set) ที่ทราบจำนวน โดยแต่ละตัวอย่างประกอบไปด้วยสถานะระบบและค่าความสำคัญของตัวอย่างนั้นๆ  $[x, \omega]$

การคำนวณค่าสถานะระบบจากกลุ่มตัวอย่างทำได้โดยการตรวจสอบการกระจายตัวและค่าความสำคัญของแต่ละตัวอย่าง ทำให้ Particle Filter สามารถแสดงให้เห็นรูปแบบที่แท้จริงของ PDF ได้ตามสภาพความเป็นจริง โดยไม่ต้องมีการสร้างโมเดลหรือสมการระบบ ซึ่งเป็นข้อดีที่เห็นได้อย่างชัดเจนเมื่อเทียบกับ Kalman Filter โดยเฉพาะอย่างยิ่งในงานด้านหุ่นยนต์ที่การหาโมเดลของระบบทำได้ยาก หรือไม่สามารทำได้ เนื่องจากความไม่แน่นอนในการวัดและทดสอบสมมุติฐานของพฤติกรรมระบบ

ในทางกลับกันจุดด้อยที่สำคัญของ Particle Filter คือ ต้องการหน่วยความจำมากเมื่อเปรียบเทียบ

กับ Kalman Filter แต่ด้วยประสิทธิภาพของคอมพิวเตอร์และราคาของหน่วยความจำในปัจจุบัน ทำให้ Particle Filter ได้รับการพัฒนาและประยุกต์ใช้งานอย่างกว้างขวางในงานวิจัยด้านหุ่นยนต์ [55, 27, 43]

### 3.4.1 ส่วนประกอบขั้นตอนการทำงานของ Particle Filter

เช่นเดียวกับ Gaussain Filter ในการประมาณค่า PDF ของ Bayes Filter ในสมการที่ 3.4 และ 3.5 ส่วนประกอบหลักของ Particle Filter คือ เซตตัวอย่างสถานะระบบ ที่กระจายตัวอยู่ในปริภูมิสถานะ (State Space) และค่าความสำคัญของตัวอย่างนั้น (Importance Factor, Weight) เช่น เซตตัวอย่างของการระบุตำแหน่งหุ่นยนต์ คือ  $S = \langle \langle x_1, \omega_1 \rangle, \langle x_2, \omega_2 \rangle, \dots, \langle x_N, \omega_N \rangle \rangle$  ที่ประกอบด้วย  $x$  ที่แทนตำแหน่งของหุ่นยนต์และ  $\omega$  ที่แทนความสำคัญของตำแหน่งนั้นจำนวน  $N$  ตัวอย่าง

ขั้นตอนการทำงานพื้นฐานของ Particle Filter เป็นไปตามระเบียบวิธีการที่ 1 ซึ่งประกอบด้วยสามขั้นตอนหลัก คือ

1. ประมาณค่าสถานะระบบของแต่ละตัวอย่าง  $x_k^{[i]}$  ในบรรทัดที่ 3 ด้วยสถานะระบบก่อนหน้า  $x_{k-1}^{[i]}$  โดยอาศัยข้อมูลการควบคุมล่าสุด  $u_k$  ในการประมาณค่า PDF ของ  $P(x_k | u_k, x_{k-1})$
2. คำนวณค่าความสำคัญของแต่ละตัวอย่าง ในบรรทัดที่ 4 โดยอาศัยค่าสถานะระบบจากการวัด  $z_k$  ว่าสถานะระบบในแต่ละตัวอย่างมีลักษณะใกล้เคียงกับความเป็นจริงที่วัดได้โดย  $z_k$  มากเพียงใด
3. Resampling เซตตัวอย่าง ในบรรทัดที่ 7 ถึง 14 ถือได้ว่าเป็นขั้นตอนที่สำคัญที่สุดในกลไกการทำงานของ Particle Filter ในการเปลี่ยนรูปแบบความเชื่อมั่นในสถานะระบบที่เกิดจากการประมาณในขั้นตอนที่ 1 ไปเป็นความเชื่อมั่นในสถานะระบบที่มีข้อมูลจากการวัดค่าเข้ามามีส่วนร่วมในขั้นตอนที่ 2 ผ่านการเลือกตัวอย่างตามค่าความสำคัญของตัวอย่างนั้นๆ เพื่อให้ได้เซตของกลุ่มตัวอย่างที่มีขนาดเดิมแต่มีลักษณะการกระจายตัวในปริภูมิสถานะ ตามการกระจายตัวของค่าความสำคัญ การใช้งาน Particle Filter โดยทั่วไปมีการกำหนดค่าขั้นต่ำของจำนวนตัวอย่าง (Effective Sample Size:ESS) ก่อนทำการ Resample เพื่อให้แต่ละตัวอย่างมีโอกาสในการรวบรวมข้อมูลจากการวัดค่ามาปรับค่าความสำคัญก่อนที่จะถูกกำจัดโดยกระบวนการ Resample ซึ่งช่วยเพิ่มโอกาสให้ Particle Filter สามารถทำงานได้ตามจุดประสงค์ ในระบบที่มีความไม่แน่นอนของการสังเกตและวัดข้อมูลสูง

### 3.5 FastSLAM

จากรูปแบบปัญหาการระบุตำแหน่งและสร้างแผนที่ในหัวข้อ 3.2 และหลักการพื้นฐานในการประมาณค่า PDF ตามหัวข้อที่ 3.3 และ 3.4 กลุ่มนักวิจัยซึ่งนำทีมโดย S. Thrun ได้นำเสนอวิธีการใช้งาน Particle Filter และ Extended Kalman Filter ในการแก้ปัญหาการระบุตำแหน่งและสร้างแผนที่ด้วยระเบียบวิธีการ FastSLAM [44] ซึ่งเป็นการแก้ปัญหาการระบุตำแหน่งและสร้างแผนที่ ด้วยการแบ่งปัญหาออกเป็นสองส่วนที่ประกอบด้วย การระบุตำแหน่งของหุ่นยนต์โดยใช้ Particle Filter และการวัดตำแหน่งของจุดสังเกตทั้งหมดในแผนที่ของแต่ละตัวอย่างใน Particle Filter ด้วย EKF

จากหัวข้อที่ 3.2 ปัญหาการระบุตำแหน่งและสร้างแผนที่ที่สามารถเขียนได้ในเชิงความน่าจะเป็นตาม



ในรูป 3.26 เมื่อรู้ความสัมพันธ์ระหว่างค่าที่วัดได้และจุดสังเกต (Correspondence) ที่แทนด้วย  $c$

$$P(x_{0:k}, M | Z_{0:k}, U_{0:k}, c_{0:k}) \quad (3.26)$$

การประมาณตำแหน่งของแต่ละจุดสังเกต  $m$  เป็นอิสระต่อกัน เมื่อทราบเส้นทางการเคลื่อนที่  $x_{0:k}$  และความสัมพันธ์ของการวัดตำแหน่งกับจุดสังเกต  $c_{0:k}$  ทำให้สามารถเขียน 3.26 ในรูปการกระจาย (Factored Form) ได้ตามสมการที่ 3.27 โดยกำหนดให้จำนวนจุดสังเกตทั้งหมดเป็น  $I$  ทำให้วิธีการ FastSLAM แบ่งปัญหาการประมาณค่าออกเป็น  $I + 1$  ปัญหา โดยแบ่งเป็นการประมาณเส้นทางการเคลื่อนที่ และการประมาณตำแหน่งของจุดสังเกตจำนวน  $I$  จุด

$$P(x_{0:k}, M | Z_{0:k}, U_{0:k}, c_{0:k}) = P(x_{0:k} | Z_{0:k}, U_{0:k}, c_{0:k}) \prod_i P(m_i | x_{0:k}, Z_{0:k}, U_{0:k}, c_{0:k}) \quad (3.27)$$

### 3.5.1 ส่วนประกอบและขั้นตอนการทำงานของ FastSLAM

ส่วนประกอบหลักของ FastSLAM คือ เซตตัวอย่าง (Particle Set) ที่ใช้ในการประมาณค่าสถานะของระบบ ซึ่งได้แก่ตำแหน่งของหุ่นยนต์และแผนที่ ซึ่งมีโครงสร้างตามสมการที่ 3.28 โดยกำหนดให้จำนวนตัวอย่างมี  $I$  ตัวอย่าง และมีจุดสังเกตทั้งหมด  $N$  จุด โดย  $[i]$  เป็นลำดับของตัวอย่าง และ  $\mu_{n,k}^{[i]}, \Sigma_{n,k}^{[i]}$  แทนค่าเฉลี่ยและความแปรปรวนร่วมของตำแหน่งจุดสังเกตที่  $n$

$$S_k^{[i]} = \langle x_k^{[i]}, N_k^{[i]}, \omega_k^{[i]}, \mu_{1,k}^{[i]}, \Sigma_{1,k}^{[i]}, \mu_{2,k}^{[i]}, \Sigma_{2,k}^{[i]}, \dots, \mu_{N,k}^{[i]}, \Sigma_{N,k}^{[i]} \rangle \quad (3.28)$$

การประมาณค่าตำแหน่งและแผนที่ ที่ทำได้โดยอาศัยเซตตัวอย่างในอดีต  $S_{k-1}$  ร่วมการควบคุมหุ่นยนต์  $u_k$  และการวัด  $z_k$  ล่าสุด ตามลำดับดังนี้

1. ประมาณตำแหน่งใหม่ของหุ่นยนต์สำหรับแต่ละตัวอย่างโดยใช้การควบคุมหุ่นยนต์ล่าสุด  $u_k$  ใน

---

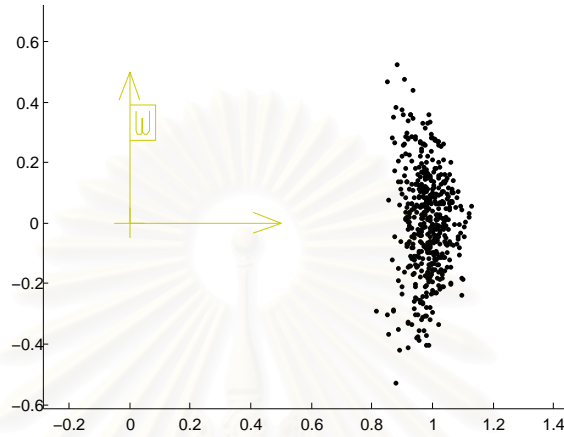
#### Algorithm 1 ParticleFilter( $S_{k-1}, u_k, z_k$ )

---

- 1:  $S_{tmp} = S_k = \emptyset$
  - 2: **for**  $i = 1$  to  $N$  **do**
  - 3:   sample  $x_k^{[i]} \sim P(x_k | u_k, x_{k-1}^{[i]})$  {predict system state}
  - 4:    $\omega_k^{[i]} = P(z_k | x_k^{[i]})$  {update by measurement}
  - 5:    $S_{tmp} = S_{tmp} + \langle x_k^{[i]}, \omega_k^{[i]} \rangle$  {add new particle to temporary set}
  - 6: **end for**
  - 7: {check effective sample size}
  - 8: **if**  $ESS(S_{tmp}) < N_{ESS}$  **then**
  - 9:   {resample if effective sample size < threshold}
  - 10:   draw  $i$  with probability  $\propto \omega_k^{[i]}$
  - 11:   add  $x_k^{[i]}$  to  $S_k$
  - 12: **else**
  - 13:    $S_k = S_{tmp}$ ; {create new particle set from temporary set}
  - 14: **end if**
-

การประมาณตำแหน่งใหม่ของหุ่นยนต์  $x_k^{[i]}$  จากตำแหน่งเก่า  $x_{k-1}^{[i]}$  ในแต่ละตัวอย่างตามสมการ 3.29 โดยอาศัยโมเดลการเคลื่อนที่ของหุ่นยนต์ โดยผลที่ได้มีลักษณะดังรูปที่ 3.1 เมื่อสั่งให้หุ่นยนต์เคลื่อนที่เป็นเส้นตรงโดยอาศัยโมเดลการเคลื่อนที่จากหัวข้อ 4.3.1

$$x_k^{[i]} \sim P(x_k | x_{k-1}^{[i]}, u_k) \quad (3.29)$$



รูปที่ 3.1: ตัวอย่างตำแหน่งของหุ่นยนต์ที่เกิดจากการประมาณค่า เมื่อสั่งหุ่นยนต์เคลื่อนที่จากจุดเริ่มต้นไป 1 เมตร โดยมี  $\sigma_{tran} = 5\text{cm/m}$  และ  $\sigma_{drift} = 10^\circ/\text{m}$

2. **ปรับแก้ข้อมูล การวัด ตำแหน่ง ของจุดสังเกต** แต่ละจุดที่ประกอบด้วย  $\mu_{n,k-1}^{[i]}$  และ  $\Sigma_{n,k-1}^{[i]}$  จากตำแหน่งของหุ่นยนต์ที่เกิดจากการประมาณค่า  $x_k^{[i]}$  โดยอาศัยค่าจากเซนเซอร์  $z_k$  ที่ตั้งอยู่บนสมมุติฐานว่า การวัดค่าหนึ่งครั้งสามารถให้ข้อมูลที่เพียงพอสำหรับการบอกตำแหน่งของจุดสังเกต และการวัดตำแหน่งจะเกิดกับจุดสังเกตเพียงหนึ่งจุดต่อรอบการทำงาน โดยมี  $c_k$  เป็นตัวบอกความสัมพันธ์ของการวัดค่า  $z_k$  กับจุดสังเกต  $m_n$  ทำให้สามารถแบ่งผลการวัดตำแหน่งได้เป็นสามลักษณะ ดังนี้

- **จุดสังเกตในตัวอย่างที่ไม่ถูกวัดค่า** เนื่องจาก  $c_k \neq n$  หรือในอีกความหมายหนึ่ง คือ จุดสังเกต  $n$ อยู่นอกระยะเวลาการทำงานของเซนเซอร์ ณ เวลา  $k$  ทำให้ความเชื่อมั่นในตำแหน่งของจุดสังเกตนั้นไม่มีการเปลี่ยนแปลง

$$\langle \mu_{n,k}^{[i]}, \Sigma_{n,k}^{[i]} \rangle = \langle \mu_{n,k-1}^{[i]}, \Sigma_{n,k-1}^{[i]} \rangle \quad (3.30)$$

- **จุดสังเกตในตัวอย่างที่มีการวัดตำแหน่งมาแล้ว** หรือ  $c_k = n$  ทำให้เกิดการปรับแก้ค่าการวัดตำแหน่งของจุดสังเกตนั้นด้วยวิธีการ EKF ที่ประมาณค่าการวัดตำแหน่งของจุดสังเกตโดยอาศัยตำแหน่งของหุ่นยนต์ในแต่ละตัวอย่างจากสมการ 3.31 แล้วปรับแก้ด้วยค่าที่วัดได้จากเซนเซอร์ ทำให้ได้ค่าเฉลี่ยของตำแหน่งและความแปรปรวนร่วมตามสมการ 3.34 และ 3.35 โดยมี  $K$  เป็นเกนของความเชื่อมั่นระหว่างการวัดตำแหน่งจุดสังเกตและการควบคุมหุ่นยนต์

$$\hat{z}_k^{[m]} = h(\mu_{n,k-1}^{[i]}, x_k^{[i]}) \quad (3.31)$$

$$H_k^{[i]} = h'(x_k^{[i]}, \mu_{n,k-1}^{[i]}) \quad (3.32)$$

$$K_k^{[i]} = \Sigma_{n,k-1}^{[i]} H_k^{[i]} (H_k^{[i]T} \Sigma_{n,k-1}^{[i]} H_k^{[i]} + R_k)^{-1} \quad (3.33)$$

$$\mu_{n,k}^{[i]} = \mu_{n,k-1}^{[i]} + K_k^{[i]} (z_k - \hat{z}_k^{[m]})^T \quad (3.34)$$

$$\Sigma_{n,k}^{[i]} = (I - K_k^{[i]} H_k^{[i]T}) \Sigma_{n,k-1}^{[i]} \quad (3.35)$$

- จุดสังเกตใหม่ที่เพิ่งตรวจพบ หรือ  $c_k = N + 1$  ทำให้เกิดการเพิ่มจุดสังเกตใหม่เข้าไปในตัวอย่าง โดยกำหนดให้ค่าตั้งต้นสำหรับกระบวนการระบุตำแหน่งด้วยวิธี EKF เป็นไปตามค่าที่วัดได้จากเซนเซอร์และตำแหน่งของหุ่นยนต์ที่เกิดจากการประมาณ ตามสมการ 3.36 และ 3.37

$$\mu_{N+1,k}^{[i]} = h^{-1}(z_k, x_k^{[i]}) \quad (3.36)$$

$$\Sigma_{N+1,k}^{[i]} = (H_{N+1}^{[i]} R^{-1} H_{N+1}^{[i]T})^{-1} \quad (3.37)$$

3. **Resampling** มีจุดประสงค์เช่นเดียวกับการ Resampling ในวิธีการ Particle Filter จากหัวข้อ 3.4 เพื่อปรับรูปแบบการกระจายตัวของกลุ่มตัวอย่าง ที่เกิดจากการประมาณตำแหน่งด้วยค่าความสำคัญ ( $\omega$ ) ของแต่ละตัวอย่าง จากข้อมูลเซนเซอร์ การคำนวณค่าความสำคัญของแต่ละตัวอย่างใน Fast-SLAM ใช้ค่าความเชื่อมั่นในตำแหน่งของหุ่นยนต์ ที่คำนวณกลับจากข้อมูลการวัดตำแหน่งจุดสังเกตจากเซนเซอร์ มาเทียบกับความเชื่อมั่นในตำแหน่งของหุ่นยนต์ที่คำนวณกลับจากข้อมูลจุดสังเกตที่มีในแต่ละตัวอย่างตามสมการที่ 3.38 ที่ยากต่อการประมาณค่าโดยตรง แต่สามารถปรับรูปแบบโดยอาศัยกฎของ Bayes ออกมาในรูป 3.39 แล้วประมาณค่า PDF โดยอาศัยการกระจายแบบปรกติตามสมการ 3.40 ที่มี  $Q_k^{[i]} = H_k^{[i]T} \Sigma_{n,k-1}^{[i]} H_k^{[i]} + R_k$

$$\omega_k^{[i]} = \frac{P(x_{0:k}^{[i]} | z_{0:k}, u_{0:k}, c_{0:k})}{P(x_{0:k}^{[i]} | z_{0:k-1}, u_{0:k}, c_{0:k-1})} \quad (3.38)$$

$$= \eta P(z_k | x_{0:k}^{[i]}, z_{0:k-1}, c_{0:k}) \quad (3.39)$$

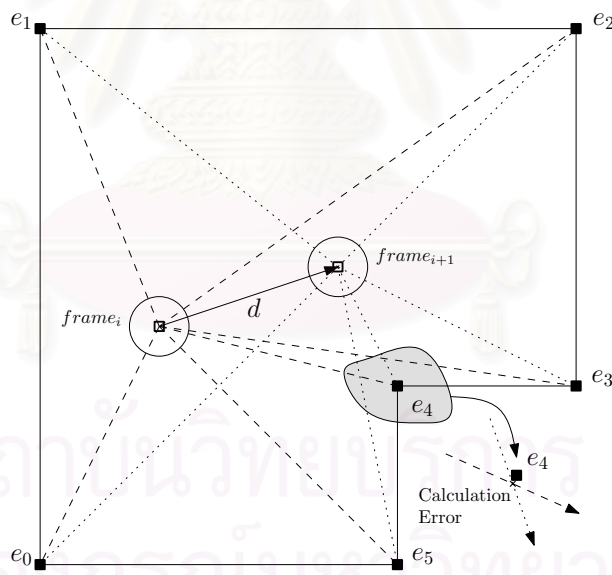
$$\approx |2\pi Q_k^{[i]}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(z_k - \hat{z}_k^{[i]})^T Q_n^{-1} (z_k - \hat{z}_k^{[i]})\right) \quad (3.40)$$

## บทที่ 4

### การระบุตำแหน่งและสร้างแผนที่โดยอาศัยข้อมูลจากกล้องอ้อมนิ

จากจุดประสงค์ของงานวิจัย ในการออกแบบวิธีการนำข้อมูลจากกล้องอ้อมนิเพียงอย่างเดียว มาใช้ในการระบุตำแหน่งและสร้างแผนที่โดยการทำงานร่วมกันของหุ่นยนต์สองตัว ในสภาพแวดล้อมภายในอาคารโดยอาศัยเส้นแนวตั้งเป็นจุดสังเกตหลัก ทำให้สามารถออกแบบวิธีการแก้ปัญหา ซึ่งสามารถแบ่งออกได้เป็นสามส่วนหลัก คือ การระบุเส้นแนวตั้ง การหาทิศทางและระยะทางของเส้นแนวตั้ง และการนำข้อมูลมาสร้างแผนที่และระบุตำแหน่งของหุ่นยนต์ดังรูปที่ 4.1 โดยมีการทำงานร่วมกันของหุ่นยนต์เป็นส่วนเพิ่มในหัวข้อสุดท้ายของบทนี้ซึ่งแบ่งเนื้อหาเป็นสามส่วนตามลำดับดังนี้

- วิธีการหาและระบุตำแหน่งเส้นแนวตั้ง
- การนำข้อมูลเส้นแนวตั้งมาใช้ในการระบุตำแหน่งและสร้างแผนที่ด้วยวิธีการ FastSLAM สำหรับหุ่นยนต์หนึ่งตัว
- การทำงานร่วมกันของหุ่นยนต์สองตัวในการระบุตำแหน่งและสร้างแผนที่ โดยอาศัยข้อมูลตำแหน่งและแผนที่จากวิธีการ FastSLAM



รูปที่ 4.1: รูปแบบการระบุตำแหน่งและสร้างแผนที่

#### 4.1 การตรวจจับเส้นแนวตั้งในภาพจากกล้องอ้อมนิ

ข้อมูลหลักที่ใช้เป็นจุดสังเกตในสิ่งแวดล้อมในงานวิจัยชิ้นนี้ คือ เส้นแนวตั้งซึ่งตรวจพบในภาพจากกล้องอ้อมนิ ตามคุณสมบัติของเส้นตรงในกล้องอ้อมนิในบทที่ 2 โดยใช้ข้อมูลภาพในมุมกว้างจากกล้องอ้อมนิที่ผ่านการแปลงด้วย Mapping Function จากการหาค่าพารามิเตอร์ในหัวข้อ 2.4.1 มาทำการหาเส้นขอบแนวตั้งซึ่งมีขั้นตอนการทำงานและรายละเอียดตามลำดับขั้นตอนดังนี้

