



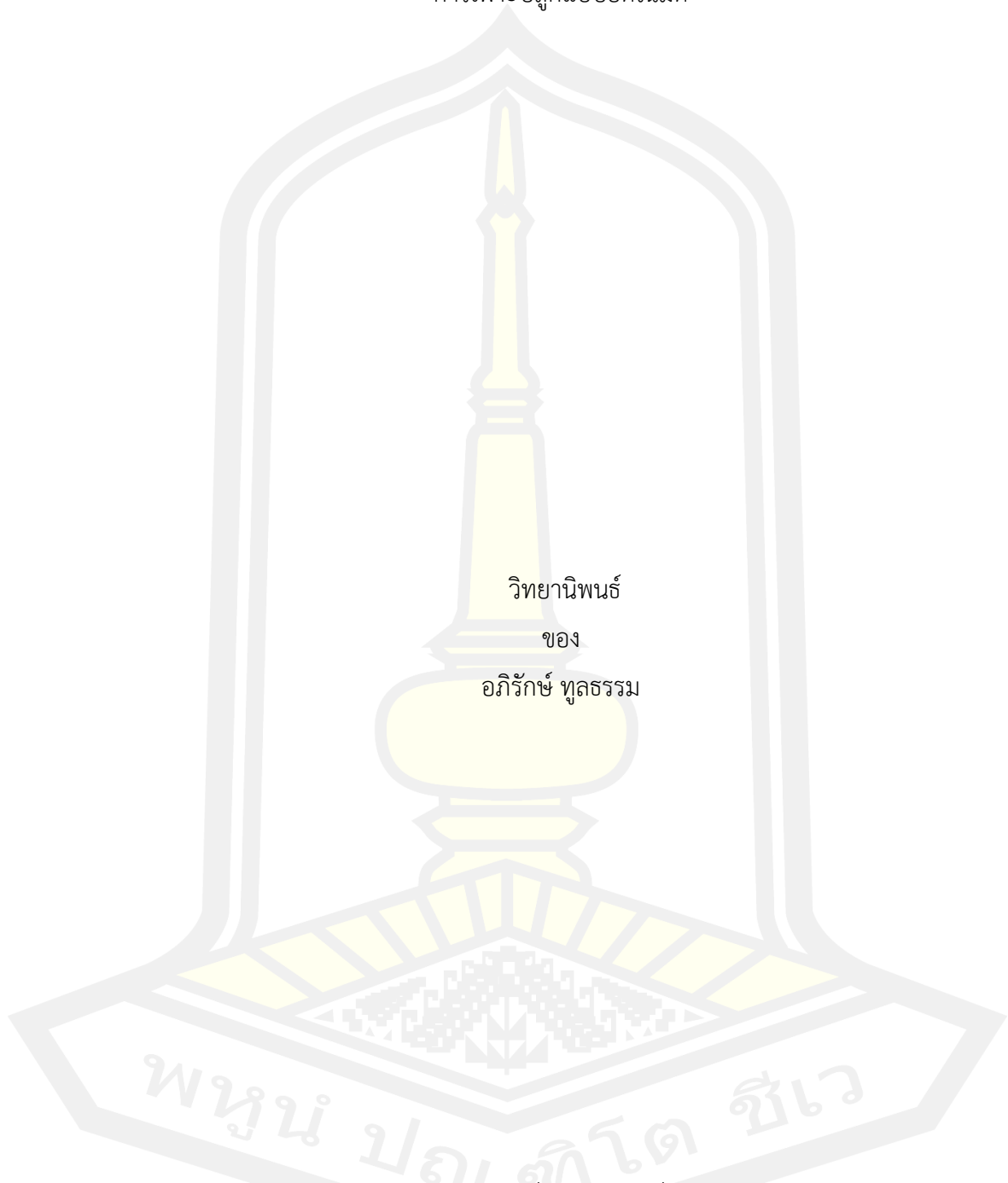
อินเทอร์เน็ตของสรรพสิ่งสำหรับหุ่นยนต์ปัญญาประดิษฐ์เพื่อสนับสนุน  
การเพาะปลูกแบบอัตโนมัติ

วิทยานิพนธ์  
ของ  
อภิรักษ์ ทูลธรรม

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
ธันวาคม 2565

ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม

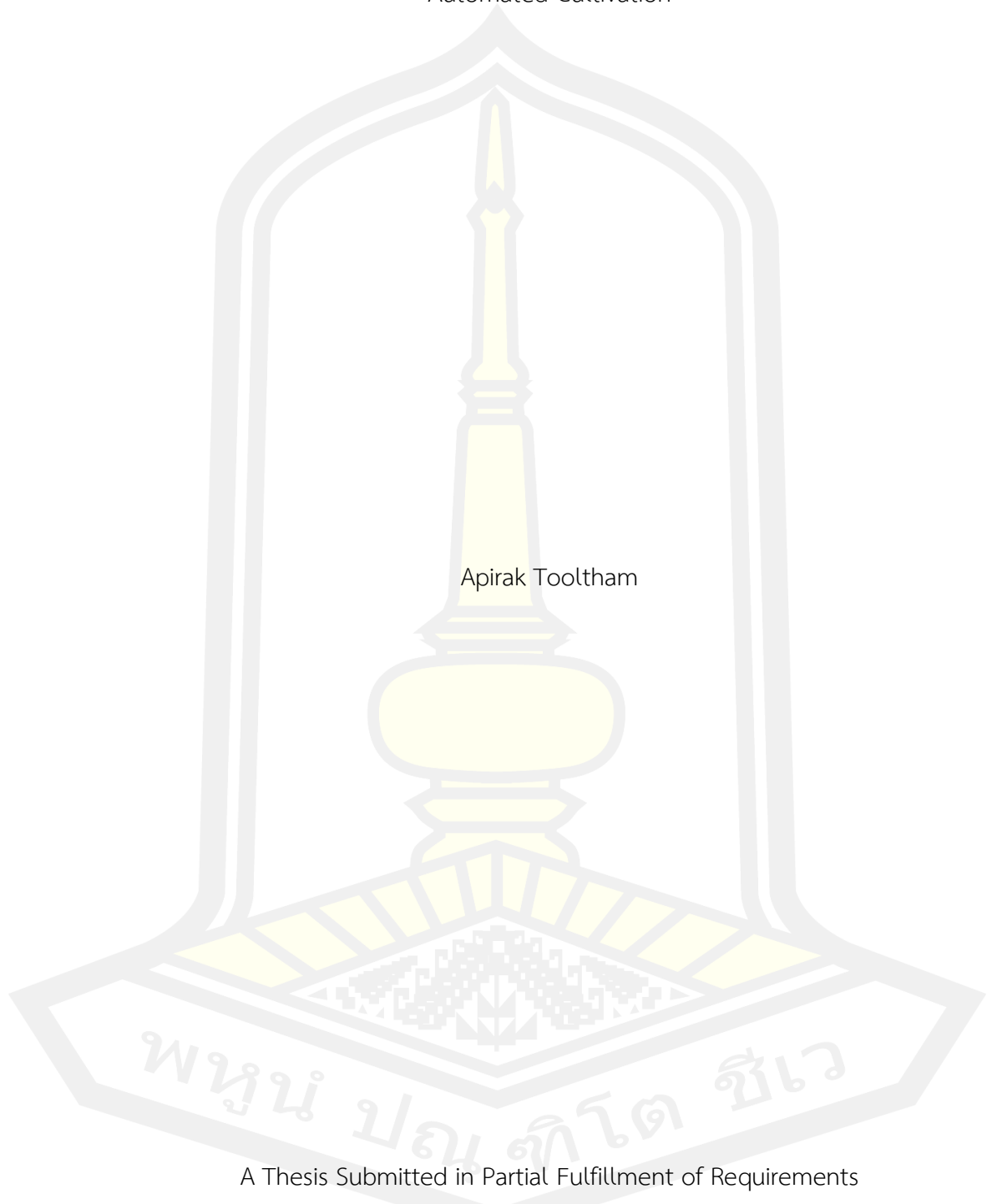
อินเทอร์เน็ตของสรรพสิ่งสำหรับหุ่นยนต์ปัญญาประดิษฐ์เพื่อสนับสนุน  
การเพาะปลูกแบบอัตโนมัติ



เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์  
ธันวาคม 2565

ลิขสิทธิ์เป็นของมหาวิทยาลัยมหาสารคาม

The Internet of Things for Artificial Intelligence Robots to Support  
Automated Cultivation



Apirak Tooltham

A Thesis Submitted in Partial Fulfillment of Requirements  
for Doctor of Philosophy (Computer Science)

December 2022

Copyright of Mahasarakham University



คณะกรรมการสอบวิทยานิพนธ์ ได้พิจารณาวิทยานิพนธ์ของนายอภิรักษ์ ทูลธรรม แล้ว  
เห็นสมควรรับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาปรัชญาดุษฎีบัณฑิต สาขาวิชา  
วิทยาการคอมพิวเตอร์ ของมหาวิทยาลัยมหาสารคาม

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ

(รศ. ดร. ธัชพงศ์ กัตัญญกุล )

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(ผศ. ดร. สุชาติ คุ่มมะณี )

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

(ผศ. ดร. ฉัตรเกล้า เจริญผล )

..... กรรมการ

(รศ. ดร. จันทิมา พลพินิจ )

..... กรรมการ

(ผศ. ดร. มนัสวี แก่นอำพรพันธ์ )

..... กรรมการ

(ผศ. ดร. พัฒนพงษ์ ชมภูวิเศษ )

มหาวิทยาลัยขอนแก่นให้รับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร  
ปริญญา ปรัชญาดุษฎีบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ของมหาวิทยาลัยมหาสารคาม

.....  
(รศ. ดร. จันทิมา พลพินิจ )

คณบดีคณะวิทยาการสารสนเทศ

.....  
(รศ. ดร. กริสน์ ชัยมูล )

คณบดีบัณฑิตวิทยาลัย

<b>ชื่อเรื่อง</b>	อินเทอร์เน็ตของสรรพสิ่งสำหรับหุ่นยนต์ปัญญาประดิษฐ์เพื่อสนับสนุน การเพาะปลูกแบบอัตโนมัติ		
<b>ผู้วิจัย</b>	อภิรักษ์ ทูลธรรม		
<b>อาจารย์ที่ปรึกษา</b>	ผู้ช่วยศาสตราจารย์ ดร. สุชาติ คุ้มมะณี ผู้ช่วยศาสตราจารย์ ดร. ฉัตรเกล้า เจริญผล		
<b>ปริญญา</b>	ปรัชญาดุษฎีบัณฑิต	<b>สาขาวิชา</b>	วิทยาการคอมพิวเตอร์
<b>มหาวิทยาลัย</b>	มหาวิทยาลัยมหาสารคาม	<b>ปีที่พิมพ์</b>	2565

### บทคัดย่อ

ปัจจุบันภาคเกษตรกรรมของโลกเริ่มเข้าสู่ยุคของการใช้หุ่นยนต์แล้ว หลังจากแรงงานในภาคเกษตรกรรมมีแนวโน้มสูงที่จะขาดแคลนในอนาคต ซึ่งเหตุผลมาจากคนวัยทำงานไม่เลือกประกอบอาชีพด้านการเกษตร เพราะเป็นงานหนัก ซ้ำซาก และเสี่ยงการสัมผัสสารเคมี รวมถึงเกษตรกรในปัจจุบันก็เริ่มย่างเข้าสู่วัยผู้สูงอายุ ในขณะที่สินค้าเกษตรเป็นสินค้าที่ตลาดโลกมีความต้องการอย่างต่อเนื่อง และมีการบริโภคเพิ่มขึ้นทุกขณะ ซึ่งงานวิจัยก่อนหน้านี้ได้ทำการศึกษาเพื่อประเมินและแก้ไขปัญหามาแล้ว แต่พบว่าการแก้ไขดังกล่าวเป็นการพัฒนาต้นแบบที่ทำงานเฉพาะด้าน รวมถึงยังมีปัจจัยที่ต้องพัฒนาเพิ่มเติมต่อ เช่น การทำงานแบบอัตโนมัติ ความยืดหยุ่นในการทำงาน การเพิ่ม-ลดพื้นที่การทำงานและต้นทุนของการพัฒนา ในวิทยานิพนธ์นี้ได้ทำการพัฒนาหุ่นยนต์เพาะปลูกพืชแบบอัตโนมัติ ซึ่งประยุกต์การทำงานของหุ่นยนต์เข้ากับอินเทอร์เน็ตของสรรพสิ่ง การประมวลผลภาพ และปัญญาประดิษฐ์ เพื่อหยุดเมล็ดตกลงบนกระถางในแปลงปลูก ซึ่งผลของการพัฒนาดังกล่าวแสดงให้เห็นว่า หุ่นยนต์ที่ถูกพัฒนาขึ้นมีความเร็วในการเคลื่อนที่ 360 เมตรต่อชั่วโมง และมีอัตราการหยุดเมล็ด 13.37 วินาทีต่อกระถาง หรือ 257 กระถางต่อชั่วโมง ซึ่งในแต่ละรอบมีระยะเวลาในการทำภารกิจเฉลี่ย 12.53 นาที โดยระยะเวลาในการทำงานต่อการชาร์จหนึ่งครั้งประมาณ 2 ชั่วโมง ซึ่งหุ่นยนต์จะสามารถหยุดเมล็ดได้ 486 กระถาง ที่ความแม่นยำในการหยุดเมล็ดเท่ากับ 0.72 นอกจากนี้หุ่นยนต์ต้นแบบยังมีความยืดหยุ่นและปรับขนาดพื้นที่การทำงานได้โดยที่มีขนาดเล็ก น้ำหนักเบาเพียง 20 กิโลกรัม และมีต้นทุนในการพัฒนาที่ต่ำ

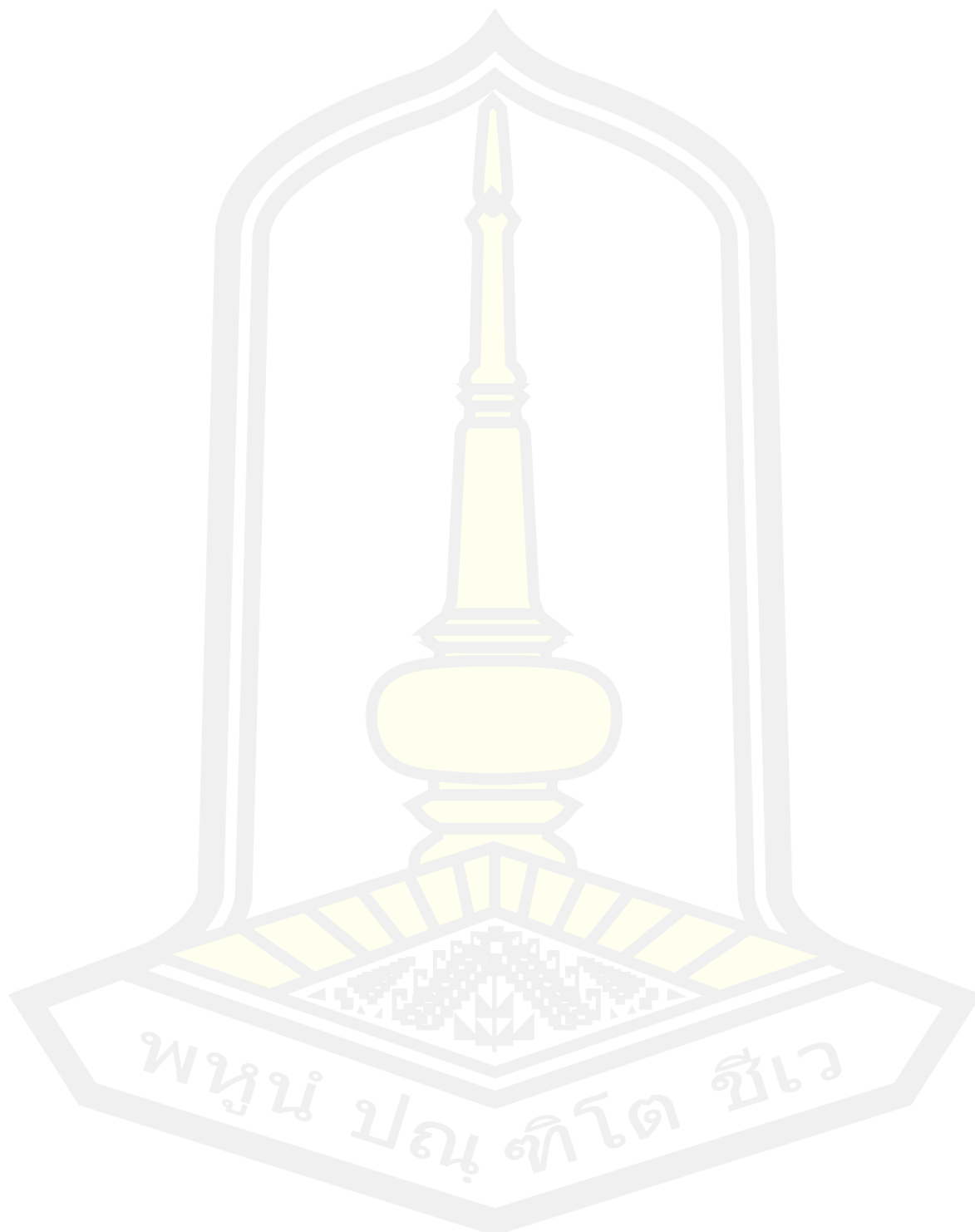
คำสำคัญ : หุ่นยนต์, การเพาะปลูกแบบอัตโนมัติ, อินเทอร์เน็ตของสรรพสิ่ง, การประมวลผลภาพ

<b>TITLE</b>	The Internet of Things for Artificial Intelligence Robots to Support Automated Cultivation		
<b>AUTHOR</b>	Apirak Tooltham		
<b>ADVISORS</b>	Assistant Professor Suchart Khummanee , Ph.D. Assistant Professor Chatklaw Jareanpon , Ph.D.		
<b>DEGREE</b>	Doctor of Philosophy	<b>MAJOR</b>	Computer Science
<b>UNIVERSITY</b>	Maharakham University	<b>YEAR</b>	2022

### ABSTRACT

The world's agricultural sector has entered the era of robotics, after laboring in this sector has a high probability of shortages. This is because working people do not choose to pursue a career in agriculture. Due because it is hard work, repetitive patterns, risks of exposure to pesticides and herbicides, and laborers enter the elderly age. While agricultural products are still in demand, the quantity of consumption is increasing every moment. Several previous studies have been done to solve the problems. However, those solutions were to develop prototypes working on specific tasks. Some factors need to be further developed and modified, such as automation, the flexibility of work, increasing and reducing working space, and the cost of development. In this thesis, an automated cultivation robot was developed by integrating robotics with the Internet of Things, Image Processing, and Artificial Intelligence to sow the seeds on the pots. The results showed that the velocity of the prototype was 360 meters per hour. The sowing speed was about 13.37 seconds per pot or 257 pots per hour. The seeding task for each round takes an average of 12.53 minutes. The operating time of the robot per charge is approximately 2 hours (the robot can seed 486 pots on a single charge). The seeding accuracy is 0.72. Moreover, the workspace can be flexible and scaled up. The robot is small, lightweight (only 20 kg), and has low development costs.

