

หุ่นยนต์ภาคพื้นเพื่อการสำรวจสวนผลไม้  
Field Robot for Fruit tree Orchard Inspection

จิราเจตน์ บัวกิ่ง  
จิรายุ พันธุมิตร  
สหรัฐ ไมลาศรี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมเกษตร  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2560

หุ่นยนต์ภาคพื้นเพื่อการสำรวจสวนผลไม้

Field Robot for Fruit tree Orchard Inspection

จิราเจตน์ บัวกิ่ง

จิรายุ พันธุมิตร

สหรัฐ ไมลาศรี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมเกษตร

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

# Field Robot For Fruit Tree Orchard Inspection

Jirajet Buaking

Jirayu Phantumitr

Saharath Mailasri

A THESIS SUBMITTED IN PARTIAL FULFILLMENT  
OF THE REQUIREMENT FOR THE DEGREE OF  
BACHELOR OF ENGINEERING IN AGRICULTURAL ENGINEERING  
FACULTY OF ENGINEERING  
KING MONGKUTS INSTITUTE OF TECHNOLOGY LADKRABANG  
ACADEMIC YEAR 2017

ปริญญาานิพนธ์ปีการศึกษา 2560

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญาานิพนธ์

หัวข้อปริญญาานิพนธ์ หุ่นยนต์ภาคพื้นเพื่อการสำรวจสวนผลไม้  
Field Robot for Fruit Tree Orchard Inspection


นักศึกษาผู้จัดทำ นายจิราเจตน์ บัวกิ่ง รหัสนักศึกษา 57010197  
นายจิรายุ พันธุ์มิตร รหัสนักศึกษา 57010201  
นายสหรัฏฐ์ ไมลาศรี รหัสนักศึกษา 57011327

ปริญญา วิศวกรรมศาสตรบัณฑิต (วิศวกรรมเกษตร)

หลักสูตร วิศวกรรมเกษตร

ภาควิชา วิศวกรรมเกษตร

ปีการศึกษา 2560

อาจารย์ผู้ควบคุมปริญญาานิพนธ์	ลายมือชื่อ
ดร.วสุ อุดมเพทายกุล	

หัวข้อปริญญานิพนธ์	หุ่นยนต์ภาคพื้นเพื่อการสำรวจสวนผลไม้		
นักศึกษาผู้จัดทำ	นายจิราเจตน์	บัวกิ่ง	รหัสนักศึกษา 57010197
	นายจิรายุ	พันธุมิตร	รหัสนักศึกษา 57010201
	นายสหรัฐ	ไมลาศรี	รหัสนักศึกษา 57011327
อาจารย์ที่ปรึกษา	ดร.วสุ อุดมเพทายกุล		
ปีการศึกษา	2560		

### บทคัดย่อ

โครงการวิศวกรรมเกษตรนี้มีวัตถุประสงค์เพื่อออกแบบและสร้างต้นแบบหุ่นยนต์ขับเคลื่อน 4 ล้อ เพื่อการสำรวจสวนผลไม้ด้วยวิธีการประมวลผลภาพซึ่งเป็นการถ่ายภาพจากแนวราบทำให้ได้ภาพมาทำการวิเคราะห์หาสิ่งแปลกปลอมภายในสวนผลไม้และสามารถระบุตำแหน่งของสิ่งแปลกปลอม

หุ่นยนต์ขับเคลื่อน 4 ล้อ เพื่อการสำรวจสวนผลไม้ประกอบด้วย 2 ส่วนหลัก คือ ระบบควบคุมการเคลื่อนที่ของหุ่นยนต์และระบบประมวลผลภาพ โดยหลักการทำงานของการทำงานของการเคลื่อนที่คือ Joystick ทำการรับคำสั่งการเคลื่อนที่แล้วส่งคำสั่งต่อไปยังไมโครคอนโทรลเลอร์ทำการส่งคำสั่งให้ motor driver และ motor driver ทำการสั่งให้มอเตอร์หมุนตามที่ต้องการ และในส่วนของการประมวลผลภาพจะทำการทดสอบโดยใช้หลักการของกล้องอินฟราเรดและคำนวณของระยะห่างระหว่างกล้องกับสิ่งแปลกปลอมเพื่อระบุตำแหน่งของสิ่งแปลกปลอมนั้น

Thesis Title	Automatic Ground Robot for Fruit Tree Orchard Inspection		
Authors	Mr.Jirajet	Buaking	57010197
	Mr.Jirayu	Phantumitr	57010201
	Mr.Saharath	Milasri	57011327
Thesis Advisor	Dr.Vasu	Udompetaikul	
Year	2017		

### Abstract

The objective of this agricultural engineering project was to design and build a four-wheeled robotic platform for orchard of fruit tree inspection. Visualization and image processing technique were used to identifiable the specified targets.

The system consists of two main parts, robotic system and image processing system. Motion of the robot was controlled by a microcontroller system with motor/wheel driving units and a wireless joystick. In image processing part, we used a Microsoft Kinect camera which applying infrared mapping concept to determine the distance between camera and the target.

## กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วง โดยได้รับคำแนะนำ ความช่วยเหลือ และความร่วมมือจาก บุคคลหลายฝ่าย

บุคคลแรกที่ต้องขอขอบพระคุณเป็นอย่างสูงคือ ดร.วสุ อุดมเพทายกุล อาจารย์ที่ปรึกษาที่ให้ คำแนะนำ ความช่วยเหลือ กำลังใจ และดูแลเอาใจใส่ตลอดมา

ขอขอบพระคุณคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้งบประมาณสนับสนุน

ขอขอบพระคุณคณาจารย์หลักสูตรวิศวกรรมเกษตร คณะวิศวกรรมศาสตร์ สถาบัน วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่านสำหรับคำแนะนำ ความรู้ทางด้านวิชาการ และกำลังใจในการทำงาน

ขอขอบพระคุณบุคลากรหลักสูตรวิศวกรรมเกษตร คณะวิศวกรรมศาสตร์ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ให้คำแนะนำ ความช่วยเหลือและ อำนวยความสะดวกในการทำงาน

ขอขอบคุณเพื่อนๆหลักสูตรวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ วิศวกรรมศาสตร์ สถาบัน เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้การสนับสนุน ช่วยเหลือและกำลังใจในการทำงาน

สุดท้ายนี้ต้องขอขอบพระคุณบุคคลที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ คือ บิดา มารดา อันเป็นที่เคารพ รักยิ่ง ซึ่งได้ดูแลเลี้ยงดูมาเป็นอย่างดี ให้ความเอาใจใส่เสมอมา และให้การสนับสนุนในทุกเรื่อง พร้อมทั้ง ให้ออกาสทางการศึกษาอย่างเต็มที่ ข้าพเจ้าขอกราบพระคุณมา ณ ที่นี้

นายจิราเจตน์ บัวกิ่ง

นายจิรายุ พันธุ์มิตร

นายสหรัฐ ไม้ลาศรี

## สารบัญ

	หน้าที่
บทคัดย่อภาษาไทย/บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญภาพ	VIII
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตการศึกษา	2
1.4 ผลที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนการดำเนินงาน	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	
2.1 หลักการเคลื่อนที่ของหุ่นยนต์ขับเคลื่อน 4 ล้ออัตโนมัติ	3
2.2 มอเตอร์ไฟฟ้า (Electric motor)	5
2.3 ชุดขับเคลื่อนมอเตอร์กระแสตรง	8
2.4 Kinect Xbox 360	9
2.5 การประมวลผลภาพ	9
2.6 ภาพรวมระบบของการทำงานหุ่นยนต์ขับเคลื่อน 4 ล้อ	11
บทที่ 3 การออกแบบและการสร้าง	
3.1 ขั้นตอนการออกแบบและสร้างหุ่นยนต์ขับเคลื่อน 4 ล้อ	12
3.2 ขั้นตอนการประกอบหุ่นยนต์	15
3.3 ส่วนของการวิเคราะห์และระบุตำแหน่ง	17
บทที่ 4 วิธีการทดสอบ	
4.1 การทดสอบสมรรถนะของหุ่นยนต์ขับเคลื่อน 4 ล้อ	19
4.2 การทดสอบการวิเคราะห์และระบุตำแหน่งสิ่งแปลกปลอม	20

บทที่ 5 ผลการทดสอบและวิเคราะห์ผลการทดสอบ

5.1 ผลการทดสอบการสิ้นเปลืองของล้อและ การรับน้ำหนักของหุ่นยนต์ขับเคลื่อน 4 ล้อ	22
5.2 ผลการทดสอบการวิเคราะห์และระบุตำแหน่งสิ่งแปลกปลอม	25

บทที่ 6 สรุปผลการศึกษา

เอกสารอ้างอิง

ภาคผนวก

ภาคผนวก ก. รายละเอียดของอุปกรณ์อิเล็กทรอนิกส์	31
ภาคผนวก ข. โปรแกรมที่ใช้ในการควบคุมมอเตอร์	34
ภาคผนวก ค. โปรแกรมที่ใช้ในการวิเคราะห์และระบุสิ่งแปลกปลอม	37
ภาคผนวก ง. ข้อมูลจากการทดสอบ	40

## สารบัญตาราง

	หน้าที่
ตารางที่ 1 การทดสอบการสิ้นไกลของล้อยและ	
การรับน้ำหนักในพื้นที่คอนกรีตเรียบขัดเงา	22
ตารางที่ 2 การทดสอบการสิ้นไกลของล้อยและ	
การรับน้ำหนักในพื้นที่คอนกรีต	23
ตารางที่ 3 ผลการทดลองการวิเคราะห์และระบุ	
ตำแหน่งเป้าหมาย	24

## สารบัญรูป

	หน้าที่
รูปที่ 2.1 แสดงความแตกต่างระหว่าง holonomic กับ non-holonomic	3
รูปที่ 2.2 differential drive	4
รูปที่ 2.3 skid steer	4
รูปที่ 2.4 ตัวอย่างมอเตอร์ไฟฟ้ากระแสตรง	5
รูปที่ 2.5 การทำงานของวงจร H-Bridge สำหรับควบคุมการหมุนของ DC Motor	8
รูปที่ 2.6 การเชื่อมต่อ Microcontroller กับชุดขับกระแสดีซีมอเตอร์ แบบ H-Bridge	8
รูปที่ 2.7 ส่วนประกอบของกล้อง Kinect Xbox 360	9
รูปที่ 2.8 ตัวอย่างการ Threshold	10
รูปที่ 2.9 ภาพรวมระบบของการทำงานหุ่นยนต์ขับเคลื่อน 4 ล้อ	11
รูปที่ 3.1 หุ่นยนต์ขับเคลื่อน 4 กล้อง	12
รูปที่ 3.2 โครงของหุ่นยนต์ขับเคลื่อน 4 ล้อ	12
รูปที่ 3.3 แผนอลูมิเนียม	13
รูปที่ 3.4 มอเตอร์กระแสตรง 12 โวลต์ 400 รอบ	13
รูปที่ 3.5 H-bridge motor driver รุ่น BTS7960B	14
รูปที่ 3.6 เพลา ล้อ และเบริงตุ๊กตา	15
รูปที่ 3.7 มอเตอร์และชุดเฟืองทด พร้อมทั้งโซ่	15
รูปที่ 3.8 H-bridge motor driver	15
รูปที่ 3.9 Raspberry Pi 3 Model B	16
รูปที่ 3.10 การติดตั้งแผ่นอะคริลิก	16
รูปที่ 3.11 ลักษณะการติดตั้งกล้อง Kinect Xbox 360	16
รูปที่ 3.12 กล้อง Kinect Xbox 360	17

## สารบัญรูป ( ต่อ )

	หน้าที่
รูปที่ 3.13 Flow chartแสดงการทำงานของระบบประมวลผลภาพ	17
รูปที่ 4.1 การทดสอบการสิ้นไถลขณะเคลื่อนที่และการรับน้ำหนักของหุ่นยนต์ ขับเคลื่อน 4 ล้อ	19
รูปที่ 4.2การทดสอบการสิ้นไถลขณะเคลื่อนที่และการรับน้ำหนักของหุ่นยนต์ ขับเคลื่อน 4 ล้อ	20
รูปที่ 4.3 การเพิ่มน้ำหนักในการทดสอบการสิ้นไถลขณะเคลื่อนที่และการรับน้ำหนักของ หุ่นยนต์ขับเคลื่อน 4 ล้อ	20
รูปที่ 4.4 ตำแหน่งการวางลูกบอลทดสอบ	21
รูปที่ 4.5 การทดสอบการวิเคราะห์และระบุตำแหน่งสิ่งแปลกปลอมในตำแหน่งที่แตกต่างกัน	22
รูปที่ 5.1 ความสัมพันธ์ระหว่าง Load กับ Velocity	24
รูปที่ 6.1 หุ่นยนต์ขับเคลื่อน 4 ล้อเพื่อการสำรวจสวนผลไม้	26

## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญ

ประเทศไทยเป็นประเทศเกษตรกรรมที่มีประชาชนในภาคการเกษตรเป็นจำนวนมาก ซึ่งในปัจจุบันได้มีการนำเทคโนโลยีมาใช้พัฒนาในหลายๆด้านมากยิ่งขึ้น เพราะการใช้เทคโนโลยีจะมีส่วนช่วยให้ผลผลิตมีคุณภาพ โดยในด้านเกษตรกรรมประเทศไทยมีแนวโน้มที่จะเปลี่ยนจากการทำเกษตรกรรมแบบดั้งเดิมไปสู่เกษตรกรรมแบบสมัยใหม่ ที่จะเน้นการบริหารจัดการและการใช้เทคโนโลยีหรือ Smart Farming ในการประยุกต์ใช้กับเกษตรกรรมเพื่อให้ได้มาซึ่งผลผลิตที่มีคุณภาพมากยิ่งขึ้น

จากข้อมูลของสำนักงานเศรษฐกิจการเกษตรของประเทศไทยได้กล่าวถึงผลผลิตที่สำคัญหลายอย่างซึ่งผลผลิตที่สำคัญทางเศรษฐกิจของประเทศไทยคือผลไม้จัดได้ว่าเป็นผลผลิตที่ต้องการของตลาดส่งออกทั้งในและต่างประเทศแถบเอเชียและยุโรป<sup>[1]</sup> โดยสถิติจากสำนักงานเศรษฐกิจการเกษตรในปี พ.ศ.2559 ผลผลิตของผลไม้มีปริมาณที่ลดลงจากปีที่ผ่านมา โดยหนึ่งในสาเหตุที่ทำให้ผลผลิตของผลไม้ลดลงคือ โรคและแมลง เช่น โรคแอนแทรกโนส โรคราดำ โรคราแป้ง เป็นต้น<sup>[2]</sup> ซึ่งเป็นสิ่งที่เข้าทำลายส่วนประกอบของต้นและผลผลิตโดยสาเหตุนี้สามารถแก้ไขได้ทัน ถ้าเกษตรกรสามารถรับรู้ปัญหาได้ทันเวลาที่ และเข้าแก้ไขปัญหาที่เกิดขึ้นด้วยการพ่นยาหรือตัดส่วนที่เสียหายได้ทันเวลา โดยปกติแล้วเกษตรกรจะทำการลงพื้นที่สำรวจความผิดปกติโดยสังเกตจากลักษณะอาการที่เกิดขึ้นจากโรคและแมลงบริเวณใบ ลำต้น ดอกและผล แต่เนื่องจากแรงงานมีน้อยแต่การปลูกเชิงเศรษฐกิจจะปลูกในพื้นที่กว้างและปริมาณมาก ซึ่งในบางครั้งอาจไม่ทั่วถึงและเกิดความผิดพลาดในการสำรวจตรวจสอบ

ดังนั้นเพื่อแก้ไขปัญหาที่กล่าวมาข้างต้น จึงมีแนวคิดที่จะออกแบบและพัฒนาหุ่นยนต์ขับเคลื่อนสี่ล้ออัตโนมัติเพื่อช่วยเกษตรกรในการเข้าสำรวจพื้นที่สวนผลไม้เพื่อเพิ่มประสิทธิภาพในการสำรวจของเกษตรกรและเพื่อช่วยลดระยะเวลาในการตรวจสอบ ด้วยวิธีการถ่ายภาพและมาวิเคราะห์ภาพถึงความผิดปกติของพืชผล

#### 1.2 วัตถุประสงค์

- 1) พัฒนาหุ่นยนต์ขับเคลื่อน4ล้อสำรวจภาคพื้นเพื่อใช้ในการสำรวจสวนผลไม้
- 2) พัฒนาวีธีจำแนกและระบุตำแหน่งของเป้าหมายที่กำหนด

### 1.3 ขอบเขตการศึกษา

- 1) เป็นการพัฒนาระบบต้นแบบและทดสอบในห้องปฏิบัติการโดยใช้เป้าหมายที่มีสีแตกต่างจากสภาพแวดล้อม

### 1.4 ผลที่คาดว่าจะได้รับ

- 1) ได้ต้นแบบและแนวทางในการพัฒนาหุ่นยนต์ขับเคลื่อน 4 ล้อเพื่อการสำรวจสวนผลไม้

### 1.5 ขั้นตอนการดำเนินงาน

- 1) ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวกับโรคและแมลงของต้นมะม่วง, หุ่นยนต์อัตโนมัติ, การประมวลผลภาพ
- 2) ออกแบบระบบและเลือกอุปกรณ์จากทฤษฎีและการวิจัยต่างๆ
- 3) ประกอบอุปกรณ์ต่างๆและเขียนชุดคำสั่งเพื่อควบคุมความเร็วการเคลื่อนที่ของหุ่นยนต์และทดสอบหาข้อบกพร่อง
- 4) ปรับปรุงระบบการควบคุมของหุ่นยนต์สำรวจ
- 5) ทดสอบการเคลื่อนที่สู่ตำแหน่งที่ต้องการ และความแม่นยำของการประมวลผลภาพของ

## บทที่ 2

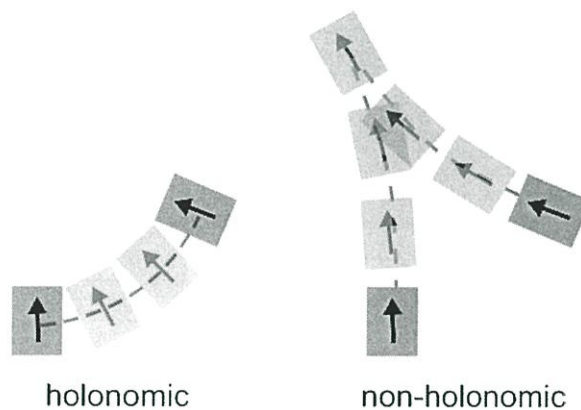
### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

#### 2.1 หลักการเคลื่อนที่ของหุ่นยนต์ขับเคลื่อน 4 ล้ออัตโนมัติ

Holonomic locomotion คือหุ่นยนต์ที่มีองศาอิสระที่ควบคุมได้ (controllable degree of freedom) เท่ากับองศาอิสระทั้งหมดของหุ่นยนต์ (total degree of freedom) หุ่นยนต์ที่เป็น holonomic จะสามารถเคลื่อนที่ไปยังทิศทางใด ๆ ก็ได้ตามที่กำหนด<sup>[3]</sup>

##### 2.1.1 Non - Holonomic locomotion

คือหุ่นยนต์ที่มีองศาอิสระที่ควบคุมได้ (controllable degree of freedom) น้อยกว่าองศาอิสระทั้งหมดของหุ่นยนต์ (total degree of freedom)<sup>[3]</sup> หุ่นยนต์ที่เป็น non-holonomic จะไม่สามารถเคลื่อนที่ไปยังทิศทางใด ๆ ก็ได้ตามใจชอบ จะต้องยุ่งยาก ตัวอย่างเช่น รถยนต์ อยู่บนระนาบ 2 มิติ จะมีองศาอิสระทั้งหมด 3 องศา คือ การเคลื่อนที่ตามแนว X, Y และองศาของการหมุนรอบตัว รถยนต์มีองศาอิสระที่ควบคุมได้แค่ 2 องศา คือ การเคลื่อนที่ไปหน้า-หลัง และองศาการเลี้ยว<sup>[4]</sup> ดังนั้น รถยนต์ไม่สามารถเคลื่อนที่ไปในทิศใด ๆ ที่ต้องการได้ตามใจชอบ แต่อาจจะต้องเลี้ยวขยับไปมาหลายรอบ เช่น ตอนจอดรถแบบขนาน จะต้องเดินหน้า ถอยหลังหลายรอบ ถ้ารถสามารถเคลื่อนที่แนวซ้าย-ขวาได้(holonomic) คล่องตัวมากขึ้น<sup>[2]</sup>



รูปที่ 2.1 แสดงความแตกต่างระหว่าง holonomic กับ non-holonomic

ที่มา kamolc(2017)

### 2.1.2 Differential drive

เป็นรูปแบบการเคลื่อนที่ด้วยล้อที่ง่ายที่สุด คือ มีล้อ 2 ล้อ ล้อซ้าย และล้อขวา ถ้าต้องการเคลื่อนที่ไปข้างหน้าตรง ๆ ก็หมุนล้อซ้ายและขวาไปทางเดียวกันด้วยความเร็วเท่ากัน ถอยหลังตรงๆก็หมุนล้อซ้ายและขวาถอยหลังทั้งคู่ถ้าอยากเลี้ยวซ้ายก็หมุนล้อขวาไปข้างหน้าให้เร็วกว่าล้อซ้ายอยากเลี้ยวขวาก็หมุนล้อซ้ายไปข้างหน้าให้เร็วกว่าล้อขวาหรือจะหมุนตัวอยู่กับที่ก็ได้ก็ให้หมุนล้อข้างหนึ่งไปข้างหน้า และล้ออีกข้างหมุนกลับหลังด้วยความเร็วเท่ากัน จะเห็นได้ว่าการขับเคลื่อนแบบ differential drive นั้น ไม่สามารถเคลื่อนที่ไปทิศใด ๆ ได้ตามต้องการในทันที ต้องทำการเลี้ยว หมุนตัวก่อน ดังนั้น differential drive จึงเป็นการเคลื่อนที่แบบ non-holonomic<sup>[4]</sup>



รูปที่ 2.2 differential drive ที่มา kamolc(2017)

### 2.1.3 Skid steer

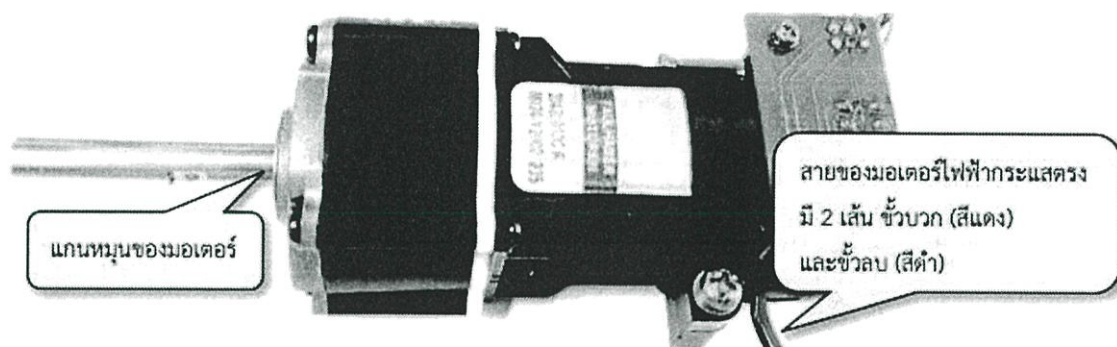
คล้ายกันกับ differential drive ใช้หลักการเดียวกันเรื่องการหมุนล้อเดินทาง ถอยหลังเพื่อเคลื่อนที่และเลี้ยว ต่างกันตรงที่ skid steer จะใช้เรียกกับหุ่นยนต์ที่ใช้มากกว่า 2 ล้อ เช่น 4, 6, 8 ล้อและตีนตะขาบ เนื่องจากในหุ่นยนต์ที่มีล้อมากๆหรือสายพานตีนตะขาบ เวลาเลี้ยวหรือหมุนตัว จะเกิดลื่นไถล (slip) บนล้อเทียบกับพื้น แน่นนอนว่าการ slip นี้ทำให้ skid steer ใช้พลังงานในการเลี้ยวที่มากกว่า differential drive แต่ด้วยล้อที่มากกว่า มีพื้นที่สัมผัสพื้นมากกว่า ก็ทำให้มีแรงขับเคลื่อนมากกว่า<sup>[4]</sup>



รูปที่ 2.3 skid steer ที่มา kamolc(2017)

## 2.2 มอเตอร์ไฟฟ้า (Electric motor)

มอเตอร์ไฟฟ้ากระแสตรง หรือดีซีมอเตอร์ (DC Motor) เป็นอุปกรณ์ที่แปลงพลังงานไฟฟ้าให้เป็นพลังกลโครงสร้างภายใน<sup>[5]</sup> DC motor ประกอบด้วยส่วนหลักๆ สองส่วน ได้แก่ แม่เหล็กถาวรและแกนขดลวดนอกจากนี้ยังมีแปรงถ่าน(Brush)ซึ่งเป็นส่วนเชื่อมต่อเพื่อรับพลังงานไฟฟ้าภายนอกไปยังขดลวดของมอเตอร์เมื่อขดลวดได้รับไฟฟ้ากระแสตรงจะมีถูกเหนี่ยวนำให้เกิดสนามแม่เหล็กรอบๆ ขดลวด