1. ลดสัญญาณรบกวนและความไม่สม่ำเสมอของภาพ ด้วยวิธีการ Gaussian Blur ซึ่งให้ผลตามรูป 4.2 เนื่องจากภาพที่แปลงจากภาพรอบทิศทางที่มีความละเอียดต่ำไปเป็นภาพในมุมมองกว้างซึ่งมีความละเอียดสูงขึ้น ทำให้เกิดความไม่ต่อเนื่องของข้อมูลพิกเซลซึ่งมีผลเสียโดยตรงต่อการหาเส้นขอบ



(ก) ภาพมุมมองกว้างจากกล้องออมนิ



(ข) ภาพมุมมองกว้างที่ผ่านการทำ Gaussian Blur

รูปที่ 4.2: ความแตกต่างของภาพก่อนและหลังทำ Gaussian Blur

2. แปลงภาพสีเป็นภาพโทนสีเทา  $RGB \rightarrow Grayscale$  เพื่อใช้การหาเส้นขอบแนวตั้งตามรูปที่ 4.3



รูปที่ 4.3: ภาพที่ผ่านการแปลงเป็นโทนสีเทา

3. ตรวจจับเส้นขอบแนวตั้ง ด้วยวิธีการ Sobel Edge Detection โดยใช้ Kernel สองลักษณะซึ่งตอบสนองการเปลี่ยนแปลงจากบวกเป็นลบ และลบเป็นบวก ซึ่งให้ผลตามรูปที่ 4.4 สาเหตุการใช้ Kernel สองแบบเกิดจากเส้นแนวตั้งในสภาพแวดล้อมบางส่วน มีลักษณะเป็นแถบ เช่น กรอบอลูมิเนียม หรือจุดเชื่อมจากกันห้อง ซึ่งทำให้เกิดขอบขึ้นทั้งทางซ้ายและขวา จึงจำเป็นต้องใช้ขอบทั้ง



สองฝั่งมาใช้ เพื่อให้สามารถประมาณตำแหน่งในภาพได้แม่นยำขึ้นในขั้นตอนที่ 6

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

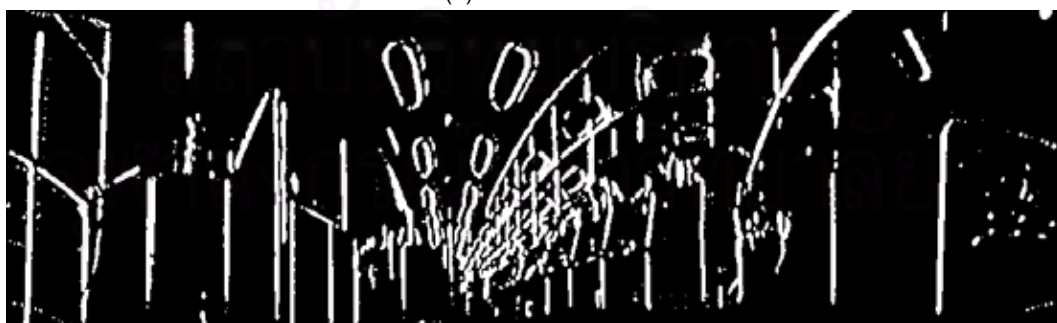


รูปที่ 4.4: ภาพขอบแนวตั้งที่เกิดจากขอบซ้ายและขวาของเส้นแนวตั้งในภาพ

4. แยกเส้นขอบที่เด่นชัด ด้วยวิธีการ Binary Thresholding เพื่อให้เหลือเพียงเส้นซึ่งเป็นเส้นขอบแนวตั้งที่ชัดเจนการแยกเส้นแนวตั้งที่เกิดกับขอบด้านซ้ายและขวาตามลักษณะของ Kernel ในขั้นตอนที่ 3 ซึ่งให้ผลดังรูปที่ 4.5



(ก) ขอบทางด้านซ้าย



(ข) ขอบทางด้านขวา

รูปที่ 4.5: ตำแหน่งเส้นแนวตั้งจากการใช้ Kernel สองแบบตามขั้นตอนที่ 3

5. หาดำแหน่งและความยาวของเส้นแนวตั้ง ในภาพโดยอาศัย Hough Line Transformation โดยกำหนดให้ค้นหาเฉพาะเส้นแนวตั้งเพียงอย่างเดียว ได้ผลตามรูปที่ 4.6

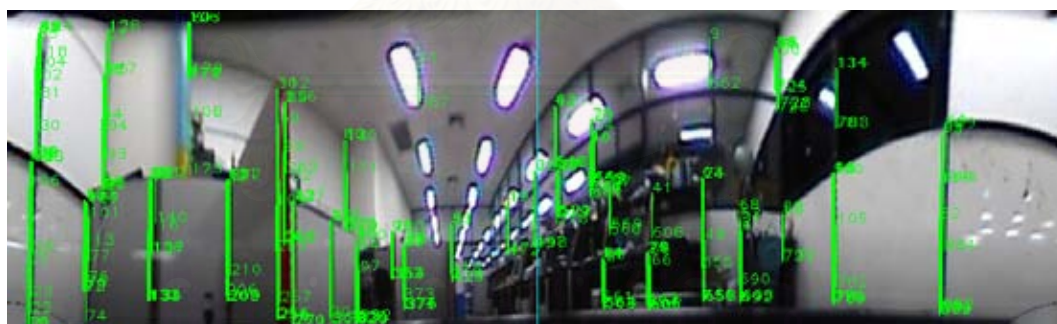
6. รวมเส้นแนวตั้งในแต่ละภาพที่มีแนวโน้มเป็นเส้นเดียวกัน โดยอาศัยระยะห่าง และความยาวของเส้นตรงที่ตรวจพบในขั้นตอนที่ 5 โดยคำนวณ ตำแหน่งในแกน  $X$  ซึ่งเป็นมุมของเส้นแนวตั้งที่มีความสำคัญต่อการระบุตำแหน่งและสร้างแผนที่เป็นอย่างมาก ด้วยการเฉลี่ยตำแหน่งแบบถ่วงน้ำหนัก (Weighted Mean) ด้วยความยาวของเส้นที่สามารถรวมกันได้ ให้ผลตามภาพที่ 4.7 (ก) และ (ข) หลังจากนั้นนำข้อมูลเส้นขอบแนวตั้งทางด้านซ้ายและขวาที่มีแนวโน้มว่าเกิดจากเส้นแนวตั้งเดียวกันซึ่งปรากฏเป็นแถบ โดยอาศัยระยะห่างและความยาว เช่นเดียวกับการรวมเส้นแนวตั้งในส่วนแรก แต่เพิ่มรายละเอียดของสีทางด้านซ้ายและขวาเพื่อความถูกต้องในการจับคู่ และได้คำตอบเป็นตำแหน่งของเส้นแนวตั้งทั้งหมดที่สามารถตรวจจับได้ดังรูปที่ 4.7 (ค)

ข้อมูลจากขั้นตอนข้างต้น คือ ทิศทาง ตำแหน่งในแนวตั้ง ความยาว และลักษณะของสีทางด้านซ้ายและขวา ของเส้นแนวตั้งทั้งหมดที่ตรวจพบ

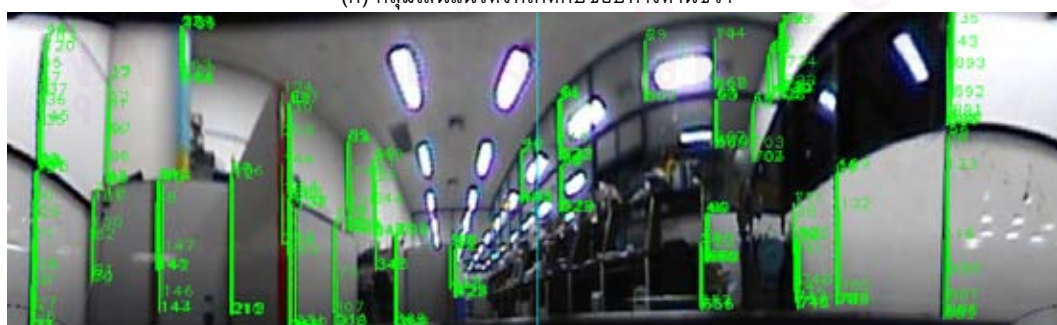
#### 4.2 การระบุตำแหน่งของเส้นแนวตั้งโดยอาศัยข้อมูลจากการวัดมุม

เงื่อนไขในการระบุตำแหน่งของเส้นแนวตั้งในงานวิจัยชิ้นนี้ คือ ต้องการลดความจำเป็นในการติดตามการเปลี่ยนแปลงของเส้นแนวตั้งที่สนใจ (Tracking) เพื่อใช้ในการคำนวณตำแหน่งให้น้อยที่สุดทำให้ง่ายต่อการออกแบบ และสามารถใช้งานได้ทั่วไปโดยไม่จำเป็นต้องทราบคุณสมบัติที่ชัดเจนของเส้นแนวตั้ง

ด้วยเงื่อนไขข้างต้นการระบุตำแหน่งเส้นแนวตั้งจะทำในขณะทีหุ่นยนต์เคลื่อนที่เป็นเส้นตรงเพียงอย่างเดียว โดยใช้ข้อมูลเส้นแนวตั้งจากสองช่วงเวลาติดกันเท่านั้นในการคำนวณตำแหน่ง และไม่ทำการวัด



(ก) กลุ่มเส้นแนวตั้งที่เกิดกับขอบทางด้านขวา



(ข) กลุ่มเส้นแนวตั้งที่เกิดกับขอบทางด้านซ้าย

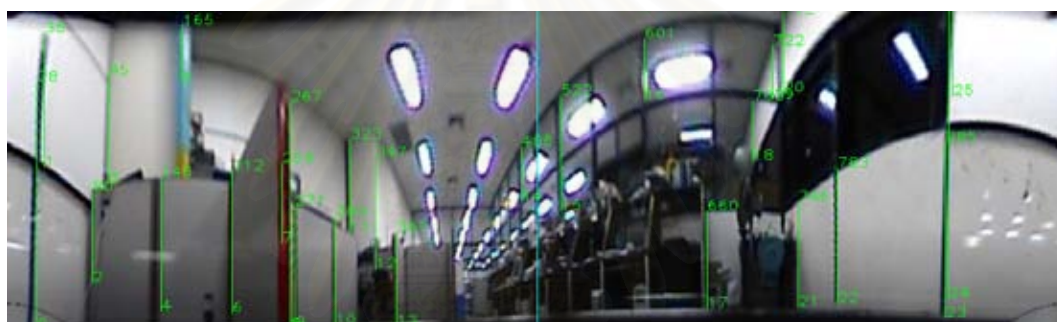
รูปที่ 4.6: เส้นแนวตั้งจากกระบวนการ Hough Line Transformation

ตำแหน่งเส้นแนวตั้งเมื่อหุ่นยนต์หมุนตัวเพื่อลดความจำเป็นในการติดตามตำแหน่งของเส้นแนวตั้งที่สนใจ

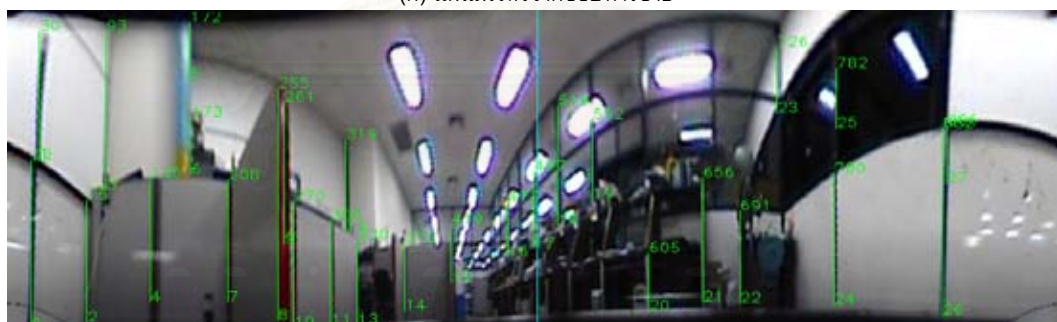
การคำนวณตำแหน่งของเส้นแนวตั้งทำได้โดยอาศัยทิศทางของเส้นแนวตั้งซึ่งวัดได้จากสองตำแหน่ง ( $\theta_i, \theta_{i+1}$ ) ร่วมกับระยะห่างของสองตำแหน่ง ( $d$ ) ในการทำ Triangulation เพื่อหาตำแหน่งของเส้นแนวตั้งตามรูปที่ 4.8

ส่วนสำคัญของการวิธีการระบุตำแหน่ง คือ การระบุว่าเส้นแนวตั้งสองเส้นใดๆ ซึ่งตรวจพบระหว่างสองช่วงเวลาถัดกัน (Time Step) เป็นเส้นเดียวกันเพื่อความถูกต้องในการวัดตำแหน่ง โดยสามารถทำได้ด้วยการตรวจสอบ ทิศทางที่เปลี่ยนไป ความยาว และตำแหน่ง ตามรูปที่ 4.9 ร่วมกับการตรวจสอบสีทางด้านซ้ายและขวาของเส้นแนวตั้งที่สนใจตามวิธีการในงานวิจัย [16]

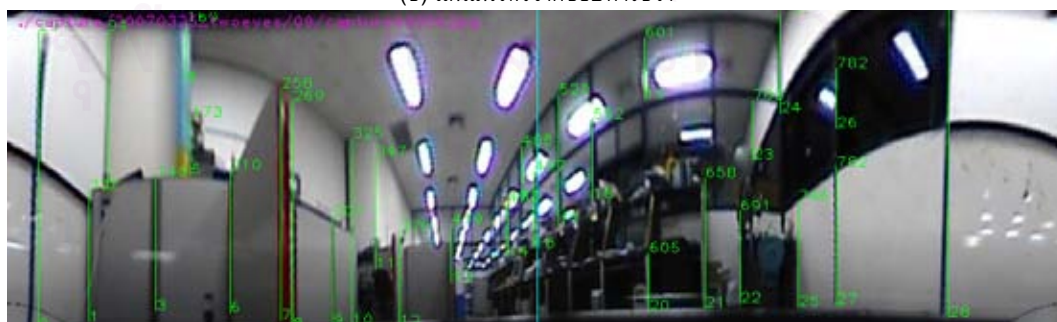
อย่างไรก็ตามวิธีการดังกล่าวย่อมมีความผิดพลาดเกิดขึ้นได้ โดยเฉพาะการวัดตำแหน่งเส้นแนวตั้งในระยะไกล ซึ่งเกิดจากสิ่งแวดล้อมที่มีลักษณะคล้ายกัน เช่น แนวนั่งกระจกที่มีโครงสร้างเป็นอลูมิเนียมหรือขาโต๊ะในระยะไกล ข้อมูลจุดสังเกตที่ผิดพลาดดังกล่าว มีผลเป็นอย่างมากต่อการระบุตำแหน่งและ



(ก) เส้นแนวตั้งจากขอบทางซ้าย



(ข) เส้นแนวตั้งจากขอบทางขวา



(ค) เส้นแนวตั้งจากขอบทางซ้ายและขวา และตำแหน่งของเส้นแนวตั้งที่ปรากฏเป็นแถบ

รูปที่ 4.7: การระบุตำแหน่งของเส้นแนวตั้งในภาพ



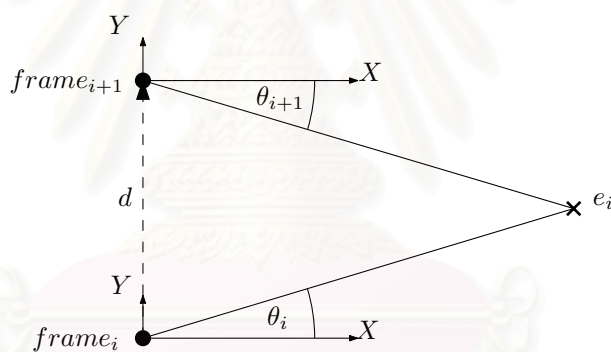
สร้างแผนที่ของหุ่นยนต์ตามการทดลองในบทที่ 5 แต่สามารถแก้ไขได้ด้วยวิธี การกำจัดข้อมูลการวัดตำแหน่งที่ผิดพลาด ซึ่งเพิ่มเติมเข้าไปในวิธีการ FastSLAM พื้นฐานและจะกล่าวถึงในหัวข้อถัดไป

เมื่อผ่านขั้นตอนทั้งหมดแล้วจะได้ตำแหน่งสัมพัทธ์ของเส้นแนวตั้งทั้งหมดเทียบกับตัวกล้องออปติคัลซึ่งสามารถแปลงให้อยู่ในรูปของระยะทางและมุม (Range and Bearing Measurement) เพื่อใช้ในการระบุตำแหน่งและสร้างแผนที่ในงานวิจัย

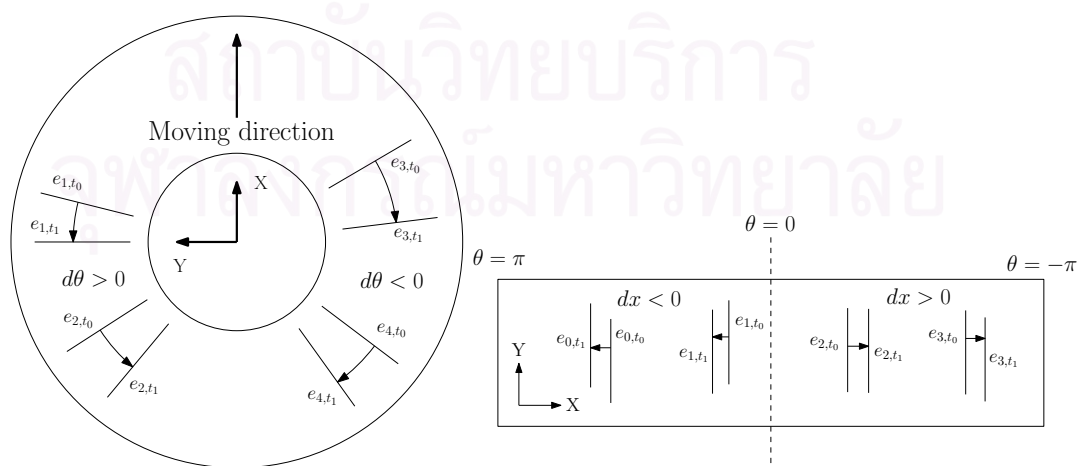
### 4.3 FastSLAM โดยใช้ข้อมูลเส้นแนวตั้งในสภาพแวดล้อม

ในสองหัวข้อที่ผ่านมาเป็นการออกแบบวิธีดึงข้อมูลซึ่งจำเป็นสำหรับการระบุตำแหน่งและสร้างแผนที่โดยไม่คำนึงถึงความไม่แน่นอนที่เกิดขึ้น เช่น การจับคู่เส้นเพื่อหาตำแหน่งผิด การคำนวณระยะทางและมุมของเส้นแนวตั้งซึ่งผิดพลาดจากสัญญาณรบกวนจาก CCD และสภาพแสง และความไม่แน่นอนในการเคลื่อนที่ของหุ่นยนต์ ซึ่งต้องรวมความไม่แน่นอนทั้งหมดที่เป็นไปได้ เข้าไปในระเบียบวิธีการสำหรับแก้ปัญหา แต่การแก้ปัญหาข้างต้นเป็นสิ่งที่ระบุได้ยากหรือทำไม่ได้ในทางปฏิบัติ เนื่องจากรูปแบบของความไม่แน่นอนทั้งหมดไม่สามารถหาได้โดยง่าย

แต่ในการใช้งานจริงเราสามารถประมาณได้จากการสังเกตและลดปัจจัยซึ่งทำให้เกิดความไม่แน่นอน



รูปที่ 4.8: การหาตำแหน่งของเส้นแนวตั้งด้วยการ Triangulation



(ก) การเปลี่ยนแปลงมุมเมื่อหุ่นยนต์เคลื่อนที่

(ข) ตำแหน่งที่เปลี่ยนไปในภาพมุมกว้าง

รูปที่ 4.9: ความเปลี่ยนแปลงของเส้นแนวตั้งในภาพจากกล้องออปติคัลเมื่อหุ่นยนต์เคลื่อนที่

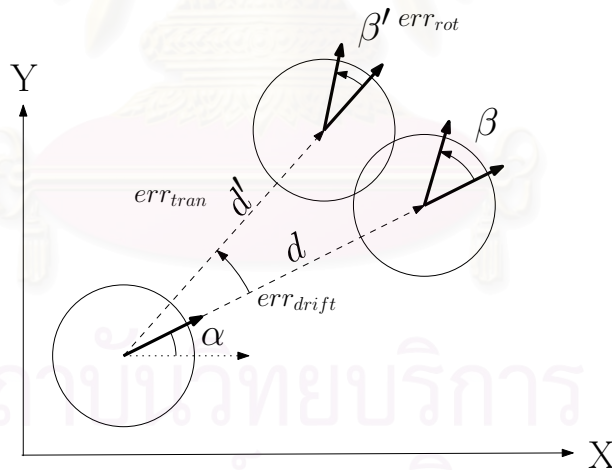
ที่ยากต่อการประมาณค่า เช่น รูปการเคลื่อนที่ของหุ่นยนต์ในสองมิติสามารถลดให้เหลือเพียงสองรูปแบบ คือ การเคลื่อนที่เป็นเส้นตรงและการหมุนตัว เพื่อให้สามารถสร้างโมเดลของของ ความไม่แน่นอนได้ ใกล้เคียงความเป็นจริง

เนื้อหาในส่วนนี้ เป็น รายละเอียด ของ โมเดล การเคลื่อนที่ ของ หุ่นยนต์ การ วัด ตำแหน่ง ใน เซิง ความน่าจะเป็นจากการทดลอง และการนำระเบียบวิธีการ FastSLAM ที่อยู่ในรูปโค้ดจำลอง (Psudo Code) จากงานวิจัย [44] มาใช้งาน โดยตัดแปลงและเพิ่มเติมขั้นตอนการทำงานให้เหมาะสมกับการใช้งานของงานวิจัยชิ้นนี้