Keyword : Robotics, Automated Cultivation, Internet of Things, Image Processing



## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยความช่วยเหลือจาก ผู้ช่วยศาสตราจารย์ ดร.สุชาติ คุ้มมะณี และผู้ช่วยศาสตราจารย์ ดร.ฉัตรเกล้า เจริญผล โดยมี รองศาสตราจารย์ ดร.รัชพงศ์ กัตัญญกุล ประธานกรรมการสอบ รองศาสตราจารย์ ดร.จันทิมา พลพิณิช ผู้ช่วยศาสตราจารย์ ดร.มนัสวี แก่นอำพรพันธ์ และผู้ช่วยศาสตราจารย์ ดร.พัฒน์พงษ์ ชมภูวิเศษ กรรมการสอบ ซึ่งเป็นผู้ให้คำชี้แนะเพื่อปรับปรุงข้อบกพร่องของวิทยานิพนธ์ฉบับนี้ ขอขอบคุณ อาจารย์มนตรี โนนพะยอม วิทยาลัยการอาชีพขอนแก่น สำหรับคำชี้แนะแนวทางในการทำวิจัย

ขอขอบคุณมหาวิทยาลัยนครพนมซึ่งเป็นผู้ให้ทุนการศึกษาต่อระดับปริญญาเอกในครั้งนี้ และวิทยาลัยเทคโนโลยีอุตสาหกรรมศรีสงคราม ที่ให้การสนับสนุนอุปกรณ์ในการทำวิจัยและอำนวยความสะดวกในเรื่องต่าง ๆ มาโดยตลอด

ขอบคุณบิดา มารดา ภรรยา และลูกทั้งสองที่ให้กำลังใจและเปิดโอกาสให้สามารถทำงานวิจัยได้อย่างเต็มที่โดยไม่ต้องกังวลกับสิ่งใด

โครงการวิจัยนี้ได้รับทุนอุดหนุนการทำกิจกรรมส่งเสริมและสนับสนุนการวิจัยและนวัตกรรมจากสำนักงานการวิจัยแห่งชาติ

อภิรักษ์ ทุลธรรม

พนุน ปณู จิตโต ชีเว

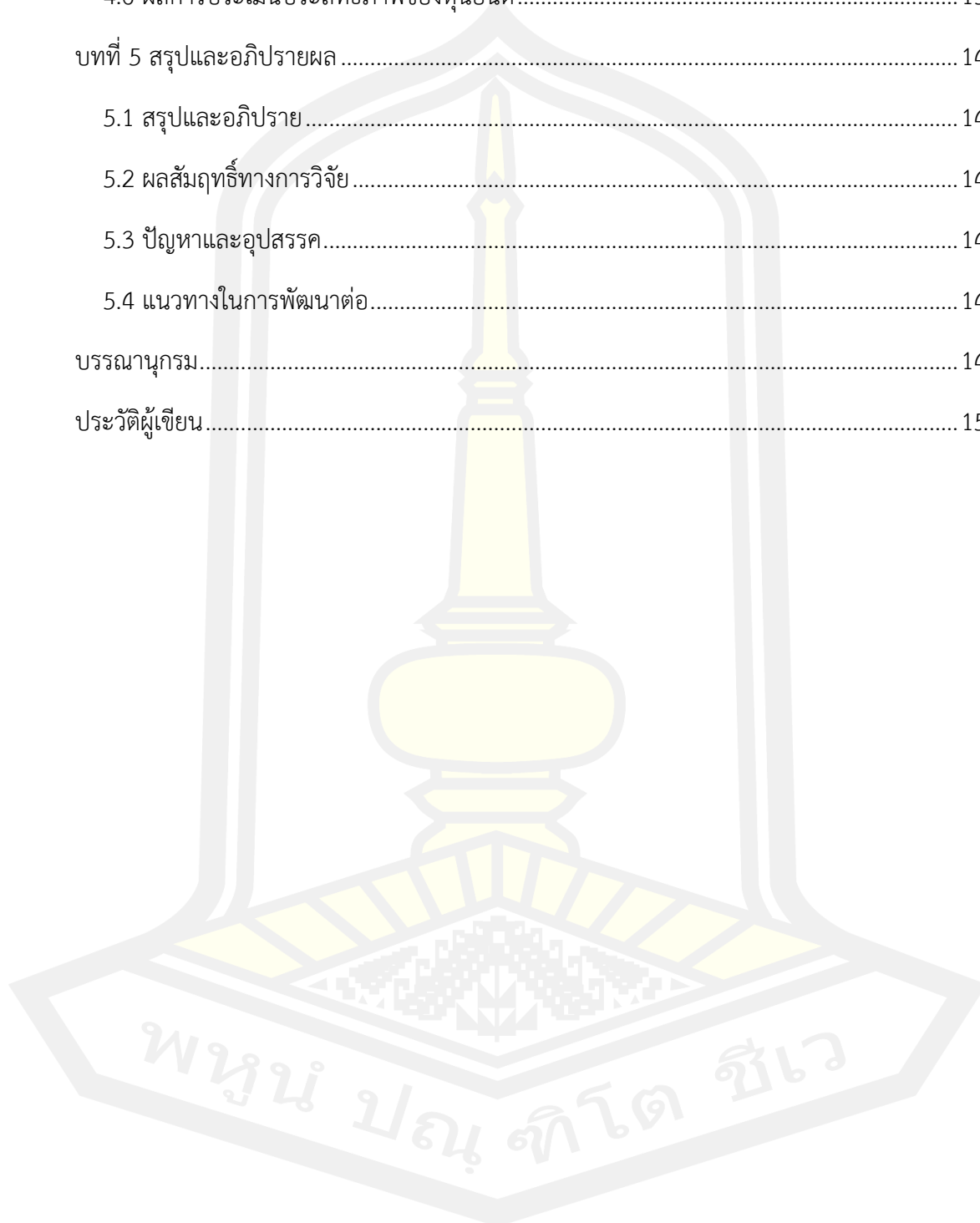


## สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ง
บทคัดย่อภาษาอังกฤษ.....	จ
กิตติกรรมประกาศ.....	ช
สารบัญ.....	ซ
สารบัญตาราง.....	ฅ
สารบัญภาพประกอบ.....	ฉ
บทที่ 1 บทนำ .....	1
1.1 หลักการและเหตุผล .....	1
1.2 วัตถุประสงค์ของการวิจัย .....	2
1.3 ความสำคัญของการวิจัย.....	3
1.4 ขอบเขตของการวิจัย.....	3
1.5 นิยามศัพท์เฉพาะ .....	4
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง .....	6
2.1 หุ่นยนต์.....	6
2.2 หุ่นยนต์แบบอนุกรม.....	9
2.3 โครงสร้างของหุ่นยนต์.....	11
2.4 จลนศาสตร์ของหุ่นยนต์ .....	15
2.5 ระบบการมองเห็นของหุ่นยนต์.....	17
2.6 แนวทางการควบคุมหุ่นยนต์โดยภาพ .....	18
2.7 การประยุกต์ใช้หุ่นยนต์ในภาคการเกษตร.....	21
2.8 การประมวลผลภาพ.....	27

2.9 ภาพดิจิทัล .....	28
2.10 การหาขอบภาพ.....	31
2.11 การหาขอบภาพโดยวิธีโซเบล.....	33
2.12 การหาขอบภาพโดยวิธีแคนนี่.....	35
2.13 Hough Transform.....	37
2.14 ระบบสมองกลฝังตัว.....	40
2.15 เซ็นเซอร์.....	42
2.16 อินเทอร์เน็ตของสรรพสิ่ง .....	44
2.17 Message Queuing Telemetry Transport.....	46
2.18 การประเมินประสิทธิภาพ .....	48
2.19 งานวิจัยที่เกี่ยวข้อง .....	52
บทที่ 3 วิธีดำเนินการวิจัย.....	71
3.1 การออกแบบและพัฒนาหุ่นยนต์ .....	72
3.2 การออกแบบซอฟต์แวร์ควบคุมการทำงานของหุ่นยนต์.....	87
3.3 การออกแบบและพัฒนาระบบอินเทอร์เน็ตของสรรพสิ่ง .....	91
3.4 สภาพแวดล้อมที่ใช้ในการทดสอบ .....	96
3.5 การประเมินประสิทธิภาพ .....	97
3.6 เกณฑ์ในการประเมิน .....	98
3.7 เครื่องมือที่ใช้ในการวิจัย.....	99
บทที่ 4 ผลการวิจัย.....	100
4.1 ผลการออกแบบและพัฒนาระบบการเคลื่อนที่.....	100
4.2 ผลการออกแบบและพัฒนาระบบการรับรู้.....	107
4.3 ผลการออกแบบและพัฒนาระบบการกระทำ .....	114
4.4 ผลการออกแบบและพัฒนาระบบอินเทอร์เน็ตของสรรพสิ่ง.....	121

4.5 ผลการออกแบบและพัฒนาซอฟต์แวร์ควบคุมการทำงานของหุ่นยนต์ .....	128
4.6 ผลการประเมินประสิทธิภาพของหุ่นยนต์ .....	130
บทที่ 5 สรุปและอภิปรายผล .....	140
5.1 สรุปและอภิปราย .....	140
5.2 ผลสัมฤทธิ์ทางการวิจัย .....	143
5.3 ปัญหาและอุปสรรค .....	144
5.4 แนวทางในการพัฒนาต่อ .....	146
บรรณานุกรม .....	149
ประวัติผู้เขียน .....	157



## สารบัญตาราง

	หน้า
ตารางที่ 2.1 การเปรียบเทียบข้อดี-ข้อเสียของหุ่นยนต์ปลูกพืชในงานวิจัยที่ผ่านมา .....	68
ตารางที่ 3.1 รูปแบบที่ใช้ในการทดสอบการเคลื่อนของล้อ Mecanum Wheel.....	76
ตารางที่ 3.2 การกำหนดเงื่อนไขการทำงานของหุ่นยนต์ด้วย Barcode.....	81
ตารางที่ 3.3 เงื่อนไขในการประเมินการทำงานระบบอินเทอร์เน็ตของสรรพสิ่ง .....	93
ตารางที่ 3.4 รูปแบบในการควบคุมการทำงานของ Topic “status” .....	95
ตารางที่ 4.1 ผลการทดสอบการเคลื่อนที่ของหุ่นยนต์.....	103
ตารางที่ 4.2 ผลการทดสอบการเคลื่อนที่เชิงเส้นทั้ง 3 แกนของหุ่นยนต์.....	106
ตารางที่ 4.3 สรุปผลการทดสอบการเคลื่อนที่เชิงเส้นทั้ง 3 แกนของหุ่นยนต์ .....	106
ตารางที่ 4.4 ผลการทดสอบระบบการตรวจจับกระถางของหุ่นยนต์ .....	108
ตารางที่ 4.5 ผลทดสอบการทำงานของหุ่นยนต์ด้วยสัญลักษณ์ Barcode.....	112
ตารางที่ 4.6 ผลการทดสอบระบบควบคุมการหยุดเคลื่อนที่.....	119
ตารางที่ 4.7 ผลการทดสอบ Publish และ Subscribe.....	122
ตารางที่ 4.8 ผลการทดสอบการควบคุมหุ่นยนต์ผ่านระบบอินเทอร์เน็ตของสรรพสิ่ง .....	125
ตารางที่ 4.9 ตำแหน่งการหยุดเคลื่อนที่ในกระถางของหุ่นยนต์ .....	131
ตารางที่ 4.10 ระยะห่างของจุดหยุดเคลื่อนที่กับจุดศูนย์กลางของกระถาง .....	133
ตารางที่ 4.11 ผลการประเมินประสิทธิภาพของหุ่นยนต์.....	135
ตารางที่ 4.12 ผลการประเมินประสิทธิภาพเปรียบเทียบกับงานวิจัยที่ผ่านมา.....	138

## สารบัญภาพประกอบ

	หน้า
ภาพที่ 2.1 องค์ประกอบของหุ่นยนต์และความสัมพันธ์ระหว่างกัน.....	7
ภาพที่ 2.2 ส่วนประกอบทางกลของหุ่นยนต์.....	8
ภาพที่ 2.3 ลักษณะของ Joint และ Link เมื่อเปรียบเทียบกับแขนของมนุษย์.....	8
ภาพที่ 2.4 ลักษณะของข้อต่อที่ถูกใช้ในแขนหุ่นยนต์.....	9
ภาพที่ 2.5 แนวคิดของ Close and Open Kinematic Chain.....	10
ภาพที่ 2.6 แขนหุ่นยนต์แบบพับ (Articulated Arm).....	12
ภาพที่ 2.7 แขนหุ่นยนต์แบบพิกัตทรงกลม (Spherical Arm).....	12
ภาพที่ 2.8 แขนหุ่นยนต์แบบสการา (SCARA Arm).....	13
ภาพที่ 2.9 แขนหุ่นยนต์แบบพิกัตทรงกระบอก (Cylindrical Arm).....	14
ภาพที่ 2.10 แขนหุ่นยนต์แบบพิกัตคาร์ทีเซียน (Cartesian Arm).....	14
ภาพที่ 2.11 ปริภูมิการทำงานของแขนหุ่นยนต์ประเภทต่าง ๆ.....	15
ภาพที่ 2.12 จลนศาสตร์ข้างหน้าและจลนศาสตร์ผกผัน.....	16
ภาพที่ 2.13 รูปแบบการติดตั้งกล้อง.....	19
ภาพที่ 2.14 แนวทางการควบคุมโดยภาพ.....	20
ภาพที่ 2.15 หุ่นยนต์ตรวจการณ์ภาคสนาม.....	22
ภาพที่ 2.16 หุ่นยนต์กำจัดแมลง.....	22
ภาพที่ 2.17 หุ่นยนต์เก็บเกี่ยวผลผลิต.....	23
ภาพที่ 2.18 หุ่นยนต์ฉีดพ่นสารเคมี.....	24
ภาพที่ 2.19 หุ่นยนต์ตัดแต่งกิ่ง.....	24
ภาพที่ 2.20 หุ่นยนต์รีดนมวัว.....	25
ภาพที่ 2.21 หุ่นยนต์ตรวจจำแนกลักษณะพืช.....	26

ภาพที่ 2.22	หุ่นยนต์จัดเรียงต้นตอพืชเพื่อทำการทาบกิ่ง.....	26
ภาพที่ 2.23	หุ่นยนต์เพาะปลูกพืชแบบหยอดเมล็ดพันธุ์ .....	27
ภาพที่ 2.24	จุดพิกัดบนภาพดิจิทัล.....	28
ภาพที่ 2.25	เมทริกซ์ของภาพดิจิทัล ขนาด $M \times N$ .....	29
ภาพที่ 2.26	ภาพไบนารี (Binary) .....	29
ภาพที่ 2.27	ภาพระดับเทา (Grayscale Image).....	30
ภาพที่ 2.28	ภาพสี (Color Image).....	30
ภาพที่ 2.29	ภาพแบบดัชนี (Indexed Image) .....	31
ภาพที่ 2.30	แสดงตำแหน่งของตัวแปรด้วยวิธี Sobel .....	33
ภาพที่ 2.31	การหาขอบภาพด้วยวิธี Sobel Edge Detection .....	34
ภาพที่ 2.32	การหาขอบภาพด้วยวิธี Canny Edge Detection .....	37
ภาพที่ 2.33	เส้นและค่าพารามิเตอร์ของเส้น .....	38
ภาพที่ 2.34	ตัวอย่างของ Hough Line Transform.....	39
ภาพที่ 2.35	ตัวอย่างของ Hough Circle Transform .....	40
ภาพที่ 2.36	อุปกรณ์ Open Source Hardware สำหรับใช้พัฒนาระบบสมองกลฝังตัว .....	42
ภาพที่ 2.37	ประเภทอุปกรณ์เซ็นเซอร์ตามคุณสมบัติในการตรวจวัด.....	44
ภาพที่ 2.38	แนวความคิดของเทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง .....	45
ภาพที่ 2.39	MQTT Publish / Subscribe Architecture .....	46
ภาพที่ 2.40	ขั้นตอนการ Subscribe ของ MQTT Client.....	47
ภาพที่ 2.41	ขั้นตอนการ Publish ของ MQTT Client.....	47
ภาพที่ 2.42	ขั้นตอนการ Publish ของ MQTT Broker .....	47
ภาพที่ 2.43	Elements of Mobile Robot Autonomy .....	48
ภาพที่ 2.44	เวลา (Time) และ ปฏิกริยา (Reactivity) ของหุ่นยนต์ .....	50
ภาพที่ 2.45	วิธีการพิจารณาประสิทธิภาพของหุ่นยนต์ที่คาดหวัง .....	50