รูปที่ 2.4 ตัวอย่างมอเตอร์ไฟฟ้ากระแสตรงที่มา ONSemiconductor(2013)

### 2.2.1 การคำนวณการเลือกใช้มอเตอร์

-การหาแรงในการขับเคลื่อนสมการความสัมพันธ์ระหว่างกำลังของมอเตอร์กับทอร์กของมอเตอร์<sup>[6]</sup>

$$P_m = 2\pi N T_m$$

$P_m$  = กำลังมอเตอร์, watt

$N$  = อัตราเร็วรอบของมอเตอร์, rps

$T_m$  = แรงบิดมอเตอร์, N·m

สมการความสัมพันธ์ระหว่างทอร์กของล้อกับทอร์กของมอเตอร์

$$T_w = \eta_t \times i_g \times i_f \times T_m \quad \text{หรือ} \quad \eta_t \times i_0 \times T_m$$

$T_w$  = แรงบิดที่ล้อ, N·m

$\eta_t$  = ประสิทธิภาพการส่งกำลัง

$i_g = \frac{N_m}{N_w}$  = อัตราการทดระหว่างจำนวนฟันเฟืองขับมอเตอร์และเฟืองขับล้อ

$i_f = \frac{d_s}{d_w}$  = อัตราการทดระหว่างเส้นผ่านศูนย์กลางเฟืองขับล้อกับเส้นผ่านศูนย์กลางของล้อ

$i_0 = i_g \times i_f$  = อัตราทดของระบบส่งกำลัง

สมการความสัมพันธ์แรงขับเคลื่อนกับทอร์กของล้อ

$$F = \frac{T_w}{r} \text{ หรือ } \frac{\eta_t i_g i_f T_m}{r}$$

$F$  = แรงขับเคลื่อนที่ต้องการเอาชนะแรงต้าน, N

$r$  = รัศมีของล้อ, m

การคำนวณหาแรงต้านการเคลื่อนที่

$$F_r = K_r N = (0.015 + 0.00016V) mg = \text{แรงต้านจากการหมุน}$$

$K_r$  = สัมประสิทธิ์แรงต้านจากการหมุนของล้อ

$V$  = อัตราเร็ว, km/hr

$N$  = แรงปฏิกิริยาที่กระทำพื้นกระทำต่อหุ่นยนต์, N·m

$$F_a = K_a A V^2 = (0.03) (W \times L) V^2 = \text{แรงต้านอากาศ}$$

$K_a$  = สัมประสิทธิ์แรงต้านอากาศ

$A$  = พื้นที่หน้าตัดของหุ่นยนต์,  $m^2$

$W$  = ความกว้างของหุ่นยนต์, m

$L$  = ความสูงของหุ่นยนต์, m

$$F_a = mg \sin(\theta) = \text{แรงต้านจากทางชัน}$$

แรงต้านการเคลื่อนที่รวม

$$F = F_r + F_a + F_g$$

การหาความเร็วรอบที่ต้องการ

$$\text{อัตราเร็วรอบของมอเตอร์} = N \text{ rps}$$

$$\text{อัตราเร็วของล้อ} = \frac{N}{i_0} \text{ rps}$$

$$\text{เส้นรอบวงของล้อ 1 รอบ} = 2\pi r m$$

$$\text{อัตราเร็วของหุ่นยนต์} = 2\pi r \left[ \frac{N}{i_0} \right] \text{ m/s หรือ } 2\pi r \left[ \frac{N}{i_0} \right] \left[ \frac{3600}{1000} \right] \text{ km/hr}$$

สมการการหาอัตราเร็วของหุ่นยนต์

$$V = \frac{3.6(2\pi r)N}{i_0} \text{ km/hr}$$

สมการการหากำลังขับเคลื่อน

$$P_w = \frac{FV}{3.6}$$

ประสิทธิภาพการส่งกำลัง

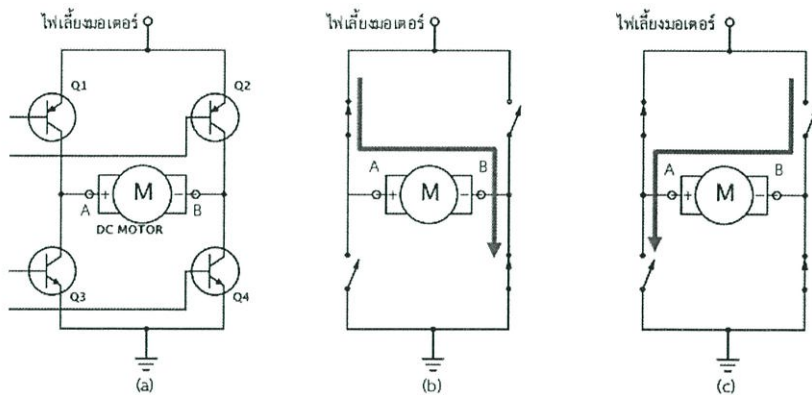
$$\eta_t = \frac{P_w}{P_m}$$

การหกำลังมอเตอร์ที่ต้องการ

$$P_m = \frac{FV}{3.6\eta_t} \text{ watt}$$

## 2.3 ชุดขับกระแสมอเตอร์กระแสตรง

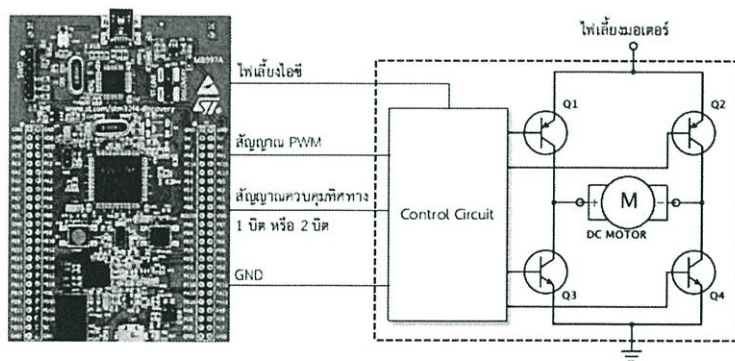
ชุดขับกระแสดีซีมอเตอร์ส่วนใหญ่จะเป็นแบบ H-Bridge ซึ่งวงจรประกอบด้วยทรานซิสเตอร์หรือมอดเฟส<sup>[7]</sup> โดยทำหน้าที่เป็นสวิตช์เปิดปิด จำนวน 4 ชุด (Q1-Q4) โดยต่อกับ DC Motor ซึ่งสามารถควบคุมการทิศทางการไหลของกระแสได้ เมื่อส่งสัญญาณควบคุมให้ทรานซิสเตอร์ Q1 และ Q4 ทำงาน และปิดการทำงานของทรานซิสเตอร์ Q2 และ Q3 กระแสจะไหลจากจุด A ไปจุด B จึงทำให้มอเตอร์เริ่มหมุนเมื่อส่งสัญญาณควบคุมให้ทรานซิสเตอร์ Q2 และ Q3 ทำงาน และปิดการทำงานของทรานซิสเตอร์ Q1 และ Q4 กระแสจะไหลจากจุด B ไปจุด A เป็นผลให้มอเตอร์หมุนกลับทิศ



รูปที่ 2.5 การทำงานของวงจร H-Bridge สำหรับควบคุมการหมุนของ DC Motor

ที่มา ONSemiconductor(2013)

ปัจจุบันวงจร H-Bridge มีอยู่ในรูปของไอซีวงจรรวม เช่น ไอซีเบอร์ L293D หรือ L298N เพื่อลดขนาดของอุปกรณ์ และง่ายต่อการใช้งาน เพียงต่อสัญญาณ PWM และสัญญาณควบคุมทิศทาง (1 บิตหรือ 2 บิต)<sup>[8]</sup>

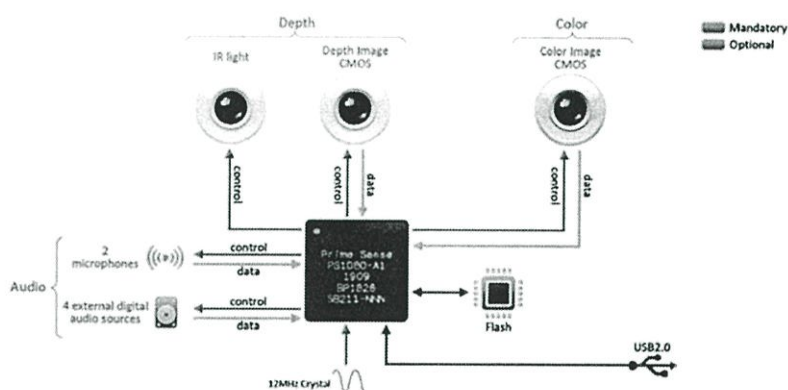


รูปที่ 2.6 การเชื่อมต่อ Microcontroller กับชุดขับกระแสดีซีมอเตอร์แบบ H-Bridge

ที่มา ONSemiconductor(2013)

## 2.4 Kinect Xbox 360

ภายใน Kinect ประกอบด้วย อุปกรณ์ฉายแสงอินฟราเรด (Infrared) กล้องวัดความลึกของภาพ (Depth Camera) กล้องวิดีโอ (Video Camera) ไมโครโฟน และเซนเซอร์ (Sensor) ประมวลผลมีการทำงานเริ่มจากการฉายแสงอินฟราเรดออกจากตัว Kinect ซึ่งไม่สามารถมองเห็นได้ด้วยตาเปล่า<sup>[9]</sup> แสงที่ถูกฉายออกมาจะมีลักษณะเป็นจุดๆตามแนวตั้ง 480 จุด แนวนอน 640 จุด แต่ละจุดห่างกัน 3 มิลลิเมตร (ที่ระยะสองเมตรจากแหล่งกำเนิดแสง)หลังจากนั้น กล้องวัดความลึกจะรับภาพ ระดับความสว่างของแสงอินฟราเรดที่ตกกระทบลงบนวัตถุ สะท้อนกลับมาที่เซนเซอร์ก็จะสามารถคำนวณพิกัด x, y และ z ของจุดกระทบนั้นได้ทำให้สามารถจำลองสภาพแวดล้อมเป็นสามมิติได้



รูปที่ 2.7 ส่วนประกอบของกล้อง Kinect Xbox 360

ที่มา: Computer Society (2017)

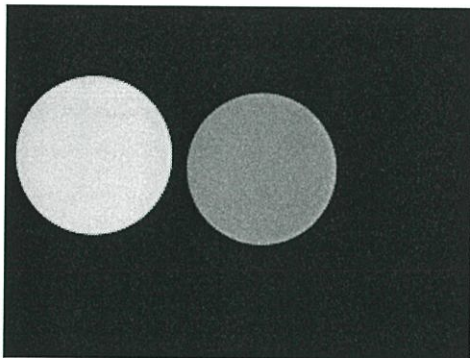
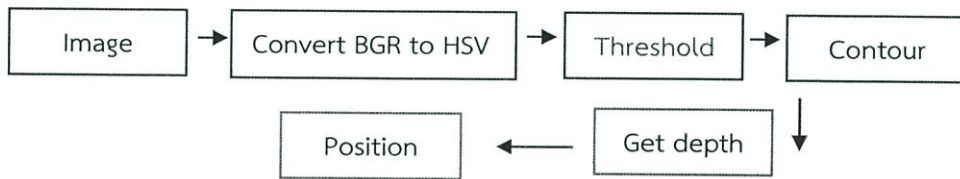
## 2.5 การประมวลผลภาพ

Image Processing หรือการประมวลผลภาพเป็นกระบวนการจัดการและวิเคราะห์สารสนเทศของรูปภาพ โดยใช้คอมพิวเตอร์ในการประมวลผล ซึ่งกระบวนการต่าง ๆ มีอยู่ด้วยกันหลายอย่าง ยกตัวอย่างเช่น การแบ่งภาพและการหาขอบภาพในวัตถุ (Image Segmentation and EdgDeTecton)<sup>[10]</sup>