#### 4.3.1 โมเดลการเคลื่อนที่ของหุ่นยนต์

ตามที่ได้กล่าวมาข้างต้น รูปแบบการเคลื่อนที่ของหุ่นยนต์ สามารถช่วยลดปัจจัยซึ่งทำให้เกิดความไม่แน่นอนในการเคลื่อนที่ลงได้ ด้วยการแบ่งการเคลื่อนที่ออกเป็นสองชั้น คือ การเคลื่อนที่เป็นเส้นตรงแล้ว จึงหมุนตัวไปในทิศทางที่ต้องการ ซึ่งสามารถอธิบายได้ตามรูปที่ 4.10 ที่กำหนดให้หุ่นยนต์เคลื่อนที่ไปเป็น ระยะทาง  $d$  แล้วจึงหมุนตัวเป็นระยะ  $\beta$  โดยมีทิศทางเริ่มต้นเป็น  $\alpha$

จากการสังเกตพบว่าการเคลื่อนที่ตามรูปแบบดังกล่าว มีความคลาดเคลื่อนเกิดขึ้นสามรูปแบบ คือ ความคลาดเคลื่อนของระยะทาง ( $err_{tran}$ ) การเบนออกจกแนวการเคลื่อนที่ ( $err_{drift}$ ) และความคลาดเคลื่อนของการหมุนตัว ( $err_{rot}$ ) ซึ่งประมาณได้ด้วยการกระจายตัวของความความน่าจะเป็นแบบปรกติที่มีค่าเฉลี่ยเป็นศูนย์  $N(0, \sigma)$  และมีค่าเบี่ยงเบนมาตรฐาน  $\sigma_{tran}$   $\sigma_{drift}$  และ  $\sigma_{rot}$



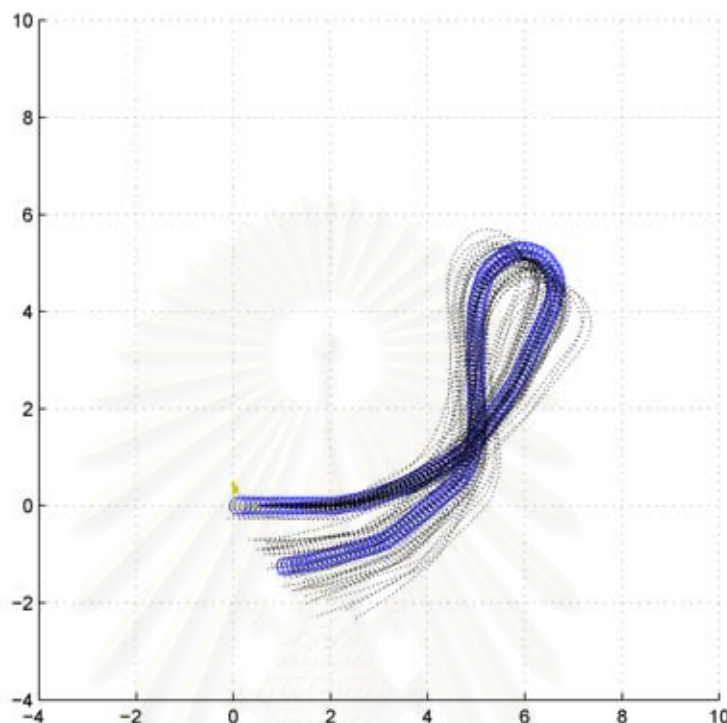
รูปที่ 4.10: การเคลื่อนที่ของหุ่นยนต์ และลักษณะของความไม่แน่นอนที่เกิดขึ้น

ด้วยรูปแบบการเคลื่อนที่ข้างต้น สามารถเขียนการเปลี่ยนแปลงตำแหน่งของหุ่นยนต์ เมื่อได้รับคำสั่งให้เคลื่อนเป็นระยะทาง  $d$  และแล้วหมุนตัวเป็นมุม  $\beta$  ได้ตามสมการ 4.1 โดยมี  $err_{tran} \sim N(0, \sigma_{tran} \cdot d)$ ,  $err_{drift} \sim N(0, \sigma_{drift} \cdot d)$  และ  $err_{rot} \sim N(0, \sigma_{rot} \cdot \beta)$  เป็นความคลาดเคลื่อนที่เกิดขึ้น

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} \approx \begin{bmatrix} x_{k-1} + (u_{tran} + err_{tran}) \cdot \cos(\theta + err_{drift}) \\ y_{k-1} + (u_{tran} + err_{tran}) \cdot \sin(\theta + err_{drift}) \\ \theta_{k-1} + u_{rot} + err_{rot} + err_{drift} \end{bmatrix} \quad (4.1)$$



เมื่อนำโมเดลการเคลื่อนที่ของหุ่นยนต์ตามสมการที่ 4.1 ไปใช้ทดสอบการประมาณเส้นทางการเคลื่อนที่ของหุ่นยนต์ เทียบกับเส้นทางซึ่งคำนวณจากตำแหน่งของ encoder พบว่าสามารถประมาณความคลาดเคลื่อนได้ครอบคลุมความเป็นไปของเส้นทางการเคลื่อนที่ได้ตามรูปที่ 4.11



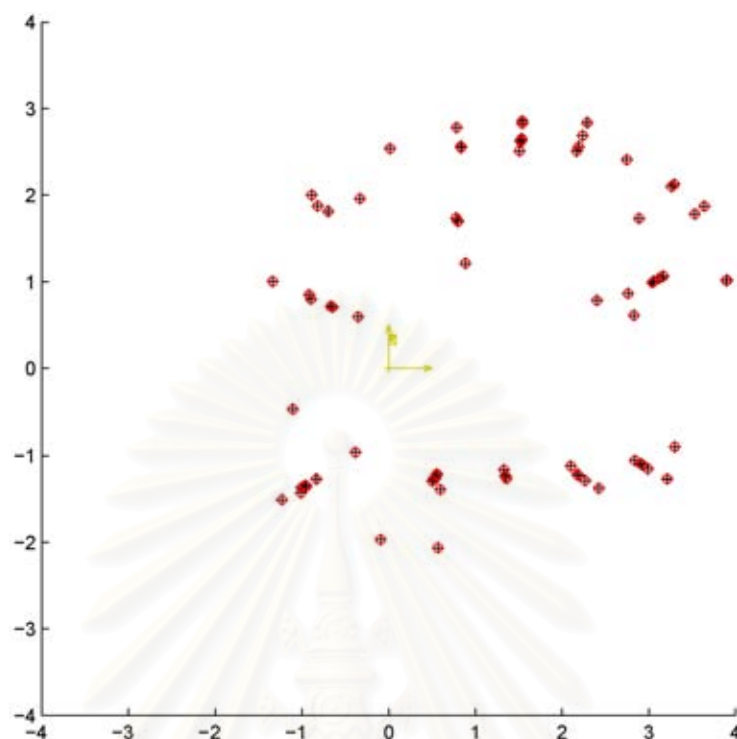
รูปที่ 4.11: เส้นทางจากค่าซึ่งวัดโดย encoder และเส้นทางทั้งหมดที่สามารถประมาณได้ผ่านโมเดลการเคลื่อนที่ของหุ่นยนต์ตามสมการ 4.1 โดยกำหนดให้  $\sigma_{tran} = 1\text{cm}/\text{m}$   $\sigma_{drift} = 5^\circ/\text{m}$  และ  $\sigma_{rot} = 5^\circ/\pi$

#### 4.3.2 โมเดลการวัดตำแหน่งของเส้นแนวตั้ง

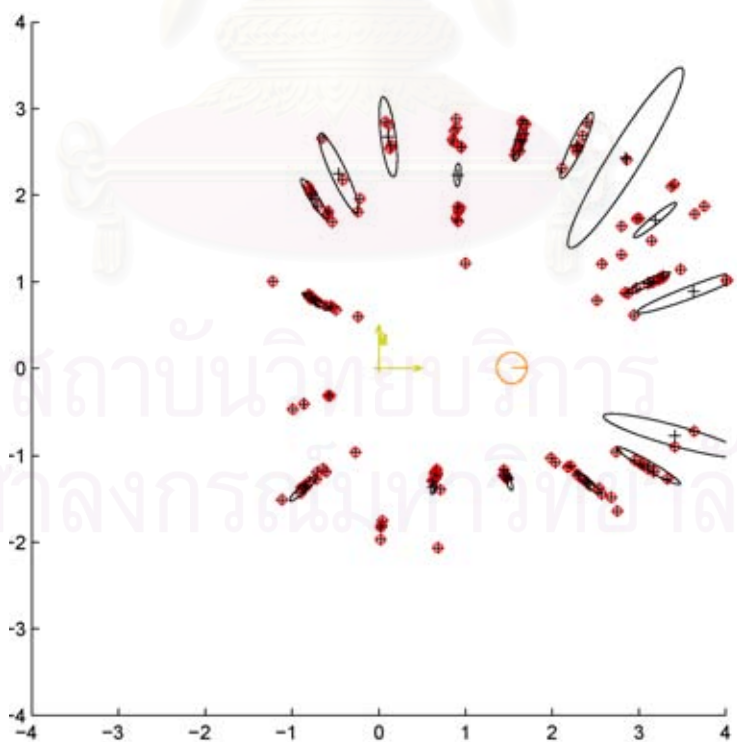
การวัดตำแหน่งของเส้นแนวตั้ง  $[x \ y]^T$  ตามหัวข้อ 4.2 โดยอาศัยมุมจากกล้องออปติ และระยะทางที่หุ่นยนต์เคลื่อนที่ เป็นตำแหน่งสัมพันธ์กับตำแหน่งของกล้องออปติ ซึ่งสามารถแปลงให้อยู่ในรูประยะทางและมุม  $[r \ \theta]^T$  ได้ตามสมการ 4.2 ทำให้สามารถประมาณรูปแบบของความผิดพลาดในการวัดค่า ออกมาได้สองส่วน คือ ความผิดพลาดของระยะทางและมุมที่วัดได้ ซึ่งประมาณได้ด้วยการกระจายตัวของความน่าจะเป็นแบบปรกติ เช่นเดียวกับโมเดลการเคลื่อนที่ โดยมีค่าความเบี่ยงมาตรฐานเป็น  $\sigma_r$  และ  $\sigma_\theta$

$$\begin{bmatrix} r \\ \theta \end{bmatrix} = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \arctan(\frac{y}{x}) \end{bmatrix} \quad (4.2)$$

รูป 4.12 (ก) แสดงให้เห็นตำแหน่งของเส้นแนวตั้งซึ่งคำนวณจากมุมของเส้นแนวตั้งที่วัดโดยกล้องออปติ และสามารถนำไปสร้างเป็นแผนที่ได้ตามรูปที่ 4.12 (ข) โดยแสดงความเชื่อมั่นในตำแหน่งของเส้นแนวตั้งด้วยวงรี ซึ่งแทนความแปรปรวนร่วมในสองมิติของตำแหน่งเส้นแนวตั้งด้วยลักษณะเดียวกับกระจายตัวของตำแหน่งเส้นแนวที่วัดได้โดยกล้องออปติ



(ก) ภาพ แสดง การ กระจาย และ ความ คลาดเคลื่อน ของ ตำแหน่ง เส้น แนว ดิ่ง ใน สภาพแวดล้อมจากกล้องอ้อมนิ เมื่อหุ่นยนต์เคลื่อนที่เป็นระยะทาง  $1.5m$



(ข) ความ เชื่อมั่น ในตำแหน่ง ของเส้นแนวตั้งจากการสร้างแผนที่ด้วยข้อมูลชุดเดียวกัน เมื่อกำหนดให้  $\sigma_r = 0.5m$  และ  $\sigma_\theta = 1.5^\circ$

รูปที่ 4.12: ภาพเปรียบเทียบการกระจายตัวของตำแหน่งเส้นแนวตั้งจากการวัดและการประมาณด้วยโมเดลการวัดตำแหน่ง

### 4.3.3 ขั้นตอนการระบุตำแหน่งและสร้างแผนที่โดยอาศัยข้อมูลเส้นแหวดิ่งด้วยวิธีการ Fast-SLAM

ขั้นตอนการระบุตำแหน่งและสร้างแผนที่โดยอาศัยข้อมูลเส้นแหวดิ่งในสิ่งแวดล้อมด้วยวิธีการ Fast-SLAM ในงานวิจัยชิ้นนี้เป็นไปตามระเบียบวิธีการ 2 ซึ่งประยุกต์ใช้ระเบียบวิธีการ FastSLAM 1.0 สำหรับการระบุตำแหน่งและสร้างแผนที่โดยไม่รู้ความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกต (FastSLAM 1.0 with unknown data association) จากระเบียบวิธีการใน [44] โดยเพิ่มเติมรายละเอียด และขั้นตอนการทำงาน รวมถึงความหมายของตัวแปรต่างๆ เพื่อให้เหมาะสมกับการใช้ในงานวิจัยดังต่อไปนี้

#### 4.3.3.1 ความหมายของตัวแปรของระเบียบวิธีการ FastSLAM ในงานวิจัย

- สถานะระบบ ประกอบด้วยข้อมูลสามส่วน คือ

- ตำแหน่งของหุ่นยนต์ ณ เวลา  $k$  แทนด้วย

$$x_k = [x_r, y_r, \theta_r]^T$$

- ตำแหน่งของจุดสังเกต ที่  $n$  แทนด้วย

$$m_n = [x_f, y_f]^T$$

- แผนที่ คือ เซตของตำแหน่งจุดสังเกต  $m$  จำนวน  $N$  จุด แทนด้วย

$$M = \langle m_1, m_2, \dots, m_N \rangle$$

- คำสั่งควบคุม คือ คำสั่งให้หุ่นยนต์เคลื่อนที่เป็นเส้นตรงหรือหมุนตัว ณ เวลา  $k$  แทนด้วย

$$u_k = \langle tran, rot \rangle$$

- การวัดตำแหน่ง คือ เซตของตำแหน่งเส้นแหวดิ่งซึ่งสัมพันธ์กับตำแหน่งกล้อง โดยใช้ข้อมูลจากกล้องอ้อมนี้ซึ่งประกอบด้วย ระยะห่าง ( $d_f$ ) และมุม ( $\theta_f$ ) ณ เวลา  $k$  ที่มีจำนวน  $J$  จุดต่อการวัดหนึ่งครั้ง แทนด้วย

$$z_k = \langle z_1, z_2, \dots, z_J \rangle$$

- เซตกลุ่มตัวอย่าง จำนวน  $I$  ตัวอย่างมีส่วนประกอบดังนี้

$$S_k = \langle S_k^{[1]}, S_k^{[2]}, \dots, S_k^{[I]} \rangle$$

โดย

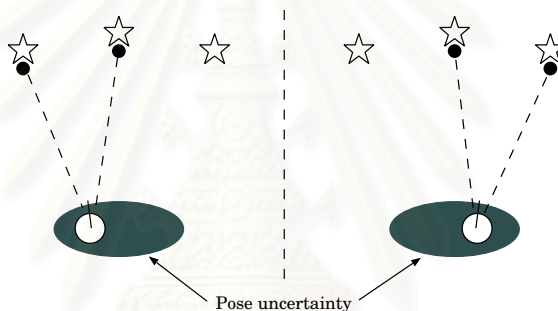
$$S_k^{[i]} = \langle x_k^{[i]}, N_k^{[i]}, \omega_k^{[i]}, feature_1, feature_2, \dots, feature_N \rangle$$

กำหนดให้  $feature_n = \langle \mu_{n,k}^{[i]}, \Sigma_{n,k}^{[i]}, cnt_{n,k}^{[i]}, isUpdated_{n,k}^{[i]} \rangle$  และ  $cnt$  แทนค่าความน่าจะเป็นในการมีอยู่ของจุดสังเกต โดยมี  $isUpdated$  แสดงว่าเป็นจุดสังเกตใหม่หรือเป็นจุดสังเกตเดิมที่ปรากฏใน

$z_k$ 

### 4.3.3.2 ปัญหาความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกต

การทำงานของระเบียบวิธีการ FastSLAM ในหัวข้อ 3.5 มีข้อจำกัดในการใช้งาน เมื่อไม่ทราบความสัมพันธ์ระหว่างตำแหน่งที่วัดได้โดยเซนเซอร์และวัตถุที่ให้ค่าการวัด (Unknown data association) เป็นปัญหาหลักของการระบุตำแหน่งและสร้างแผนที่โดยอาศัยจุดสังเกตในสภาพแวดล้อม เนื่องจากจุดสังเกตที่เหมาะสมในการนำมาใช้ ควรเป็นจุดสังเกตซึ่งตรวจพบได้ง่าย มีรูปแบบคงที่ และมีลักษณะใกล้เคียงกัน เช่น เส้นแนวตั้ง สัญลักษณ์เฉพาะ หรือใช้อุปกรณ์พิเศษที่ติดตั้งเพิ่มเติม จากคุณสมบัติที่คล้ายกันของจุดสังเกตที่ดี ทำให้ง่ายต่อการดึงข้อมูลจากเซนเซอร์ แต่ยากต่อการระบุความสัมพันธ์ระหว่างค่าจากเซนเซอร์และตำแหน่งที่แท้จริงของจุดสังเกตนั้น โดยเฉพาะเมื่อไม่ทราบตำแหน่งแน่นอนของหุ่นยนต์ตามรูปที่ 4.13 ทำให้จำเป็นต้องหาวิธีระบุความสัมพันธ์ดังกล่าว หรือในอีกนัยหนึ่ง คือ หาวิธีการระบุค่า  $c_k$  ในสมการ 3.27 ซึ่งใช้บอกความสัมพันธ์ของค่าที่วัดได้กับตำแหน่งของจุดสังเกต



รูปที่ 4.13: ปัญหาความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกต เมื่อไม่ทราบตำแหน่งแน่นอนของหุ่นยนต์ (ภาพจากงาน [44])

วิธีการพื้นฐานในการแก้ปัญหาความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกต คือ การเลือกจุดสังเกต  $m_n$  จากรายการข้อมูลตำแหน่งจุดสังเกตในแต่ละตัวอย่างของ Particle Filter ซึ่งใกล้เคียงกับค่าจากการวัดตำแหน่งมากที่สุด (Maximum Likelihood Estimator) และสามารถคำนวณได้ตามสมการ 4.3 ซึ่งเป็นรูปทั่วไปที่ต้องปรับให้เข้ากับการใช้งานจริง เมื่อ  $\hat{n}_k$  คือ ลำดับของข้อมูลจุดสังเกต ณ เวลา  $k$  ซึ่งมีตำแหน่งใกล้เคียงกับตำแหน่งจากการวัดโดยเซนเซอร์  $z_k$  ล่าสุด โดยอาศัยข้อมูลเส้นทางการเคลื่อนที่  $s_{0:k}$  การวัดตำแหน่ง  $z_{0:k-1}$  และการควบคุมทั้งหมดที่ให้กับหุ่นยนต์  $u_{0:k}$

$$\hat{n}_k = \operatorname{argmax}(P(z_k | n_k, \hat{n}_{0:k-1}, s_{0:k}, z_{0:k-1}, u_{0:k})) \quad (4.3)$$

ในการประยุกต์ใช้งานจริงในการระบุตำแหน่งและสร้างแผนที่ สามารถอธิบายการทำงานได้ตามบรรทัดที่ 6 - 14 ของระเบียบวิธีการ 2 (หน้า 40) เมื่อกำหนดให้จำนวนข้อมูลจุดสังเกตในแต่ละตัวอย่างของ Particle Filter เป็น  $N_{k-1}$  และหุ่นยนต์ได้รับคำสั่ง  $u_k$  ให้เคลื่อนที่ การหาจุดสังเกตซึ่งใกล้เคียงกับค่าจากเซนเซอร์ทำได้ตามลำดับขั้นตอนต่อไปนี้

- ประเมินค่าการวัดตำแหน่งของข้อมูลจุดสังเกตทั้งหมดที่ละจุดตามบรรทัดที่ 8 ด้วยตำแหน่งของหุ่นยนต์จากการประมาณผ่านโมเดลการเคลื่อนที่ในบรรทัดที่ 3
- คำนวณหาความแปรปรวนร่วมของการวัดตำแหน่ง ( $Q_n$ ) จากการประมาณค่าด้วยวิธีการในบรรทัดที่ 9 และ 10
- ประเมินความเชื่อมั่นในตำแหน่งของจุดสังเกตจากเซนเซอร์เทียบกับค่าซึ่งทำการประมาณ โดยอาศัยโมเดลการกระจายตัวของความน่าจะเป็นแบบปรกติในบรรทัดที่ 11 และใช้เป็นค่าความสำคัญของตัวอย่างซึ่งเกิดจากการวัดค่า  $z_k$  ในการทำ Resampling ของ FastSLAM

เมื่อทำการเปรียบเทียบกับข้อมูลทั้งหมด จะได้รายการของความเชื่อมั่น  $w$  เพื่อหา  $\hat{n}$  ซึ่งแทนความสัมพันธ์ของข้อมูลการวัดตำแหน่งจุดสังเกตในบรรทัดที่ 14 และให้ค่าที่เป็นไปได้ในสองลักษณะ คือ เป็นจุดสังเกตเดิมหรือจุดสังเกตใหม่ เช่นเดียวกับกระบวนการ *ปรับแก้ข้อมูลการวัดตำแหน่งของจุดสังเกต* ในวิธีการ FastSLAM ตามหัวข้อ 3.5 การวัดตำแหน่ง  $z_k$  จะเป็นของจุดสังเกตใหม่เมื่อค่า  $w$  ของ  $\hat{n}$  ต่ำกว่า  $P_0$  ซึ่งเป็นค่าคงในบรรทัดที่ 13