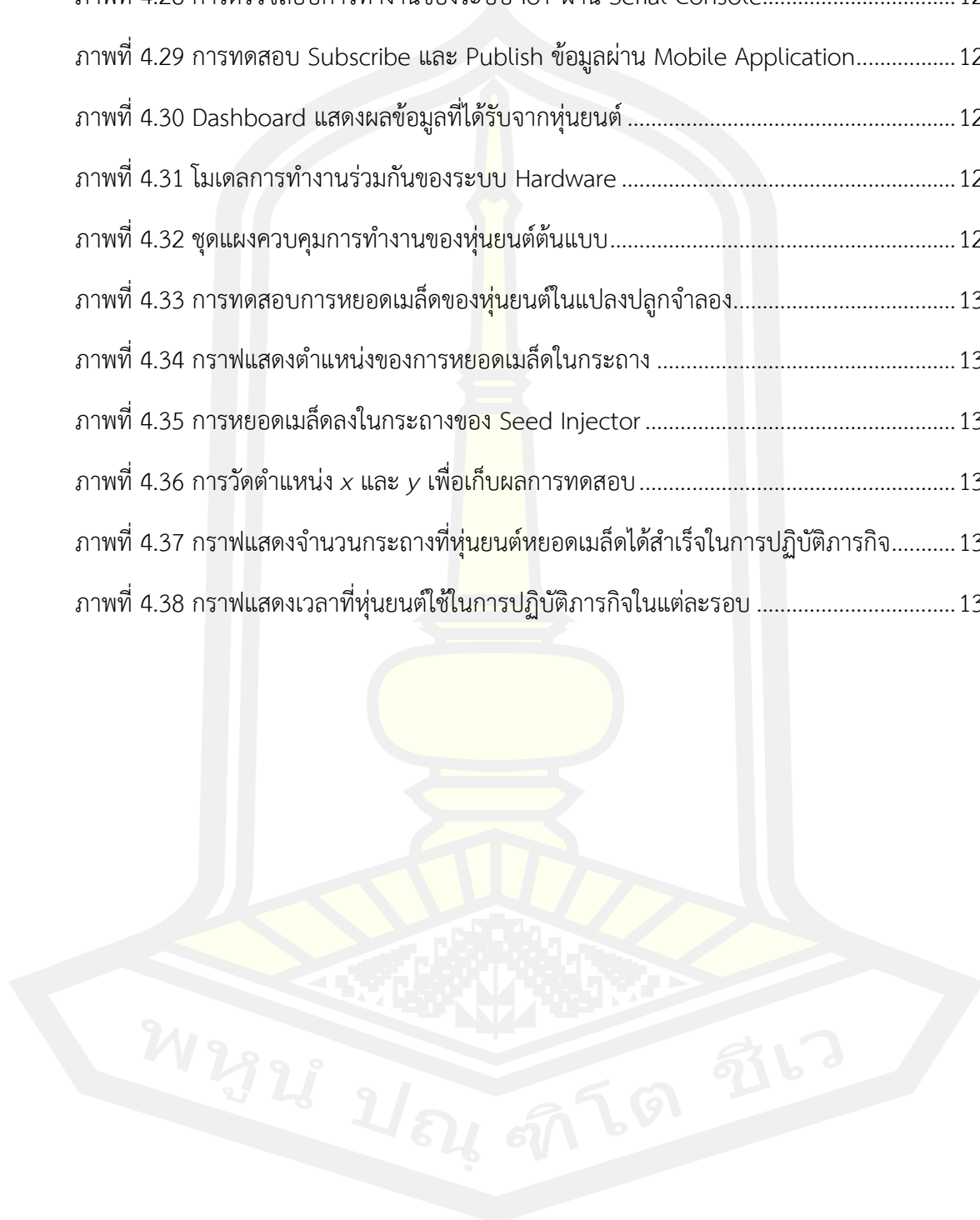
ภาพที่ 2.46 แนวความคิดเกี่ยวกับการเกษตรแบบดิจิทัลโดยเน้นบทบาทของหุ่นยนต์การเกษตร.....	57
ภาพที่ 2.47 ต้นแบบการพัฒนาแพลตฟอร์มหุ่นยนต์เคลื่อนที่แบบ 4WD.....	59
ภาพที่ 2.48 จลนศาสตร์ของกลไกการเคลื่อนที่ของหุ่นยนต์ 4WD.....	59
ภาพที่ 2.49 หุ่นยนต์ต้นแบบ CeRSAA กำลังทำงานภายในโรงเรือน .....	61
ภาพที่ 2.50 เลย์เอาต์และปฏิภูมิการทำงานของหุ่นยนต์.....	62
ภาพที่ 2.51 หุ่นยนต์ RIRRIG ของ Universidade Federal do Ceará.....	63
ภาพที่ 2.52 ต้นแบบของ FarmBot .....	64
ภาพที่ 2.53 ภาพรวมของ FarmBot Software ในระดับ High Level.....	65
ภาพที่ 2.54 การติดตั้งระบบของ SAASbot .....	67
ภาพที่ 2.55 สถาปัตยกรรมระบบโดยรวมของ SAASbot .....	67
ภาพที่ 3.1 สถาปัตยกรรมระบบโดยรวมของหุ่นยนต์ปลูกพืชแบบอัตโนมัติ.....	71
ภาพที่ 3.2 ภาพรวมการดำเนินการวิจัย.....	72
ภาพที่ 3.3 แนวคิดการออกแบบหุ่นยนต์ปลูกพืชแบบอัตโนมัติ .....	72
ภาพที่ 3.4 แบบโครงสร้างของหุ่นยนต์ปลูกพืช .....	73
ภาพที่ 3.5 โครงสร้างของหุ่นยนต์ปลูกพืช .....	74
ภาพที่ 3.6 ไดอะแกรมของวงจร H-Bridge Circuit .....	75
ภาพที่ 3.7 รูปแบบโครงสร้างของล้อ Mecanum Wheel.....	75
ภาพที่ 3.8 การเคลื่อนที่ของหุ่นยนต์ตามทิศทางและความเร็วเชิงมุมของล้อ.....	76
ภาพที่ 3.9 แสดงการเคลื่อนที่ของหุ่นยนต์ภายในแปลงปลูก.....	77
ภาพที่ 3.10 ปฏิภูมิการทำงานของหุ่นยนต์ .....	77
ภาพที่ 3.11 Raspberry Pi และ Camera Module.....	77
ภาพที่ 3.12 การตรวจจับกระถางและหาค่าจุด Centroids ด้วย HCT .....	79
ภาพที่ 3.13 แนวคิดการออกแบบระบบการรับรู้ของหุ่นยนต์.....	79
ภาพที่ 3.14 รูปแบบบาร์โค้ดของ MCU-Cam ทั้ง 16 รหัส.....	80

ภาพที่ 3.15 การทดสอบอ่านค่า Symbol ที่ใช้ในการควบคุมหุ่นยนต์.....	81
ภาพที่ 3.16 ตำแหน่งการจัดวางกระถางเพื่อใช้ในการทดสอบระบบการรับรู้.....	82
ภาพที่ 3.17 การทดสอบระบบการตรวจจับกระถางของหุ่นยนต์.....	82
ภาพที่ 3.18 หุ่นยนต์ประเภท Gantry Robot.....	83
ภาพที่ 3.19 แนวคิดในการออกแบบโครงสร้างระบบการเคลื่อนที่ในระนาบ x-y ของหุ่นยนต์.....	84
ภาพที่ 3.20 ตัวอย่างการประยุกต์ใช้ Lead Screw ในแนวแกน z.....	84
ภาพที่ 3.21 การประกอบโครงสร้างการเคลื่อนที่เชิงเส้นของหุ่นยนต์.....	84
ภาพที่ 3.22 แนวคิดการออกแบบหัวปลูก.....	85
ภาพที่ 3.23 Seed Injector ที่ใช้สำหรับหยอดเมล็ด.....	85
ภาพที่ 3.24 รูปแบบการทดสอบระบบการเคลื่อนที่ของชุดอุปกรณ์ส่วนปลาย.....	86
ภาพที่ 3.25 การเคลื่อนที่กลับไปยังตำแหน่ง Home.....	86
ภาพที่ 3.26 ตำแหน่งการจัดวางกระถางเพื่อใช้ในการทดสอบระบบการกระทำ.....	87
ภาพที่ 3.27 การออกแบบและพัฒนาซอฟต์แวร์ควบคุมการทำงาน.....	88
ภาพที่ 3.28 Flow Diagram แสดงโครงสร้างการทำงานของหุ่นยนต์ปลูกพืชอัตโนมัติ.....	90
ภาพที่ 3.29 ชุด Sensor Units ที่ใช้ในงานวิจัย.....	91
ภาพที่ 3.30 รูปแบบการทำงานของ MQTT Protocol.....	92
ภาพที่ 3.31 การติดตั้งระบบ IoT เพื่อรับส่งข้อมูลผ่าน HSPA+ บนเครือข่าย 3G.....	92
ภาพที่ 3.32 การเขียนโค้ดเพื่อ Publish และ Subscribe ไปยัง MQTT Broker.....	93
ภาพที่ 3.33 การพัฒนา Dashboard ของระบบด้วย Node-RED.....	94
ภาพที่ 3.34 สภาพแวดล้อมที่ใช้ในการทดสอบ.....	97
ภาพที่ 3.35 กระถางพลาสติกที่ใช้ในการทดสอบ.....	97
ภาพที่ 3.36 ตำแหน่งของการหยอดเมล็ดบนกระถางตามเงื่อนไขในการทดสอบ.....	98
ภาพที่ 4.1 โครงสร้าง Aluminum Profile ของหุ่นยนต์.....	100
ภาพที่ 4.2 การติดตั้งเพลทสำหรับยึดเกียร์มอเตอร์.....	101



ภาพที่ 4.3 การติดตั้งชุดขับเคลื่อนเข้ากับเฟรมของหุ่นยนต์ .....	101
ภาพที่ 4.4 ต้นแบบหุ่นยนต์เพื่อใช้ในการเพาะปลูกแบบอัตโนมัติ.....	102
ภาพที่ 4.5 รูปแบบเงื่อนไขในการทดสอบการเคลื่อนที่ของหุ่นยนต์ .....	102
ภาพที่ 4.6 ทดสอบการเคลื่อนที่ของหุ่นยนต์ด้วยการ Simulation บน Trestles.....	103
ภาพที่ 4.7 ชุดควบคุมการเคลื่อนที่เชิงเส้นของหุ่นยนต์ต้นแบบ .....	104
ภาพที่ 4.8 การทดสอบการเคลื่อนที่เชิงเส้นทั้ง 3 แกนของหุ่นยนต์ .....	105
ภาพที่ 4.9 Open Builds Control ซอฟต์แวร์ที่ใช้ในการ Simulation .....	105
ภาพที่ 4.10 การติดตั้งชุด Sensors ของ Perception Part บนแกน z.....	109
ภาพที่ 4.11 ความคลาดเคลื่อนขณะทดสอบการตรวจจับกระถาง .....	109
ภาพที่ 4.12 ผลการตรวจจับกระถางเมื่อจัดวางในรูปแบบอื่น.....	110
ภาพที่ 4.13 ผลการตรวจจับกระถางเมื่อจัดวางบนพื้นผิวที่แตกต่าง .....	111
ภาพที่ 4.14 การทดสอบการทำงานของหุ่นยนต์ด้วยสัญลักษณ์ Barcode.....	113
ภาพที่ 4.15 การติดตั้ง Ultrasonic distance sensor เพื่อใช้หาระยะของความสูง.....	114
ภาพที่ 4.16 การติดตั้ง MCU-Cam เพื่อใช้ในการตัดสินใจเคลื่อนที่ของหุ่นยนต์ .....	114
ภาพที่ 4.17 การออกแบบชุดหัวปลูกแบบ Seed Injector.....	115
ภาพที่ 4.18 ขั้นตอนการทำงานของ Seed Injector.....	116
ภาพที่ 4.19 โมลด์พลาสติกของ Seed Injector .....	116
ภาพที่ 4.20 การออกแบบเคสของ Seed Injector ด้วยซอฟต์แวร์ Autodesk Fusion 360.....	117
ภาพที่ 4.21 เคสของ Seed Injector ที่พิมพ์ด้วยเครื่อง 3D Printer.....	117
ภาพที่ 4.22 ต้นแบบของ Seed Injector .....	118
ภาพที่ 4.23 การติดตั้งชุด Seed Injector บนแกน z.....	118
ภาพที่ 4.24 การทดสอบระบบควบคุมการหยอดเมล็ด .....	120
ภาพที่ 4.25 ความแตกต่างของขนาดของเมล็ดพันธุ์ที่ถูกนำมาทดสอบ .....	120
ภาพที่ 4.26 ชุด Sensor Units ของระบบอินเทอร์เน็ทของสรรพสิ่ง .....	121

ภาพที่ 4.27 การแสดงผลข้อมูลที่ได้รับจากหุ่นยนต์ผ่าน WebSocket UI .....	123
ภาพที่ 4.28 การตรวจสอบการทำงานของระบบ IoT ผ่าน Serial Console.....	124
ภาพที่ 4.29 การทดสอบ Subscribe และ Publish ข้อมูลผ่าน Mobile Application.....	126
ภาพที่ 4.30 Dashboard แสดงผลข้อมูลที่ได้รับจากหุ่นยนต์ .....	127
ภาพที่ 4.31 โมเดลการทำงานร่วมกันของระบบ Hardware .....	129
ภาพที่ 4.32 ชุดแผงควบคุมการทำงานของหุ่นยนต์ต้นแบบ.....	129
ภาพที่ 4.33 การทดสอบการหยุดเมล็ดของหุ่นยนต์ในแปลงปลูกจำลอง.....	130
ภาพที่ 4.34 กราฟแสดงตำแหน่งของการหยุดเมล็ดในกระถาง .....	132
ภาพที่ 4.35 การหยุดเมล็ดลงในกระถางของ Seed Injector .....	134
ภาพที่ 4.36 การวัดตำแหน่ง $x$ และ $y$ เพื่อเก็บผลการทดสอบ .....	134
ภาพที่ 4.37 กราฟแสดงจำนวนกระถางที่หุ่นยนต์หยุดเมล็ดได้สำเร็จในการปฏิบัติภารกิจ.....	136
ภาพที่ 4.38 กราฟแสดงเวลาที่หุ่นยนต์ใช้ในการปฏิบัติภารกิจในแต่ละรอบ .....	136



# บทที่ 1

## บทนำ

### 1.1 หลักการและเหตุผล

เทคโนโลยีฟาร์มอัจฉริยะ (Smart Farm) [1] ตั้งอยู่บนแนวคิดของการทำเกษตรสมัยใหม่ที่เรียกว่า การเกษตรแบบแม่นยำสูง (Precision Agriculture) เป็นแนวคิดของการปลูกพืชในสภาพล้อมของดิน น้ำ แสง และอากาศ ที่มีความแตกต่างกัน ซึ่งส่งผลให้ผลผลิตที่ได้แตกต่างกัน โดยการปรับเปลี่ยนปัจจัยของการดูแลให้เหมาะสมนี้ จะทำให้สามารถสร้างผลผลิตได้อย่างมีประสิทธิภาพ จึงมีการนำเทคโนโลยีและนวัตกรรมต่าง ๆ มาใช้ในการผลิตสินค้าเกษตร ในด้านการลดต้นทุน เพิ่มคุณภาพการผลิต และมาตรฐานของสินค้า เพื่อให้เกษตรกรสามารถวางแผนถึงอุปสงค์การตลาดและเตรียมการผลิตได้ถูกต้องสอดคล้องกับความจริง รวมถึงการเพิ่มประสิทธิภาพการผลิตและการแก้ไขปัญหา โดยนำข้อมูลรอบด้านมาวิเคราะห์เพื่อเป็นองค์ประกอบในการตัดสินใจ ด้วยความก้าวหน้าของเทคโนโลยีในปัจจุบัน ส่งผลให้การทำเกษตรกรรมเริ่มก้าวเข้าสู่ยุคดิจิทัล [2] ที่มีการนำเทคโนโลยีและเครื่องจักรกลอัตโนมัติมาประยุกต์ใช้ เพื่อให้เกิดเป็นเกษตรกรรมความแม่นยำสูง [3] ที่ประกอบด้วย 5 วงจร ได้แก่ Data Collection เก็บข้อมูลของดิน น้ำ แสง ภูมิอากาศ ผลผลิต และปัจจัยที่เกี่ยวข้อง Diagnostics วินิจฉัยข้อมูล สร้าง กรองและเก็บข้อมูลเข้าสู่ฐานข้อมูล Analysis วิเคราะห์ข้อมูล ทำนายผลเชิงพื้นที่ และวางแผนจัดการ Precision Field Operations ปฏิบัติการตามแผนที่วางไว้ และ Evaluation ประเมินประสิทธิภาพของการปฏิบัติการ พร้อมทั้งประเมินความคุ้มค่าของการลงทุน โดยใช้ข้อมูลการเงินและเศรษฐศาสตร์อุตสาหกรรม

ปัจจุบันภาคเกษตรกรรมโลกได้เข้าสู่ยุคของการใช้หุ่นยนต์เพื่อการเกษตร หรือ Agricultural Robotics อย่างแท้จริง ภายหลังจากที่บุคลากรในภาคเกษตรกรรมมีแนวโน้มสูงว่ากำลังจะขาดแคลนแรงงานลงในอนาคต เนื่องจากคนวัยทำงานรุ่นใหม่ไม่เลือกทำอาชีพเกษตรกร เพราะเป็นงานหนัก และเกษตรกรปัจจุบันก็เริ่มเข้าสู่วัยชรา ในขณะที่สินค้าเกษตรและอาหารกลับเป็นสินค้าที่ตลาดโลกยังคงมีความต้องการบริโภคเพิ่มขึ้นทุกขณะ ดังนั้นปัจจัยสำคัญที่จะช่วยจัดการปัญหาดังกล่าวก็คือ การนำหุ่นยนต์เข้ามาใช้ทดแทนแรงงานมนุษย์ในกิจกรรมต่าง ๆ เช่น การหยอดเมล็ดพันธุ์ รดน้ำ ใส่ปุ๋ย กำจัดแมลงศัตรูพืช-วัชพืช เก็บเกี่ยวผลผลิต และตรวจสภาพพื้นที่การเกษตร ซึ่งนอกจากจะช่วยให้ผลิตสินค้าเกษตรได้ปริมาณมากขึ้นแล้ว ยังส่งผลให้มีต้นทุนที่ลดลง สามารถควบคุมคุณภาพการผลิตหรือการเก็บเกี่ยวได้อย่างแม่นยำอีกด้วย