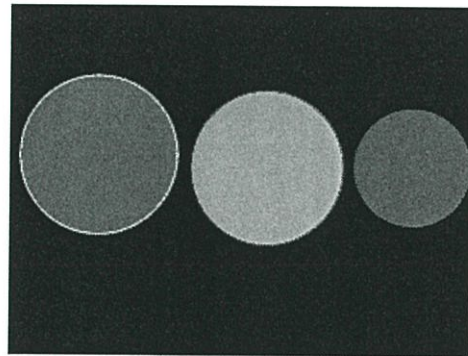
กระบวนการ Segmentation เป็นกระบวนการแยกวัตถุ หรือองค์ประกอบต่างๆ ออกจากภาพอินพุท วิธีการ Intensity thresholding เป็นการแยกแยะวัตถุจากฉากหลังโดยดูจาก Intensity ของพิกเซลเป็นหลัก ข้อดีของวิธีการนี้คือมีขั้นตอนในการทำงานที่ง่ายไม่ซับซ้อนทำงานได้รวดเร็ว แต่ข้อเสียคือไม่สามารถใช้กับภาพที่มีสัญญาณรบกวนมาก หรือภาพที่มีความสว่างไม่สม่ำเสมอ เทคนิคการทำขีดแบ่ง (Thresholding Techniques) เป็นการพิจารณาว่าจุดภาพใดควรเป็นจุดขาวหรือจุดดำ ซึ่งทำได้โดยการเปรียบเทียบระหว่างจุดภาพเริ่มต้นกับค่าคงที่ค่าหนึ่ง เรียกว่าค่าขีดแบ่ง ซึ่งเป็นค่าความ

เข้มแสงค่าหนึ่งที่ใช้แยกแยะประเภทของจุดภาพ<sup>[11]</sup> เทคนิคนี้ใช้กันมากในกรณีที่ข้อมูลภาพมีลักษณะแตกต่างกันระหว่างวัตถุและพื้นหลัง ในขั้นตอนนี้ทำการตัดพื้นหลังออกจากพื้นหน้าโดยใช้วิธี Threshold ที่มีระดับความเข้มอยู่ระหว่างกลุ่มทั้งสองของ Histogram ซึ่งค่า Threshold ที่ได้จะอยู่ระหว่าง 0-255 เท่านั้น Threshold จะถูกนำไปเพื่อเปรียบเทียบค่าของแต่ละพิกเซล หากค่า  $f(x, y)$  น้อยกว่า Threshold จุดพิกเซลนั้นจะถูกปรับให้เป็นสีดำหรือส่วนของวัตถุ และหากค่า  $f(x, y)$  มากกว่าหรือเท่ากับ Threshold จุดพิกเซลนั้นจะถูกปรับให้เป็นสีขาวหรือส่วนของพื้นหลัง สามารถเขียนแทนด้วยสมการดังนี้

$$f_{thr}(x, y) = \begin{cases} 1, & f_f(x, y) < Threshold \\ 0, & f_f(x, y) \geq Threshold \end{cases}$$



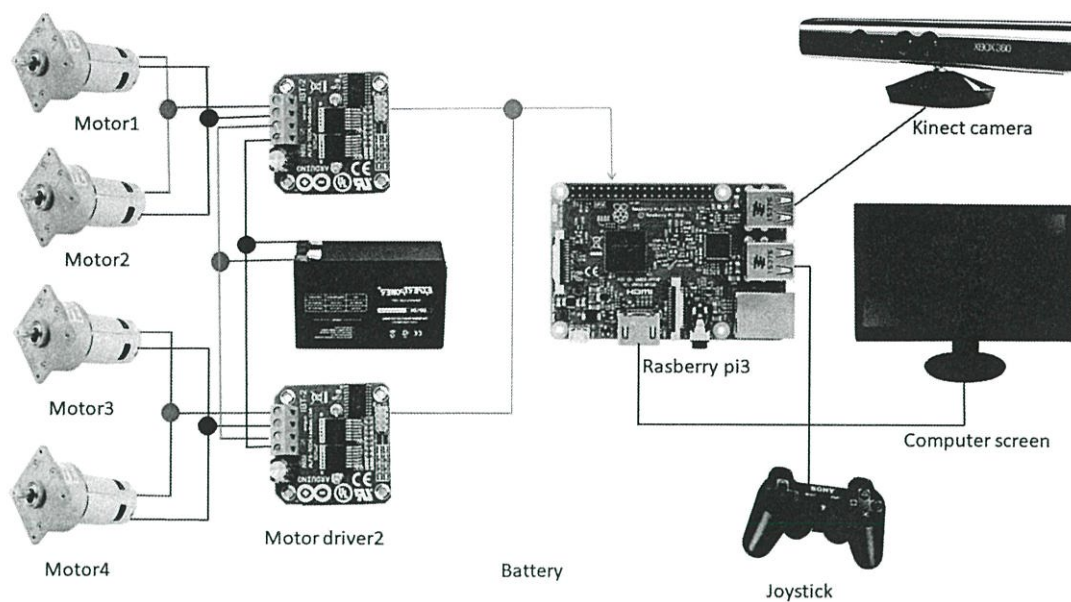
Input Image



Three level threshold output

รูปที่ 2.8 ตัวอย่างการ Threshold ที่มา Quora(2017)

## 2.6 ภาพรวมระบบของการทำงานหุ่นยนต์ขับเคลื่อน 4 ล้อ

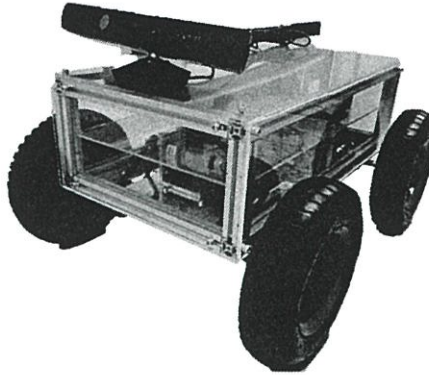


รูปที่ 2.9 ภาพรวมระบบของการทำงานหุ่นยนต์ขับเคลื่อน 4 ล้อ

### บทที่ 3

#### ขั้นตอนการดำเนินงาน

#### 3.1 ขั้นตอนการออกแบบและสร้างหุ่นยนต์ขับเคลื่อน 4 ล้อ

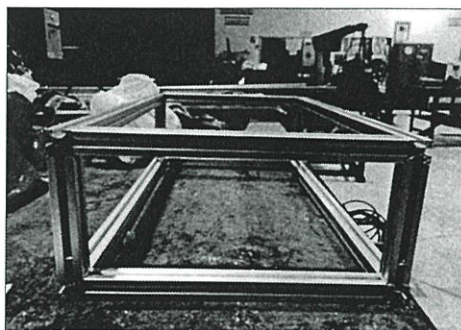


รูปที่ 3.1 หุ่นยนต์ขับเคลื่อน 4 ล้อ

#### 3.1.1 การออกแบบโครงสร้างของหุ่นยนต์ขับเคลื่อน 4 ล้อ

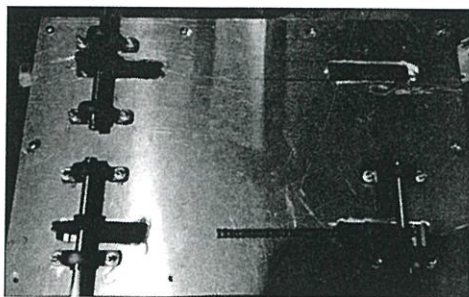
##### 3.1.1.1 การเลือกอุปกรณ์มาเป็นโครงของหุ่นยนต์

- ส่วนของตัวโครงของหุ่นยนต์ขับเคลื่อนสี่ล้อ เลือกใช้อลูมิเนียมโปรไฟล์ เนื่องจากสามารถประกอบได้รวดเร็วและมีน้ำหนักเบา



รูปที่ 3.2 โครงของหุ่นยนต์ขับเคลื่อน 4 ล้อ

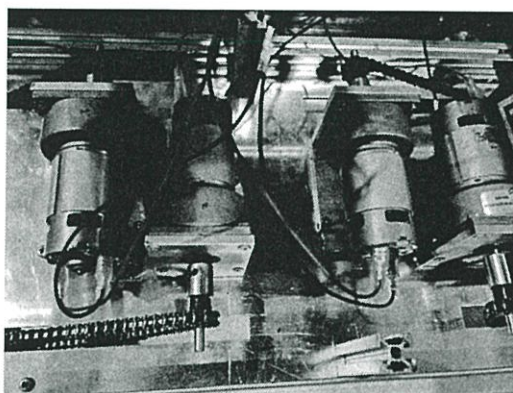
- ส่วนท้องด้านล่างของหุ่นยนต์เลือกใช้เป็นอลูมิเนียมแผ่นเนื่องจากสามารถรับน้ำหนัก อุปกรณ์ที่ใช้ในการขับเคลื่อนการเคลื่อนที่ของหุ่นยนต์ขับเคลื่อนสี่ล้อได้



รูปที่ 3.3 แผนอลูมิเนียม

### 3.1.1.2 การเลือกขนาดมอเตอร์

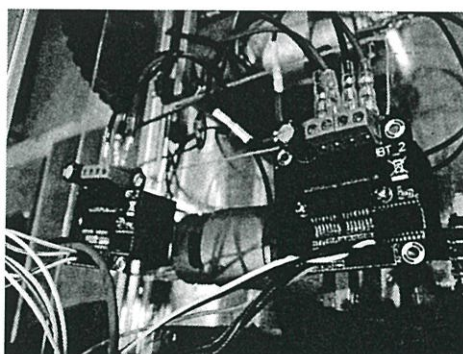
- เลือกใช้มอเตอร์ชนิด DC gear motor 12 โวลต์ 400 รอบทั้งหมด 4 ตัว เพื่อใช้เป็นตัวขับเคลื่อนแต่ละล้อ เนื่องจากมอเตอร์แต่ละตัวรับน้ำหนักได้ 4 กิโลกรัม



รูปที่ 3.4 มอเตอร์กระแสตรง 12 โวลต์ 400 รอบ

### 3.1.1.3 การเลือก H-bridge motor driver

- เลือกใช้ชุดขับเคลื่อนมอเตอร์แบบเพื่อควบคุมความเร็วและทิศทางการไหลของกระแสในมอเตอร์ซึ่งมีผลต่อการเคลื่อนที่ของหุ่นยนต์(รุ่น BTS7960B)



รูปที่ 3.5 H-bridge motor driver รุ่น BTS7960B

### 3.1.2 ส่วนฮาร์ดแวร์

- หน่วยประมวลผลกลางเลือกใช้ Raspberry Pi3 Model B
- ใช้ Wireless Joysticks ในการควบคุมหุ่นยนต์ขับเคลื่อนสี่ล้อ

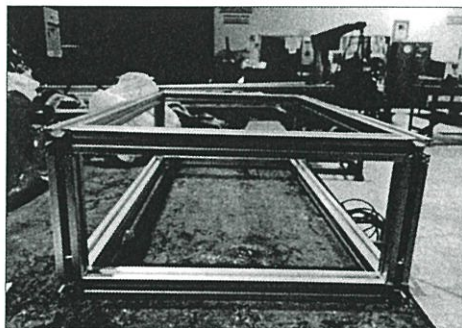
### 3.1.3 ส่วนของซอฟต์แวร์

- สำหรับซอฟต์แวร์ที่ใช้ในระบบปฏิบัติการคือ Linux
- ภาษาที่ใช้เขียนนั้นคือภาษา Python

Python คือชื่อภาษาที่ใช้ในการเขียนโปรแกรมภาษาหนึ่ง ซึ่งถูกพัฒนาขึ้นมาโดยไม่มีติดกับแพลตฟอร์ม กล่าวคือสามารถรันภาษา Python ได้ทั้งบนระบบ Unix, Linux , Windows NT, Windows 2000, Windows XP หรือแม้แต่ระบบ FreeBSD อีกอย่างหนึ่งภาษาตัวนี้ เป็น OpenSource

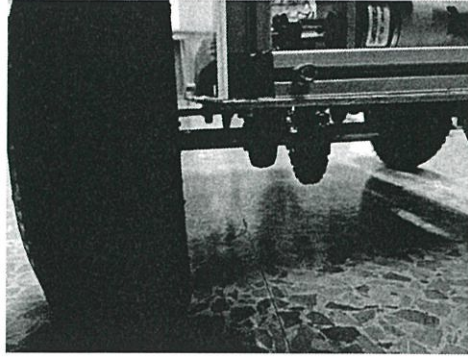
## 3.2 ขั้นตอนการประกอบหุ่นยนต์

- 1) ทำโครงของหุ่นยนต์จากอลูมิเนียมโปรไฟล์



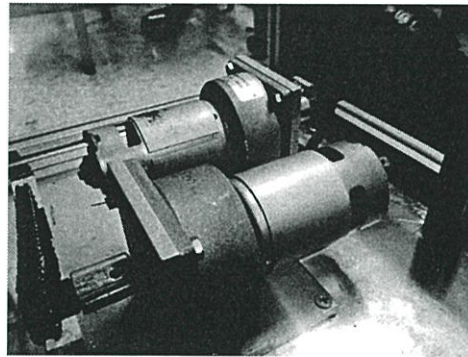
รูปที่ 3.5 โครงของหุ่นยนต์จากอลูมิเนียมโปรไฟล์