#### 4.3.3.3 การกำจัดการวัดตำแหน่งที่ผิดพลาด

เมื่อทำการหาความสัมพันธ์ของข้อมูล การวัด ตำแหน่ง จุด สังเกต แล้วได้ผล เป็น จุด สังเกต ใหม่ ใน สภาพแวดล้อม เนื่องจากค่าความเชื่อมั่นต่ำกว่า  $P_0$  จะมีขั้นตอนการกำหนดค่าความเชื่อในตำแหน่งของจุดสังเกตตามบรรทัดที่ 17 - 22 ของระเบียบวิธีการ 2 (หน้า 40) สิ่งที่ต้องคำนึงถึงในส่วนถัดไป คือ จุดสังเกตที่วัดได้มีอยู่จริงหรือเกิดจากความผิดพลาดในการทำงานของเซนเซอร์ ซึ่งเป็นปัญหาที่สำคัญของการทำงานและส่งผลโดยตรงต่อความสำเร็จในการการระบุตำแหน่งและสร้างแผนที่ [22] วิธีการแก้ปัญหาที่ตรงตัวที่สุด คือ การย้อนกลับไปตรวจสอบลักษณะการทำงานของเซนเซอร์ ว่ามีข้อจำกัดในการทำงานเป็นอย่างไร และสาเหตุของความผิดพลาดเกิดจากสาเหตุใดได้บ้าง

ในกรณีของเซนเซอร์ที่ให้ข้อมูลเป็นระยะและทิศทางของจุดสังเกตที่สามารถวัดได้รอบทิศทาง เช่น กล้องออบนิและเลเซอร์วัดระยะทาง สามารถกล่าวได้ว่า เซนเซอร์มีระยะทำการที่จำกัดและจุดสังเกตที่อยู่ในระยะทำงานของเซนเซอร์ควรมีค่าการวัดตำแหน่งเมื่อไม่ถูกบัง จากเงื่อนไขข้างต้นสามารถนำมาใช้ในการระบุจุดสังเกตนั้นมีอยู่จริงหรือเกิดจากความผิดพลาดในการทำงานของเซนเซอร์ โดยการเพิ่มตัวแปรสำหรับแต่ละจุดสังเกต  $cnt$  ซึ่งเพิ่มค่าทุกครั้งเมื่อปรากฏเป็นจุดสังเกตมีอยู่ในตัวอย่างของ Particle Filter ตามบรรทัดที่ 28 แต่ในกรณีที่จุดสังเกตอยู่ในระยะทำงานของเซนเซอร์แต่ไม่ถูกตรวจวัดในช่วงเวลานั้น  $cnt$  จะโดนลดค่าจนจุดสังเกตนั้นถูกกำจัดออกไป เมื่อค่าของ  $cnt$  ต่ำกว่าศูนย์ ตามเงื่อนไขในบรรทัดที่ 36 - 42

#### 4.3.3.4 การปรับแก้ระเบียบวิธีการ FastSLAM 1.0 เพื่อใช้ในงานวิจัย

ระเบียบวิธีการ FastSLAM 1.0 ในงานวิจัย [44] ถือว่าการวัดตำแหน่งในแต่ละช่วงเวลาเกิดกับจุดสังเกตเพียงจุดเดียวเท่านั้นเพื่อให้สะดวกต่อการอธิบายการทำงาน เงื่อนไขการทำงานดังกล่าวไม่ตรงกับความเป็นในการใช้งานของงานวิจัย เนื่องจากการวัดตำแหน่งของเส้นแนวตั้งจากกล้องออบนิเกิดขึ้นพร้อมกันหลายจุดรอบตัวหุ่นยนต์ในแต่ละช่วงเวลา จึงต้องเพิ่มลำดับขั้นตอนต่างๆ ดังต่อไปนี้



- ปรับแก้ให้สามารถรองรับการวัดค่าหลายตำแหน่งในแต่ละช่วงเวลาโดยเพิ่มลูบในบรรทัดที่ 4
- การคำนวณค่าความสำคัญของแต่ละตัวอย่าง ต้องคำนวณจากทุกการวัดตำแหน่งในแต่ละช่วงเวลา ซึ่งทำได้โดยอาศัยตัวแปร  $W$  ในบรรทัดที่ 3 ในการรวมค่าความสำคัญของตัวอย่างที่เกิดจากการวัดค่าทั้งหมดในบรรทัดที่ 15 เพื่อใช้เป็นค่าความสำคัญของตัวอย่าง
- การพิจารณาการมีอยู่จริงของจุดสังเกต ต้องพิจารณาเฉพาะข้อมูลจุดสังเกตที่ไม่ใช่จุดสังเกตใหม่และไม่โดนตรวจวัดซ้ำในช่วงเวลานั้น โดยการใช้ตัวแปร  $isUpdated$  เป็นตัวระบุว่าจุดสังเกตนั้นได้รับการตรวจวัดโดยการวัดตำแหน่งในครั้งนั้นหรือไม่

#### 4.4 การทำงานร่วมกันของหุ่นยนต์สองตัวในการระบุตำแหน่งและสร้างแผนที่

ขั้นตอนการทำงานของหุ่นยนต์ในการสำรวจสถานที่แบ่งออกเป็นสองรูปแบบ คือ การแยกกันปฏิบัติงาน และการแลกเปลี่ยนข้อมูลแผนที่ซึ่งกันและกันเมื่อหุ่นยนต์ทั้งสองสามารถสังเกตเห็นหุ่นยนต์ที่ปฏิบัติงานร่วมกัน เนื้อหาในส่วนนี้มีสองส่วน คือ การตรวจหาและวัดตำแหน่งของหุ่นยนต์ที่ปฏิบัติร่วมกันด้วยเครื่องหมายที่ติดตั้งบนหุ่นยนต์ และการรวมข้อมูลแผนที่และตำแหน่งของหุ่นยนต์ทั้งสองเข้าด้วยกัน

##### 4.4.1 การตรวจหาและวัดตำแหน่งหุ่นยนต์ที่ปฏิบัติร่วมกัน

เครื่องหมายบนหุ่นยนต์มีลักษณะตามรูปที่ 5.1 เป็นกระดาษสีทรงกระบอกสีเขียวสะท้อนแสง เพื่อให้สามารถแยกแยะจากสภาพแวดล้อมปรกติในอาคารได้โดยง่าย และสามารถตรวจจับได้ดีในสภาพแสงภายในอาคาร ตามรูปที่ 4.14 (ก) การทำงานแบ่งออกเป็นสองขั้นตอนซึ่งประกอบไปด้วย

- การระบุตำแหน่งเครื่องหมายหุ่นยนต์ในภาพ จากภาพมุมกว้างของกล้องออบนิซึ่งใช้การทำงานสองขั้นตอนต่อไปนี้
  - แปลงรูปแบบข้อมูลสี ของภาพมุมกว้างจาก  $RGB$  เป็น  $HSV$  เพื่อให้สามารถใช้ข้อมูลโทนสี (Hue) ในการแยกแยะสีของเครื่องหมายหุ่นยนต์ออกจากสภาพแวดล้อมได้โดยง่าย
  - ทำการแยกกลุ่มสี (Color Indexing) โดยอาศัยค่าของสีที่สนใจในการแบ่งกลุ่ม ซึ่งระบุโดยค่าสูงสุด และต่ำสุดของ โทนสี (Hue) ความอิ่มตัว (Saturation) และความสว่าง (Value) ในการแยกเครื่องหมายของหุ่นยนต์ออกจากสิ่งแวดลอม แล้วหาการเกาะกลุ่มของจุดสีทั้งหมด (Connected Component Labeling) เพื่อคำนวณตำแหน่งจากจุดศูนย์กลางมวลและได้ผลการทำงานตามรูปที่ 4.14 (ข) เพื่อนำไปคำนวณตำแหน่งของหุ่นยนต์
- การคำนวณหาตำแหน่งของหุ่นยนต์ โดยใช้วิธีการ Triangulation เช่นเดียวกับการคำนวณตำแหน่งของเส้นแนวตั้งตามรูปที่ 4.16 โดยอาศัยมุมจากขั้นตอนการระบุตำแหน่งเครื่องหมายหุ่นยนต์ในภาพได้ตามสมการที่ 4.4 ซึ่งกำหนดให้  $d$  เป็นระยะเคลื่อนที่เป็นเส้นตรงของหุ่นยนต์ระหว่างเวลา  $k$  และ  $k + 1$

$$\frac{\sin(\theta_1)}{L_2} = \frac{\sin(\theta_2)}{d} = \frac{\sin(\theta_3)}{L_1} \quad (4.4)$$

#### 4.4.2 วิธีการรวมข้อมูลแผนที่ของหุ่นยนต์ที่แยกกันปฏิบัติงาน

การทำงานของหุ่นยนต์แบ่งเป็นการแยกกันสำรวจพื้นที่ซึ่งอธิบายไปแล้วในหัวข้อ 4.3.3 และการทำงานร่วมกันของหุ่นยนต์ในการสำรวจสถานที่ ซึ่งถูกแบ่งออกจากกันหรือกว้างเกินกว่าระยะการทำงานของกล้องออปติคัล และส่งผลให้หุ่นยนต์เพียงหนึ่งตัวไม่สามารถทำงานได้อย่างครอบคลุม ดังตัวอย่างในรูปที่ 4.16 ซึ่งแสดงให้เห็นว่าสถานที่ทำงานถูกแบ่งออกเป็นสองส่วนด้วยแนวกำแพงตรงกลาง ซึ่งสามารถนำข้อมูลแผนที่จากหุ่นยนต์ทั้งสองมารวมกันเป็นแผนที่ของสถานที่ทั้งหมดได้โดยอาศัยการวัดตำแหน่งของกันและกัน

ในการทำงาน เมื่อหุ่นยนต์ตรวจพบกันเองในระหว่างปฏิบัติงาน หุ่นยนต์ทั้งสองทำการยืนยันว่า

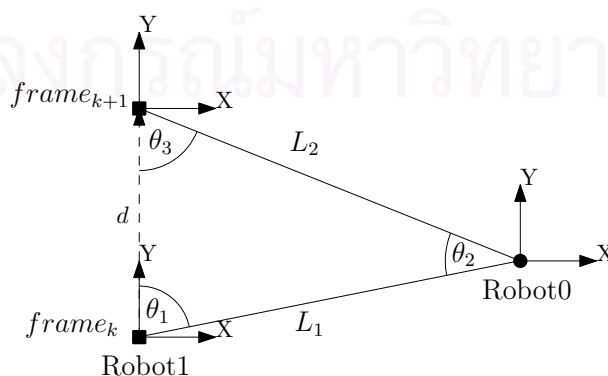


(ก) ภาพเครื่องหมายหุ่นยนต์จากกล้องออปติคัล (สีเขียว)



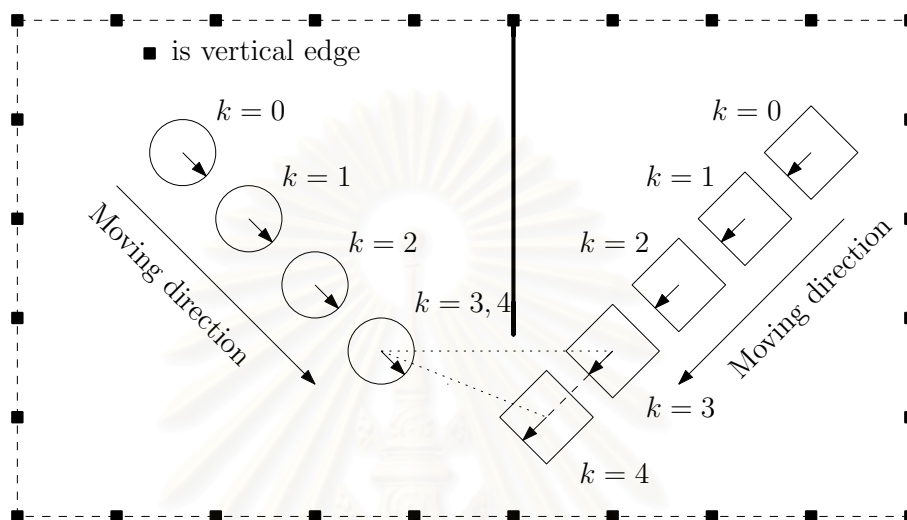
(ข) กลุ่มจากการแยกกลุ่มด้วยช่วง HSV ซึ่งกำหนดตามสีของเครื่องหมายหุ่นยนต์ และตำแหน่งที่คำนวณจากจุดศูนย์กลางมวลของกลุ่มสีที่ชัดเจนที่สุด (วงกลมสีเขียว)

รูปที่ 4.14: ลักษณะของเครื่องหมายหุ่นยนต์และตำแหน่งจากการคำนวณ



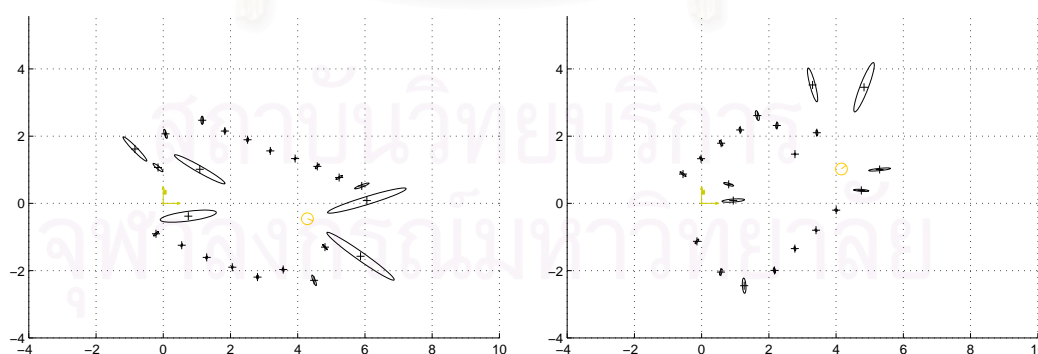
รูปที่ 4.15: การคำนวณตำแหน่งของหุ่นยนต์โดยอาศัยมุมและระยะเคลื่อนที่ของหุ่นยนต์

ตรวจพบหุ่นยนต์อีกตัวหรือไม่ หลังจากนั้นหุ่นยนต์ตัวที่หนึ่งหยุดรอหุ่นยนต์ตัวที่สองเคลื่อนที่ เพื่อทำการวัดตำแหน่งเทียบกับตำแหน่งของตัวเอง เช่นเดียวกับหุ่นยนต์ตัวที่สองซึ่งจะวัดตำแหน่งหุ่นยนต์ตัวที่หนึ่งเมื่อตัวเองเคลื่อนที่ไป ขั้นตอนดังกล่าวสามารถอธิบายได้ตามรูปที่ 4.16 หลังจากการวัดตำแหน่งของหุ่นยนต์ที่ร่วมปฏิบัติงาน หุ่นยนต์ทั้งสองทำการแลกเปลี่ยนข้อมูลแผนที่ โดยการแปลงพิกัดของจุดสังเกตในแผนที่ของหุ่นยนต์อีกตัว ไปเป็นข้อมูลในระบบพิกัดของตัวเอง ด้วยตำแหน่งและทิศทางของหุ่นยนต์ที่วัดได้



รูปที่ 4.16: ภาพแสดงการทำงานร่วมกันในการระบุตำแหน่งและสร้างแผนที่ ณ เวลา  $k = 4$  หุ่นยนต์พบกันเป็นครั้งแรก หุ่นยนต์ตัวที่หนึ่งหยุดรอตัวที่สองเคลื่อนที่ต่อไปที่เวลา  $k = 5$  เพื่อทำการวัดตำแหน่งแล้วจึงปรับปรุงข้อมูลแผนที่ร่วมกัน

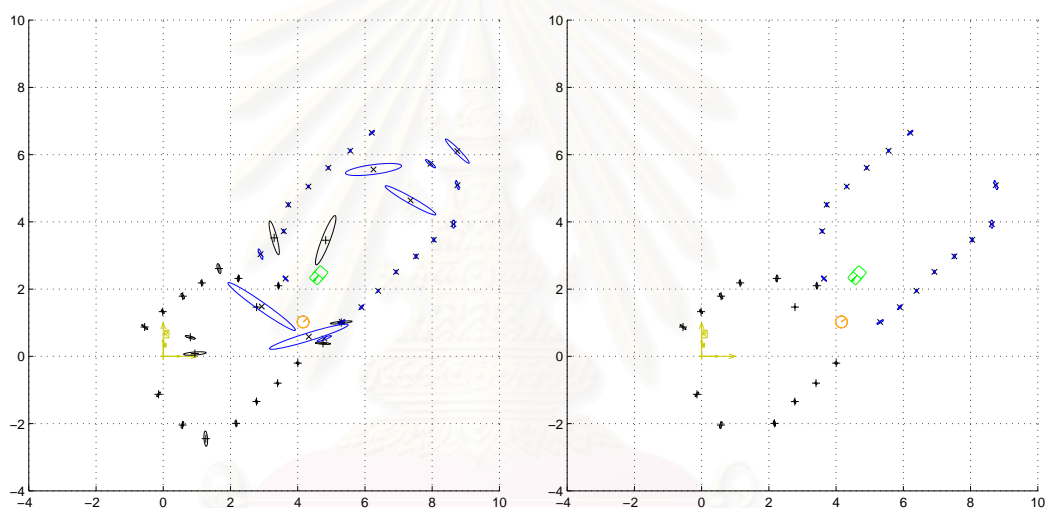
ตัวอย่างการรวมข้อมูลแผนที่จากหุ่นยนต์ทั้งสองซึ่งมีลักษณะดังรูปที่ 4.17 (ก) และ (ข) เมื่อทำการรวมแผนที่เข้าด้วยกันโดยอาศัยทิศทางของหุ่นยนต์ที่วัดได้โดยเครื่องหมายบนตัวหุ่นยนต์ได้ผลตามรูปที่ 5.24 (ก) หลังจากนั้นทำการกำจัดจุดสังเกตที่มีความแปรปรวนของตำแหน่งสูงออกได้ผลดังรูปที่ 5.24 (ข)



(ก) แผนที่ของหุ่นยนต์ตัวที่หนึ่ง

(ข) แผนที่ของหุ่นยนต์ตัวที่สอง

รูปที่ 4.17: แผนที่จากการแยกกับทำงานของหุ่นยนต์สองตัว



(ก) แผนที่ ก่อน กำจัด จุด สังกะสี ที่ มี ความ แปรปรวน ของ (ข) แผนที่ หลัง กำจัด จุด สังกะสี ที่ มี ความ แปรปรวน ของ ตำแหน่ง ตำแหน่ง สูง

รูปที่ 4.18: การรวมข้อมูลแผนที่ที่ขัดด้วยกันโดยอาศัยการวัดมุมของเครื่องหมายหุ่นยนต์

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

---

**Algorithm 2** OmniVeritcalEdgeFastSLAM( $S_{k-1}, u_k, z_k$ )
 

---

```

1: for  $i = 1$  to  $I$  do
2:    $x_k^{[i]} \approx P(x_k | x_{k-1}^{[i]}, u_k)$  {sample new pose}
3:    $W = 1$ ; {new weight of this particle}
4:   for  $j = 1$  to  $J$  do
5:      $w = []$ ; {temporary weight for each feature}
6:     for  $n = 1$  to  $N_{k-1}^{[i]}$  do
7:       {compute measurement likelihood}
8:        $\hat{z}_n = h(\mu_{n,k-1}^{[i]}, x_k^{[i]})$ ; {measurement prediction}
9:        $H_n = h'(\mu_{n,k-1}^{[i]}, x_k^{[i]})$ ; {compute jacobian}
10:       $Q_n = H_n^T \Sigma_{n,k-1}^{[i]} H_n + R_k$ ; {measurement covariance}
11:       $w[n] = |2\pi Q_n|^{-\frac{1}{2}} \exp(-\frac{1}{2}(z_{k,j} - \hat{z}_n)^T Q_n^{-1}(z_{k,j} - \hat{z}_n))$ ; {likelihood of coresspondence}
12:    end for
13:     $w[N_{k-1}^{[i]} + 1] = P_0$ ; {importance factor of new landmark}
14:     $[maxW, \hat{n}] = \max(w)$ ; {max likelihood coresspondence}
15:     $W = W \cdot maxW$ ; {update weight from measurement  $z_{k,j}$ }
16:     $N_k^{[i]} = \max(\hat{n}, N_{k-1}^{[i]})$ ; {new number of feature in map}
17:    if  $N_k^{[i]} = N_{k-1}^{[i]} + 1$  then
18:      {new feature}
19:       $\mu_{\hat{n},k}^{[i]} = h^{-1}(z_{k,j}, x_k^{[i]})$  {initialize meam}
20:       $\Sigma_{\hat{n},k}^{[i]} = (H_{\hat{n}}^{[i]} R^{-1} H_{\hat{n}}^{[i]T})^{-1}$ ; {initialize covariance}
21:       $cnt_{\hat{n},k}^{[i]} = 1$ ; {initialize counter}
22:       $isUpdated_{\hat{n},k}^{[i]} = 1$ ; {this is newly added feature}
23:    else
24:      {observed feature}
25:       $K = \Sigma_{\hat{n},k-1}^{[i]} H_{\hat{n}}^{[i]} Q_n^{-1}$ ; {compute Kalman gain}
26:       $\mu_{\hat{n},k}^{[i]} = \mu_{\hat{n},k-1}^{[i]} + K(z_{k,j} - \hat{z}_{\hat{n}})^T$  {update meam}
27:       $\Sigma_{\hat{n},k}^{[i]} = (I - K H_{\hat{n}}^{[i]T}) \Sigma_{\hat{n},k-1}^{[i]}$  {update covariance}
28:       $cnt_{\hat{n},k}^{[i]} = cnt_{\hat{n},k-1}^{[i]} + 1$ ; {update counter}
29:       $isUpdated_{\hat{n},k}^{[i]} = 1$ ; {this is observed feature}
30:    end if
31:  end for
32:  for  $n = 1$  to  $N_k^{[i]}$  do
33:    if  $isUpdated_{n,k}^{[i]} \neq 1$  then
34:       $\mu_{n,k}^{[i]} = \mu_{n,k-1}^{[i]}$  {use old mean}
35:       $\Sigma_{n,k}^{[i]} = \Sigma_{n,k-1}^{[i]}$  {use old covariance}
36:      if  $\mu_{n,k}^{[i]}$  is outside sensor range then
37:         $cnt_{n,k}^{[i]} = cnt_{n,k-1}^{[i]}$ ; {yes, do not decrease counter}
38:      else
39:         $cnt_{n,k}^{[i]} = cnt_{n,k-1}^{[i]} - 1$ ; {no, decrease counter}
40:        if  $cnt_{n,k}^{[i]} < 0$  then
41:          {remove doubtful feature  $n$  from particle  $i$ }
42:        end if
43:      end if
44:    end if
45:  end for
46:   $\omega_k^{[i]} = W$ ; {update particle weight}
47:   $S_k^{[i]} = S_{k-1}^{[i]}$ ; {update particle in  $S_k$ }
48: end for {should we process the resampling state}
49: if  $effectiveSampleSize(S_k) < N_{eff}$  then
50:    $S_k = resample(S_k)$ ;
51: end if