อย่างไรก็ตาม งานวิจัยและพัฒนาการประยุกต์ใช้งานหุ่นยนต์ในภาคการเกษตรที่ผ่านมานั้น พบว่าเป็นการพัฒนาต้นแบบ (Prototype Phase) ซึ่งออกแบบให้ทำงานเฉพาะด้านภายในฟาร์ม เช่น หุ่นยนต์ตรวจการณ์และเก็บข้อมูล (Field Scouting and Data Collection Robot) [4] หุ่นยนต์กำจัดแมลงและศัตรูพืช (Pest and Weed Control Robot) [5] หุ่นยนต์เก็บเกี่ยวผลผลิต (Harvesting Robot) [6] หุ่นยนต์ฉีดพ่นสารเคมี (Target Spraying Robot) [7] หุ่นยนต์ตัดแต่งกิ่ง (Pruning Robot) [8] หุ่นยนต์รีดนม (Milking Robot) [9] หุ่นยนต์ตรวจจำแนกลักษณะพืช (Phenotyping Robot) [10] หุ่นยนต์จัดเรียงพืช (Sorting Robot) [11] และหุ่นยนต์เพาะปลูกพืช (Seeding and Planting Robot) [12] โดยส่วนใหญ่จะถูกใช้กับพืชที่มีความสำคัญทางเศรษฐกิจของต่างประเทศในพื้นที่แปลงปลูกและฟาร์มขนาดใหญ่ ซึ่งมีความแตกต่างจากประเทศไทยที่มีการทำเกษตรกรรมในสเกลที่ขนาดเล็กกว่า บนพื้นที่ของไร่ สวน หรือโรงเรือนเพาะปลูก และพืชบางชนิด เช่น เมล่อน มะเขือเทศราชินี ผักสลัด ก็ถูกปลูกอยู่บนภาชนะอย่างกระถางหรือถาดปลูกภายในโรงเรือน

ดังนั้นในงานวิจัยนี้ จึงได้นำเสนอการพัฒนาต้นแบบหุ่นยนต์ที่ใช้สำหรับทำการเพาะปลูกพืช (Planting Robot) แบบอัตโนมัติ บนกระถางพลาสติกที่ถูกจัดวางไว้บนแปลงปลูกภายในโรงเรือน ด้วยเทคโนโลยี Internet of Things for Artificial Intelligence Robots ซึ่งเป็นการประยุกต์การทำงานของหุ่นยนต์ (Robotics) ร่วมกับอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) และปัญญาประดิษฐ์ (Artificial Intelligence) เพื่อการปลูกพืชภายในโรงเรือนทดแทนการทำงานของมนุษย์ ลดเวลา ลดความยุ่งยาก เพิ่มประสิทธิภาพในการผลิต และสามารถควบคุมคุณภาพการผลิตได้อย่างแม่นยำ โดยใช้ต้นทุนต่ำซึ่งเกษตรกรสามารถลงทุนได้ ซึ่งเป็นการสนับสนุนการปรับตัวของภาคเกษตรกรในยุคประเทศไทย 4.0 นำไปสู่การยกระดับประสิทธิภาพการผลิตและเพิ่มผลผลิตให้กับเกษตรกร จนสามารถพัฒนาเกษตรกรให้มีความยั่งยืนในอาชีพเกษตรกรรมได้ต่อไป

## 1.2 วัตถุประสงค์ของการวิจัย

1. เพื่อออกแบบและพัฒนาหุ่นยนต์เพาะปลูกพืชอัตโนมัติ โดยใช้เทคโนโลยีอินเทอร์เน็ตของสรรพสิ่งร่วมกับการประมวลผลภาพและปัญญาประดิษฐ์
2. เพื่อประเมินประสิทธิภาพของหุ่นยนต์เพาะปลูกพืชอัตโนมัติที่พัฒนาขึ้น เปรียบเทียบกับหุ่นยนต์ที่ถูกเสนอมาเพื่อแก้ไขปัญหาในก่อนหน้านี้

### 1.3 ความสำคัญของการวิจัย

ภาคเกษตรกรรมของโลกได้เริ่มเข้าสู่ยุคของการใช้หุ่นยนต์เพื่อการเกษตรแล้ว ภายหลังจากพบว่าบุคลากรในภาคเกษตรกรรม มีแนวโน้มสูงที่กำลังจะขาดแคลนแรงงานในอนาคต จากข้อมูลของกระทรวงเกษตรสหรัฐอเมริกา [13] มีรายงานว่าในช่วง 10 ปีที่ผ่านมา (พ.ศ. 2545 – 2557) แรงงานในภาคการเกษตรมีจำนวนลดลงเป็นจำนวนมาก จากจำนวน 3.4 ล้านคน ลดลงเหลือเพียง 1 ล้านคน แรงงานประจำภาคอุตสาหกรรมการเกษตรลดลงเฉลี่ยมากกว่าร้อยละ 20 สร้างความเสียหายคิดเป็นมูลค่าสูงถึงแสนล้านบาทต่อปี ทั้งนี้มีเหตุผลมาจากคนในวัยทำงานรุ่นใหม่ มักไม่เลือกประกอบอาชีพทางการเกษตรเพราะเป็นงานหนัก และพบว่าเกษตรกรปัจจุบันก็เริ่มอย่างเข้าสู่วัยผู้สูงอายุ ในขณะที่สินค้าเกษตรกลับเป็นสินค้าที่ตลาดโลกยังคงมีความต้องการ และการบริโภคก็มีปริมาณเพิ่มขึ้นทุกขณะ

ดังนั้นในงานวิจัยนี้ จึงได้เสนอการพัฒนาต้นแบบหุ่นยนต์ที่ใช้สำหรับทำการเพาะปลูกพืชแบบอัตโนมัติ ซึ่งเป็นการประยุกต์การทำงานของหุ่นยนต์ (Robotics) เข้ากับเทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) การประมวลผลภาพ (Image Processing) และปัญญาประดิษฐ์ (Artificial Intelligent) เพื่อการเพาะปลูกพืชบนกระถางพลาสติกกลมที่ถูกจัดวางไว้ในแปลงปลูกของโรงเรือน ซึ่งสามารถปรับขนาดพื้นที่ (Scalable) และปรับเปลี่ยนตำแหน่งการจัดวาง (Adjust or Alter Position) ของกระถางปลูกได้

### 1.4 ขอบเขตของการวิจัย

1. ออกแบบและพัฒนาต้นแบบหุ่นยนต์เพื่อใช้ในการเพาะปลูกแบบอัตโนมัติ โดยการประยุกต์การทำงานของหุ่นยนต์ในแพลตฟอร์ม Manipulator Robot ซึ่งมีหน้าที่ในการควบคุมอุปกรณ์ส่วนปลายที่ใช้ในการปลูก ให้ทำงานร่วมกับแพลตฟอร์มของ Mobile Robot ซึ่งมีหน้าที่ควบคุมการเคลื่อนที่ของหุ่นยนต์ภายในแปลงปลูก

2. กำหนดลักษณะของการเพาะปลูกพืชไว้บนกระถางพลาสติกกลมสีดำ ขนาดเส้นผ่านศูนย์กลาง 5 นิ้ว ที่ถูกจัดวางไว้ในแปลงปลูกพืช ขนาด 50 x 150 เซนติเมตร จำนวน 2 แปลง โดยมีระยะห่างระหว่างแปลง 50 เซนติเมตร

3. พืชที่ใช้ในการวิจัย คือ พืชตระกูลสัลด ซึ่งเป็นพืชกินใบ ขนาดเล็กที่สามารถปลูกและเติบโตบนกระถางภายในโรงเรือนแบบเปิดได้เป็นอย่างดี

4. การประเมินประสิทธิภาพ (Performance Evaluation) ในการวิจัยครั้งนี้จะทำการประเมินประสิทธิภาพ 2 ด้าน ประกอบด้วย การวัดและการทดสอบความสามารถ (Capabilities) รวมถึงพฤติกรรม (Behaviors) ของหุ่นยนต์ที่ถูกพัฒนาขึ้น ประกอบด้วยตัวชี้วัด (Metrics) ดังนี้

- 1) ความเร็ว (Velocity)
- 2) ความแม่นยำ (Accuracy)
- 3) ระยะเวลาในการทำภารกิจ (Mission Duration)
- 4) อัตราความสำเร็จของภารกิจ (Mission Success Rate)

### 1.5 นิยามศัพท์เฉพาะ

1. หุ่นยนต์ (Robot) หมายถึง เครื่องจักรที่ถูกใช้งานแทนมนุษย์ มีการออกแบบให้สามารถตั้งลำดับการทำงาน ตลอดจนมีการใช้งานในหลากหลายหน้าที่ เช่น การใช้เคลื่อนย้ายวัสดุ อุปกรณ์ ส่วนประกอบต่าง ๆ เครื่องมือ หรืออุปกรณ์พิเศษ ตลอดจนการเคลื่อนที่ไปยังตำแหน่งตามที่ตั้งลำดับการทำงานไว้ เพื่อใช้ในงานหลากหลายประเภท และหุ่นยนต์แต่ละประเภทจะมีหน้าที่การทำงานในด้านต่าง ๆ ตามการควบคุมของมนุษย์ ซึ่งการควบคุมระบบในการสั่งงานระหว่างหุ่นยนต์และมนุษย์นี้สามารถทำได้โดยตรงและแบบอัตโนมัติ

2. อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things: IoT) หมายถึง สภาพแวดล้อมซึ่งประกอบไปด้วยอุปกรณ์ต่าง ๆ ที่มีการถ่ายโอนข้อมูลร่วมกันผ่านเครือข่าย โดยไม่จำเป็นต้องใช้ปฏิสัมพันธ์ระหว่างมนุษย์กับมนุษย์ หรือระหว่างมนุษย์กับคอมพิวเตอร์ ซึ่งระบบ Internet of Things นั้น ถูกพัฒนามาจากเทคโนโลยีไร้สายและอินเทอร์เน็ต โดยคำว่า Things ใน Internet of Things คือ อุปกรณ์ต่าง ๆ ที่สามารถอ้างอิงได้ด้วยหมายเลขเลขไอพี (IP Address) และมีความสามารถในการถ่ายโอนข้อมูลระหว่างกันได้ผ่านระบบเครือข่าย

3. การประมวลผลภาพ (Image Processing) หมายถึง การแปลงข้อมูลรูปภาพให้อยู่ในรูปแบบข้อมูลดิจิทัล (Digital Format) หรือข้อมูลเชิงตัวเลข โดยผ่านขั้นตอนต่าง ๆ ด้วยคอมพิวเตอร์ เช่น การทำให้ภาพมีความคมชัด การกำจัดสัญญาณรบกวนจากภาพ การแบ่งส่วนของวัตถุที่สนใจออกมาจากภาพ เพื่อวัตถุประสงค์ในการปรับปรุงคุณภาพของภาพดังกล่าวให้มีรายละเอียดที่ชัดเจนมากยิ่งขึ้น และเพื่อให้สามารถแปลความหมายของภาพได้ (Computer Interpretation)

4. ปัญญาประดิษฐ์ (Artificial Intelligence: AI) หมายถึง ความฉลาดเทียมที่สร้างขึ้นให้กับคอมพิวเตอร์หรือเครื่องจักร เพื่อเลียนแบบกลไกของกิจกรรมที่เกี่ยวข้องกับความคิดมนุษย์ เช่น การตัดสินใจ การแก้ปัญหา หรือการเรียนรู้ โดยแบ่งออกได้ 4 ประเภท ประกอบด้วย ระบบที่คิดเหมือนมนุษย์ (Systems That Think Like Humans) ระบบที่กระทำเหมือนมนุษย์ (Systems That Act Like Humans) ระบบที่คิดอย่างมีเหตุผล (Systems That Think Rationally) และระบบที่กระทำอย่างมีเหตุผล (Systems That Act Rationally)

5. การเพาะปลูกแบบอัตโนมัติ (Automated Cultivation) หมายถึง ระบบที่ทำงานแทนมนุษย์ในขบวนการเพาะปลูกพืช ประกอบไปด้วยฮาร์ดแวร์ (Hardware) เช่น เซ็นเซอร์สำหรับวัดข้อมูลภายในพื้นที่เพาะปลูก แขนกลหรืออุปกรณ์ที่ทำหน้าที่หยิบจับเมล็ดไปปลูก พร้อมทั้งมีกล้องเพื่อใช้เทคโนโลยี Computer Vision ในการประมวลผลการทำงานและตัดสินใจ โดยการทำงานเหล่านี้ถูกควบคุมด้วยซอฟต์แวร์ (Software) ซึ่งถูกออกแบบให้สามารถตั้งลำดับการทำงานและตัดสินใจในการทำงานได้อัตโนมัติจากข้อมูลที่ได้รับจากเซ็นเซอร์



## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

เนื้อหาในบทนี้จะกล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้องในการออกแบบและพัฒนาหุ่นยนต์ภาคการเกษตร (Agricultural Robotics) สำหรับการเพาะปลูกพืชแบบอัตโนมัติ ซึ่งประกอบด้วย หุ่นยนต์ (Robotics) การประมวลผลภาพ (Image Processing) ปัญญาประดิษฐ์ (Artificial Intelligence) อินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) และเนื้อหาส่วนสุดท้าย ซึ่งจะกล่าวถึงการทบทวนวรรณกรรมงานวิจัยที่เกี่ยวข้องซึ่งถูกใช้ในงานวิจัยนี้

#### 2.1 หุ่นยนต์

วิทยาการหุ่นยนต์ (Robotics Science) เป็นศาสตร์ประยุกต์ที่เกิดขึ้นจากความรู้จากหลากหลายสาขาวิชาในการออกแบบ วิเคราะห์ และพัฒนา ซึ่งประกอบด้วยสาขาวิชาเครื่องกลที่เกี่ยวข้องกับการออกแบบและวิเคราะห์กลไกการเคลื่อนไหว สาขาวิชาไฟฟ้าที่เกี่ยวข้องกับวงจรไฟฟ้า วงจรควบคุม วงจรการสื่อสาร ตลอดจนวงจรของเซ็นเซอร์ต่าง ๆ สาขาวิชาคอมพิวเตอร์ที่เกี่ยวข้องกับการออกแบบและพัฒนาซอฟต์แวร์ ตลอดจนอัลกอริทึมในการควบคุมการทำงานของหุ่นยนต์

คำว่า “Robot” ปรากฏครั้งแรกในปี ค.ศ. 1921 จากบทละคร เรื่อง “Rossum’s Universal Robot” หรือ R.U.R. ของ Karel Capek นักประพันธ์ชาวเช็ก มีเนื้อหากล่าวถึงมนุษย์เทียมที่ถูกสร้างขึ้นเพื่อทำงานเป็นทาส เนื่องจาก Robota ในภาษาชาวสลาฟมีความหมายว่าทาส ต่อมาในปี ค.ศ. 1940 Isaac Asimov นักเขียนนวนิยายวิทยาศาสตร์ชาวรัสเซีย ได้กล่าวถึงปฏิสัมพันธ์ในสังคมของหุ่นยนต์ที่มีรูปร่างคล้ายมนุษย์แต่ปราศจากจิตใจและกฎสามข้อของหุ่นยนต์ (Three Laws Robotics) [14] ที่กล่าวว่า “หุ่นยนต์จักต้องไม่ทำให้มนุษย์ได้รับบาดเจ็บ หุ่นยนต์จักต้องเชื่อฟังคำสั่งของมนุษย์ และหุ่นยนต์ต้องปกป้องสถานะความมีตัวตนไว้ トラบเท่าที่การกระทำนั้นมิได้ขัดแย้งต่อกฎข้อแรกหรือกฎข้อที่สอง” ซึ่งได้ยึดถือมาเป็นแนวปฏิบัติในการออกแบบหุ่นยนต์

หุ่นยนต์ (Robots) [15] เป็นเครื่องจักรที่ถูกใช้งานแทนมนุษย์ มีการออกแบบให้สามารถตั้งลำดับการทำงาน ตลอดจนมีการใช้งานในหลากหลายหน้าที่ เช่น การใช้เคลื่อนย้ายวัสดุ อุปกรณ์ส่วนประกอบต่าง ๆ เครื่องมือ หรืออุปกรณ์พิเศษ ตลอดจนการเคลื่อนที่ไปยังตำแหน่งต่าง ๆ ตามที่ตั้งลำดับการทำงานไว้ เพื่อใช้ในงานหลากหลายประเภท และหุ่นยนต์แต่ละประเภทจะมีหน้าที่การทำงานในด้านต่าง ๆ ตามการควบคุมของมนุษย์ ซึ่งการควบคุมระบบต่าง ๆ ในการสั่งงานระหว่างหุ่นยนต์และมนุษย์นี้ สามารถทำได้โดยตรงและแบบอัตโนมัติ โดยทั่วไปหุ่นยนต์ถูกสร้างขึ้นเพื่อ



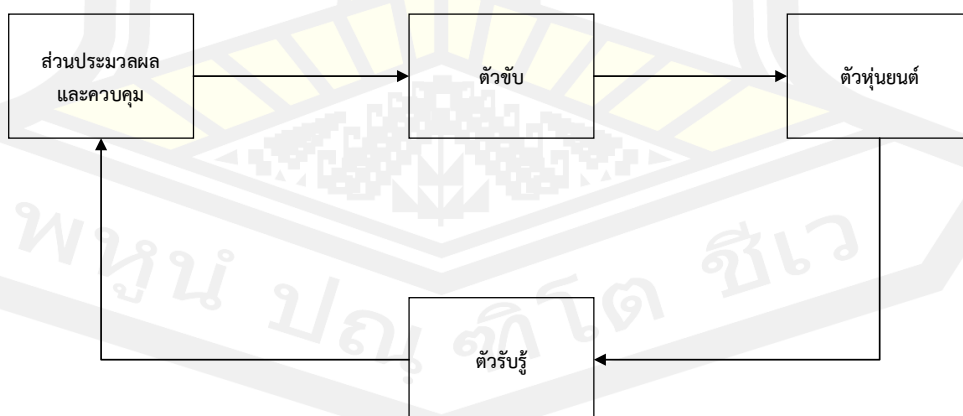
สำหรับงานที่มีความยากลำบาก ซึ่งจากนิยามดังกล่าวพบว่า หุ่นยนต์โดยทั่วไปจะมีองค์ประกอบหลักที่สำคัญ 4 ส่วน ประกอบด้วย

1. ตัวหุ่นยนต์ เป็นโครงสร้างทางกายภาพ ซึ่งทำงานตามการประมวลผลของชุดคำสั่งและสัญญาณการควบคุม โดยโครงสร้างของตัวหุ่นยนต์สามารถแบ่งออกได้เป็น 2 ส่วนหลัก คือ ส่วนขับเคลื่อน (Locomotion) เพื่อให้หุ่นยนต์สามารถเคลื่อนที่ไปยังตำแหน่งต่าง ๆ ได้ เช่น ล้อ หรือขา ซึ่งพบว่าการออกแบบและพัฒนาขาของหุ่นยนต์นั้นมีความซับซ้อนยุ่งยาก แต่จะส่งผลให้หุ่นยนต์สามารถเคลื่อนที่ได้ในรูปแบบที่อิสระขึ้น และส่วนตัวกระทำ (Manipulation) เพื่อให้หุ่นยนต์สามารถกระทำงานต่าง ๆ ได้ตามความต้องการ ตัวอย่างของกลไกส่วนนี้ คือ แขน หรือมือของหุ่นยนต์