2) ติดตั้งเพลาล้อ และแบร์ริงตุ๊กตา



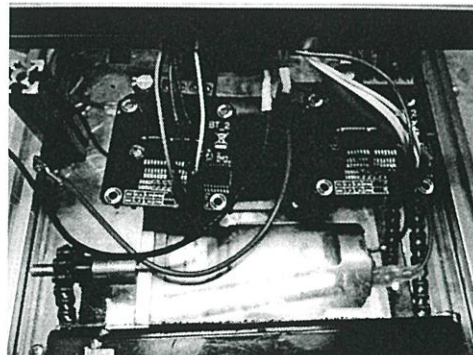
รูปที่ 3.6 เพลาล้อ และแบร์ริงตุ๊กตา

3) ติดตั้งมอเตอร์และชุดเฟืองทด พร้อมทั้งโซ่



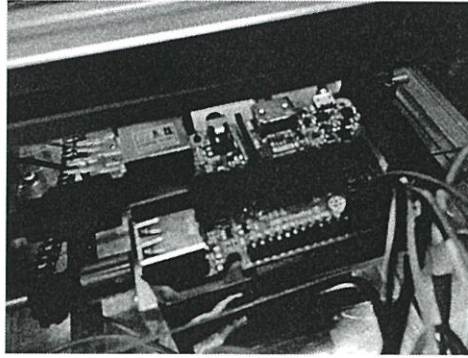
รูปที่ 3.7 มอเตอร์และชุดเฟืองทด พร้อมทั้งโซ่

4) ติดตั้ง H-bridge motor driver



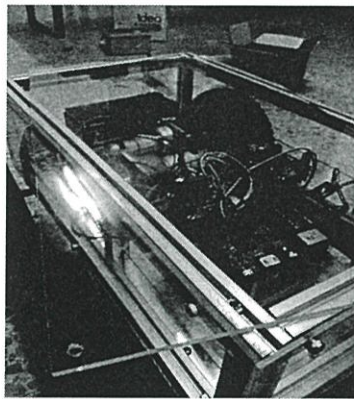
รูปที่ 3.8 H-bridge motor driver

5) ติดตั้ง Raspberry Pi 3 Model B



รูปที่ 3.9 Raspberry Pi 3 Model B

6) ติดแผ่นอะคริลิกให้กับหุ่นยนต์



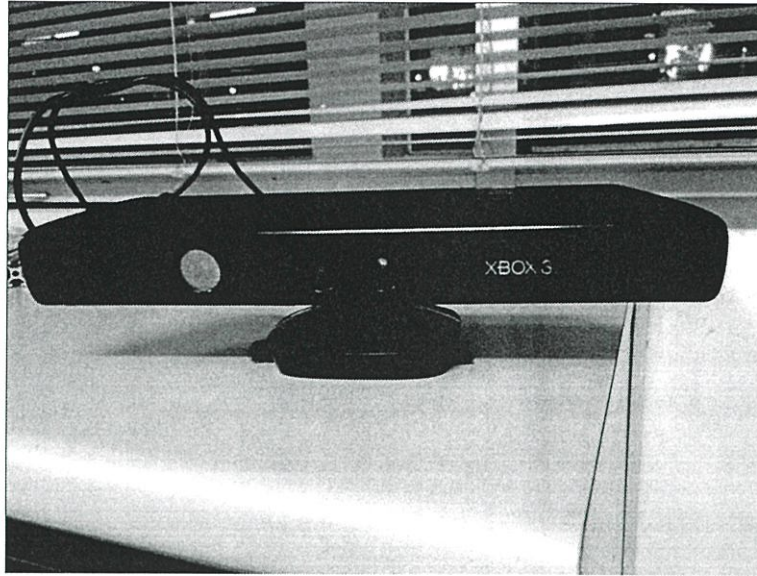
รูปที่ 3.10 การติดตั้งแผ่นอะคริลิก

7).ติดตั้งกล้อง Kinect Xbox 360



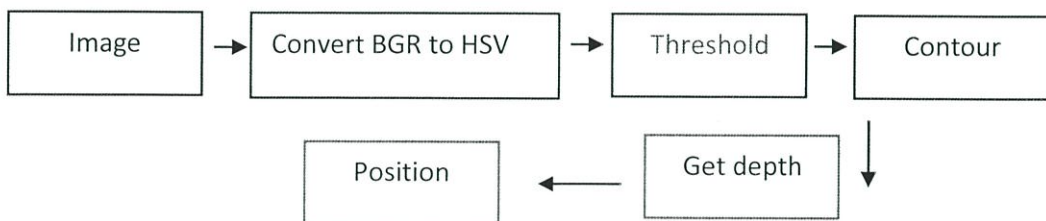
รูปที่ 3.11 ลักษณะการติดตั้งกล้อง Kinect Xbox 360

### 3.3 ส่วนของการวิเคราะห์และระบุตำแหน่ง



รูปที่ 3.12 กล้อง Kinect Xbox 360

- เลือกใช้กล้อง Kinect Xbox 360 เพราะเป็นกล้องที่สามารถฉายแสงอินฟราเรดไปกระทบกับวัตถุและมีเซ็นเซอร์วัดความลึกที่จะทำการวัดความลึกจากระดับความสว่างของแสงอินฟราเรดที่ไปตกกระทบกับวัตถุทำให้ง่ายต่อการคำนวณและระบุตำแหน่งของวัตถุ
- ซอฟต์แวร์ที่ใช้ คือ Opencv
- โดยมีขั้นตอนการวิเคราะห์และระบุตำแหน่งเป้าหมาย ดังนี้



รูปที่ 3.13 Flow chartแสดงการทำงานของระบบประมวลผลภาพ

สูตรที่ใช้ในการคำนวณตำแหน่งของเป้าหมาย

$$Z_{\text{meters}} = \frac{-D_{\text{Processed}}}{1000}$$

$$X_{\text{meters}} = 6 - 2 \left( \text{Pixel}_x - \frac{639}{2} \right) \tan\left(\frac{57^\circ}{2}\right) \frac{Z_{\text{meters}}}{640}$$

$$Y_{\text{meters}} = 4.5 - 2 \left( 479 - \text{Pixel}_y - \frac{479}{2} \right) \tan\left(\frac{43^\circ}{2}\right) \frac{Z_{\text{meters}}}{480}$$

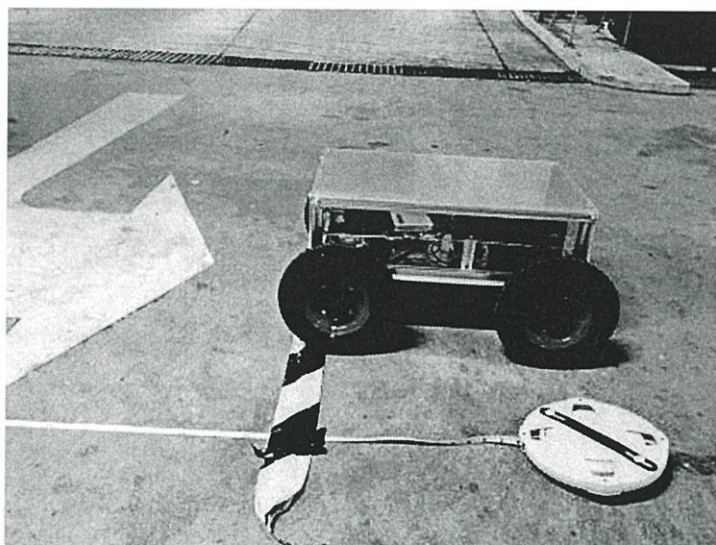
## บทที่ 4

### วิธีการทดสอบ

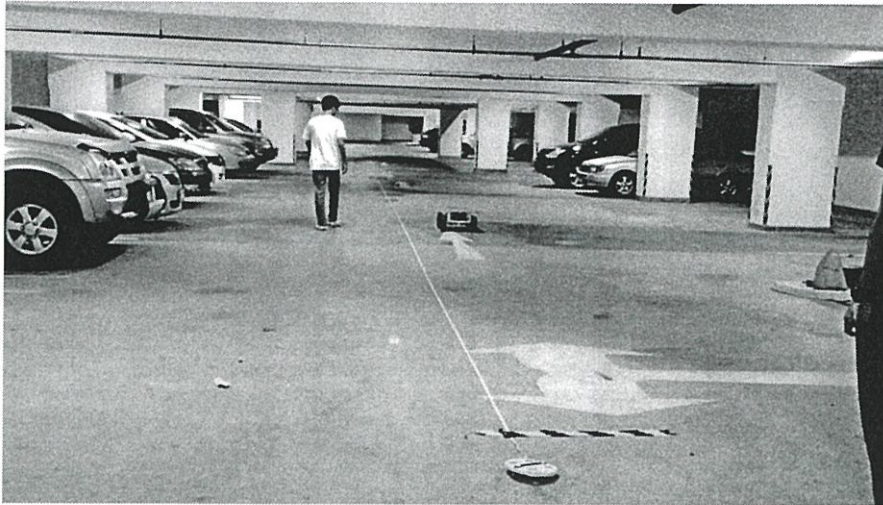
#### 4.1 การทดสอบสมรรถนะของหุ่นยนต์ขับเคลื่อน 4 ล้อ

##### 4.1.1 การทดสอบการสิ้นเปลืองขณะเคลื่อนที่และการรับน้ำหนักของหุ่นยนต์ขับเคลื่อน 4 ล้อ

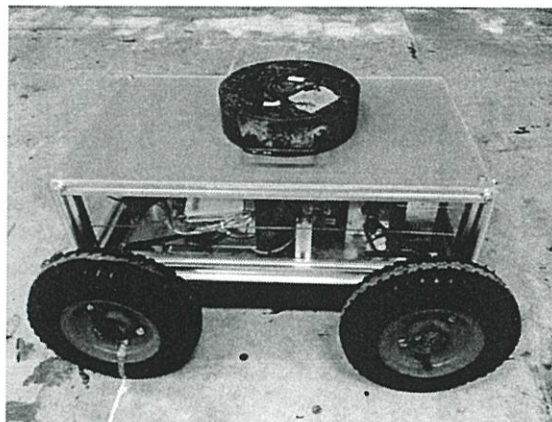
- 1) ทำการจัดเตรียมอุปกรณ์ หุ่นยนต์ และสนามทดสอบ
- 2) นำหุ่นยนต์มาจอดเทียบเส้นที่จุด start
- 2) ขับเคลื่อนหุ่นยนต์เป็นระยะทาง 20.44 เมตร พร้อมกับนำจำนวนรอบล้อหมุนจากเส้นอ้างอิง
- 3) นำค่ารอบของล้อจากจุดอ้างอิงของหุ่นยนต์มาเปรียบเทียบกับจำนวนรอบล้อที่ได้จากทฤษฎีเพื่อหาระยะการสิ้นเปลือง
- 4) ทำซ้ำในข้อ 1 และ 2 จำนวน 3 ครั้ง แล้วจดบันทึกค่า
- 5) ทำการเพิ่มน้ำหนักให้กับหุ่นยนต์ที่ละ 10 กิโลกรัม แล้วทำตามในข้อ 1 และ 2 จำนวน 3 ครั้งแล้วจดบันทึกค่า



รูปที่ 4.1 นำหุ่นยนต์มาจอดที่จุด start



รูปที่ 4.2 การขับเคลื่อนหุ่นยนต์และการนำร่องอัตโนมัติ



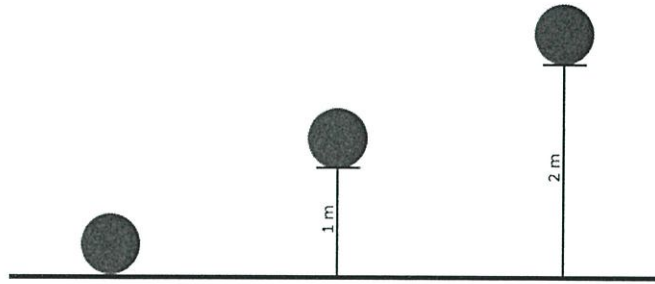
รูปที่ 4.3 การเพิ่มน้ำหนักในการทดสอบการสิ้นไกลขณะเคลื่อนที่และการรับน้ำหนักของหุ่นยนต์ขับเคลื่อน 4 ล้อ