```

---



## บทที่ 5

### การทดลองและผลการทำงาน

เป้าหมายหลักของการทดลอง คือ การแสดงให้เห็นว่าข้อมูลเส้นแนวตั้งจากวิธีการที่นำเสนอสามารถนำมาใช้เป็นข้อมูลหลักในการระบุตำแหน่งและสร้างแผนที่ของหุ่นยนต์ได้จริง ด้วยการทดลองในสภาพแวดล้อมภายในอาคาร รูปแบบการนำเสนอข้อมูลจากการทดลอง เป็นการนำเสนอในรูปแบบการอธิบายการทดลองร่วมกับข้อมูลภาพ ซึ่งได้จากข้อมูลการระบุตำแหน่งและสร้างแผนที่ โดยไม่เน้นการอธิบายข้อมูลในเชิงสถิติและประสิทธิภาพในเชิงเวลาของการสร้างแผนที่ เนื่องจากการประยุกต์ใช้งานระเบียบวิธีการ FastSLAM ซึ่งมีการทดลองและรายงานผลการเปรียบเทียบกับวิธีการอื่นไว้อย่างชัดเจนแล้วในงานวิจัย [44]

การทดลองทำโดยการเก็บข้อมูลภาพจากกล้องออปติคัลและข้อมูลการสั่งงานหุ่นยนต์ ด้วยการบังคับหุ่นยนต์ผ่านโปรแกรมควบคุม ให้เคลื่อนที่ไปในห้องแลปวิจัย แล้วนำข้อมูลมาประมวลผลด้วยโปรแกรมสำหรับแยกและคำนวณตำแหน่งของเส้นแนวตั้ง ซึ่งเขียนขึ้นด้วยภาษา C++ ร่วมกับ OpenCV [56] เพื่อนำข้อมูลไปใช้ในการระบุตำแหน่งและสร้างแผนที่ของหุ่นยนต์ซึ่งเขียนด้วย Matlab [50] โดยอาศัยฟังก์ชันการแสดงผลจาก Toolbox [57] และแนวทางในการออกแบบระเบียบวิธีการ FastSLAM บน Matlab จากโปรแกรมจำลองการทำงานของ SLAM โดย [58]

#### 5.1 อุปกรณ์สำหรับการทดลอง

##### 5.1.1 หุ่นยนต์

เป็นหุ่นยนต์สำหรับงานวิจัยรุ่น Pioneer DX3 ของ [59] จำนวนสองตัวซึ่งมีลักษณะดังรูปที่ 5.1 ใช้การขับเคลื่อนแบบสองล้ออิสระ ตัวหุ่นยนต์มีขนาด  $44.5\text{cm} \times 40.0\text{cm}$  สามารถติดตั้งอุปกรณ์เพิ่มเติมได้ 23 กิโลกรัม บนตัวหุ่นยนต์มีคอมพิวเตอร์รุ่น Pentium III ความเร็ว 1 GHz หน่วยความจำ 256 MB ระบบปฏิบัติการ WindowsXP ซึ่งควบคุมผ่าน Wireless LAN ความเร็ว 11 MB/s และมีการติดตั้งเครื่องหมายซึ่งทำจากกระดาษทรงกระบอกสี่เหลี่ยมบนกล้องออปติคัล เพื่อใช้ในการบ่งบอกตำแหน่งของหุ่นยนต์ที่ทำงานร่วมกัน

##### 5.1.2 กล้องออปติคัล

กล้องออปติคัลในงานวิจัยเป็นกล้อง รุ่น OmniView360 ของ RemoteReality [49] ซึ่งเดิมติดตั้งเข้ากับตัวหุ่นยนต์โดยตรง ทำให้แนวแกนของกล้องไม่ตั้งฉากกับระนาบของการเคลื่อนที่ ส่งผลให้เส้นแนวตั้งในภาพซึ่งเป็นส่วนสำคัญในการหาตำแหน่งของเส้นแนวตั้งในงานวิจัยชิ้นนี้ไม่ปรากฏในแนวรัศมี จึงต้องแก้ไขด้วยการออกแบบแผ่นยึดซึ่งสามารถปรับระดับและมุมของกล้องได้ตามรูปที่ 5.2 เพื่อใช้ในการปรับแต่งทิศทางและตำแหน่งของกล้องให้เหมาะสม

## 5.2 สถานที่ทดลอง

การทดลองทำในห้องแลปวิจัยขนาดประมาณ  $12m \times 5m$  ตามแผนผังในรูป 5.3 ซึ่งประกอบด้วยฉากกั้นห้องและโต๊ะทำงานตามรูปที่ 5.4 (ก) และ (ข) ในการทดลองมีการแก้ไขเส้นแนวตั้งเดิมด้วยแถบกระดาษสีขาว เนื่องจากความสว่างไม่เพียงพอสำหรับการทำงานของกล้องอ้อมนิตามรูปที่ 5.4 (ค) รวมถึงการเพิ่มเส้นแนวตั้งบริเวณแนวกำแพงยาวเพราะต้องการแสดงลักษณะของห้องในผลการทำงานให้ชัดเจนขึ้นตามรูปที่ 5.4 (ง) การทดลองทั้งหมดใช้การตรวจจับเส้นแนวตั้งโดยอาศัยเส้นขอบแนวตั้งที่ตรวจพบในภาพ โดยไม่ใช่ข้อมูลส่วนเพิ่มซึ่งเกิดจากแถบกระดาษที่ติดตั้งเพิ่มเข้าไป เช่น สีและความยาวของแถบกระดาษ ในการทำงาน



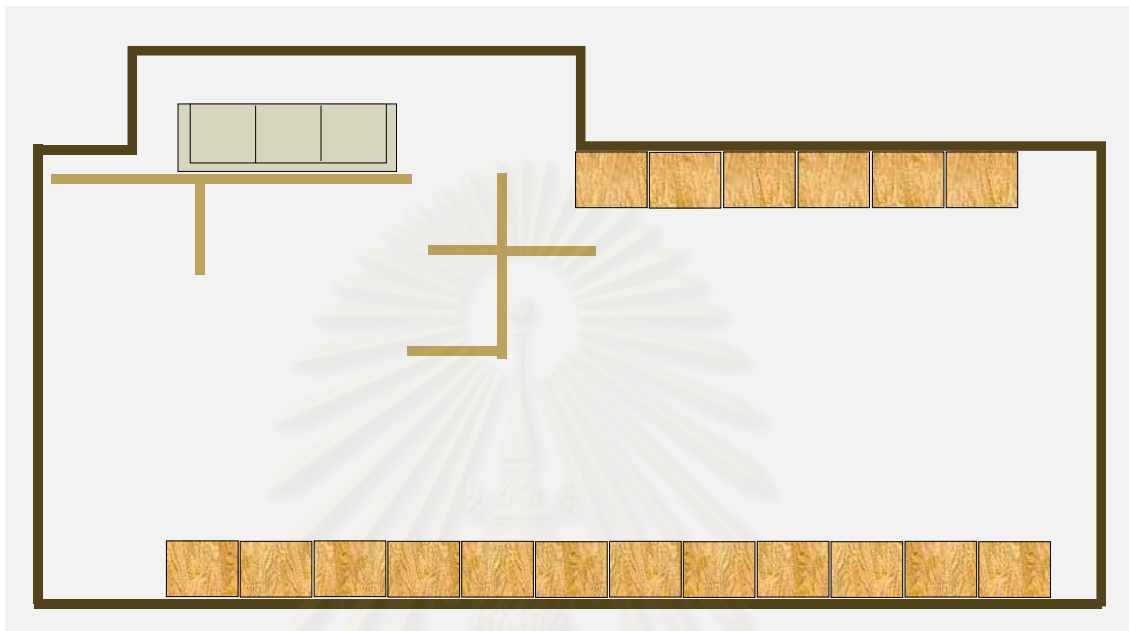
รูปที่ 5.1: หุ่นยนต์ทั้งสองตัวที่ใช้ในการทดลอง



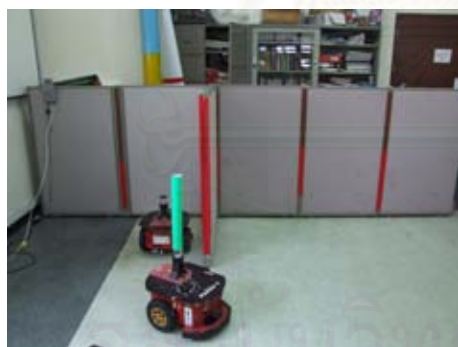
รูปที่ 5.2: แผ่นยึดกล้องอ้อมนิปรับระดับได้

### 5.3 การทดลองระบุตำแหน่งและสร้างแผนที่ด้วยหุ่นยนต์หนึ่งตัว

การทดลองในส่วนนี้เป็นการบังคับหุ่นยนต์ให้เคลื่อนที่ไปรอบสถานที่ทดลอง เพื่อทดสอบว่าข้อมูลตำแหน่งของเส้นแนวตั้งจากกล้องออบนิซึ่งมีความแม่นยำเชิงมุมสูง แต่ขาดความแน่นอนในการวัดระยะ



รูปที่ 5.3: แผนผังสถานที่ทดลองบนห้องแลป ISL2



(ก) ผงกั้นห้อง



(ข) โต๊ะคอมพิวเตอร์



(ค) แถบสีขาวที่ติดตั้งเพิ่ม

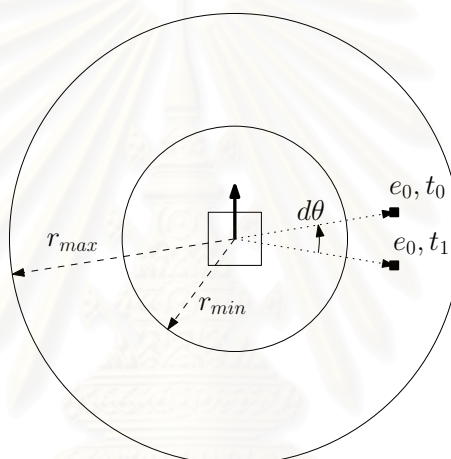


(ง) เส้นแนวตั้งเพิ่มเติมบนแนวกำแพง

รูปที่ 5.4: รายละเอียดของสถานที่ทดลอง

ตารางที่ 5.1: พารามิเตอร์ของโปรแกรมหาและระบุตำแหน่งเส้นแนวตั้ง

พารามิเตอร์	ค่าในการทดลอง	หน่วย	ผลต่อการทำงาน
$r_{max}$	2, 3, 4	$m$	จำนวน เส้น แนว ตั้ง ที่ ตรวจ พบ ใน รัศมีการทำงานของกล้องอ้อมนิ
$r_{min}$	0.4	$m$	เช่นเดียวกับ $r_{max}$
$d\theta$	$\pi/36$	$rad$	ความถูกต้องในการจับคู่เส้นแนวตั้ง เพื่อคำนวณตำแหน่ง
$d$ จากสมการ 4.1	0.1	$m$	ความ ถูกต้อง ของ การ หา ตำแหน่ง เส้นแนวตั้งตามหัวข้อ 4.2
$\beta$ จากสมการ 4.1	5	$deg$	ความ ถูกต้อง ของ การ ระบุ ตำแหน่ง และสร้างแผนที่ เนื่องจากไม่มีการ วัดตำแหน่งและติดตามเส้นแนวตั้ง เมื่อหุ่นยนต์หมุนตัวตามหัวข้อ 4.2



รูปที่ 5.5: พารามิเตอร์ที่มีผลต่อข้อมูลการวัดตำแหน่งเส้นแนวตั้ง

ทาง สามารถนำมาใช้เป็นจุดสังเกตในวิธีการ FastSLAM ได้จริง และการเคลื่อนที่แบบวนกลับมาจุดเดิม ในสถานที่ทดลองยังเป็นการทดสอบว่าข้อมูลเส้นแนวตั้งสามารถแก้ปัญหาการ Close Loop ซึ่งเป็นปัญหาหลักของการระบุตำแหน่งพร้อมการสร้างแผนที่ [60] ซึ่งต้องการความถูกต้องของข้อมูลสูงได้

ข้อมูลสำหรับการทดลองเป็นข้อมูลชุดเดียวกันทั้งหมดจากการบังคับหุ่นยนต์ให้เคลื่อนที่รอบสถานที่ทดลอง ความแตกต่างของแต่ละการทดลอง คือ ค่าพารามิเตอร์ของโปรแกรม ซึ่งประกอบด้วยพารามิเตอร์ของโปรแกรมหาและระบุตำแหน่งเส้นแนวตั้งที่มีคุณสมบัติตามตาราง 5.1 และรูป 5.5 และพารามิเตอร์ของโปรแกรม FastSLAM ซึ่งมีพารามิเตอร์ที่มีผลต่อการทำงานตามตารางที่ 5.2

วิธีการแสดงให้เห็นถึงความแตกต่างของผลการทำงานซึ่งเกิดจากพารามิเตอร์แต่ละตัว เป็นการเปรียบเทียบการทำงานกับข้อมูลการระบุตำแหน่งและสร้างแผนที่ โดยใช้ค่าพารามิเตอร์ซึ่งทดสอบแล้วให้ผลดีกับข้อมูลชุดนั้น ตามตารางที่ 5.3 โดยมีผลการทำงานตามรูปที่ 5.6 เป็นข้อมูลอ้างอิง นอกจากนี้ การทดลองเป็นการทดลองเพื่อให้เห็นถึงความเป็นไปได้ในการใช้ข้อมูลจากกล้องอ้อมนิเพียงอย่างเดียว ในการสร้างแผนที่ จึงทำการเปรียบเทียบแผนที่จากกระบวนการระบุตำแหน่งพร้อมสร้างแผนที่กับแผนผังห้องและเปรียบเทียบตำแหน่งของหุ่นยนต์โดยอาศัยตำแหน่งของกระเบื้องบนพื้นในสถานที่ทดลองตามรูป

ตารางที่ 5.2: พารามิเตอร์ของโปรแกรมระบุตำแหน่งและสร้างแผนที่

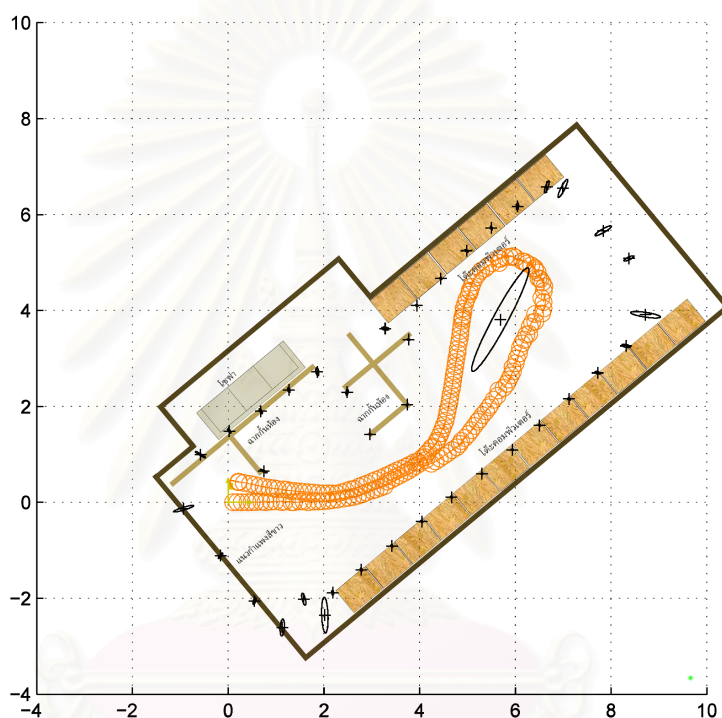
พารามิเตอร์	ค่าในการทดลอง	หน่วย	ผลต่อการทำงาน
$\sigma_{tran}$	5	cm/m	มีผลตามหัวข้อ 4.3.1
$\sigma_{rot}$	5	deg/ $\pi$	มีผลตามหัวข้อ 4.3.1
$\sigma_{drift}$	5	deg/ $\pi$	มีผลตามหัวข้อ 4.3.1
$\sigma_r$	0.5	m	มีผลต่อความถูกต้องในการสร้างแผนที่และระบุตำแหน่ง เมื่อมีค่าน้อย ความเชื่อมั่นในตำแหน่งของหุ่นยนต์และจุดสังเกตจะเปลี่ยนแปลงตามค่าที่วัดได้อย่างรวดเร็ว ซึ่งอาจนำไปสู่ความผิดพลาดในการทำงานได้ง่าย เมื่อความคลาดเคลื่อนจริงของเซนเซอร์แตกต่างจากค่าพารามิเตอร์มาก
$\sigma_{theta}$	1.5	deg	ให้ผลเช่นเดียวกับ $\sigma_r$
$I$	20, 50, 100, 200	-	จำนวนที่น้อยเกินไป มีผลต่อความถูกต้องในการประมาณ PDF ของระบบ และจำนวนที่มากขึ้นส่งผลโดยตรงต่อเวลาในการประมวลผล
$Max_{range}$	เปลี่ยนตามค่า $r_{max}$ ในตาราง 5.1	m	เป็นค่าสำหรับใช้ในการกำจัดจุดสังเกตที่เกิดจากความผิดพลาดในการวัดระยะ เช่น การจับคู่เส้นแนวตั้งเพื่อกำหนดตำแหน่งผิด โดยตั้งอยู่บนพื้นฐานว่าจุดสังเกตที่อยู่ในระยะทำงานของเซนเซอร์ควรถูกตรวจวัดทุกครั้ง แต่จุดสังเกตที่เกิดจากความผิดพลาดควรหายไปเมื่อไม่สามารถตรวจวัดได้



ตารางที่ 5.3: พารามิเตอร์ของแผนที่อ้างอิงจากการทดลอง

$r_{max}$	$r_{min}$	$d\theta$	$d$	$\beta$
3m	0.4m	$\pi/36$	0.1m	$5^\circ$
$\sigma_{tran}$	$\sigma_{drift}$	$\sigma_{rot}$	$\sigma_r$	$\sigma_{theta}$
5cm/m	$5^\circ/m$	$5^\circ/\pi$	0.5m	$1.5^\circ$
$I$	$Max_{range}$			
200	3m	-	-	-

ที่ 5.7 (ก) และ 5.7 (ข)



รูปที่ 5.6: ภาพแสดงตำแหน่งของหุ่นยนต์และแผนที่จากการทดลองเทียบกับแผนที่ของสถานที่ทดลอง

1. ทดสอบปรับค่าความเชื่อมั่นของการวัดตำแหน่งเส้นแนวตั้ง เพื่อสอบความทนทานต่อความผิดพลาดของโมเดลการวัดตำแหน่งของเส้นแนวตั้งในสิ่งแวดล้อม เมื่อข้อมูลจริงมีความถูกต้องเพียงพอต่อการทำงาน โดยผลการทดลองเป็นไปตามรูปที่ 5.8 ถึง 5.10 และมีรายละเอียดของผลการทดลองดังนี้

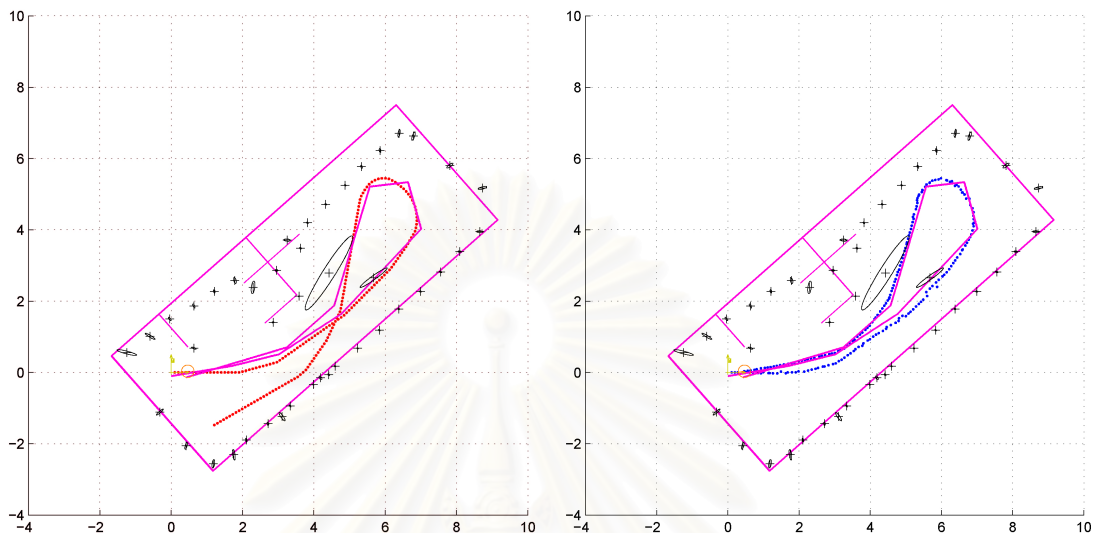
รูป 5.8 กำหนดให้การวัดตำแหน่งมีความคลาดเคลื่อนต่ำมาก ทั้งระยะทางและมุม ซึ่งผิดไปจากความเป็นจริงของเซนเซอร์ที่มีความผิดพลาดของการวัดระยะทางสูงแต่มีความแม่นยำในการวัดมุม เป็นผลให้การระบุตำแหน่งและสร้างแผนที่ล้มเหลวโดยสิ้นเชิง

รูปที่ 5.9 กำหนดให้ความคลาดเคลื่อนของการวัดระยะทางผิดไปจากความเป็นจริง ปรากฏว่าไม่สามารถทำการระบุตำแหน่งและสร้างแผนที่ได้อย่างถูกต้อง