2. ตัวขับเคลื่อน (Actuators) เป็นส่วนที่ทำให้หุ่นยนต์สามารถเคลื่อนที่และทำงาน ซึ่งพบว่าจากข้อจำกัดทางด้านเทคโนโลยีส่งผลให้ต้นทุนส่วนใหญ่ของหุ่นยนต์ คือ มอเตอร์ (Motor) โดยอาจจะมีการใช้มอเตอร์ประกอบเพื่อทำหน้าที่ลดทอนความเร็วหรือเพิ่มแรงขับให้มีความเหมาะสมกับภาระ (Load) อาจจะมีการขับเคลื่อนด้วยแรงดันจากของไหล (Fluid Power) ในงานที่มีภาระหนัก

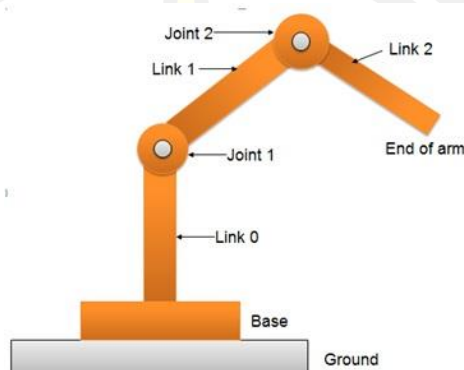
3. ตัวรับรู้ หรือเซ็นเซอร์ (Sensor) ทำหน้าที่คล้ายประสาทสัมผัสของหุ่นยนต์ในการตรวจวัด (Acquire) ข้อมูลที่เกี่ยวข้องกับระบบและสภาพแวดล้อมในแบบทันที (Real Time) เพื่อส่งต่อข้อมูลไปควบคุมการทำงานหรือประมวลผลต่อ เช่น ข้อมูลตำแหน่ง หรือความเร็วของชุดขับเคลื่อน ระยะห่างระหว่างวัตถุกับหุ่นยนต์ และข้อมูลภาพที่ได้รับจากกล้องที่ติดตั้งในหุ่นยนต์ เป็นต้น

4. ส่วนประมวลผลและควบคุม เป็นส่วนที่ทำหน้าที่คล้ายกับสมอง โดยจะนำข้อมูลที่ได้รับจากเซ็นเซอร์ต่าง ๆ มาประมวลผลและคำนวณสัญญาณควบคุม (Control Signal) เพื่อส่งต่อไปยังชุดตัวขับเคลื่อนให้ทำงานตามความต้องการ ซึ่งมักนำเอาคอมพิวเตอร์มาใช้ดำเนินการในส่วนนี้



ภาพที่ 2.1 องค์ประกอบของหุ่นยนต์และความสัมพันธ์ระหว่างกัน

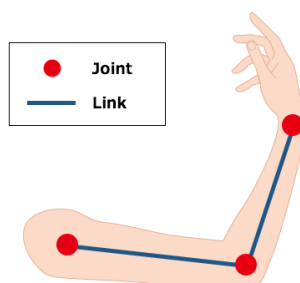
โดยทั่วไปแล้วองค์ประกอบทั้ง 4 ส่วนมักมีความสัมพันธ์กันดังแสดงในภาพที่ 2.1 ระบบหุ่นยนต์จะรับคำสั่ง แล้วส่วนประมวลผลและควบคุมจะทำหน้าที่แปลงคำสั่งดังกล่าวเป็นคำสั่งปฐมฐาน (Primitive Command) และนำไปประมวลผลกับข้อมูลที่ได้รับจากตัวรับรู้ เพื่อนำไปคำนวณเป็นสัญญาณไปควบคุมตัวขับให้หุ่นยนต์ทำตามคำสั่งต่อไป



ภาพที่ 2.2 ส่วนประกอบทางกลของหุ่นยนต์

ที่มา: [17]

ภาพที่ 2.2 แสดงส่วนประกอบหลักทางกล [18] ที่มีความสำคัญในการวิเคราะห์แขนหุ่นยนต์ เริ่มจากฐาน (Base) ที่ยึดติดกับพื้น ผนัง หรือเพดาน ถ้าต้องการเพิ่มขอบเขตการทำงานอาจติดตั้งล้อ หรือรางเลื่อนก็ได้ ข้อต่อ (Joints) จะเป็นตัวขับที่ทำให้แขนเคลื่อนที่ได้ คล้ายคลึงกับการทำงานของข้อมือ ข้อศอก และไหล่ของมนุษย์ ส่วนที่ใช้เชื่อมต่อระหว่างข้อต่อเรียกว่า ก้านต่อ (Links) เปรียบได้กับแขนมนุษย์ ดังภาพที่ 2.3 และส่วนที่ใช้สำหรับกับชิ้นงานเรียกว่า ตัวทำงานส่วนปลาย (End-Effector) หรือเครื่องมือปลายแขน (End of Arm Tooling) เป็นส่วนที่ติดตั้งเครื่องมือที่มีความแตกต่างกันตามลักษณะของงาน เช่น มือจับ (Gripper) สำหรับใช้ในงานจับหรือเคลื่อนย้ายวัตถุ

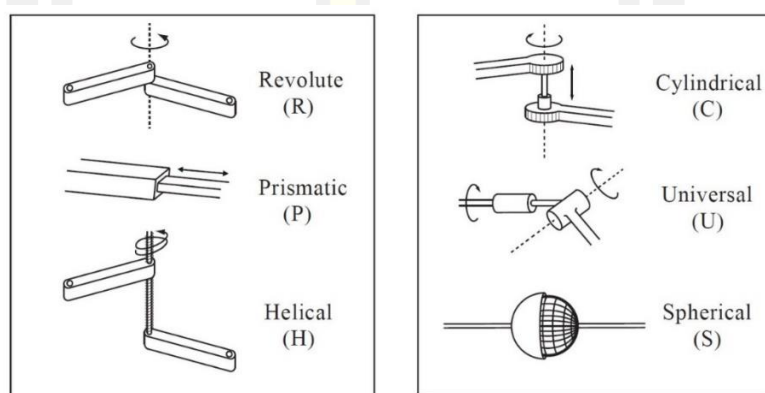


ภาพที่ 2.3 ลักษณะของ Joint และ Link เมื่อเปรียบเทียบกับแขนของมนุษย์

ที่มา: [19]

ข้อต่อที่ใช้ในแขนหุ่นยนต์ [20] สามารถแบ่งออกได้ 6 ลักษณะ คือ

- 1) ข้อต่อแบบหมุน (Revolute)
- 2) ข้อต่อแนวเส้นตรง (Prismatic)
- 3) ข้อต่อแบบเกลียว (Helical)
- 4) ข้อต่อทรงกระบอก (Cylindrical)
- 5) ข้อต่ออเนก (Universal)
- 6) ข้อต่อทรงกลม (Spherical)



ภาพที่ 2.4 ลักษณะของข้อต่อที่ถูกใช้ในแขนหุ่นยนต์

ที่มา: [20]

ปัจจุบันพบว่า ข้อต่อที่ใช้ในแขนหุ่นยนต์โดยทั่วไปมักเป็นแบบ ข้อต่อแบบหมุน ซึ่งแทนด้วยสัญลักษณ์ “R” หรือรูปร่างกลม หรือทรงกระบอก และข้อต่อแนวเส้นตรง ซึ่งแทนด้วยสัญลักษณ์ “P” หรือรูปสี่เหลี่ยม โดยข้อต่อทั้งสองลักษณะนี้จะมีลูกศรเพื่อแสดงทิศทางการเคลื่อนที่ เช่นเดียวกับโมเดลที่สร้างโดยซอฟต์แวร์คอมพิวเตอร์ ซึ่งในการแสดงโมเดลโครงสร้างของหุ่นยนต์จะแทนก้านต่อด้วยเส้นตรงเพื่อความเรียบง่าย และละเว้นการวาดตัวทำงานส่วนปลาย เนื่องจากโมดูลตัวทำงานส่วนปลายสามารถติดตั้งเครื่องมือได้หลากหลายประเภท ซึ่งไม่ใช่มีข้อจำกัด

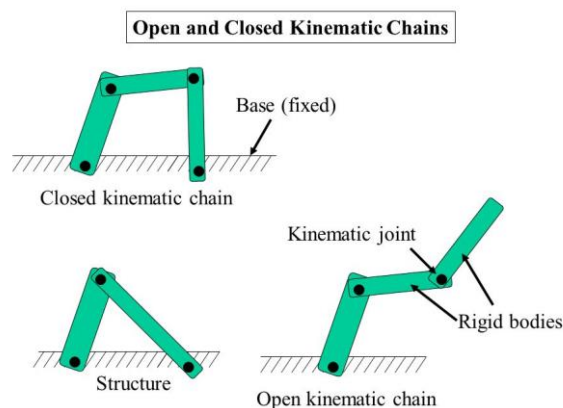
## 2.2 หุ่นยนต์แบบอนุกรม

หุ่นยนต์แบบอนุกรมเป็นหุ่นยนต์ที่มีโครงสร้างหลักทางกลซึ่งประกอบขึ้นจากการต่อกันของก้านต่อ (Link) ต่าง ๆ ด้วยข้อต่อ (Joint) จากส่วนฐาน (Base) ซึ่งอาจไม่อยู่หนึ่งกับที่ เรียงลำดับไปจนถึงส่วนปลายแขน (End Effector) ลักษณะการต่อกันของก้านต่อลักษณะดังกล่าวจะทำให้มีเส้นทางเชื่อมระหว่างจุด 2 จุดใด ๆ บนโครงสร้างเส้นทางเดียว เรียกว่า “โครงสร้างแบบโซ่เปิด” ดัง

ภาพที่ 2.5 ซึ่งสามารถพบเห็นได้ทั่วไป เช่น แขนหรือขาของสิ่งมีชีวิต ซึ่งหุ่นยนต์แบบอนุกรมจะมีองค์ประกอบที่สำคัญ ดังนี้

1. ก้านต่อ (Link) เป็นชิ้นส่วนสำคัญที่ประกอบขึ้นเป็นตัวหุ่นยนต์ โดยผลิตจากวัสดุที่แข็งแรง ในการวิเคราะห์เบื้องต้นจึงนิยมสมมติให้ก้านต่อเป็นวัตถุที่แข็งเกร็ง ซึ่งอาจมีรูปร่างที่แตกต่างกันออกไปตามการออกแบบและรูปแบบการใช้งาน แต่ก้านต่อทุกชิ้นจะต้องมีส่วนที่ใช้สำหรับเชื่อมต่อกันต่อชิ้นอื่น อย่างน้อย 1 จุด โดยก้านต่อ 2 ชิ้น ที่มีการเชื่อมต่อกันทางกายภาพ แต่ไม่ได้มีการเคลื่อนที่สัมพัทธ์ (Relative Motion) กันเลย อาจถือได้ว่าเป็นก้านต่อชิ้นเดียวกัน

2. ข้อต่อ (Joint) เกิดจากการประสานก้านต่อตั้งแต่ 2 ชิ้น เข้าด้วยกัน โดยอาจจะมีชิ้นส่วนเสริม เช่น เพลา หรือบุชชิ่ง ช่วยในการเคลื่อนที่สัมพัทธ์ ในด้านของจลศาสตร์ ข้อต่อเป็นชิ้นส่วนที่จำกัดการเคลื่อนที่สัมพัทธ์ของก้านต่อ หากไม่มีข้อต่อเลย ก้านต่อ 2 ชิ้นจะไม่ถูกจำกัดการเคลื่อนที่ และสามารถเคลื่อนที่ได้อย่างอิสระ หรือในอีกด้าน ข้อต่ออาจจะจำกัดการเคลื่อนที่สัมพัทธ์ของก้านต่ออย่างสมบูรณ์ เช่น ข้อต่อจากการเชื่อมก้านเข้าด้วยกัน ข้อต่อที่มักพบโดยทั่วไปในหุ่นยนต์มักเป็นข้อต่ออย่างง่าย (Simple Joint) ซึ่งเป็นข้อต่อที่จำกัดจำนวนองศาเสรีของการเคลื่อนที่สัมพัทธ์ จาก 6 ลดลงเหลือ 1 (Single Degree of Freedom)



ภาพที่ 2.5 แนวคิดของ Close and Open Kinematic Chain

ที่มา: [21]

หุ่นยนต์ทั่วไปมักจะติดตั้งตัวขับที่ข้อต่อ เนื่องจากสามารถควบคุมการเคลื่อนที่สัมพัทธ์ของก้านต่อได้สะดวกที่สุด หากเป็นข้อต่อเชิงมุม ก้านต่อสัมพัทธ์รอบแกนข้อต่อ (Joint Axis) ที่ทับกัน (Coincide) กับแนวเพลา และแกนของเพลามอเตอร์ ในกรณีของข้อต่อแบบเชิงเส้น ก้านต่อจะเคลื่อนที่เชิงเส้นตามแนวแกนข้อต่อที่ทับกับแนวบุชชิ่ง และแกนของตัวขับเชิงเส้น สำหรับหุ่นยนต์ที่มี

โครงสร้างแบบโซ่เปิดนั้น จำนวนองศาเสรีของหุ่นยนต์จะเท่ากับผลรวมของจำนวนองศาเสรีของทุก ๆ ข้อต่อ ดังนั้นหุ่นยนต์ที่ได้รับการออกแบบสำหรับงานทั่วไป จึงควรมีจำนวนองศาเสรีขั้นต่ำเท่ากับ 6 โดยแบ่งเป็นส่วนของการเคลื่อนที่เชิงเส้นเท่ากับ 3 และการเคลื่อนที่เชิงมุมเท่ากับ 3

3. ฐาน (Base) เป็นส่วนที่ติดกับพื้น มักจะไม่เคลื่อนไหว หรืออาจจะเคลื่อนไหวได้ เช่น ฐานของหุ่นยนต์ชนิดเคลื่อนที่ได้ (Mobile Robot) ฐานที่เป็นส่วนที่รองรับแรงปฏิกิริยาที่เกิดขึ้นเนื่องจากแรงภายนอกและแรงพลศาสตร์ของหุ่นยนต์ ดังนั้นฐานจะต้องได้รับการออกแบบให้มีความแข็งแรงอย่างพอเพียง นอกจากนี้ ฐานยังมีความสำคัญที่มักใช้เป็นตำแหน่งของเฟรมอ้างอิง (Reference Frame)

4. ข้อมือ (Wrist) เป็นส่วนของโครงสร้างที่ประกอบขึ้นมาจากก้านต่อและข้อต่อ ที่มักให้การเคลื่อนที่สัมผัสแบบหมุนรอบจุดศูนย์กลางของข้อมือ (Wrist Point) เรียกว่า “ข้อมูลแบบทรงกลม (Spherical Wrist)” ข้อมือมักได้รับการติดตั้งที่ปลายแขนส่วนล่าง (Forearm) ก่อนถึงส่วนปลายแขน ข้อมือที่มีการเคลื่อนที่สัมผัสแบบหมุนรอบจุดร่วม จะมีแกนข้อต่อทำงาน (Effective Joint Axis) ตัดกันที่จุดร่วมนี้ โครงสร้างดังกล่าวจะช่วยให้การวิเคราะห์ทางจลศาสตร์กระทำได้ง่ายขึ้น จะทำให้การวิเคราะห์ตำแหน่ง (Position) และการหมุน (Orientation) ของส่วนปลายแขนนั้น แยกออกจากกันได้ ส่งผลให้สามารถหาผลเฉลยปิด (Closed-form Solution) ของปัญหาจลศาสตร์ ผกผันได้

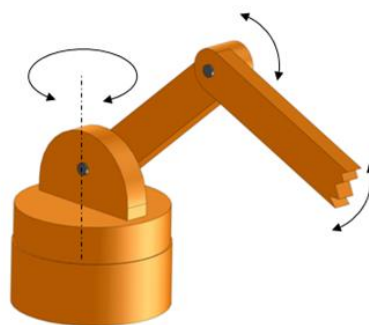
5. ส่วนปลายแขน (End Effector) เป็นส่วนที่มีกติดั้งถัดจากข้อมือเพื่อใช้ทำงานที่ต้องการ ส่วนปลายแขนมีหลายรูปแบบขึ้นอยู่กับลักษณะงาน ส่วนปลายแขนที่ง่ายที่สุด ได้แก่ ตัวจับ (Gripper) ที่มีจำนวนองศาเสรีเท่ากับ 1 สำหรับการเปิดและปิด เพื่อจับวัตถุที่ต้องการ โดยทั่วไป หุ่นยนต์มักถูกออกแบบให้สามารถเปลี่ยนส่วนปลายแขนเพื่อทำงานที่หลากหลาย มือหุ่น (Hand) ที่ประกอบไปด้วยนิ้วและฝ่ามือ

## 2.3 โครงสร้างของหุ่นยนต์

หุ่นยนต์แบบอนุกรมนั้น ถึงแม้ว่าจะสามารถสร้างขึ้นมาด้วยการจัดเรียงกันของข้อต่ออย่างง่ายและก้านต่อในรูปแบบต่าง ๆ ก็ตาม แต่ในทางปฏิบัติ กลับมีโครงสร้างทางจลศาสตร์เพียงบางรูปแบบเท่านั้นที่ถูกนำมาใช้งานจริง เนื่องจากสาเหตุบางประการ เช่น ความยากในการออกแบบและพัฒนา หรือมีสมบัติทางพลศาสตร์ที่ไม่พึงประสงค์ ซึ่งโครงสร้างทางจลศาสตร์ที่นิยมใช้ในแขนหุ่นยนต์แบบอนุกรม [16, 22, 23] สามารถจำแนกเป็นประเภท ดังนี้

1. แขนแบบพับ (Articulated Arm) ซึ่งเป็นแขนที่มีความคล้ายคลึงกับแขนของมนุษย์ (Anthropomorphic Arm) โดยก้านต่อในโครงสร้างมีการเคลื่อนไหวคล้ายคลึงกับ ลำตัว (Body)