#### 4.2 การทดสอบการวิเคราะห์และระบุตำแหน่งสิ่งแปลกปลอมในตำแหน่งที่แตกต่างกัน

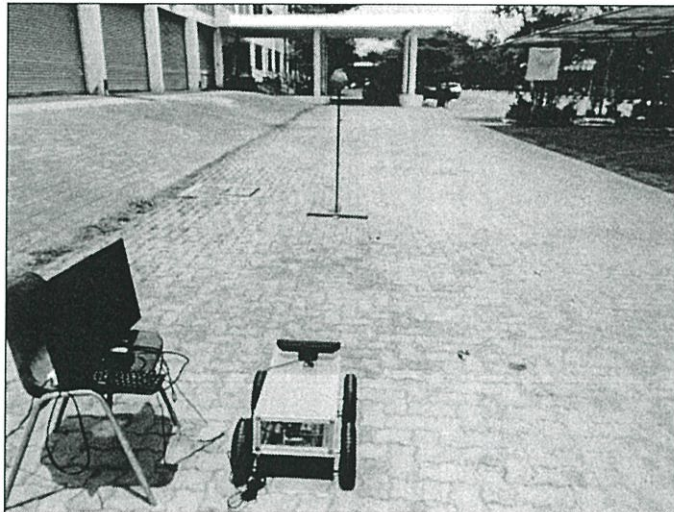
เป็นการทดสอบเพื่อตรวจสอบการวิเคราะห์และระบุตำแหน่งของสิ่งแปลกปลอม โดยจำลองสิ่งแปลกปลอมเป็นเป้าทรงกลมสีแดง โดยมีการวิธีการทดสอบ ดังนี้

- 1) จัดเตรียมอุปกรณ์ในการเก็บข้อมูล กล้อง เป้าจำลองและระบบวิเคราะห์
- 2) ทำการวางเป้าจำลองในตำแหน่งที่แตกต่างกัน 18 จุด โดยให้กล้องวางอยู่กับที่และวิเคราะห์ตำแหน่งของเป้าจำลอง
- 3) ทดสอบซ้ำในข้อ 2 ทั้งหมด 3 ครั้ง จนครบทุกตำแหน่งของเป้าจำลองที่กำหนดในแผนการทดลอง

- 4) บันทึกผลการทดสอบ ทำการวิเคราะห์ เปรียบเทียบความผิดพลาดระหว่างระยะจากกล้องถึงเป้าจำลองทางปฏิบัติและทฤษฎี



รูปที่ 4.4 ตำแหน่งการวางลูกบอลทดสอบ



รูปที่ 4.5 การทดสอบการวิเคราะห์และระบุตำแหน่งสิ่งแปลกปลอมในตำแหน่งที่

แตกต่างกัน

## บทที่ 5

## ผลการทดสอบและวิเคราะห์ผลการทดสอบ

## 5.1 ผลการทดสอบการลื่นไถลของล้อและการรับน้ำหนักของหุ่นยนต์ขับเคลื่อน 4 ล้อ

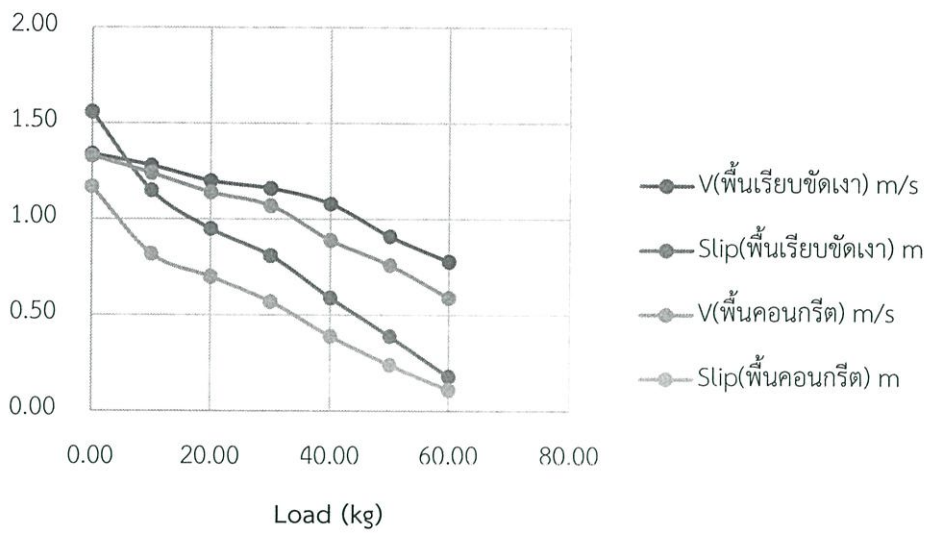
ได้พัฒนาหุ่นยนต์ขับเคลื่อน 4 ล้อที่มีน้ำหนัก 12.3 kg สามารถวิ่งด้วยความเร็ว 1.24 m/s ที่โหลด 10kg และวิ่งด้วยความเร็ว 0.76 m/s ที่โหลด 50 kg บนพื้นผิวคอนกรีต

## ตารางที่ 1 การทดสอบการลื่นไถลของล้อและการรับน้ำหนักในพื้นที่คอนกรีตเรียบขัดเงา

Load (kg)	Repeat	Rev at 32		Time (s)	Velocity (m/s)	Slip (m)	T เฉลี่ย	V เฉลี่ย m/s	Slip เฉลี่ย m
		Actual	Theory						
0	1	19.11	20.44	13.77	1.39	1.33	14.11	1.34	1.56
	2	18.55	20.44	14.26	1.30	1.89			
	3	18.98	20.44	14.30	1.33	1.46			
10	1	18.78	20.44	15.39	1.22	1.66	15.07	1.28	1.15
	2	19.58	20.44	14.74	1.33	0.86			
	3	19.50	20.44	15.09	1.29	0.94			
20	1	19.42	20.44	16.07	1.21	1.02	16.14	1.21	0.95
	2	19.64	20.44	16.21	1.21	0.80			
	3	19.42	20.44	16.15	1.20	1.02			
30	1	19.73	20.44	16.98	1.16	0.71	17.00	1.16	0.81
	2	19.50	20.44	17.24	1.13	0.94			
	3	19.66	20.44	16.77	1.17	0.78			
40	1	19.97	20.44	17.98	1.11	0.47	18.42	1.08	0.59
	2	19.80	20.44	18.84	1.05	0.64			
	3	19.77	20.44	18.43	1.07	0.67			
50	1	19.97	20.44	21.64	0.92	0.47	22.14	0.91	0.39
	2	19.99	20.44	22.88	0.87	0.45			
	3	20.18	20.44	21.91	0.92	0.26			
60	1	20.32	20.44	25.11	0.81	0.12	25.84	0.78	0.18
	2	20.20	20.44	26.41	0.76	0.24			
	3	20.25	20.44	25.99	0.78	0.19			

ตารางที่ 2 การทดสอบการลื่นไถลของล้อและการรับน้ำหนักในพื้นคอนกรีต

Load (kg)	Repeat	Rev at 32		Time (s)	Velocity (m/s)	Slip (m)	Tเฉลี่ย	Vเฉลี่ย m/s	Slipเฉลี่ย (m)
		Actual	Theory						
0	1	19.23	20.44	15.19	1.27	1.21	14.68	1.31	1.17
	2	19.35	20.44	14.52	1.33	1.09			
	3	19.22	20.44	14.33	1.34	1.22			
10	1	19.66	20.44	16.17	1.22	0.78	15.87	1.24	0.82
	2	19.62	20.44	15.89	1.23	0.82			
	3	19.58	20.44	15.55	1.26	0.86			
20	1	20.13	20.44	16.88	1.19	0.31	17.38	1.14	0.70
	2	19.34	20.44	18.22	1.06	1.10			
	3	19.74	20.44	17.04	1.16	0.70			
30	1	19.82	20.44	19.04	1.04	0.62	18.87	1.05	0.57
	2	19.76	20.44	18.89	1.05	0.68			
	3	20.02	20.44	18.67	1.07	0.42			
40	1	20.19	20.44	22.45	0.90	0.25	22.70	0.88	0.39
	2	20.09	20.44	22.65	0.89	0.35			
	3	19.88	20.44	23.01	0.86	0.56			
50	1	20.21	20.44	26.45	0.76	0.23	26.58	0.76	0.24
	2	20.05	20.44	26.76	0.75	0.39			
	3	20.33	20.44	26.54	0.77	0.11			
60	1	20.29	20.44	32.76	0.62	0.15	34.45	0.59	0.11
	2	20.34	20.44	34.94	0.58	0.10			
	3	20.35	20.44	35.65	0.57	0.09			



รูปที่ 5.1 ความสัมพันธ์ระหว่าง Load กับ Velocity

## 5.2 ผลการทดสอบการวิเคราะห์และระบุตำแหน่งสิ่งแปลกปลอมในตำแหน่งที่แตกต่างกัน

จากการทดสอบการวิเคราะห์และระบุตำแหน่งพบว่าจากการวางสิ่งแปลกปลอมจำลองที่ตำแหน่งแตกต่างกัน 18 จุดโดยเมื่อระยะห่างของการวางสิ่งแปลกปลอมจำลองจากตัวกล้องเพิ่มขึ้น ความคลาดเคลื่อนของการวิเคราะห์และการระบุตำแหน่งก็จะเพิ่มขึ้นเช่นกัน

ตารางที่ 3 ผลการทดลองการวิเคราะห์และระบุตำแหน่งเป้าหมาย

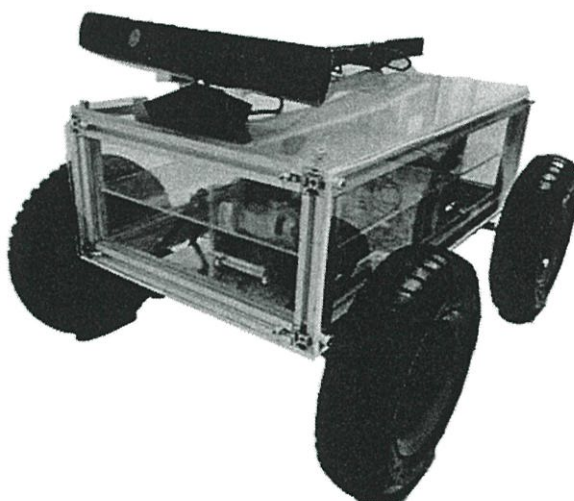
	Actual (m)			Detect (m)			Error (m)		
	X	Y	Z	x	Y	z	x	y	Z
1	0.00	2.00	0.00	0.02	2.04	0.05	0.02	0.04	0.05
2	0.00	2.00	1.00	0.01	2.06	0.95	0.01	0.06	-0.05
3	0.00	2.00	2.00	-	-	-	-	-	-
4	0.00	4.00	0.00	0.07	3.94	0.01	0.07	-0.06	0.01
5	0.00	4.00	1.00	0.08	4.12	1.11	0.08	0.12	0.11
6	0.00	4.00	2.00	0.04	4.22	2.23	0.04	0.22	0.23
7	0.00	6.00	0.00	0.05	6.23	0.12	0.05	0.23	0.12
8	0.00	6.00	1.00	0.09	6.37	1.37	0.09	0.37	0.37
9	0.00	6.00	2.00	-	-	-	-	-	-
10	1.50	2.00	0.00	1.56	2.06	0.08	0.06	0.06	0.08
11	1.50	2.00	1.00	1.62	2.13	1.19	0.12	0.13	0.19
12	1.50	2.00	2.00	-	-	-	-	-	-
13	1.50	4.00	0.00	1.68	3.94	0.09	0.18	-0.06	0.09
14	1.50	4.00	1.00	1.59	4.05	1.27	0.09	0.05	0.27
15	1.50	4.00	2.00	1.62	4.18	2.37	0.12	0.18	0.37
16	1.50	6.00	0.00	1.78	6.33	0.17	0.28	0.33	0.17
17	1.50	6.00	1.00	1.69	6.43	1.53	0.19	0.43	0.53
18	1.50	6.00	2.00	-	-	-	-	-	-