รูปที่ 5.10 กำหนดให้ความคลาดเคลื่อนในการวัดมุมให้มากกว่าความเป็นจริง หุ่นยนต์สามารถระบุตำแหน่งในสภาพแวดล้อมได้ แต่ไม่สามารถให้ตำแหน่งที่ถูกต้องของจุดสังเกตได้

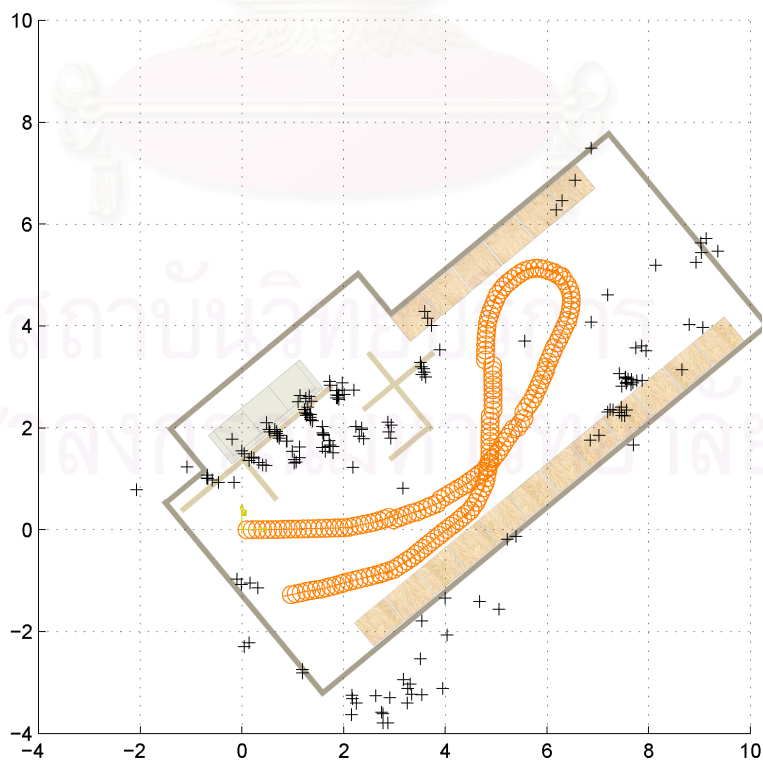
ผลการทดลองจึงแสดงให้เห็นอย่างชัดเจนว่าความผิดพลาดของโมเดลการวัดตำแหน่งที่สูงหรือต่ำเกินไป มีผลต่อการระบุตำแหน่งและสร้างแผนที่เป็นอย่างมาก

## 2. ทดสอบปรับค่าความเชื่อมั่นของการควบคุมหุ่นยนต์ จากการทดลองในรูปที่ 5.11 ถึง 5.13



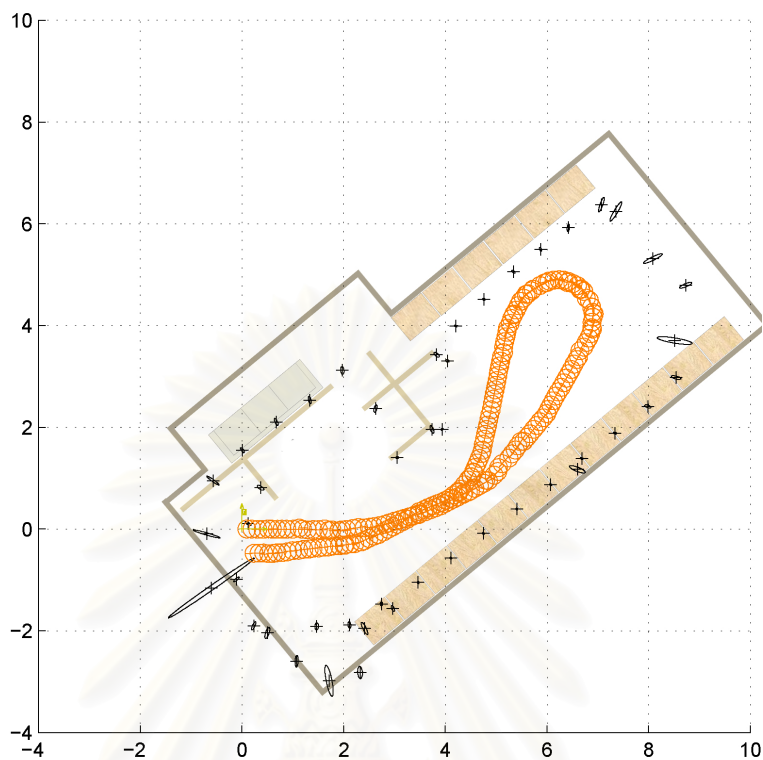
(ก) เส้นทางของหุ่นยนต์จากการวัดตำแหน่งของแผนกระเบื้อง (ข) เส้นทางของหุ่นยนต์จากการวัดตำแหน่งของแผนกระเบื้องและเส้นทางจากการระบุตำแหน่งพร้อมสร้างแผนที่

รูปที่ 5.7: ภาพเปรียบเทียบเส้นทางการเคลื่อนที่ของหุ่นยนต์กับตำแหน่งจริงซึ่งประมาณจากตำแหน่งของหุ่นยนต์บนกระเบื้อง

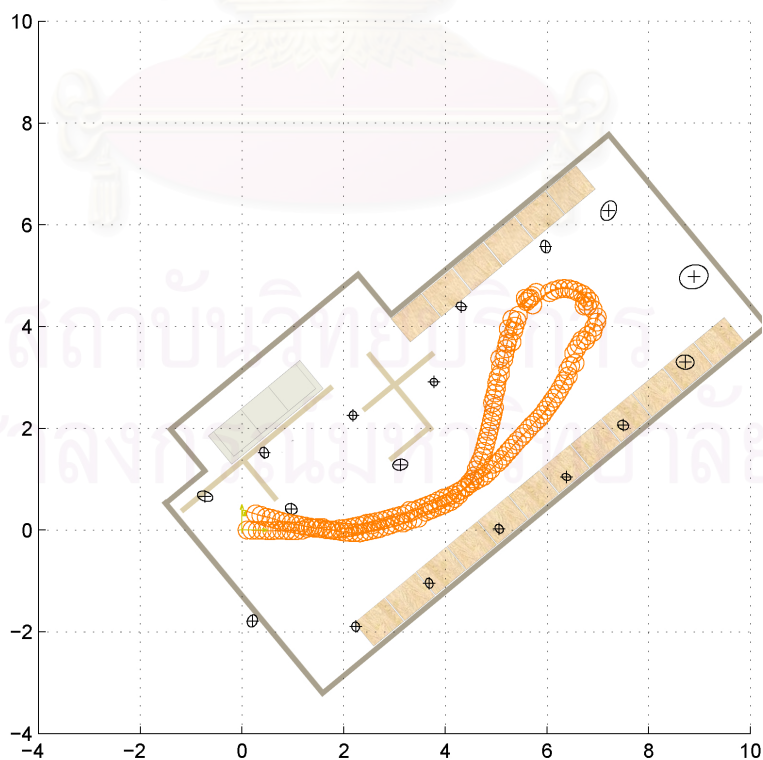


รูปที่ 5.8: กำหนดให้โมเดลการวัดตำแหน่งมี  $\sigma_r = 0.001m, \sigma_{\theta} = 0.001^\circ$

แสดงให้เห็นว่าโมเดลการควบคุมหุ่นยนต์ที่ผิดพลาด มีผลต่อการทำงานของ FastSLAM ในสอง



รูปที่ 5.9: กำหนดให้โมเดลการวัดตำแหน่งมี  $\sigma_r = 1m, \sigma_{theta} = 1.5^\circ$



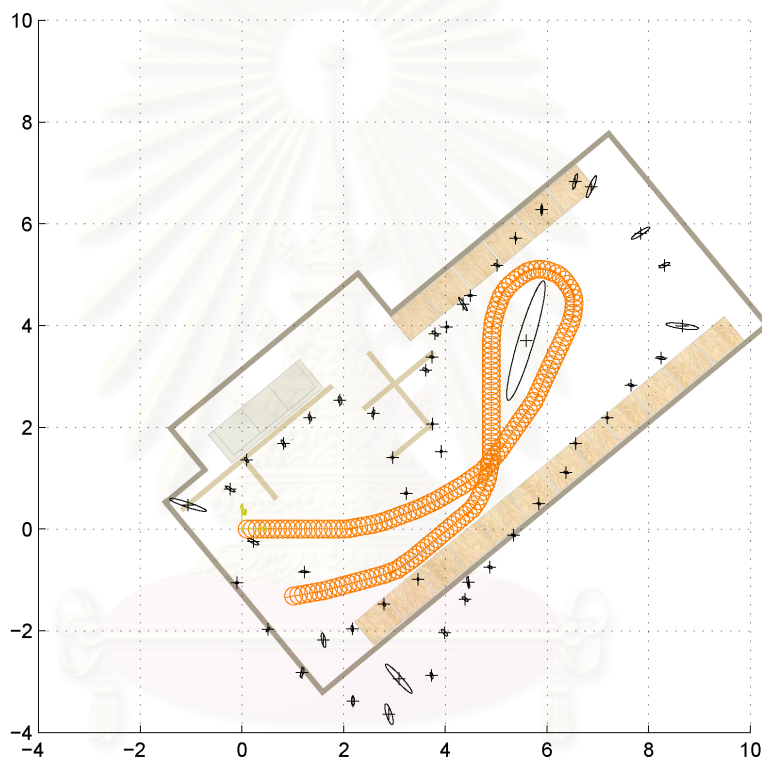
รูปที่ 5.10: กำหนดให้โมเดลการวัดตำแหน่งมี  $\sigma_r = 0.5m, \sigma_{theta} = 10^\circ$

ลักษณะ

รูปที่ 5.11 แสดงให้เห็นว่าความเชื่อมั่นที่มากเกินไปต่อการควบคุม ส่งผลต่อความถูกต้องในการทำงานเป็นอย่างมาก

รูปที่ 5.13 กำหนดให้โมเดลการควบคุมมีความผิดพลาดมาก กลับประสบความสำเร็จในการระบุตำแหน่งและสร้างแผนที่ ซึ่งให้ผลการทำงานไม่ต่างจากรูปที่ 5.12 ซึ่งเป็นผลการทำงานโดยโมเดลการเคลื่อนที่จากการทดลอง

ผลการทดลองแสดงให้เห็นว่าข้อมูลตำแหน่งเส้นแนวตั้งจากกล้องออปติ มีความแม่นยำเพียงพอสำหรับงานระบุตำแหน่งและสร้างแผนที่ เมื่อไม่สามารถหาโมเดลความผิดพลาดของการควบคุมที่ชัดเจนได้



รูปที่ 5.11: กำหนดให้โมเดลการเคลื่อนที่มี  $\sigma_{tran} = 0m, \sigma_{drift} = 0^\circ/m, \sigma_{rot} = 0^\circ/\pi$

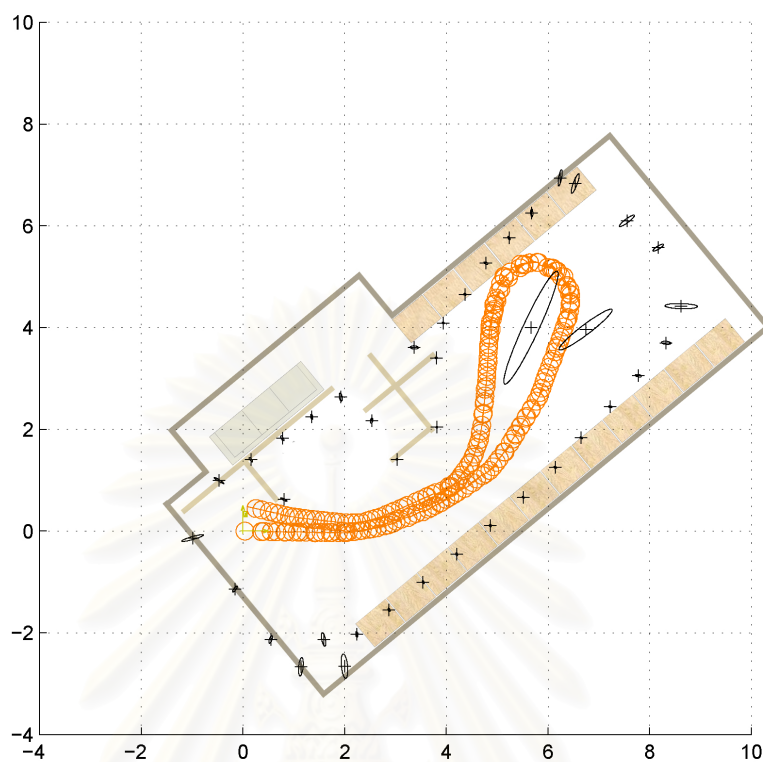
- ทดสอบปรับจำนวนตัวอย่างใน Particle Filter เพื่อทดสอบผลการทำงานของ FastSLAM ในเรื่องของจำนวนตัวอย่างของ Particle Filter ต่อผลการทำงานของ FastSLAM เมื่อนำมาใช้ร่วมกับข้อมูลจากกล้องออปติ ซึ่งมีความผิดพลาดของการวัดระยะทางสูงแต่มีความแม่นยำในการวัดทิศทาง

รูปที่ 5.14 แสดงให้เห็นว่าการทำงานเมื่อใช้ 10 ตัวอย่างไม่ประสบความสำเร็จในการระบุตำแหน่งและสร้างแผนที่

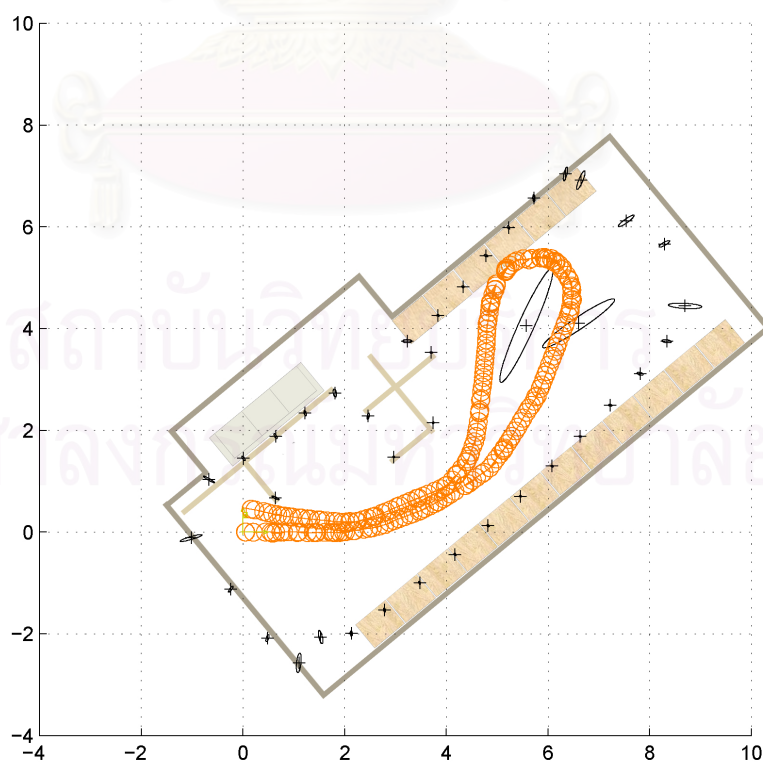
รูปที่ 5.15 ถึง 5.18 ใช้จำนวนตัวอย่างตั้งแต่ 20 ถึง 200 ตัวอย่างในการทำงาน สามารถทำการระบุตำแหน่งและสร้างแผนที่ได้โดยไม่มีความแตกต่างในระดับที่สังเกตได้

การทดลองแสดงให้เห็นว่าข้อมูลเส้นแนวตั้งจากกล้องออปติ ให้ผลการระบุตำแหน่งและสร้างแผนที่ใน FastSLAM ด้วยแนวโน้มเดียวกับเซนเซอร์ราคาสูง เช่น เลเซอร์ ในการทดลองของ [44]

4. ทดสอบระยะทำงานของเซนเซอร์ เพื่อทดสอบระยะการทำงานของเซนเซอร์หรือระยะไกลสุดของ



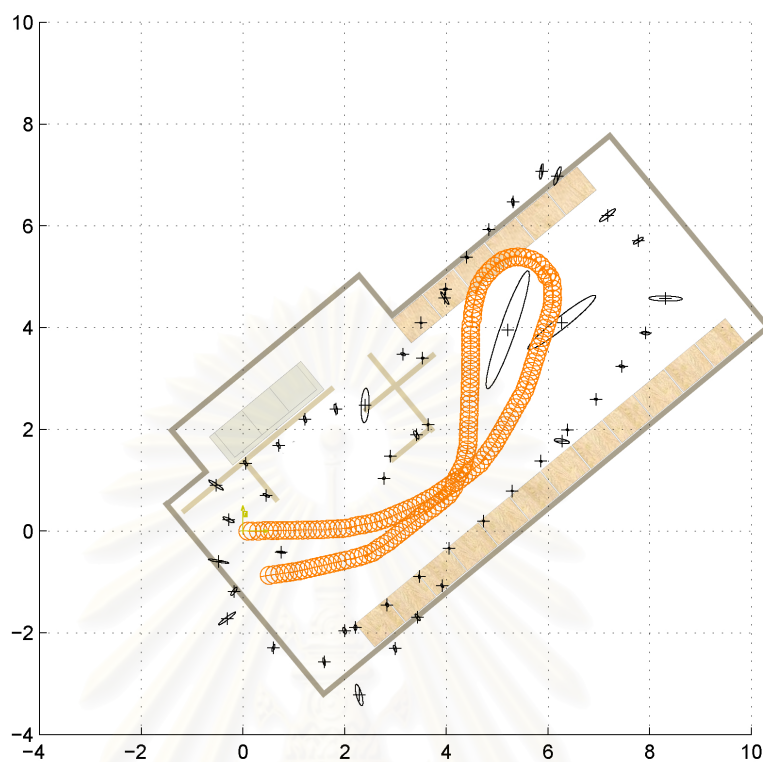
รูปที่ 5.12: กำหนดให้โมเดลการเคลื่อนที่มี  $\sigma_{tran} = 0.5m, \sigma_{drift} = 5^\circ/m, \sigma_{rot} = 5^\circ/\pi$



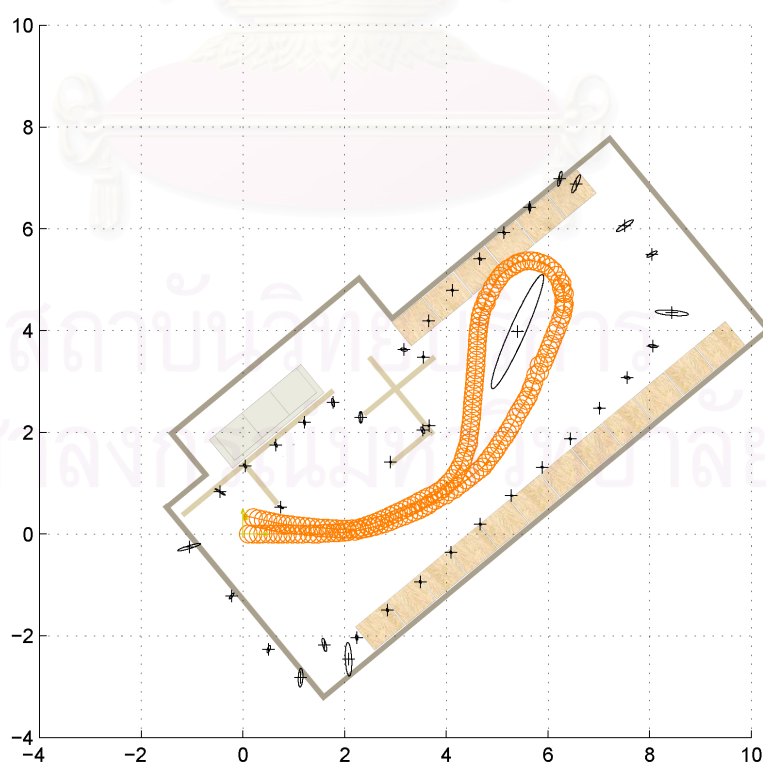
รูปที่ 5.13: กำหนดให้โมเดลการเคลื่อนที่มี  $\sigma_{tran} = 1m, \sigma_{drift} = 20^\circ/m, \sigma_{rot} = 20^\circ/\pi$