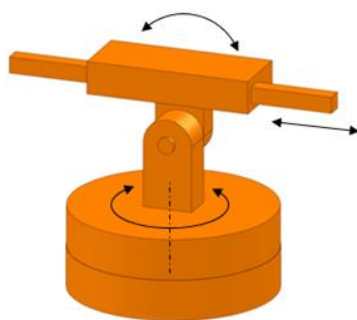
แขนส่วนบน (Upper Arm) และแขนส่วนล่าง (Forearm) ของมนุษย์ ส่วนข้อต่อในโครงสร้างทำหน้าที่คล้ายคลึงกับ เอว (Waist) หัวไหล่ (Shoulder) และข้อศอก (Elbow) ซึ่งแขนแบบพับนี้มีแกนข้อต่อของหัวไหล่และศอกที่ขนานกัน และตั้งฉากกับแกนข้อต่อของเอว ดังภาพที่ 2.6 โครงสร้างนี้มีจุดเด่นที่ปริภูมิทำงานที่มีขนาดใหญ่เมื่อเทียบกับขนาดของแขน ตัวอย่างหุ่นยนต์ชนิดนี้คือ งานประกอบ งานเชื่อม และงานขนถ่ายวัสดุ เป็นต้น



ภาพที่ 2.6 แขนหุ่นยนต์แบบพับ (Articulated Arm)

ที่มา: [17]

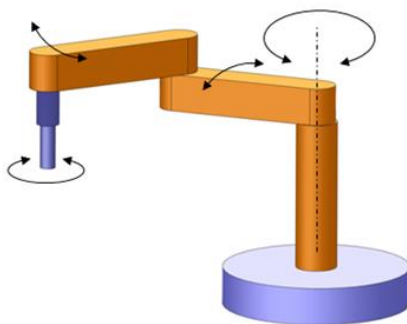
2. แขนแบบพิกัดทรงกลม (Spherical Arm) เป็นแขนที่มีข้อต่อหลัก ถูกจัดเรียงกันตามพิกัดทรงกลม โดยจะมีพิกัดข้อต่อ (Joint Coordinates) สอดคล้องกับมุมแอสซิมาท (Azimuth Angle) มุมเงย (Elevation Angle) และระยะรัศมี (Radial Distance) ของระบบพิกัดทรงกลม ดังภาพที่ 2.7 จากโครงสร้างดังกล่าวจึงทำให้ปริภูมิการทำงานของแขนมีรูปร่างเป็นทรงกลมกลวง เนื่องจากขีดจำกัดบนและล่างของระยะชักในข้อต่อ ตัวอย่างหุ่นยนต์ชนิดนี้คือ บันไดของรถดับเพลิง และ Material Transfer เป็นต้น



ภาพที่ 2.7 แขนหุ่นยนต์แบบพิกัดทรงกลม (Spherical Arm)

ที่มา: [17]

3. แขนกลแบบสการา (SCARA Arm) หรือ Selective Compliant Articulated Robot for Assembly ซึ่งมีจุดเด่นของโครงสร้างที่แกนข้อต่อหลักถูกวางตัวในแนวขนานกันทั้งหมด โดย 2 ข้อต่อแรกเป็นข้อต่อแบบเชิงมุมที่สามารถเคลื่อนที่ได้ครอบคลุมในระนาบหนึ่งได้ ส่วนข้อต่อที่ 3 เป็นข้อต่อแบบเชิงเส้น หรือข้อต่อแบบเกลียว (Helical Joint) ดังภาพที่ 2.8 เพื่อให้ส่วนปลายแขนสามารถเคลื่อนที่ในแนวตั้งฉากกับระนาบนั้นได้ จึงเหมาะสำหรับงานประเภทการหยิบจับและวาง (Pick and Place Operation) ซึ่งบริเวณข้อมือของแขนหุ่นยนต์อาจมีการติดตั้งอุปกรณ์ปรับความยืดหยุ่นที่เรียกว่า Remote Center Compliance (RCC) เพื่อให้หุ่นยนต์สามารถทำงานประกอบชิ้นส่วน (Assembly Task) ที่มีค่าระยะเผื่อ (Tolerance) ที่ค่อนข้างจำกัดได้โดยไม่ติดขัด ตัวอย่างหุ่นยนต์ชนิดนี้คือ Automatic Assembly, Die Casting หรือ Welding เป็นต้น



ภาพที่ 2.8 แขนหุ่นยนต์แบบสการา (SCARA Arm)

ที่มา: [17]

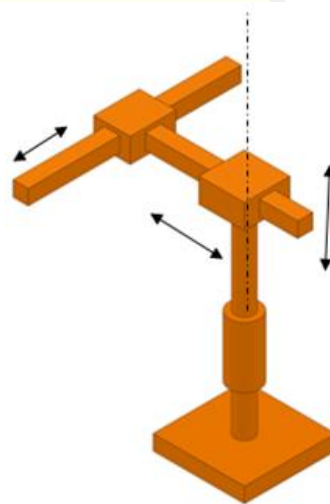
4. แขนแบบพิกัดทรงกระบอก (Cylindrical Arm) ซึ่งโครงสร้างแขนที่มีข้อต่อหลักจัดเรียงกันตามพิกัดทรงกระบอก สอดคล้องกับมุมกวาด (Sweep Angle) ความสูง (Height) และระยะรัศมี (Radial Distance) ของระบบพิกัดทรงกระบอก ดังภาพที่ 2.9 เนื่องจากปัจจุบันโมดูลตัวขับเคลื่อนเชิงเส้นสามารถหาได้ง่าย จึงทำให้การสร้างหุ่นยนต์ประเภทนี้สร้างได้ง่ายและสะดวก เช่น การนำเอาตัวขับเคลื่อนบอลสกรู (Ball Screw Joint) มาประกอบเข้าด้วยกัน และเนื่องจากขีดจำกัดบนและล่างของระยะชักในข้อต่อ จึงส่งผลกระทบต่อประสิทธิภาพการทำงานของแขนมีรูปร่างเป็นทรงกระบอกกลาง ตัวอย่างหุ่นยนต์ชนิดนี้ คือ เครื่องสำหรับสร้างตึกสูง หรือ อุปกรณ์ขนถ่ายวัสดุทั่วไป



ภาพที่ 2.9 แขนหุ่นยนต์แบบพิกัดทรงกระบอก (Cylindrical Arm)

ที่มา: [17]

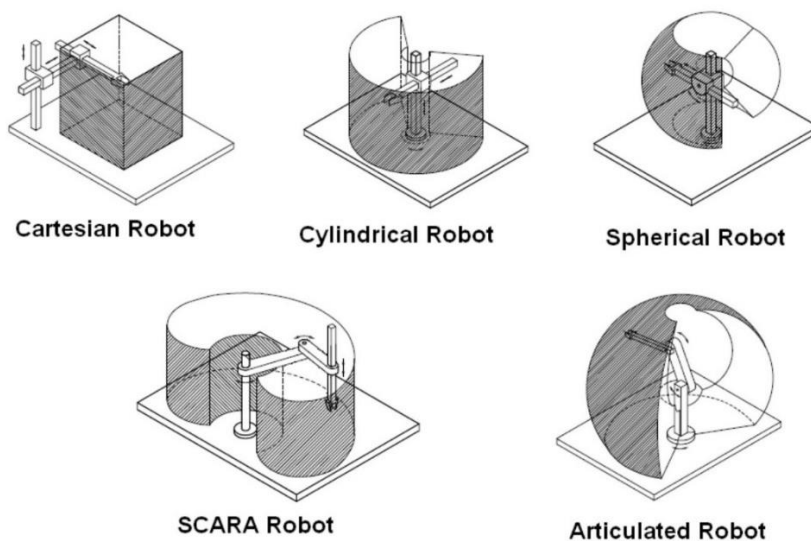
5. แขนแบบพิกัดคาร์ทีเซียน (Cartesian Arm) โครงสร้างของแขนประเภทนี้ได้รับแนวคิดมาจากการเคลื่อนที่ในปริภูมิสามมิติ ด้วยพิกัดคาร์ทีเซียน  $x$ ,  $y$  และ  $z$  ซึ่งข้อต่อหลักทั้งสามเป็นข้อต่อแบบเชิงเส้นทั้งหมด โดยสร้างจากการนำเอาโมดูลตัวขับเคลื่อนเชิงเส้นมาประกอบกัน ดังภาพที่ 2.10 มักพบในเครื่องจักร CNC หรือเครนแบบแกนทรี (Gantry Crane) หรือ Inspection หุ่นยนต์ประเภทนี้สามารถออกแบบให้รับแรงมากได้



ภาพที่ 2.10 แขนหุ่นยนต์แบบพิกัดคาร์ทีเซียน (Cartesian Arm)

ที่มา: [17]





ภาพที่ 2.11 ปฏิภูมิการทำงานของแขนหุ่นยนต์ประเภทต่าง ๆ

ที่มา: [24]

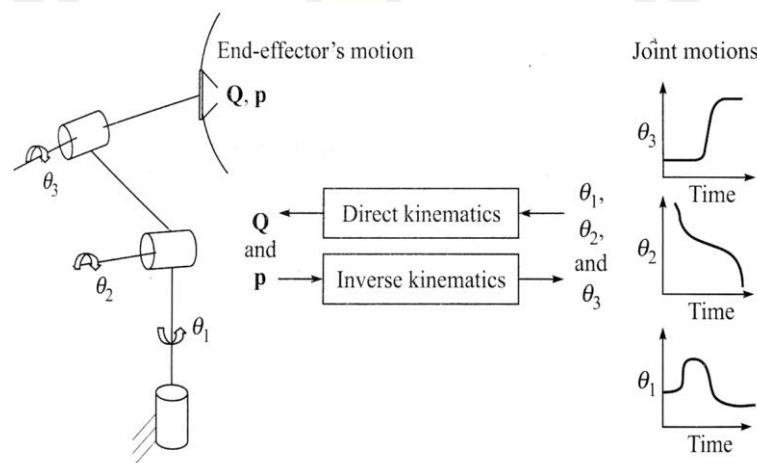
## 2.4 จลนศาสตร์ของหุ่นยนต์

จลนศาสตร์ (Kinematics) [16, 18] เป็นเครื่องมือทางคณิตศาสตร์ที่ใช้สำหรับศึกษาเกี่ยวกับตำแหน่ง (Position) ทิศทางการหมุน (Orientation) และการเคลื่อนที่ทั้งเชิงเส้น (Translation Motion) และเชิงมุม (Rotational Motion) ของปลายแขนหุ่นยนต์ (End-Effectors) โดยไม่พิจารณาถึงปัจจัยที่ก่อให้เกิดการเคลื่อนที่ ซึ่งโดยพื้นฐานแล้วหุ่นยนต์เป็นระบบพลศาสตร์ที่ต้องเกี่ยวข้องกับการเคลื่อนที่ผ่านตัวขับ และเพื่อให้หุ่นยนต์สามารถทำงานได้หลากหลาย หุ่นยนต์จะต้องสามารถเคลื่อนที่ในปริภูมิสามมิติได้ ดังนั้นการศึกษาเกี่ยวกับการเคลื่อนที่ของหุ่นยนต์จึงจำเป็นต้องเริ่มจากจลนศาสตร์ของปริภูมิสามมิติ ดังภาพที่ 2.12

เนื่องจากจลนศาสตร์เป็นการนิยามการเคลื่อนที่ของวัตถุโดยไม่พิจารณาแรงที่กระทำกับโครงสร้างหรือพลวัต ดังนั้นตำแหน่งทางจลนศาสตร์จะขึ้นอยู่กับตำแหน่งทางเรขาคณิตของวัตถุ หากเปรียบเทียบกับหุ่นยนต์ก็จะขึ้นอยู่กับโครงสร้างของข้อต่อและก้านต่อ ซึ่งแบ่งออกเป็น 2 ประเภทได้แก่

1. จลนศาสตร์ข้างหน้า (Forward Kinematics: FK) คือ การคำนวณตำแหน่ง ความเร็ว และความเร่งของเฟรมเครื่องมืออ้างอิงกับเฟรมฐานหรือเฟรมตรึง เมื่อทราบค่าตำแหน่ง ความเร็ว และความเร่งของตัวแปรข้อต่อ ซึ่งเป็นการหาค่าพิกัดและทิศทางในรูปแบบของการแปลงเอกพจน์ของตัวทำงานส่วนปลาย เมื่อทราบตำแหน่งและมุมของข้อต่อทั้งหมด โดยการวิเคราะห์ดังกล่าว เป็นการวิเคราะห์การเคลื่อนที่และทิศทางของตัวทำงานที่ปลายแขน ผู้ใช้จะป้อนคำสั่งอินพุตให้แก่ละแกนของ

หุ่นยนต์โดยตรง หากข้อต่อหุ่นยนต์เป็นแกนแบบเลื่อน (Prismatic Link) ค่าพารามิเตอร์ที่ได้จะบอกเป็นค่าตำแหน่งในแนวแกนเลื่อนนั้น หรือถ้าข้อต่อเป็นแกนแบบหมุน (Revolute Link) ค่าที่ได้จะบอกเป็นมุมองศาของการหมุน ขณะที่ข้อต่อของหุ่นยนต์เคลื่อนที่ไปในปริภูมิสามมิติ การวิเคราะห์จลนศาสตร์ตรงนี้ ต้องทราบตัวแปรซึ่งเป็นค่าตำแหน่งของแต่ละข้อต่อของหุ่นยนต์ทั้งหมด ทำการคำนวณตามลำดับ (Robotic Chain) ของข้อต่อไปยังปลายแขนของหุ่นยนต์ โดยค่าตัวแปรต่าง ๆ นี้ขึ้นอยู่กับลักษณะโครงสร้างของหุ่นยนต์แต่ละประเภท ดังนั้น การเคลื่อนที่ในลักษณะต่าง ๆ ของหุ่นยนต์จึงส่งผลให้ค่าพารามิเตอร์เปลี่ยนแปลงไปตามความแตกต่างกันของแต่ละชนิดข้อต่อ การคำนวณค่าในเมทริกซ์จะวางเรียงกันตามลำดับ มีความต่อเนื่องกันของตัวแปรข้อต่อต่าง ๆ จากซ้ายไปขวา (Post Matrix) โดยมี ลักษณะแบบอนุกรม ซึ่งแกนหุ่นยนต์ในแต่ละแกนจะถูกขับเคลื่อนโดย ตัวขับเคลื่อน (Actuator) เช่น มอเตอร์ไฟฟ้า กระจบอกลูกสูบไฮดรอลิกส์ หรือนิวเมตริกซ์ เป็นต้น



ภาพที่ 2.12 จลนศาสตร์ข้างหน้าและจลนศาสตร์ผกผัน

ที่มา: [25]

2. จลนศาสตร์ผกผัน (Inverse Kinematics: IK) คือ การคำนวณตำแหน่ง ความเร็ว และความเร่งของตัวแปรข้อต่อ เมื่อทราบค่าตำแหน่ง ความเร็ว และความเร่งของเฟรมเครื่องมือ ซึ่งเป็นการหาค่าตำแหน่งและมุมของข้อต่อ เมื่อทราบค่าเมทริกซ์เอกพันธ์ของตัวทำงานส่วนปลายเทียบกับเฟรมฐาน ซึ่งเป็นปัญหาที่มีความยากกว่าจลนศาสตร์ข้างหน้า โดยวิเคราะห์ตำแหน่งที่ปลายแขนหุ่นยนต์ วิธีการสั่งการให้ปลายแขนหุ่นยนต์เคลื่อนที่ไปยังตำแหน่งเป้าหมายที่ต้องการ ตามพิกัด  $x, y, z$  ทันทีแล้วทำการหยิบจับชิ้นงาน หรือทำงานอื่นตามลักษณะงานที่ต้องการ กระทำได้โดยกำหนดคำสั่งอินพุตที่ป้อนเป็นตำแหน่งที่ปลายแขน ส่วนที่ทำหน้าที่ควบคุมระบบ (Control System) จะทำการคำนวณค่ามุมแต่ละมุมของข้อต่อแต่ละส่วนที่ต้องการให้เคลื่อนที่ไปยังวัตถุนั้น เพื่อให้ส่วนปลาย

แขนหุ่นยนต์เคลื่อนที่ไปยังตำแหน่งเป้าหมายที่ต้องการ การคำนวณแบบนี้เป็นลักษณะการคำนวณหา ย้อนกลับ (Inverse) จากตำแหน่งเป้าหมายของปลายแขนหุ่นยนต์ ที่ถูกระบุไว้เทียบกับเฟรมอ้างอิงที่ ฐานหุ่นยนต์เรียกการควบคุมแบบนี้ว่า “จลนศาสตร์ผกผัน” ในทางปฏิบัติกับการควบคุมหุ่นยนต์จะให้ความสำคัญกับจลนศาสตร์ผกผันมากกว่าจลนศาสตร์ตรง ผู้ปฏิบัติงานจะกำหนดตำแหน่งที่ปลายแขน หุ่นยนต์ เพื่อให้ระบบหุ่นยนต์ ทำการคำนวณหาตำแหน่งหรือมุมในแต่ละแกนเอง โดยที่ผู้ปฏิบัติงาน ไม่ต้องทราบถึงความสัมพันธ์ระหว่างตำแหน่งปลายแขนและตำแหน่งหรือองศาของแกนเลย เนื่องจาก บริษัทผู้ออกแบบหุ่นยนต์ได้ทำการเขียนโปรแกรมคำนวณความสัมพันธ์ดังกล่าวเพื่อแก้ปัญหา จลนศาสตร์ผกผันนี้ไว้แล้วซึ่งการหาคำตอบจลนศาสตร์ผกผัน สามารถแยกออกเป็น 2 ประเภทคือ รูปแบบการคำนวณจลนศาสตร์ผกผันแบบเรขาคณิต (Geometric) และแบบเลขคณิต (Arithmetic) โดยรูปแบบเรขาคณิตจะคิดโดยใช้ภาพหรือการเขียนโครงสร้าง ส่วนรูปแบบเลขคณิตจะอาศัยสมการ เป็นสำคัญแต่ทั้งสองวิธีได้คำตอบเหมือนกัน การเลือกวิธีการใดไปใช้ขึ้นอยู่กับความถนัดของผู้ใช้เอง อย่างไรก็ตามการคำนวณแบบเรขาคณิตนั้นมีข้อจำกัดในเรื่องความซับซ้อนหรือในกรณีที่มีจำนวนแกน มาก ๆ การเขียนรูปโครงสร้างเพื่อให้ได้มาซึ่งคำตอบนั้นจะเป็นการยากต่อการทำความเข้าใจ