## บทที่ 6

### สรุปผลการทดลอง

#### 6.1 การสร้างหุ่นยนต์ต้นแบบขับเคลื่อน 4 ล้อ เพื่อการสำรวจสวนผลไม้

การสร้างหุ่นยนต์ต้นแบบขับเคลื่อน 4 ล้อ เพื่อการสำรวจสวนผลไม้ ได้สร้างหุ่นยนต์ที่มีน้ำหนัก 12.3 กิโลกรัม มีความกว้าง 30 เซนติเมตร มีความยาว 50 เซนติเมตร และมีอุปกรณ์ที่ติดตั้งได้แก่ มอเตอร์กระแสตรง 12 โวลต์, 400 รอบ, H-bridge motor driver, Raspberry Pi model b Kinect โดยหุ่นยนต์ขับเคลื่อน 4 ล้อ ทำการควบคุมด้วย Wireless joystick และมีกล้อง Kinect Xbox 360 ที่นำมาประยุกต์ใช้ในระบบการวิเคราะห์และระบุตำแหน่ง



รูปที่ 6.1 หุ่นยนต์ขับเคลื่อน 4 ล้อเพื่อการสำรวจสวนผลไม้

#### 6.2 สรุปผลการทดลอง

ผลการศึกษาการทดสอบการสิ้นเปลืองของล้อและการรับน้ำหนักของหุ่นยนต์พบว่าที่ Load 0 กิโลกรัม หุ่นยนต์มีความเร็วการเคลื่อนที่ 1.34 เมตร/วินาที มีค่าการสิ้นเปลือง 1.56 เมตร และเมื่อเพิ่ม Load 30 กิโลกรัมให้กับหุ่นยนต์หุ่นยนต์มีความเร็วการเคลื่อนที่ 1.16 เมตร/วินาที มีค่าการสิ้นเปลือง 0.81 เมตร และเมื่อเพิ่ม Load 60 กิโลกรัม หุ่นยนต์มีความเร็วการเคลื่อนที่ 0.78 เมตร/วินาที มีค่าการสิ้นเปลือง 0.18 เมตร ที่พื้นคอนกรีตเรียบขัดเงา และสำหรับในพื้นที่คอนกรีตพบว่า Load 0 กิโลกรัม หุ่นยนต์มีความเร็วการเคลื่อนที่ 1.31 เมตร/วินาที มีค่าการสิ้นเปลือง 1.17 เมตร และเมื่อเพิ่ม Load 30

กิโลกรัมให้กับหุ่นยนต์หุ่นยนต์มีความเร็วการเคลื่อนที่ 1.05 เมตร/วินาที มีค่าการสิ้นเปลือง 0.57 เมตร และเมื่อเพิ่ม Load 60 กิโลกรัม หุ่นยนต์มีความเร็วการเคลื่อนที่ 0.59 เมตร/วินาที มีค่าการสิ้นเปลือง 0.11 เมตร

ในส่วนของการวิเคราะห์และระบุตำแหน่งเป้าหมายพบว่าจากการทดลองการวิเคราะห์และระบุตำแหน่งพบว่าจากการวางสิ่งแปลกปลอมจำลองที่ระยะ 2, 4 และ 6 เมตร เมื่อระยะห่างของการวางสิ่งแปลกปลอมจำลองจากตัวกล้องเพิ่มขึ้นความคลาดเคลื่อนของการวิเคราะห์และการระบุตำแหน่งก็จะเพิ่มขึ้นเช่นกัน

### 6.3 ข้อเสนอแนะ

- จากการสร้างและพัฒนาหุ่นยนต์ขับเคลื่อน 4 ล้อพบว่าควรเพิ่มระบบการระบุตำแหน่งของหุ่นยนต์และควรพัฒนาให้สามารถเคลื่อนที่โดยอัตโนมัติแทนการควบคุมด้วย Wireless Joysticks

- ควรใช้กล้องที่มีประสิทธิภาพมากกว่าเดิม เนื่องจากการทดสอบการวิเคราะห์และระบุตำแหน่งเป้าหมายพบว่ากล้องที่ใช้ยังมีความสามารถในการโฟกัสสีในระยะที่ไกลเกิน 6 เมตรได้ดีไม่เพียงพอทำให้เกิดความผิดพลาดในการวิเคราะห์

## เอกสารอ้างอิง

- [1]จรรยา สุทธิไชยา. 2560. สถิติการเกษตรของประเทศไทย ปี ๒๕๕๙. [Online].Available :  
[http://www.oae.go.th/download/download\\_journal/2560/yearbook59.pdf](http://www.oae.go.th/download/download_journal/2560/yearbook59.pdf)
- [2]ประสาธ เกศวิทักษ์. 2555. สินค้าเกษตร.....ก้าวสู่โลกยุคใหม่. [Online].Available :  
[http://www.thaifert.com/knowledge\\_detail.php?id=12](http://www.thaifert.com/knowledge_detail.php?id=12)
- [3]kamolc. 2556. การเคลื่อนที่บนพื้นด้วยล้อและสายพาน. [Online].Available :  
<https://www.thairobotics.com/2013/10/11/byob-wheel-and-track/>
- [4]kamolc. 2556. การเคลื่อนที่บนพื้นด้วยล้อและสายพาน. [Online].Available :  
<https://www.thairobotics.com/2013/10/25/byob-wheel-and-track-2/>
- [5]Factomart. 2560. มอเตอร์ไฟฟ้า. [Online].Available  
[:https://www.factomart.com/th/rotary-encoder/](https://www.factomart.com/th/rotary-encoder/)
- [6]อินทนะเนียร์น้อย. 2560. สูตรคำนวณแรงบิดมอเตอร์. [Online].Available :  
<http://thaiengineering.com/board/index.php?topic=2312.0>
- [7]ChaiyapornSilawatchananai. 2556. การควบคุมมอเตอร์ไฟฟ้ากระแสตรง. [Online].Available  
[: http://aimagin.com/blog/motor/?lang=th](http://aimagin.com/blog/motor/?lang=th)
- [8]Andras Tantos. 2011. H-bridge motor driver. [Online].Available :  
<http://www.modularcircuits.com/blog/articles/h-bridge-secrets/h-bridges-the-basics/>
- [9]PlayStation®Move. 2553. Kinect xbox 360. [Online].Available :  
<http://asia.playstation.com/move/th/>
- [10]Stephanie Crawford. 2554. How Microsoft Kinect Works. [Online].Available :  
<http://electronics.howstuffworks.com/microsoftkinect2.htm>
- [11]Doxygen. 2561. Image Thresholding. [Online].Available :  
[https://docs.opencv.org/3.4/d7/d4d/tutorial\\_py\\_thresholding.html](https://docs.opencv.org/3.4/d7/d4d/tutorial_py_thresholding.html)

ภาคผนวก

ภาคผนวก ก

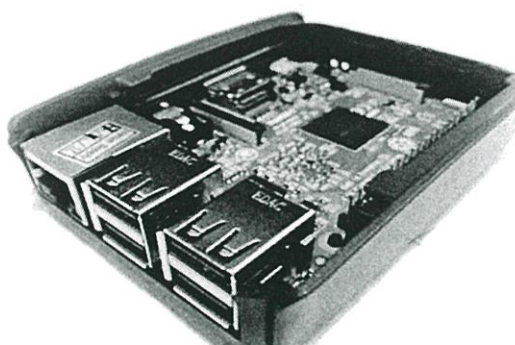
รายละเอียดของอุปกรณ์อิเล็กทรอนิกส์

## Raspberry Pi 3 Model B

Raspberry Pi เป็นเครื่องคอมพิวเตอร์ขนาดเล็กที่มีราคาถูกสามารถต่อเข้ากับจอคอมพิวเตอร์หรือจอทีวีที่รองรับ HDMI และสามารถต่อเมาส์และคีย์บอร์ดผ่านพอร์ต USB การเชื่อมต่ออินเทอร์เน็ตของ Raspberry Pi สามารถเชื่อมต่อด้วย Wi-fi และ Bluetooth ซึ่งเป็นการเชื่อมต่อแบบไร้สายง่ายต่อการนำไปใช้งานกับเซนเซอร์ต่างๆทำให้ Raspberry Pi มีการนำไปประยุกต์ใช้งานที่หลากหลาย แต่ในงานของเรานั้นจะนำรหัสเบอร์รีพายมาใช้ในการเขียนโปรแกรมควบคุมการเคลื่อนที่ของหุ่นยนต์และการประมวลผลภาพโดยใช้ภาษา Python ในการเขียนโปรแกรม

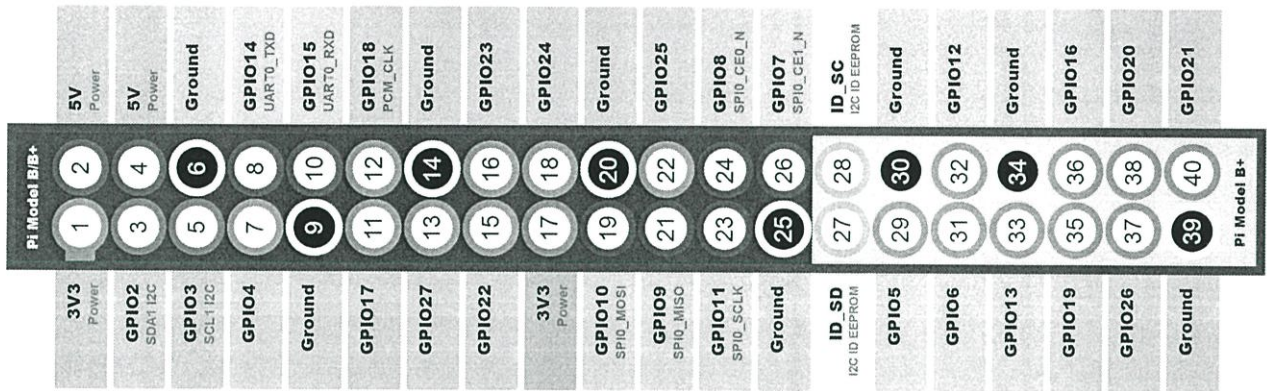
คุณสมบัติของ Raspberry Pi Model B

- CPU: Quad-core 1.2 GHz ARM Cortex-A53 แบบ 64 bits
- GPU: Broadcom VideoCore IV @ 400 MHz
- Memory ขนาด 1 GB (LPDDR2-900 SDRAM)
- หน่วยความจุแบบ MicroSD
- 4 USB ports
- 1 Ethernet port
- 802.11n Wireless LAN
- Bluetooth 4.0
- รองรับ HDMI/Composite ผ่านทาง RCA Jack
- GPIO 40 pins



รูปที่ ก.1 Raspberry Pi Model B

General Purpose Input/Output(GPIO) เป็นพอร์ตอินพุทเอาต์พุท อเนกประสงค์ เราสามารถรับและส่งข้อมูลต่างๆได้ผ่านทางพอร์ตนี้โดยในแต่ละพอร์ตของ GPIO มีการใช้งานที่จำเพาะ โดยในแต่ละรุ่นของ Raspberry Pi จะมีจำนวน ขาของ GPIO ที่ต่างกัน แต่โดยพื้นฐานจะมีขาที่เหมือนกันอยู่ คือ Power+, GND, I2C, UART, SPI,และ GPIO และในรุ่นของ model B นั้นมีทั้งสิ้น 40 GPIO



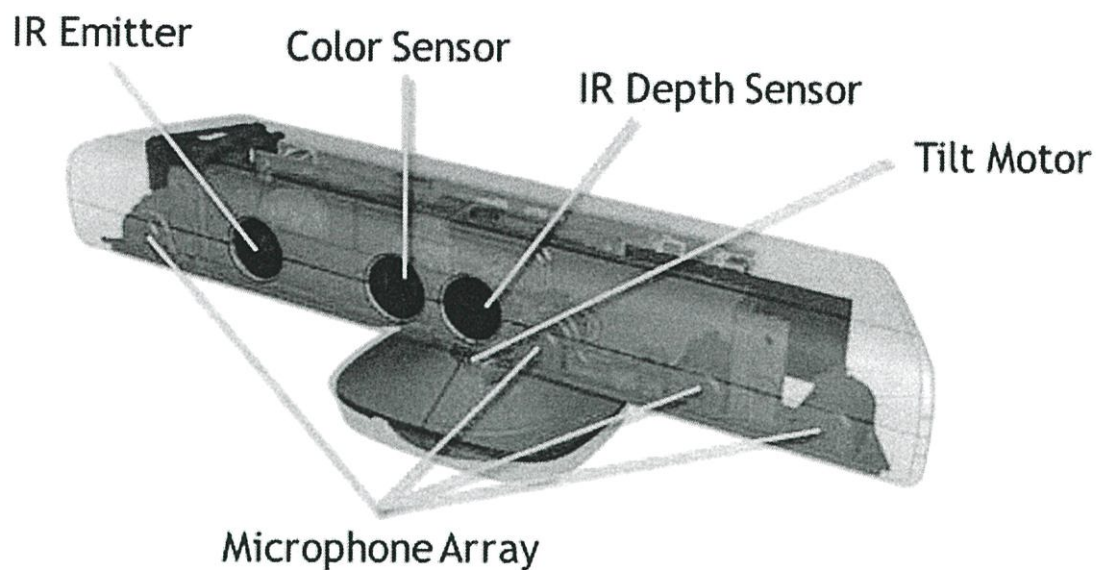
รูปที่ ก.2 GPIOที่มา raspberrypi-spy.co.uk