ตำแหน่งเส้นแนวตั้ง ที่นำมาใช้งานว่ามีผลต่อการระบุตำแหน่งและสร้างแผนที่อย่างไร

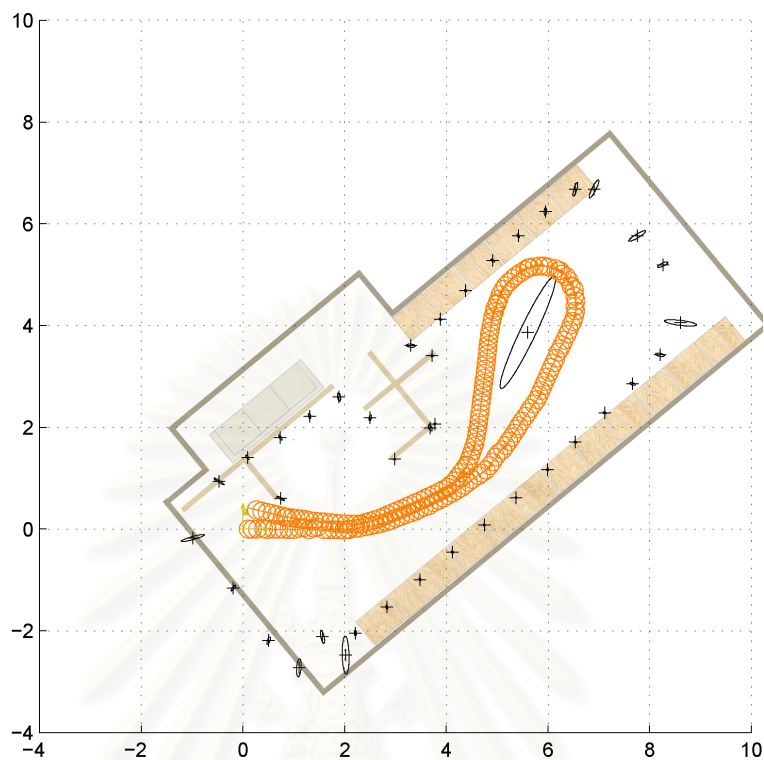


รูปที่ 5.14: กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น  $I = 10$

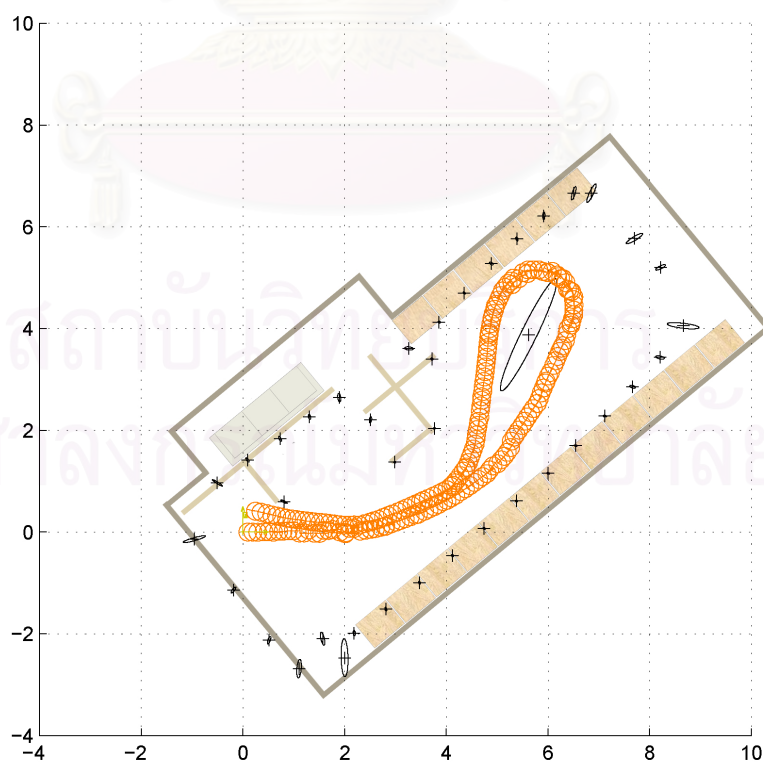


รูปที่ 5.15: กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น  $I = 20$

รูปที่ 5.19 เมื่อกำหนดให้ระยะทำงานของเซนเซอร์เป็นสองเมตร ไม่สามารถทำการระบุตำแหน่งและ



รูปที่ 5.16: กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น  $I = 50$



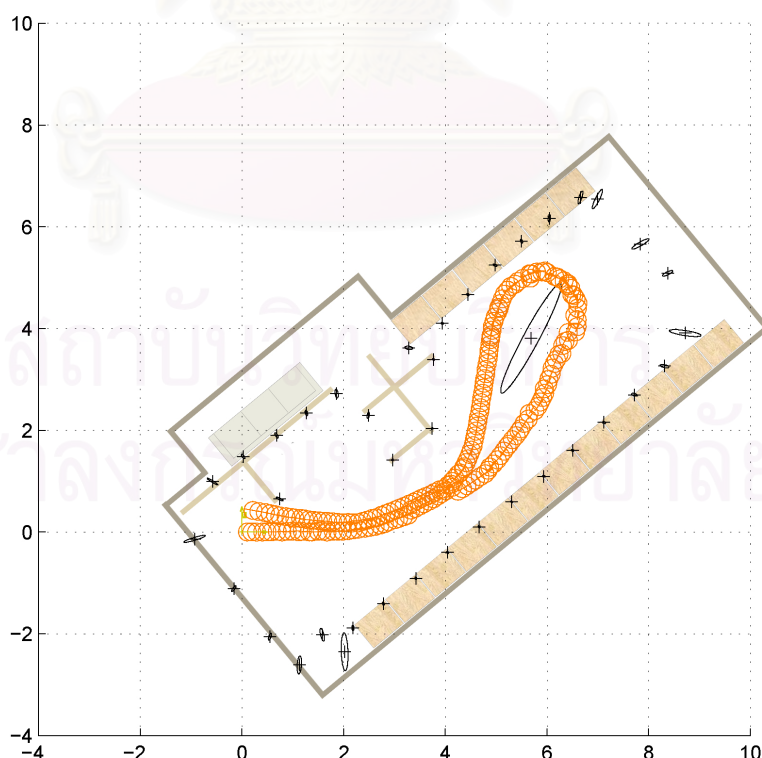
รูปที่ 5.17: กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น  $I = 100$

สร้างแผนที่ได้เนื่องจากระยะห่างของเส้นแนวตั้งโดยเฉลี่ยจากตำแหน่งที่หุ่นยนต์อยู่มีระยะทางมากกว่าสองเมตร ทำให้การระบุตำแหน่งในหลายจุดไม่สามารถทำได้ เนื่องจากข้อมูลการวัดตำแหน่งจุดสังเกตไม่เพียงพอ เป็นผลให้ความเชื่อมั่นในตำแหน่งของหุ่นยนต์ผิดไปเป็นอย่างมากในหลายจุดตามรูปที่ 5.19 (ข)

รูปที่ 5.21 กำหนดให้ ระยะ การทำงาน ของ เซนเซอร์ มี ระยะ ทางสี่ เมตร ที่ ระยะ ดังกล่าว การเปลี่ยนแปลงมุมของเส้นแนวตั้งในสิ่งแวดล้อมเมื่อหุ่นยนต์เคลื่อนที่ต่ำมาก เทียบกับความละเอียดของภาพจากกล้องออปติ เป็นผลให้เกิดความผิดพลาดในการวัดตำแหน่งเส้นแนวตั้งในด้านไกลเมื่อหุ่นยนต์เคลื่อนจากตำแหน่งเริ่มต้นบริเวณตำแหน่ง (0, 0) ไปยังด้านไกลบริเวณตำแหน่ง (6, 4) ส่งผลให้แนวเส้นแนวตั้งบริเวณ (6, 6) เบนออกจากแนวที่ควรจะเป็น และจุดสังเกตบางจุดหายไป เนื่องจากความผิดพลาดของการวัดระยะห่างของเส้นแนวตั้ง

ผลการทดลองชี้ให้เห็นข้อจำกัดของระยะการทำงานของกล้องออปติ โดยเฉพาะในการใช้สำหรับแยกแยะข้อมูลซึ่งมีลักษณะใกล้เคียงกันในระยะไกล เนื่องจากความละเอียดของตัวกล้อง และลักษณะของข้อมูลในหัวข้อ 2.21

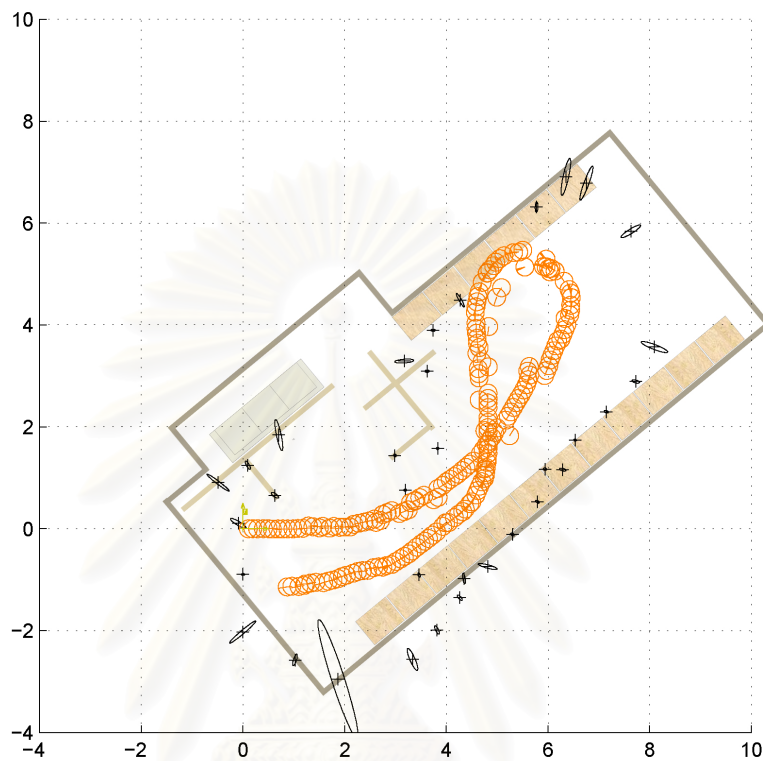
5. ทดสอบการกำจัดจุดสังเกตซึ่งเกิดจากความผิดพลาดในการวัดตำแหน่งเส้นแนวตั้ง เป็นการทดสอบความจำเป็นในการกำจัดข้อมูลการวัดตำแหน่งที่ผิดพลาดของเส้นแนวตั้งซึ่งเกิดขึ้นได้ง่ายเมื่อเส้นแนวตั้งที่ตรวจพบเป็นเส้นที่อยู่ในระยะไกลและอยู่ติดกันเป็นจำนวนมาก การทดลองแสดงให้เห็นว่าการไม่กำจัดข้อมูลดังกล่าวออกจากกระบวนการระบุตำแหน่งและสร้างแผนที่ ส่งผลให้ประสิทธิภาพการระบุตำแหน่งและสร้างแผนที่ตามรูปที่ 5.22



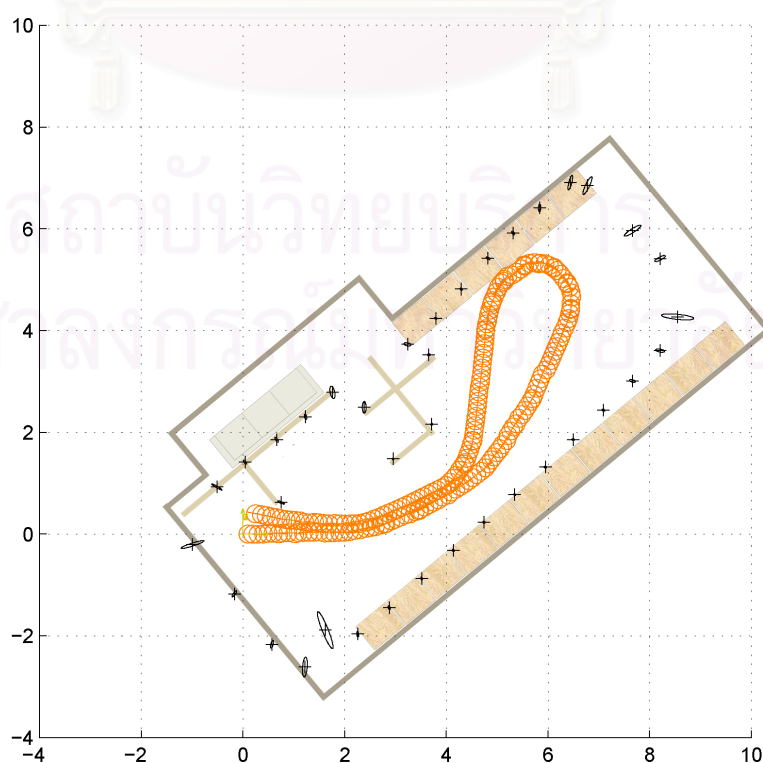
รูปที่ 5.18: กำหนดให้จำนวนตัวอย่างใน Particle Filter เป็น  $I = 200$

#### 5.4 การทดลองระบุตำแหน่งและสร้างแผนที่โดยการทำงานร่วมกันของหุ่นยนต์สองตัว

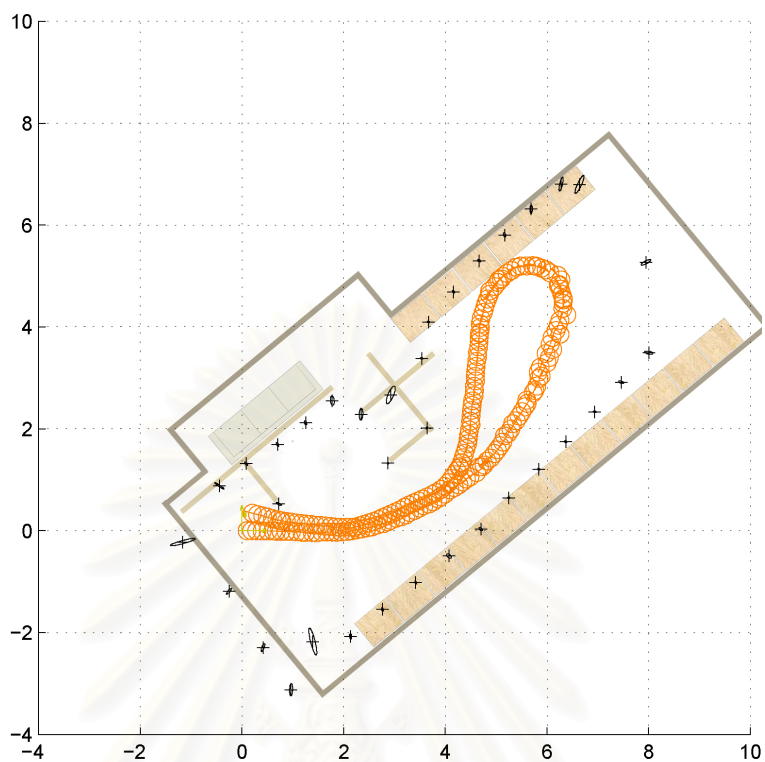
การทดลองในส่วนนี้ต้องการแสดงให้เห็นขั้นตอนการรวมแผนที่จากหุ่นยนต์ซึ่งทำงานแยกกันในสถานที่ทดลอง ในการทดลองหุ่นยนต์ทั้งสองเคลื่อนที่จากคนละฟากของสถานที่ทดลองแล้วมาพบกัน



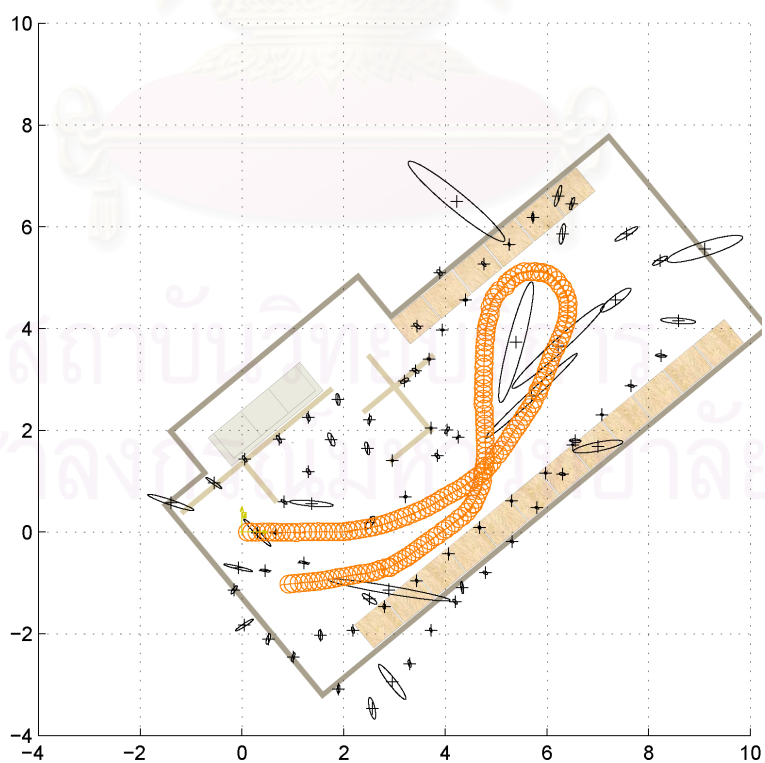
รูปที่ 5.19: กำหนดให้ระยะทำงานของเซนเซอร์เป็น  $r_{max} = 2m$



รูปที่ 5.20: กำหนดให้ระยะทำงานของเซนเซอร์เป็น  $r_{max} = 3m$



รูปที่ 5.21: กำหนดให้ระยะทำงานของเซนเซอร์เป็น  $r_{max} = 4m$

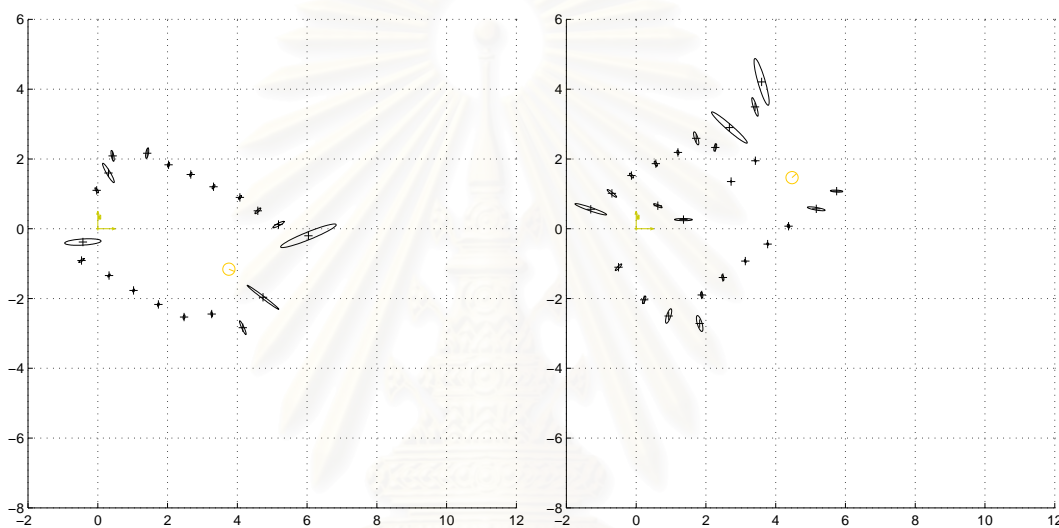


รูปที่ 5.22: กำหนดให้ระยะทำงานของเซนเซอร์เป็น  $r_{max} = 3m$  และระยะของเซนเซอร์ในระเบียบวิธีการ FastSLAM เป็น  $Max_{range} = 0.5$



บริเวณกึ่งกลางสถานที่ทดลองเพื่อทำการวัดตำแหน่งของกันและกัน แล้วนำข้อมูลแผนที่มารวมกัน

รูปที่ 5.23 แสดงให้เห็นแผนที่ซึ่งสร้างโดยหุ่นยนต์ทั้งสองตัวที่มีระบบพิกัดต่างกัน และมีรูปที่ 5.24 (ก) แสดงผลการรวมแผนที่โดยอาศัยข้อมูลการวัดตำแหน่งของหุ่นยนต์แต่ละตัวด้วยเครื่องหมายบนตัวหุ่นยนต์เพียงอย่างเดียว ซึ่งสังเกตได้ว่าตำแหน่งจุดสังเกตบริเวณรอยต่อของแผนที่ที่มีความคลาดเคลื่อนเนื่องจากความเชื่อมั่นในตำแหน่งของหุ่นยนต์แต่ละตัวบนแผนที่ของตัวเองไม่ตรงกับตำแหน่งที่วัดได้โดยหุ่นยนต์อีกตัวหนึ่ง จึงต้องทำการปรับแก้ค่าพิกัดของหุ่นยนต์เพื่อให้ได้ผลตามรูปที่ 5.24 (ข) หลังจากนั้นทำการกำจัดจุดสังเกตที่มีค่าความแปรปรวนของตำแหน่งสูงตามรูปที่ 5.25 (ก) แล้วทำการเปรียบเทียบกับตำแหน่งในแผนผังสถานที่ทดลองใน 5.25 (ข) ซึ่งเห็นได้ชัดว่ามีความคลาดเคลื่อนมากกว่ารูปที่ 5.6 ซึ่งทำงานโดยหุ่นยนต์ตัวเดียวที่เคลื่อนที่รอบสถานที่ทดลอง

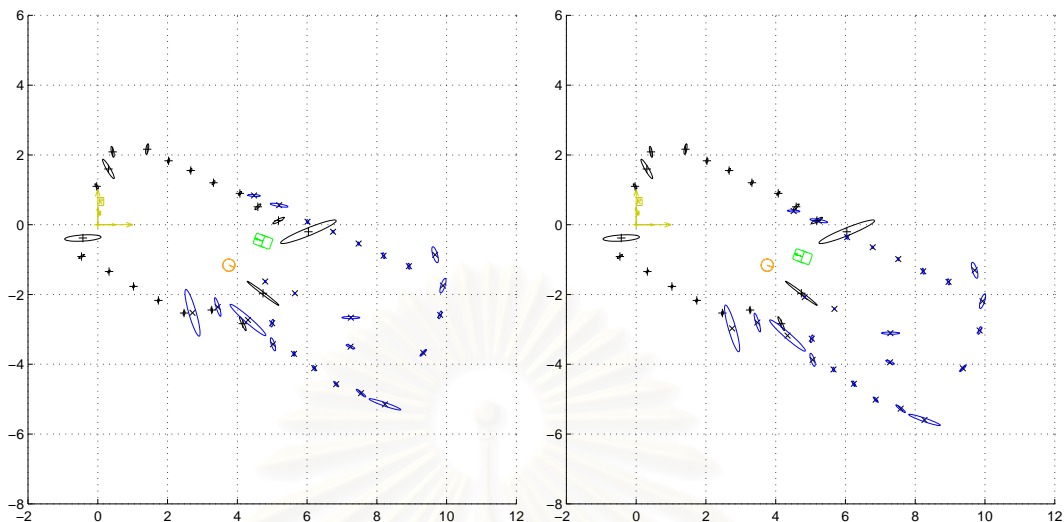


(ก) แผนที่จากการหุ่นยนต์ตัวที่ 1

(ข) แผนที่จากการหุ่นยนต์ตัวที่ 2

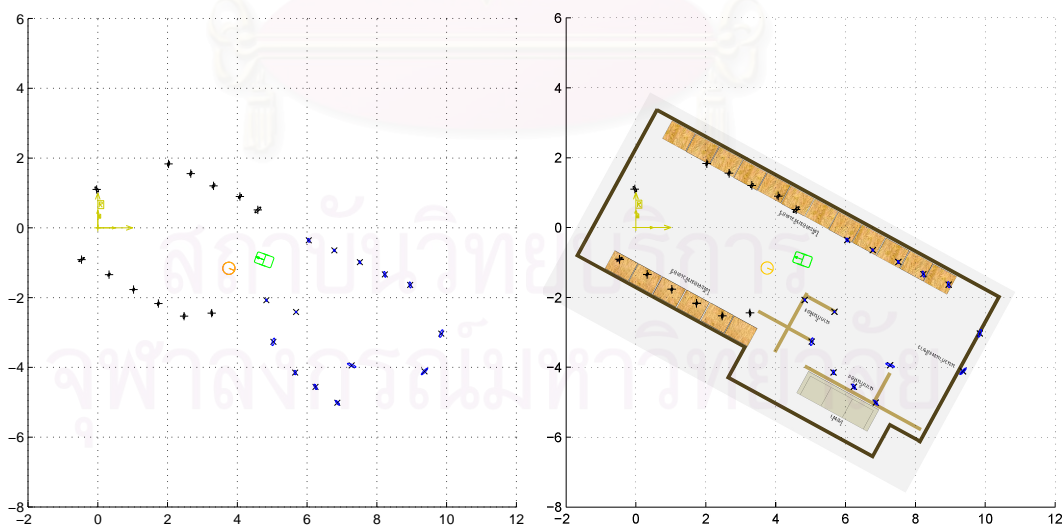
รูปที่ 5.23: ลักษณะของแผนที่จากหุ่นยนต์ทั้งสองตัว

สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย



(ก) แผนที่ยังรวมข้อมูลแผนที่ของหุ่นยนต์ทั้งสองเข้าด้วยกันโดยตรง (ข) แผนที่จากรูป 5.24 (ก) ซึ่งผ่านการปรับแก้ตำแหน่ง

รูปที่ 5.24: ลักษณะของแผนที่ก่อนและหลังการปรับแก้ค่าพิกัดของหุ่นยนต์



(ก) แผนที่ซึ่งกำจัดจุดสังเกตที่มีความความแปรปรวนร่วมของตำแหน่งสูงกว่าที่กำหนด (ข) แผนที่จากหุ่นยนต์ทั้งสองเทียบกับแผนผังห้อง

รูปที่ 5.25: แผนที่รวมจากหุ่นยนต์ทั้งสองเมื่อผ่านการปรับแก้ค่าและกำจัดจุดสังเกตที่มีความแปรปรวนสูง

## บทที่ 6

### สรุปผลของงานวิจัยและสิ่งที่ได้รับ

เนื้อหาในส่วนนี้เป็นการสรุปรายละเอียดทั้งหมดของวิทยานิพนธ์ฉบับนี้ ซึ่งประกอบไปด้วยผลการใช้ข้อมูลจากกล้องออบนิสำหรับการระบุตำแหน่งพร้อมสร้างแผนที่สำหรับหุ่นยนต์ การนำข้อมูลแผนที่มาจากหุ่นยนต์ที่ร่วมกันปฏิบัติงานในสถานที่ทดลองมาใช้ในการสร้างแผนที่รวม และงานวิจัยที่สามารถต่อยอดได้จากงานในวิทยานิพนธ์ฉบับนี้

#### 6.1 สรุปผลการทดลอง

จากการทดลองเราสามารถสรุปรายละเอียดและคุณสมบัติของกล้องออบนิในการใช้งานสำหรับการระบุตำแหน่งพร้อมการสร้างแผนที่ได้สองส่วน คือ การนำข้อมูลภาพจากกล้องออบนิมาใช้งาน และการประมวลผลข้อมูลข้างต้นเป็นข้อมูลสำหรับการระบุตำแหน่งพร้อมสร้างแผนที่

ในขั้นตอนการทดลองและศึกษาข้อมูลพบว่าการใช้งานข้อมูลภาพจากกล้องออบนิสามารถแบ่งได้เป็นสองลักษณะ คือ การนำข้อมูลภาพมาใช้สำหรับการประมวลผลภาพเพื่อการตรวจสอบสีหรือลักษณะของวัตถุเพื่อใช้ในการทำงานทั่วไป และการใช้ข้อมูลภาพมาประมวลผลเพื่อหาข้อมูลทิศทางและระยะทางของวัตถุเพื่อใช้ในการควบคุมและสั่งงานหุ่นยนต์ การใช้งานทั้งสองรูปแบบต้องการ Mapping Function เพื่อบอกความสัมพันธ์ระหว่างตำแหน่งของจุดในภาพและพิกัดในสามมิติสำหรับการแปลงภาพให้อยู่ในรูปแบบตามจุดประสงค์ของการใช้งาน เช่น การแปลงภาพรอบทิศทางเป็นภาพในมุมมองกว้างเพื่อการวัดมุม เส้นแนวตั้งหรือแปลงเป็นภาพในมุมมอง Perspective เพื่อใช้สำหรับการควบคุมการเคลื่อนที่ของหุ่นยนต์

การนำข้อมูลจากกล้องออบนิมาใช้ในการระบุตำแหน่งพร้อมสร้างแผนที่ของหุ่นยนต์โดยอาศัยข้อมูลเส้นแนวตั้ง ต้องการข้อมูลการวัดมุมที่แม่นยำเพื่อใช้ในการคำนวณตำแหน่งของเส้นแนวตั้งด้วยวิธีการ Trianguration ซึ่งจำเป็นต้องทำการปรับแก้ความผิดพลาดในการเรียงตัวของส่วนประกอบของกล้อง และหา Mapping Function ที่แม่นยำเพื่อแปลงข้อมูลภาพให้เป็นภาพในมุมมองกว้างซึ่งสามารถทำได้ด้วยวิธีปรับแก้ค่าโดยใช้ซอฟต์แวร์จาก [46] ซึ่งใช้ในวิทยานิพนธ์ฉบับนี้

ข้อมูลตำแหน่งของเส้นแนวตั้งซึ่งคำนวณจากระยะเคลื่อนที่ของหุ่นยนต์และการวัดมุมของเส้นแนวตั้งจากกล้องออบนิที่ทำการปรับแก้ความผิดพลาดแล้ว สามารถนำไปใช้เป็นข้อมูลสำหรับการระบุตำแหน่งและสร้างแผนที่ได้อย่างแม่นยำและใช้งานได้จริงในสภาพแวดล้อมในอาคารตามการทดลองที่นำเสนอ และผลการทดลองยังแสดงให้เห็นว่าการวัดตำแหน่งของหุ่นยนต์ซึ่งปฏิบัติงานร่วมกันในสถานที่ปฏิบัติงาน แล้วตรวจพบกันและกันระหว่างปฏิบัติงาน สามารถนำมาใช้ในการรวมแผนที่ของหุ่นยนต์แต่ละตัวเข้าเป็นแผนที่ของสถานที่ปฏิบัติงานได้จริง

จากข้อมูลทั้งหมดเราสามารถสรุปรายละเอียดอีกครั้งได้ดังนี้

- ข้อมูลเส้นแนวตั้งจากกล้องออบนิสามารถนำมาใช้ในการแก้ปัญหาการระบุตำแหน่งพร้อมสร้างแผนที่ได้
- ข้อมูลภาพจากกล้องออบนิที่ผ่านการปรับแก้ความผิดพลาดของการประกอบชิ้นส่วนและหา Map-

ping Function ที่ถูกต้องแล้วจะให้ข้อมูลเชิงมุมที่ความแม่นยำสูง

- การใช้กล้องอ้อมนิวัตระยะห่างของจุดสังเกตที่ทราบความสูงโดยตรง สามารถทำได้ในระยะจำกัด เนื่องจากลักษณะการรับภาพผ่านกระจกโค้งของกล้องอ้อมนิวัตส่งผลให้จำนวนพิกเซลต่อระยะทางบริเวณใกล้จุดศูนย์กลางภาพสูงกว่าบริเวณเส้นรอบวงตามรูปที่ 2.8
- ข้อมูลแผนที่ของหุ่นยนต์ซึ่งแยกกันทำงานอย่างอิสระสามารถนำมารวมกันเพื่อสร้างแผนที่รวมของสถานที่ปฏิบัติงานได้ด้วยอาศัยการวัดตำแหน่งสัมพัทธ์ของหุ่นยนต์แต่ละตัวโดยอาศัยเครื่องหมายที่ติดตั้งไว้บนหุ่นยนต์

## 6.2 แนวทางงานวิจัยในขั้นถัด

ผลการทดลองในงานวิทยานิพนธ์ฉบับนี้ร่วมกับความรู้และความเข้าใจในการใช้งานกล้องอ้อมนิวัตในงานระบุตำแหน่งพร้อมการสร้างแผนที่และการแลกเปลี่ยนข้อมูลระหว่างหุ่นยนต์ที่ร่วมกันปฏิบัติงานสามารถนำไปใช้เป็นหลักการพื้นฐานในการออกแบบระบบหุ่นยนต์ เพื่อการระบุตำแหน่งและสร้างแผนที่ในสภาพแวดล้อมที่มีความยากลำบากในการทำงานมากขึ้น เช่น การทำงานในสภาพแวดล้อมจริงภายนอกอาคารโดยอาศัยข้อมูลจากกล้องอ้อมนิวัตเพียงอย่างเดียว และการทำงานกับการเคลื่อนที่บนสภาพพื้นผิวที่มีความลาดเอียงและไม่สม่ำเสมอ นอกจากนั้นยังสามารถต่อยอดงานวิจัย โดยการเพิ่มความสามารถในการระบุตำแหน่งจุดสังเกตในสามมิติและมีการบังคับเซนเซอร์โดยอาศัยการทำงานร่วมกันของหุ่นยนต์มากกว่าหนึ่งตัว ได้อีกด้วย

## 6.3 สิ่งที่ได้รับจากงานวิจัย

สิ่งที่ได้รับจากงานวิจัยนอกเหนือจากองค์ความรู้ของวิธีการ หลักการ และความเป็นไปได้ในการใช้กล้องอ้อมนิวัตสำหรับงานระบุตำแหน่งพร้อมสร้างแผนที่ทั้งสำหรับหุ่นยนต์หนึ่งตัวและแนวทางการใช้งานประกอบด้วยผลงานทางวิชาการสองชิ้นซึ่งมีรายละเอียดดังนี้

- *An Indoor Mapping for a Mobile Robot Using an Omnidirectional Camera* จาก The 9th National Computer Science and Engineering Conference จัดโดยมหาวิทยาลัยหอการค้า กรุงเทพมหานคร ตุลาคม 2548 งานวิจัยชิ้นนี้เป็นการทดสอบการทำงานเบื้องต้นของกล้องอ้อมนิวัตสำหรับการสร้างแผนที่โดยอาศัยข้อมูลเส้นแนวตั้งโดยหุ่นยนต์หนึ่งตัวในสภาพแวดล้อมภายในอาคาร
- *Cooperative Localization and Mapping using the Vertical Edge Detection from an Omnidirectional Camera* จาก The 10th National Computer Science and Engineering Conference จัดโดยมหาวิทยาลัยขอนแก่น ณ จังหวัดขอนแก่น ตุลาคม 2549 งานวิจัยชิ้นนี้เสนอความเป็นไปได้ในการใช้หุ่นยนต์สองตัวร่วมกันระบุตำแหน่งพร้อมกับการสร้างแผนที่ในสภาพแวดล้อมภายในอาคาร

## รายการอ้างอิง

- [1] Yasushi Yagi and Shinjiro Kawato. Panorama scene analysis with conic projection. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1990.
- [2] Shree K. Nayar. Catadioptric omnidirectional camera. In IEEE Conf. on Computer Vision and Pattern Recognition, 1997.
- [3] Simon Baker and Shree K. Nayar. A theory of single-viewpoint catadioptric image formation. International Journal of Computer Vision, 35(2):1 -- 22, 1999.
- [4] Cyril Cauchois, Eric Brassart, Claude Pegard, and Arnaud Clerentin. Technique for calibrating an omnidirectional sensor. In IEEE/RSJ International Conference on Intelligent Robots and Systems, 1999.
- [5] Adam Kropp, Neel Master, and Seth Teller. Acquiring and rendering high-resolution spherical mosaics. In IEEE Workshop on Omnidirectional Vision, Hilton Head Island, South Carolina, 2000.
- [6] Sing Bing Kang. Catadioptric self-calibration. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2000.
- [7] Yasushi Yagi and Masahiko Yachida. Real-time omnidirectional image sensors. Int. Journal of Computer Vision, 58(3):172--207, 2004.
- [8] Radu Orghidan, El Mustapha Mouaddib, and Joaquim Salvi. Omnidirectional depth computation from a single image. In IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005.
- [9] Koji Yoshida, Hajime Nagahara, and Masahiko Yachida. An omnidirectional vision sensor with single viewpoint and constant resolution. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2006.
- [10] Christopher Mei and Patrick Rives. Single view point omnidirectional camera calibration from planar grids. In IEEE Int. Conf. on Robotics and Automation, 2007.
- [11] Yasushi Yagi and Masahiko Yachida. Real-time generation of environmental map and obstacle avoidance using omnidirectional image sensor with conic mirror. In IEEE Conf. on Computer Vision and Pattern Recognition, 1991.
- [12] Claude Pégard and El Mustapha Mouaddib. A mobile robot using a panoramic view. In IEEE Int. Conf. on Robotics and Automation, 1996.
- [13] Cyril Drocourt, Laurent Delahoche, Claude Pegard, and Cyril Cauchois. Localization method based on omnidirectional stereoscopic vision and dead-reckoning. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1999.



- [14] Amaud Clerentin, Laurent Delahoche, and Eric Brassart. Cooperation between two omnidirectional perception systems for mobile robot localization. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2000.
- [15] Young Jin Lee and Myung Jin Chung. A reliable feature matching method in omnidirectional views for autonomous map generation of a mobile robot. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2001.
- [16] Lixin Tang and Shin'ichi Yuta. Indoor navigation for mobile robots using memorized omni-directional images and robot's motion. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2002.
- [17] Jose-Joel Gonzalez-Barbosa and Simon Lacroix. Rover localization in natural environments by indexing panoramic images. In IEEE International Conference on Robotics and Automation, Washington, DC, USA, 2002.
- [18] Yasushi Yagi, Kousuke Imai, and Masahiko Yachida. Iconic memory-based omnidirectional route panorama navigation. In IEEE International Conference on Robotics and Automation, Taipei, Taiwan, 2003.
- [19] C. Sagüés, A.C. Murillo, J.J. Guerrero, T. Goedemé, T. Tuytelaars, and L. Van Gool. Localization with omnidirectional images using the radial trifocal tensor. In IEEE Int. Conf. on Robotics and Automation, 2006.
- [20] Kostas Daniilidis and Christopher Geyer. Omnidirectional vision: Theory and algorithms. In IEEE Computer Society Int. Conf. on Pattern Recognition, 2000.
- [21] Roland Siegwart and Illah R. Nourbakhsh. Introduction to Autonomous Mobile Robots (Intelligent Robotics and Autonomous Agents). The MIT Press, 2004.
- [22] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. Probabilistic Robotics. The MIT Press, 2005.
- [23] Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: Part i, ii. IEEE Robotics & Automation Magazine, June June, September 2006.
- [24] Akihisa Ohya, Akio Kosaka, and Avinash Kak. Vision-based navigation by a mobile robot with obstacle avoidance using single-camera vision and ultrasonic sensing. IEEE Transactions on Robotics and Automation, 14(6):969--978, 1998.
- [25] Frank Dellaert, Dieter Fox, Wolfram Burgard, and Sebastian Thrun. Monte carlo localization for mobile robots. In IEEE Int. Conf. on Robotics and Automation, 1999.
- [26] Kai O. Arras, Nicola Tomatis, and Roland Siegwart. Multisensor on-the-fly localization using laser and vision. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2000.
- [27] Ioannis Rekleitis. Cooperative Localization and Multi-Robot Exploration. PhD thesis, School of Computer Science, McGill University, 2003.

- [28] Robert Sim, Pantelis Elinas, Matt Griffin, Alex Shyr, and James J. Little. Design and analysis of a framework for real-time vision-based slam using rao-blackwellised particle filters. In The 3rd Canadian Conference on Computer and Robot Vision, 2006.
- [29] Robert Sim. Stable exploration for bearings-only slam. In IEEE Int. Conf. on Robotics and Automation, 2005.
- [30] John W. Fenwick, Paul M. Newman, and John J. Leonard. Cooperative concurrent mapping and localization. In IEEE International Conference on Robotics and Automation, Washington, DC, USA, 2002.
- [31] Yasushi Yagi, Shinichi Izuhara, and Masahiko Yachida. The integration of an environmental map observed by multiple mobile robots with omnidirectional image sensor copis. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1996.
- [32] Robocup federation. In <http://www.robocup.org>.
- [33] Bruno Marhic, El Mustapha Mouaddib, and Claude Pegard. A localisation method with an omnidirectional vision sensor using projective invariant. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1998.
- [34] Yoshio Matsumoto, Kazunori Ikeda, Masayuki Inaba, and Hirochika Inoue. Visual navigation using omnidirectional view sequence. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1999.
- [35] Jae-Hean Kim and Myung-Jin Chung. Map generation from unknown planar motion using omnidirectional vision. In IEEE/RSJ International Conference on Intelligent Robots and Systems, Wailea, Hawaii, 2001.
- [36] Jun Miura, Yoshiro Negishi, and Yoshiaki Shirai. Mobile robot map generation by integrating omnidirectional stereo and laser range finder. In IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 2002.
- [37] Adriana Tapus and Roland Siegwart. Incremental robot mapping with fingerprints of places. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2005.
- [38] Mahisorn Wongphati, Nattee Niparnan, and Attawith Sudsang. An indoor mapping for a mobile robot using an omnidirectional camera. In The 9th National Computer Science and Engineering Conference, Bangkok, Thailand, 2005.
- [39] Jae-Hean Kim and Myung Jin Chung. Slam with omni-directional stereo vision sensor. In IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, Nevada, 2003.
- [40] Cyril Drocourt, Laurent Delahoche, Bruno Marhic, and Amaud Clerentin. Simultaneous localization and map construction method using omnidirectional stereoscopic information. In IEEE Int. Conf. on Robotics and Automation, 2002.

- [41] Type "localization mapping omni directional camera". In <http://scholar.google.com/>.
- [42] Peter Biber, Henrik Andreasson, Tom Duckett, and Andreas Schilling. 3d modeling of indoor environments by a mobile robot with a laser scanner and panoramic camera. In IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004.
- [43] Henrik Andreasson, Andre Treptow, and Tom Duckett. Localization for mobile robots using panoramic vision, local features and particle filter. In IEEE International Conference on Robotics and Automation, Barcelona, Spain, 2005.
- [44] Sebastian Thrun, Michael Montemerlo, Daphne Koller, Ben Wegbreit, Juan Nieto, and Eduardo Nebot. Fastslam: An efficient solution to the simultaneous localization and mapping problem with unknown data association. Journal of Machine Learning Research, 2004.
- [45] Simon Baker and Shree K. Nayar. A theory of catadioptric image formation. In IEEE Int. Conf. on Computer Vision, 1998.
- [46] Davide Scaramuzza, Agostino Martinelli, and Roland Siegwart. A toolbox for easily calibrating omnidirectional cameras. In IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2006.
- [47] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. IEEE Journal of Robotics and Automation, 3(4):323--344, 1987.
- [48] Yasushi Yagi, Kazuya Sato, and Masahiko Yachida. Evaluating effectivity of map generation by tracking vertical edges in omnidirectional image sequence. In IEEE Int. Conf. on Robotics and Automation, 1995.
- [49] Remotereality. In <http://www.remotereality.com/>.
- [50] Matlab. In <http://www.mathworks.com/>.
- [51] Kazumasa Yamazawa, Yasushi Yagi, and Masahiko Yachida. Omnidirectional imaging with hyperboloidal projection. In IEEE/RSJ International Conference on Intelligent Robots and Systems, Yokohama, Japan, 1993.
- [52] Christopher Geyer and Kostas Daniilidis. Catadioptric camera calibration. In IEEE International Conference on Computer Vision, 1999.
- [53] Peter S Maybeck. Stochastic models, estimation and control. Academic Press, 1979.
- [54] Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, SIGGRAPH Tutorial, Coures 8, 2001.
- [55] Dieter Fox, Wolfram Burgard, Hannes Kruppa, and Sebastian Thrun. A monte carlo algorithm for multi-robot localization. In AAAI 11th Conference on Innovative Applications of Artificial Intelligence, Orlando, Florida, 1999. AAI.

- [56] Open computer vision library (opencv). In <http://www.intel.com/technology/computing/opencv/>.
- [57] Kai O. Arras. The cas robot navigation toolbox. In <http://www.openslam.org/cas-rnt.html>.
- [58] Tim Bailey. Slam package of tim bailey. In <http://www.openslam.org/bailey-slam.html>.
- [59] Mobilerobots inc. In <http://www.mobilerobots.com>.
- [60] Dirk Hahnel, Wolfram Burgard, Dieter Fox, and Sebastian Thrun. An efficient fastslam algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In [IEEE/RSJ Int. Conf. on Intelligent Robots and Systems](#), 2003.



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย

## ประวัติผู้เขียนวิทยานิพนธ์

นายมหิศร ว่องผาติ เกิดเมื่อวันที่ 24 มิถุนายน 2524 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาตรีวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัยในปีการศึกษา 2547

มีความสนใจในงานวิจัยและงานประดิษฐ์ที่เกี่ยวข้องกับหุ่นยนต์ โดยเฉพาะหุ่นยนต์ทำงานอัตโนมัติ ผลงานประดิษฐ์ด้านหุ่นยนต์ที่โดดเด่น คือ เป็นผู้ก่อตั้งและควบคุมทีมโรบอคัพขนาดเล็ก (RoboCup Small Size League) ชื่อ Plasma-Z ของชมรมนักประดิษฐ์วิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ซึ่งแข่งขันได้อันดับสามของโลกในปี 2549 (2006) ณ เมืองเบอร์เมน ประเทศเยอรมัน นอกจากนี้ยังมีประสบการณ์การทำงานเกี่ยวกับหุ่นยนต์ในเชิงอุตสาหกรรม เช่น หุ่นยนต์เจียร์พลอย และโปรแกรมควบคุมเครื่องยิงเลเซอร์สำหรับขัดแป้นติดหัวอ่านฮาร์ดดิส



สถาบันวิทยบริการ  
จุฬาลงกรณ์มหาวิทยาลัย