## 2.5 ระบบการมองเห็นของหุ่นยนต์

ระบบ Machine Vision System [26] หรือ MV เป็นระบบการมองเห็นของเครื่องจักรหรือ หุ่นยนต์เพื่องานที่ต้องการความสลับซับซ้อน หรือความแม่นยำสูง ด้วยการใช้การมองเห็นของ คอมพิวเตอร์ (Computer Vision) ในการประมวลผลภาพ (Image Processing) จากกล้องถ่ายภาพ ดิจิทัลที่ส่งไปยังชิ้นงาน แล้วทำการปรับปรุงคุณภาพของภาพ เพื่อให้หุ่นยนต์สามารถมองเห็นได้ผ่าน ทางซอฟต์แวร์ นำไปใช้ในการหาความเหมือน การจดจำ และการแยกแยะวัตถุ เพื่อให้หุ่นยนต์ ตัดสินใจทำงานได้ถูกต้องตามคำสั่งที่ต้องการ

ระบบการมองเห็นของเครื่องจักรประกอบด้วย การทำเทรชโฮลด์ (Thresholding) การ แยกภาพออกเป็นส่วน ๆ (Segmentation) การหารอยเป็อน (Blob Extraction) การรู้จำแบบ (Pattern Recognition) การอ่านบาร์โค้ด (Barcode Reading) การรับรู้อักขระด้วยแสง (Optical Character Recognition: OCR) การตรวจวัดขนาดของวัตถุ (Size) การหารูปร่างของภาพ (Shape Detection) การหาขอบภาพ (Edge Detection) และการเข้าคู่รูปแบบ (Template Matching) เป็น ต้น โดยองค์ประกอบระบบการมองเห็นของเครื่องจักร (Components of a Machine Vision System) นั้นจะประกอบด้วย

1. เซ็นเซอร์การมองเห็น (Optical Sensor) โดยจะทำหน้าที่ตรวจสอบว่า มีวัตถุอยู่ใน ตำแหน่งหรือไม่ หากมีก็จะกระตุ้นกล้องให้ถ่ายภาพของวัตถุนั้น

2. กล้องถ่ายภาพ (Camera) เป็นกล้องถ่ายภาพแบบดิจิทัล ในการใช้งานนั้น เมื่อจับภาพได้แล้วจะต้องต่อเข้ากับการ์ดเชื่อมต่อกับกล้อง (Frame Grabber) และให้โปรแกรมคอมพิวเตอร์เป็นตัวประมวลผล และทำการวิเคราะห์ข้อมูลภาพต่อไป

3. หลอดไฟ (Lamp) ทำหน้าที่ส่องสว่างไปยังวัตถุที่สนใจ เพื่อการตรวจจับภาพที่มีคุณภาพเหมาะสม ซึ่งจะต้องมีการจัดวางให้มีตำแหน่งที่เหมาะสมและมั่นคง โดยมีการติดตั้งหลอดไฟ 5 ลักษณะ ได้แก่ การติดตั้งหลอดไฟด้านหน้า การติดตั้งหลอดไฟด้านหลัง การติดตั้งหลอดไฟด้านข้าง การติดตั้งหลอดไฟแบบโครงสร้าง และการติดตั้งหลอดไฟแบบสโตรบ (Strobe)

4. การ์ดเชื่อมต่อกับกล้อง (Frame Grabber Card) เป็นอุปกรณ์คอมพิวเตอร์ที่เชื่อมต่อระหว่างกล้องกับคอมพิวเตอร์ โดยภาพจากกล้องที่เป็นสัญญาณอนาล็อกหรือสัญญาณดิจิทัลจะถูกจับโดย Frame Grabber Card ให้เป็นภาพนิ่ง แล้วให้เอาต์พุตที่เป็นรูปแบบดิจิทัล

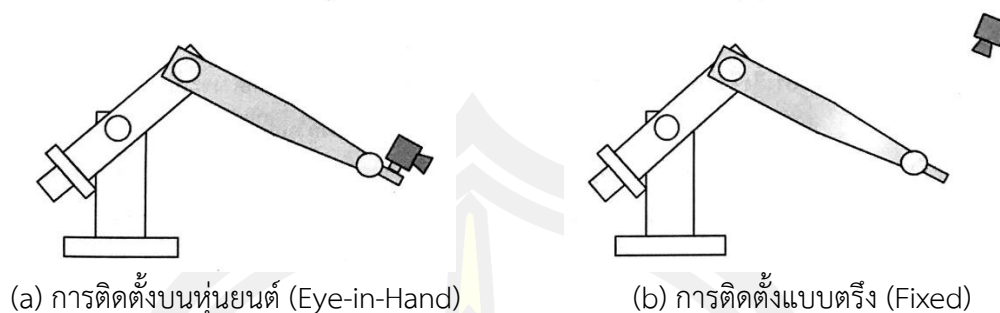
5. ซอฟต์แวร์ในการประมวลผลภาพ ซึ่งจะเริ่มต้นแบบที่มีขั้นตอนการประมวลผลหลายขั้นตอน ภาพจะถูกลดทอนสัญญาณรบกวน หรือเปลี่ยนเงาจำนวนมากของเฉดสีเทาไปเป็นภาพขาวดำ ตามด้วยซอฟต์แวร์จะทำการนับ วัด หรือทำเอกลักษณ์ของวัตถุในภาพ ในขั้นตอนสุดท้าย

6. ฮาร์ดแวร์ปรุงแต่งสัญญาณดิจิทัลหรือเครือข่ายเชื่อมต่อกับการรายงานผล

## 2.6 แนวทางการควบคุมหุ่นยนต์โดยภาพ

ภายหลังจากการดึงข้อมูลของลักษณะเด่นของวัตถุเป้าหมายออกจากภาพ ในส่วนนี้จะกล่าวถึงการควบคุมหุ่นยนต์ให้ทำตามคำสั่งโดยอ้างอิงกับข้อมูลลักษณะเด่น กรณีที่พบบ่อย คือ การเคลื่อนตัวทำงานส่วนปลายไปยังตำแหน่งที่วัตถุตั้งอยู่ เช่น การถอดอุปกรณ์อิเล็กทรอนิกส์ออกจากแผงวงจรพิมพ์ หรือการหยิบจับวัตถุด้วยคีมจับ ปัญหาการควบคุมแบบนี้เรียกโดยรวมว่า การควบคุมโดยพื้นฐานภาพ (Vision-base Control) สิ่งที่แตกต่างกันจากการควบคุมตำแหน่งและแรง คือ ค่าที่ต้องการควบคุมไม่ได้มาจากการวัดโดยตรงจากตัวรับรู้ แต่จะได้จากการฉายวัตถุสามมิติลงบนระนาบของภาพ และผ่านขั้นตอนการประมวลผลอื่นจนกระทั่งได้ลักษณะเด่นที่เป็นตัวแทนของวัตถุ

สำหรับรูปแบบการติดตั้งกล้องเพื่อการควบคุมโดยภาพ สิ่งแรกที่ต้องพิจารณา คือ ตำแหน่งของการติดตั้งกล้อง โดยปกติมี 2 รูปแบบ ได้แก่ การติดตั้งบนตัวหุ่นยนต์ใกล้ตัวทำงานส่วนปลายเรียกว่า แบบตาในมือ (Eye-in-Hand) และการตรึงกล้องไว้ที่ตำแหน่งคงที่ในปริภูมิการทำงานเรียกว่า กล้องตรึง (Fixed Camera) ดังแสดงในภาพที่ 2.13



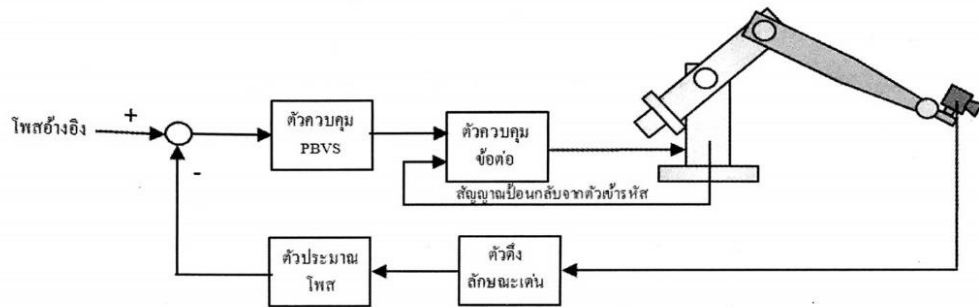
ภาพที่ 2.13 รูปแบบการติดตั้งกล้อง

ที่มา: [18]

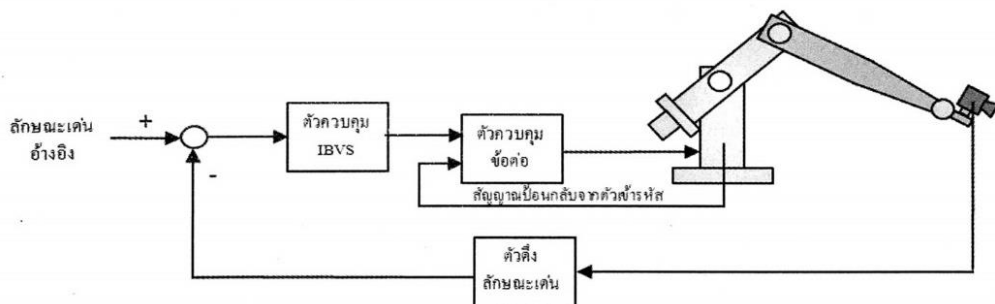
ระบบที่ติดตั้งกล้องที่ตำแหน่งคงที่ในปริภูมิการทำงาน จะต้องหาตำแหน่งที่เหมาะสมในการตรวจสอบทั้งหุ่นยนต์และวัตถุที่ทำงานด้วย ข้อได้เปรียบของวิธีการนี้ คือ เนื่องจากมีตำแหน่งกล้องคงที่ ระยะเวลาการมองเห็น (Field of View) จะไม่มีการเปลี่ยนแปลงเมื่อหุ่นยนต์เคลื่อนที่ รวมถึงความสัมพันธ์ทางเรขาคณิตที่คงที่ระหว่างกล้องกับปริภูมิการทำงาน แต่มีข้อเสีย คือ โอกาสที่บางส่วนของหุ่นยนต์อาจจะเคลื่อนที่ไปบังในขณะการทำงาน เช่น การสอดหมดลงไปในรูเจาะ ซึ่งในกรณีดังกล่าวนี้เป็นไปได้ยากที่จะหาตำแหน่งของการติดตั้งกล้องที่มองเห็นได้ตลอดเวลาการทำงาน โดยไม่ถูกบังจากตัวทำงานส่วนปลาย

ส่วนระบบการติดตั้งกล้องบนหุ่นยนต์ เมื่อทำการติดตั้งใกล้กับตัวทำงานส่วนปลายจะมีข้อดีคือ แก้ไขปัญหาเรื่องหุ่นยนต์บังการทำงานของวัตถุ แต่จะพบความยุ่งยากที่มากกว่าระบบกล้องแบบตรึง คือ ระยะเวลาการมองเห็นภาพ ตลอดจนความสัมพันธ์ทางเรขาคณิตระหว่างกล้องกับปริภูมิการทำงานของหุ่นยนต์ที่มีการเปลี่ยนแปลงไปตามการเคลื่อนที่ ซึ่งปัจจัยนี้จะขึ้นอยู่กับโครงสร้างและตำแหน่งที่ติดตั้งกล้อง ในบางกรณี การเคลื่อนเพียงเล็กน้อย เช่น การหมุนกล้อง สามารถทำให้ระยะเวลาการมองเห็นภาพเปลี่ยนแปลงไปอย่างมาก แต่อย่างไรก็ตาม ในการติดตั้งกล้องทั้ง 2 รูปแบบนี้ มีสิ่งที่เหมือนกันก็คือ การเคลื่อนที่ของหุ่นยนต์จะทำให้เกิดการเปลี่ยนแปลงของภาพจากกล้อง การวิเคราะห์ความสัมพันธ์ระหว่างการเคลื่อนที่ของหุ่นยนต์กับการเปลี่ยนแปลงของภาพทั้ง 2 รูปแบบ กรณีจะคล้ายกัน

สำหรับปัญหาการควบคุมโดยพื้นฐานภาพแบ่งออกเป็น 2 แนวทางหลัก ตามลักษณะการใช้ข้อมูลที่ได้จากกล้องเพื่อป้อนกลับให้ตัวควบคุม โดยสามารถรวมทั้งสองแนวทางเข้าด้วยกันเป็นแผนการควบคุมแบบแบ่งส่วน โดยภาพที่ 2.14 จะแสดงแผนการควบคุมสองแนวทาง เรียกว่า แบบ Position-Based Visual Servo (PBVS) และแบบ Image-Based Visual Servo (IBVS)



(a) การควบคุมแบบ Position-Based Visual Servo (PBVS)



(b) การควบคุมแบบ แบบ Image-Based Visual Servo (IBVS)

ภาพที่ 2.14 แนวทางการควบคุมโดยภาพ

ที่มา: [18]

การควบคุมแบบ PBVS นั้นพบว่า ข้อมูลลักษณะเด่นของภาพจะถูกนำมาสร้างตัวแทนสามมิติในเฟรมโลกเรียกว่า โพส (Pose) ซึ่งโพสที่สร้างนี้เป็นเพียงการประมาณค่าส่วนที่สำคัญของโพสที่ติดตั้งกับวัตถุจริงบนโลกและเปรียบเทียบกับโพสอ้างอิงซึ่งเป็นเป้าหมายการเคลื่อนที่หุ่นยนต์ ค่าแตกต่างจะเป็นอินพุตของตัวควบคุม PBVS ที่เหมือนวงนอกในการสร้างแนววิถีให้กับตัวควบคุม ข้อต่อทำหน้าที่ขับเคลื่อนหุ่นยนต์เข้าสู่ตำแหน่งและทิศทางของโพสอ้างอิง ถึงแม้ว่าแนวคิดนี้ง่ายต่อการทำความเข้าใจ แต่ในทางปฏิบัติจะประสบปัญหาเนื่องจากความยากในการประมาณตัวแทนโพสใน 3 มิติแบบทันที นอกจากนั้นวิธีการ PBVS จะไม่มีการควบคุมภาพโดยตรง ดังนั้นปัญหาที่มักเกิดขึ้นเสมอก็คือ การเคลื่อนที่ของกล้อง ซึ่งทำให้วัตถุเป้าหมายหลุดออกนอกกรอบการมองเห็นของกล้อง

แนวทางต่อมา คือ การควบคุมแบบ IBVS จะแตกต่างจากวิธี PBVS คือ ใช้ข้อมูลลักษณะเด่นของภาพในการควบคุมหุ่นยนต์โดยตรง โดยค่าความแตกต่างเป็นฟังก์ชันของจำนวนที่สามารถวัดได้ในภาพ และกฎการควบคุมจะคำนวณการเคลื่อนที่ของหุ่นยนต์จากค่าที่แตกต่างนี้ วิธีการที่นิยมในปัจจุบัน คือ กำหนดจุดที่สังเกตเห็นได้ง่ายบนภาพวัตถุเป็นจุดลักษณะเด่น และฟังก์ชันค่าแตกต่างจะเป็นเวกเตอร์ระหว่างตำแหน่งที่ต้องการกับตำแหน่งที่จับภาพได้ ณ เวลานั้น และมักใช้กฎการควบคุมที่ค่อนข้างง่ายเพื่อสามารถคำนวณได้แบบทันที

## 2.7 การประยุกต์ใช้หุ่นยนต์ในภาคการเกษตร

การประยุกต์ใช้หุ่นยนต์ในภาคการเกษตร (Agricultural Robotics) [27] เป็นรูปแบบของการนำเอาสหวิทยาการ รวมถึงเทคโนโลยีที่หลากหลายมาทำงานร่วมกัน ได้แก่ เซ็นเซอร์อัจฉริยะ (Smart Sensor) ปัญญาประดิษฐ์ (Artificial Intelligence) ระบบการควบคุมแบบอัตโนมัติ (Automatic and Control) อินเทอร์เน็ตของสรรพสิ่ง (Internet of Thing) ข้อมูลขนาดใหญ่ (Big Data) อากาศยานไร้คนขับ (Unmanned Aerial Vehicles) การเรียนรู้เชิงลึก (Deep Learning) เครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Networks) และ การจัดการความแม่นยำ (Precision Management) ซึ่งพบว่ามีบทบาทหลักอยู่ 2 ประเด็น ประกอบด้วย

1. สภาพแวดล้อมที่ไม่มีโครงสร้างหรือรูปแบบแน่ชัด (Unconstructed Environment)
2. งานที่มีความคลุมเครือไม่แน่นอน (Uncertain Task)

เนื่องจากการเกษตรกรรมเป็นงานที่มีลักษณะรูปแบบที่มีการเปลี่ยนแปลง (Dynamic Task) ดังนั้น เมื่อมีการกำหนดพันธุ์พืช ตลอดจนพื้นที่ในการเพาะปลูก จะส่งผลกระทบต่อให้เกิดการเปลี่ยนแปลงเงื่อนไขที่แตกต่างกันออกไป ตามความแตกต่างกันของพืช รูปร่าง ลำต้น กิ่ง ใบ สีของผล ผิวสัมผัส รวมถึงอิทธิพลของสภาพอากาศ เพื่อให้สามารถเพาะปลูกได้อย่างมีประสิทธิภาพในเงื่อนไขจริง ซึ่งการนำหุ่นยนต์มาใช้ในภาคการเกษตรนั้น ก็เพื่อคาดหวังให้ผลผลิตที่ได้มีปริมาณสูงขึ้น มีคุณภาพที่ดี ลดต้นทุนและค่าใช้จ่ายในการลงทุน รวมทั้งลดการใช้แรงงานมนุษย์ในภาคการเกษตรลง

ปัจจุบัน อุตสาหกรรมในภาคการเกษตรของโลกต่างมีความต้องการหุ่นยนต์ระบบอัตโนมัติ (Full Automation Robotics) ที่สามารถทำงานได้ทั้งระบบ กล่าวคือ การทำงานเริ่มตั้งแต่ขบวนการเพาะปลูก การดูแลรดน้ำ การให้ปุ๋ย การเก็บข้อมูลพืช การกำจัดแมลงและวัชพืช และการเก็บเกี่ยวผลผลิต แต่จากงานวิจัยและพัฒนาที่ผ่านมา พบว่าการพัฒนาหุ่นยนต์ด้านการเกษตรยังอยู่ในช่วงของการพัฒนาต้นแบบ ซึ่งถูกพัฒนาและออกแบบให้ทำงานต่าง ๆ เฉพาะด้าน ภายในฟาร์ม ซึ่งแบ่งออกได้เป็น 9 ประเภทหลัก ดังนี้