Pin No. 5V คือ หมายเลข 2 และ 4

Pin No. 3.3V คือ หมายเลข 1 และ 1

Pin No. GND คือ หมายเลข 6, 9, 14, 20, 25, 30, 34, และ 39

## Kinect Xbox 360



รูปที่ ก.3 ส่วนประกอบของ Kinect xbox 360

## PRODUCT SPECIFICATION

Property	PrimeSensor Spec
Field of View (Horizontal, Vertical, Diagonal)	58° H, 40° V, 70° D
Depth image size	VGA (640x480)
Spatial x/y resolution (@2m distance from sensor)	3mm
Depth z resolution (@2m distance from sensor)	1cm
Maximal image throughput (frame rate)	60fps
Average image latency in full VGA resolution	40msec
Operation range	0.8m - 3.5m

Property	PrimeSensor Spec
Color image size	UXGA (1600x1200)
Audio: built-in microphones	2 mics
Audio: digital inputs	4 inputs
Data interface	USB 2.0
Power supply	USB 2.0
Power consumption	2.25W
Dimensions (Width x Height x Depth)	14cm x 3.5cm x 5cm
Operation environment (every lighting condition)	indoor
Operating temperature	0°C - 40°C

รูปที่ ก.4 คุณสมบัติของ Kinect xbox 360

ภาคผนวก ข

โปรแกรมที่ใช้ในการควบคุมมอเตอร์

```
import RPi.GPIO as gpio

import pygamepio.setmode(gpio.BOARD)

pinR1 = 7

pinR2 = 11

pinL1 = 13

pinL2 = 15

gpio.setup(pinL1, gpio.OUT)

gpio.setup(pinL2, gpio.OUT)

gpio.setup(pinR1, gpio.OUT)

gpio.setup(pinR2, gpio.OUT)

pL1 = gpio.PWM(pinL1, 100)

pL2 = gpio.PWM(pinL2, 100)

pR1 = gpio.PWM(pinR1, 100)

pR2 = gpio.PWM(pinR2, 100)

pL1.start(0)

pL2.start(0)

pR1.start(0)

pR2.start(0)

pygame.init()

done = False #Loop until the user clicks the close button. clock = pygame.time.Clock()
# Used to manage how fast the screen updates pygame.joystick.init() # Initialize the
joysticks
```

```
# ----- Main Program Loop ----- while done==False: # EVENT PROCESSING STEP
for event in pygame.event.get(): # User did something if event.type == pygame.QUIT: #
If user clicked close done=True # Flag that we are done so we exit this loop joystick =
pygame.joystick.Joystick(0) joystick.init()

LR = joystick.get_axis(0)

FB = - joystick.get_axis(1)

gear = -0.5 * joystick.get_axis(2) +0.5

if LR > 0:

    ratioL = 1

    ratioR = -2*LR+1

else:

    ratioL = 2*LR+1

    ratioR = 1

    pwrL = FB * ratioL *100 * gear

    pwrR = FB * ratioR *100 * gear

if pwrL> 0:

pwmL1 = pwrL

pwmL2 = 0

else:

    pwmL1 = 0

    pwmL2 = -pwrL

if pwrR> 0:
```

```
pwmR1 = pwrR
```

```
pwmR2 = 0
```

```
else:
```

```
    pwmR1 = 0
```

```
    pwmR2 = -pwrR
```

```
pL1.ChangeDutyCycle(pwmL1)
```

```
pL2.ChangeDutyCycle(pwmL2)
```

```
pR1.ChangeDutyCycle(pwmR1)
```

```
pR2.ChangeDutyCycle(pwmR2)
```

```
## print ("LR: {:.2f} FB: {:.2f} ratio: {:.2f}, {:.2f} pwr: {:.1f}, {:.1f} pwmL: {:.1f}, {:.1f}
```

```
pwmR: {:.1f},
```

```
{:.1f}".format(LR,FB,ratioL,ratioR,pwrL,pwrR,pwmL1,pwmL2,pwmR1,pwmR2))
```

```
clock.tick(20) # Limit to 20 frames per second
```

```
pygame.quit ()
```

## ภาคผนวก ค

โปรแกรมที่ใช้ในการการวิเคราะห์และระบุสิ่งแปลกปลอม

```

# -*- coding: utf-8 -*-

from freenect import*
from numpy import*
from cv2 import*
from math import*
from time import*

import sys

#Funcion de Adquisicion RGB kinect
def frame_RGB():
array,_ = sync_get_video()
    array = cvtColor(array,COLOR_RGB2BGR)
    return array

def frame_depth():
array,_ = sync_get_depth()
    #array = array.astype(uint8)
    return array

def findxyzm(d,xpx,ypx):
    Y=(1)*d
    X=(2*(xpx-(639/2))*tan(radians(57/2))*Y/640)
    Z=(2*(479-ypx-(479/2))*tan(radians(43/2))*Y/480)+0.35
    return X,Y,Z

def detred(img):
    lb1=array([0,50,80])
    ub1=array([10,255,255])
    lb2=array([170,50,80])

```

```

ub2=array([180,255,255])
kop=ones((10,10))
kcl=ones((5,5))
imgHSV=cvtColor(img,COLOR_BGR2HSV)
mask1=inRange(imgHSV,lb1,ub1)
mask2=inRange(imgHSV,lb2,ub2)
mask=mask1+mask2
mop=morphologyEx(mask,MORPH_OPEN,kop)
mcl=morphologyEx(mop,MORPH_CLOSE,kcl)
#mop=morphologyEx(mcl,MORPH_OPEN,kop)
maskfinal=mcl
_,conts,h=findContours(maskfinal.copy(),RETR_EXTERNAL,CHAIN_APPROX_NONE)
#drawContours(img,conts,-1,(0,255,0),3)
for i in range(len(conts)):
sumx=0
sumy=0
x,y,w,h=boundingRect(conts[i])
    rectangle(img,(x,y),(x+w,y+h),(255,0,0),3)
sumx=sumx+(x+(w/2))
sumy=sumy+(y+(h/2))
    #print(sumx,sumy)
    #print(depth[sumy,sumx])
    #dep = ((4-0.8)/2048)*(depth[sumx,sumy]+1)+0.8
    dep = 1/(depth[sumy,sumx]*(-0.0028642) + 3.15221)
    print(findxym(dep,sumx,sumy))
    #print(dep)
return img

while True:

```

```
frame = frame_RGB() #leo frame
depth = frame_depth() #leoprofundidad depth
#depth = resize(depth,(0,0),fx=0.5, fy=0.5)
showdepth = depth.astype('uint8')
showdepth = cvtColor(showdepth, COLOR_GRAY2BGR)
imshow("mask",detred(frame))

    key = waitKey(1)
    if key == ord("q"):
        break

sys.exit()
destroyAllWindows()
VideoCapture.release()
```

ภาคผนวก ง  
ข้อมูลจากการทดสอบ

ตารางที่ 1 การทดสอบการลื่นไถลของล้อและการรับน้ำหนักในพื้นที่คอนกรีตเรียบขัดเงา

Load (kg)	Repeat	Rev at 32		Time (s)	Velocity (m/s)	Slip (m)	Tเฉลี่ย	Vเฉลี่ย m/s	Slip เฉลี่ย m
		Actual	Theory						
0	1	19.11	20.44	13.77	1.39	1.33	14.11	1.34	1.56
	2	18.55	20.44	14.26	1.30	1.89			
	3	18.98	20.44	14.30	1.33	1.46			
10	1	18.78	20.44	15.39	1.22	1.66	15.07	1.28	1.15
	2	19.58	20.44	14.74	1.33	0.86			
	3	19.50	20.44	15.09	1.29	0.94			
20	1	19.42	20.44	16.07	1.21	1.02	16.14	1.21	0.95
	2	19.64	20.44	16.21	1.21	0.80			
	3	19.42	20.44	16.15	1.20	1.02			
30	1	19.73	20.44	16.98	1.16	0.71	17.00	1.16	0.81
	2	19.50	20.44	17.24	1.13	0.94			
	3	19.66	20.44	16.77	1.17	0.78			
40	1	19.97	20.44	17.98	1.11	0.47	18.42	1.08	0.59
	2	19.80	20.44	18.84	1.05	0.64			
	3	19.77	20.44	18.43	1.07	0.67			
50	1	19.97	20.44	21.64	0.92	0.47	22.14	0.91	0.39
	2	19.99	20.44	22.88	0.87	0.45			
	3	20.18	20.44	21.91	0.92	0.26			
60	1	20.32	20.44	25.11	0.81	0.12	25.84	0.78	0.18
	2	20.20	20.44	26.41	0.76	0.24			
	3	20.25	20.44	25.99	0.78	0.19			

ตารางที่ 2 การทดสอบการลื่นไถลของล้อและการรับน้ำหนักในพื้นที่คอนกรีต

Load (kg)	Repeat	Rev at 32		Time (s)	Velocity (m/s)	Slip (m)	Tเฉลี่ย	Vเฉลี่ย m/s	Slipเฉลี่ย (m)
		Actual	Theory						
0	1	19.23	20.44	15.19	1.27	1.21	14.68	1.31	1.17
	2	19.35	20.44	14.52	1.33	1.09			
	3	19.22	20.44	14.33	1.34	1.22			
10	1	19.66	20.44	16.17	1.22	0.78	15.87	1.24	0.82
	2	19.62	20.44	15.89	1.23	0.82			
	3	19.58	20.44	15.55	1.26	0.86			
20	1	20.13	20.44	16.88	1.19	0.31	17.38	1.14	0.70
	2	19.34	20.44	18.22	1.06	1.10			
	3	19.74	20.44	17.04	1.16	0.70			
30	1	19.82	20.44	19.04	1.04	0.62	18.87	1.05	0.57
	2	19.76	20.44	18.89	1.05	0.68			
	3	20.02	20.44	18.67	1.07	0.42			
40	1	20.19	20.44	22.45	0.90	0.25	22.70	0.88	0.39
	2	20.09	20.44	22.65	0.89	0.35			
	3	19.88	20.44	23.01	0.86	0.56			
50	1	20.21	20.44	26.45	0.76	0.23	26.58	0.76	0.24
	2	20.05	20.44	26.76	0.75	0.39			
	3	20.33	20.44	26.54	0.77	0.11			
60	1	20.29	20.44	32.76	0.62	0.15	34.45	0.59	0.11
	2	20.34	20.44	34.94	0.58	0.10			
	3	20.35	20.44	35.65	0.57	0.09			

ตารางที่ 3 ผลการทดลองการวิเคราะห์และระบุตำแหน่งเป้าหมาย

	Actual (m)			Detect (m)			Error (m)		
	X	Y	Z	X	Y	Z	x	y	Z
1	0.00	2.00	0.00	0.02	2.04	0.05	0.02	0.04	0.05
2	0.00	2.00	1.00	0.01	2.06	0.95	0.01	0.06	-0.05
3	0.00	2.00	2.00	-	-	-	-	-	-
4	0.00	4.00	0.00	0.07	3.94	0.01	0.07	-0.06	0.01
5	0.00	4.00	1.00	0.08	4.12	1.11	0.08	0.12	0.11
6	0.00	4.00	2.00	0.04	4.22	2.23	0.04	0.22	0.23
7	0.00	6.00	0.00	0.05	6.23	0.12	0.05	0.23	0.12
8	0.00	6.00	1.00	0.09	6.37	1.37	0.09	0.37	0.37
9	0.00	6.00	2.00	-	-	-	-	-	-
10	1.50	2.00	0.00	1.56	2.06	0.08	0.06	0.06	0.08
11	1.50	2.00	1.00	1.62	2.13	1.19	0.12	0.13	0.19
12	1.50	2.00	2.00	-	-	-	-	-	-
13	1.50	4.00	0.00	1.68	3.94	0.09	0.18	-0.06	0.09
14	1.50	4.00	1.00	1.59	4.05	1.27	0.09	0.05	0.27
15	1.50	4.00	2.00	1.62	4.18	2.37	0.12	0.18	0.37
16	1.50	6.00	0.00	1.78	6.33	0.17	0.28	0.33	0.17
17	1.50	6.00	1.00	1.69	6.43	1.53	0.19	0.43	0.53
18	1.50	6.00	2.00	-	-	-	-	-	-