- 1) หุ่นยนต์ตรวจการณ์และเก็บข้อมูลภาคสนาม (Field Scouting and Data Collection Robot) [4] ถูกพัฒนาขึ้นใช้ในการลงพื้นที่การเพาะปลูก แปลงปลูก หรือภายในบริเวณโรงเรือน เพื่อทำการสำรวจสภาพของพื้นที่ ตรวจวัดค่าพารามิเตอร์ต่าง ๆ ในสภาพแวดล้อม เช่น อุณหภูมิ ความชื้น ความเร็วลม หรือปริมาณแสง จากนั้นทำการบันทึกข้อมูลและส่งข้อมูลกลับมาประมวลในระบบ หุ่นยนต์ประเภทนี้มักมีแพลตฟอร์มในลักษณะของ Mobile Robot ที่สามารถเคลื่อนที่ได้อย่างคล่องแคล่วบนทุกสภาพพื้นผิว ซึ่งถูกสร้างให้มีกล้องและเซ็นเซอร์ตรวจวัดประเภทต่าง ๆ ติดตั้งอยู่บนโครงสร้าง ดังภาพที่ 2.15 นอกจากนี้ยังรวมถึงโดรนที่ติดเซ็นเซอร์เพื่อบินขึ้นไป

เก็บข้อมูลและถ่ายภาพจากมุมสูงของพื้นที่ เพื่อส่งข้อมูลไปตรวจวิเคราะห์สภาพแวดล้อม สภาพอากาศ ผลผลิตและปัจจัยต่าง ๆ



ภาพที่ 2.15 หุ่นยนต์ตรวจการณ์ภาคสนาม

ที่มา: [4]

2) หุ่นยนต์กำจัดแมลงและศัตรูพืช (Pest and Weed Control Robot) [5] เป็นหุ่นยนต์ที่เคลื่อนที่เข้าไปในพื้นที่เพาะปลูกเพื่อตรวจจับแมลงหรือวัชพืช โดยอาศัยการทำงานของกล้องหรือเซ็นเซอร์ที่ถูกติดตั้งบนโครงสร้าง ซึ่งใช้การประมวลผลภาพเข้ามาช่วยทำการวิเคราะห์ว่าสิ่งที่ตรวจจับได้ดังกล่าวนั้น เป็นแมลงหรือวัชพืชหรือไม่ ก่อนที่จะนำผลลัพธ์ที่ได้จากการวิเคราะห์นั้นส่งต่อไปยังส่วนควบคุมอื่น ๆ เพื่อทำงานตามหน้าที่ต่อไป เช่น กรณีตรวจพบว่ามีแมลงศัตรูพืชอยู่ในแปลงปลูก หุ่นยนต์ก็จะทำการฉีดพ่นยาปราบศัตรูพืชในบริเวณนั้นอย่างแม่นยำ ตรงจุด หรือกรณีพบว่าเป็นวัชพืชขึ้นปะปนกับพืชที่ปลูก หุ่นยนต์ก็จะทำการทำลายวัชพืชเหล่านั้นในทันที ก่อนที่จะเคลื่อนที่ผ่านไปยังบริเวณที่อยู่ถัดไป ตัวอย่างของหุ่นยนต์กำจัดแมลงและศัตรูพืชแสดงดังภาพที่ 2.16



ภาพที่ 2.16 หุ่นยนต์กำจัดแมลง

ที่มา: [5]



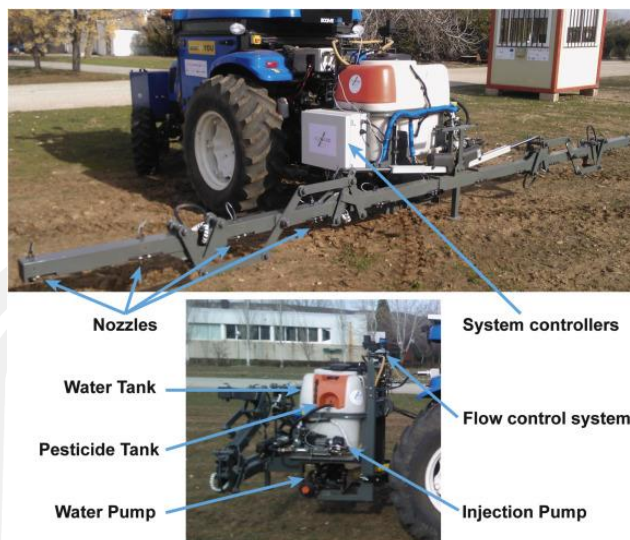
3) หุ่นยนต์เก็บเกี่ยวผลผลิต (Harvesting Robot) [6] ทำหน้าที่ในการเก็บเกี่ยวผลผลิตภายในพื้นที่เพาะปลูก มีรูปแบบการทำงานที่ประสานกันของหลายส่วน เนื่องจากหุ่นยนต์ประเภทนี้จะต้องเคลื่อนที่ภายในแปลง จากนั้นทำการตรวจจับ แยกแยะ และวิเคราะห์ผลผลิตที่ได้ตรวจพบก่อนทำการประมวลผลว่าผลผลิตดังกล่าว อยู่ในระยะเวลาที่พร้อมเก็บเกี่ยวหรือไม่ เพื่อตัดสินใจให้อุปกรณ์ที่เป็นชุดเก็บเกี่ยวเคลื่อนที่ไปทำการเก็บเกี่ยวผลผลิตนั้นต่อไป เนื่องจากพืชแต่ละประเภทมีความแตกต่างกันทางพันธุกรรม จึงส่งผลให้หุ่นยนต์เก็บเกี่ยวผลผลิตจะถูกออกแบบมาเฉพาะพืชชนิดนั้น ดังภาพที่ 2.17 เป็นการทดสอบแขนกลหุ่นยนต์เก็บเกี่ยวผลแอปเปิล โดยทำงานร่วมกับคอมพิวเตอร์และกล้องถ่ายภาพสีในการจำแนกตำแหน่งและผลผลิตที่เหมาะสมในการเก็บ



ภาพที่ 2.17 หุ่นยนต์เก็บเกี่ยวผลผลิต

ที่มา: [6]

4) หุ่นยนต์ฉีดพ่นสารเคมี (Target Spraying Robot) [7] เป็นแพลตฟอร์มหุ่นยนต์รูปแบบ Mobile Robot ซึ่งเคลื่อนแบบโดยอัตโนมัติในพื้นที่เพาะปลูก ทำหน้าที่ในการรดน้ำให้ปุ๋ย ตลอดจนฉีดพ่นสารเคมีหรือยาปราบศัตรูพืชแทนแรงงานมนุษย์ เพื่อป้องกันอันตรายในการสัมผัสกับสารเคมี และลดการใช้แรงงานมนุษย์ลง ดังภาพที่ 2.18 เป็นหุ่นยนต์ที่ใช้ฉีดพ่นยาปราบศัตรูพืชในแปลงขนาดใหญ่ ที่ถูกติดตั้งบนระบบขับเคลื่อนแบบใช้ล้อยางรถแทรกเตอร์ ทำให้หุ่นยนต์สามารถเคลื่อนที่ได้อย่างมีประสิทธิภาพบนพื้นผิวดินของแปลงปลูก



ภาพที่ 2.18 หุ่นยนต์ฉีดพ่นสารเคมี

ที่มา: [7]

5) หุ่นยนต์ตัดแต่งกิ่ง (Pruning Robot) [8] ถูกใช้ประโยชน์ในการตัดแต่งกิ่งพืช ซึ่งหุ่นยนต์ประเภทนี้มีการออกแบบในหลากหลายลักษณะขึ้นอยู่กับพื้นที่ที่ต้องการให้หุ่นยนต์เข้าไปทำงาน เช่น การเคลื่อนที่ในแนวราบเพื่อตัดกิ่งต้นไม้ที่ยื่นออกมาเลยแนวอ่าอิง หรือเคลื่อนที่ขึ้นลงในแนวตั้งไปตามลำต้นของพืช เพื่อไปตัดกิ่งในระดับความสูงที่กำหนดเอาไว้ จากตัวอย่างเป็นหุ่นตัดแต่งกิ่งที่ถูกติดเข้ากับลำต้นของพืช ใช้การไอบรตของโครงสร้างเพื่อยึดตัวหุ่นกับลำต้น จากนั้นเคลื่อนที่ขึ้นลงโดยอาศัยการหมุนของล้อยางที่สัมพันธ์กับการปรับขนาดในการรัดหรือคลายการไอบรตของโครงสร้างวงแหวนของหุ่นยนต์ เมื่อหุ่นยนต์ได้ขึ้นถึงจุดที่กำหนด ส่วนปลายของ End Effector ที่มีเลื่อยติดตั้งอยู่จะถูกใช้ในการตัดกิ่งต่อไป ดังแสดงในภาพที่ 2.19



ภาพที่ 2.19 หุ่นยนต์ตัดแต่งกิ่ง

ที่มา: [8]

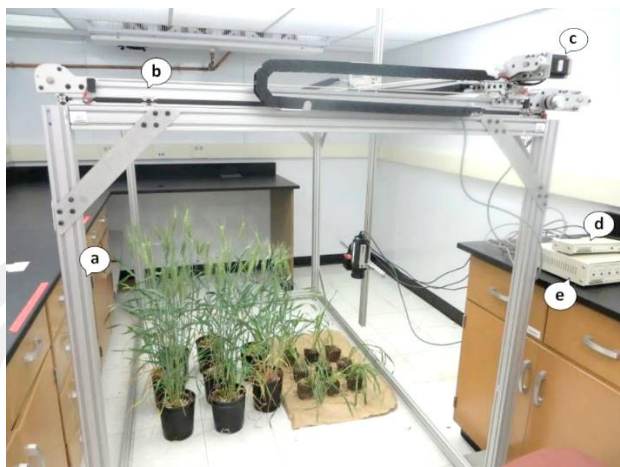
6) หุ่นยนต์รีดนม (Milking Robot) [9] เป็นหุ่นยนต์ที่ถูกพัฒนาขึ้นเพื่อใช้ในอุตสาหกรรมการผลิตนม ซึ่งหุ่นยนต์ประเภทนี้ปัจจุบันมีการใช้งานอย่างกว้างขวางในหลายประเทศ จากภาพที่ 2.20 แสดงการทำงานของหุ่นยนต์รีดนมภายในโรงเรือน โดยลักษณะของการทำงานนั้นจะมีการเคลื่อนที่ไปตามราง ซึ่งอยู่ด้านหลังของคอก เมื่อหุ่นยนต์เคลื่อนที่มาถึงจุดที่โคนมยืนอยู่ โครงสร้างของหุ่นยนต์จะเลื่อนออกชูดรีดนมออกไปด้านหน้าเพื่อนำอุปกรณ์ชูดรีดนมไปสวมใส่ยังตัวของโคนม จากนั้นจึงทำการรีดน้ำนมโคเพื่อไปเก็บไว้ในถังเก็บน้ำนม เมื่อทำการรีดนมโคเสร็จแล้ว ชุดอุปกรณ์รีดนมก็จะเคลื่อนที่ถอยหลังกลับไปยังโครงสร้างตามเดิม ก่อนเคลื่อนที่ไปตามรางเพื่อไปรีดนมโคตัวที่อยู่ถัดไป



ภาพที่ 2.20 หุ่นยนต์รีดนมวัว

ที่มา: [28]

7) หุ่นยนต์ตรวจจำแนกลักษณะฟิซ (Phenotyping Robot) [10] เป็นหุ่นยนต์ที่ถูกสร้างขึ้นเพื่อใช้ในการจำแนกลักษณะทางกายภาพของพันธุ์กรรมฟิซ โดยทำการวิเคราะห์ แยกแยะหรือจัดประเภทฟิซที่ถูกนำมาตรวจสอบ เปรียบเทียบกับฐานข้อมูลที่มีอยู่ในระบบ ในภาพที่ 2.21 เป็นหุ่นยนต์ตรวจจำแนกลักษณะฟิซซึ่งใช้ข้อมูลภาพถ่ายจากกล้องที่ติดอยู่บนแขน End Effector ของโครงสร้าง โดยสามารถเคลื่อนที่ได้ตามแนวแกน X, Y และ Z เพื่อนำมาวิเคราะห์โดยใช้เทคนิคของการประมวลผลภาพดิจิทัล จากนั้นจึงนำผลลัพธ์ดังกล่าวมาทำการจำแนกลักษณะฟิซเพื่อส่งต่อไปยังส่วนแสดงผลและส่วนคัดแยกฟิซต่อไป



ภาพที่ 2.21 หุ่นยนต์ตรวจจำแนกลักษณะพืช

ที่มา: [10]

8) หุ่นยนต์จัดเรียงพืช (Sorting Robot) [11] ทำหน้าที่ในการตรวจสอบและจัดเรียงพืชตามเงื่อนไขที่ถูกกำหนดไว้ โดยอาศัยการวิเคราะห์ข้อมูลพืชผ่านทางกล้องหรือเซ็นเซอร์ต่าง ๆ ด้วยการทำงานร่วมกันของการประมวลผลภาพและเทคโนโลยีปัญญาประดิษฐ์ ซึ่งหุ่นยนต์ในภาพที่ 2.22 นั้น เป็นตัวอย่างของการประยุกต์ใช้หุ่นยนต์ประเภทนี้ เพื่อทำการตรวจสอบและจัดเรียงกิ่งพันธุ์พืชที่มีลักษณะทางกายภาพ เช่น ความยาว หรือเส้นผ่านศูนย์กลางของกิ่ง ซึ่งตรงตามเงื่อนไขที่กำหนดไว้ในระบบ เพื่อส่งต่อไปทำการตัด บาก และเสียบทาบกิ่ง



ภาพที่ 2.22 หุ่นยนต์จัดเรียงต้นตอพืชเพื่อทำการทาบกิ่ง

ที่มา: [11]

9) หุ่นยนต์เพาะปลูกพืช (Seeding and Planting Robot) [12] เป็นหุ่นยนต์ที่ทำหน้าที่ปลูกพืชแทนมนุษย์ เพื่อลดเวลาและแรงงานในการหยอดเมล็ดพันธุ์พืชบนพื้นที่เพาะปลูก นอกจากหน้าที่ในการหยอดเมล็ดแล้ว หุ่นยนต์เพาะปลูกพืชบางตัวถูกออกแบบให้สามารถทำการขุดรูดหยอดเมล็ด กลบดิน รวมทั้งรดน้ำได้ในตัวเดียวกัน จากภาพที่ 2.23 เป็นหุ่นยนต์หยอดเมล็ดพันธุ์ข้าว ซึ่งสามารถปลูกข้าวได้เป็นแถวเรียงกันอย่างเป็นระเบียบ มีระยะห่างระหว่างหลุมที่สม่ำเสมอ และกำหนดจำนวนเมล็ดพันธุ์ที่หยอดในแต่ละหลุมได้



ภาพที่ 2.23 หุ่นยนต์เพาะปลูกพืชแบบหยอดเมล็ดพันธุ์

ที่มา: [12]

## 2.8 การประมวลผลภาพ

การประมวลผลภาพ (Image Processing) [29] คือ การแปลงข้อมูลรูปภาพให้อยู่ในรูปแบบข้อมูลดิจิทัล (Digital Format) หรือข้อมูลเชิงตัวเลข โดยผ่านขั้นตอนต่าง ๆ ด้วยคอมพิวเตอร์ เช่น การทำให้ภาพมีความคมชัด การกำจัดสัญญาณรบกวนจากภาพ การแบ่งส่วนของวัตถุที่สนใจออกจากภาพ เพื่อวัตถุประสงค์ในการปรับปรุงคุณภาพของภาพดังกล่าวให้มีรายละเอียดที่ชัดเจนมากยิ่งขึ้น และเพื่อให้สามารถแปลความหมายของภาพได้ (Computer Interpretation) ซึ่งจะเห็นได้ว่าขั้นตอนการประมวลผลภาพดิจิทัลมีหลายขั้นตอน แต่ในการทำงานจริงพบว่าในงานแต่ละประเภทไม่จำเป็นต้องทำทุกขั้นตอนขึ้นอยู่กับวัตถุประสงค์ของงานนั้น ๆ โดยขั้นตอนพื้นฐานของการประมวลผลภาพด้วยคอมพิวเตอร์ ประกอบด้วย

1. การจัดสภาพแวดล้อม (Scene Constraint) เพื่อลดความซับซ้อนในการประมวลผลภาพให้มากที่สุด เนื่องจากความสามารถในการมองเห็นและรับรู้ของอุปกรณ์ประมวลผลมีจำกัดและไม่เทียบเท่ามนุษย์ จึงต้องช่วยลดความยุ่งยากของการประมวลผล

2. การดึงข้อมูลภาพ (Image Acquisition) เป็นกระบวนการที่เริ่มตั้งแต่การถ่ายภาพด้วยกล้อง ตลอดจนการดึงข้อมูลภาพเข้าสู่คอมพิวเตอร์ หรืออุปกรณ์ประมวลผล

3. การประมวลผลภาพเบื้องต้น (Pre-processing) มีด้วยกันหลากหลายกระบวนการ เช่น การแปลงสี การวิเคราะห์ภาพในเชิงความถี่ การลดสัญญาณรบกวนที่ปรากฏในภาพ การตรวจจับขอบของวัตถุที่อยู่ในภาพ การบีบอัดภาพ และการแปลงคุณสมบัติทางกายภาพ ด้วยการหมุน การเลื่อน การย่อ หรือการขยาย เป็นต้น

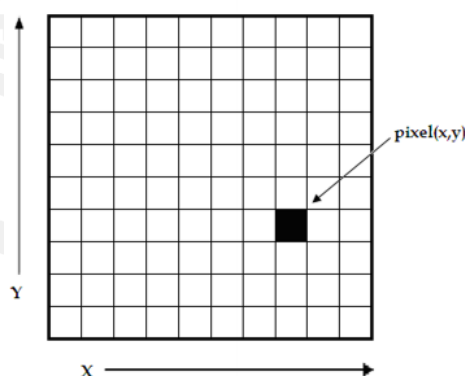
4. การแยกบริเวณ (Segmentation) เป็นกระบวนการแยกบริเวณภาพที่มีลักษณะร่วมกัน ออกเป็นส่วน ๆ ซึ่งมุ่งเน้นในการแยกวัตถุออกจากพื้นหลัง โดยกระบวนการแยกมี 2 วิธี คือ การแยกบริเวณโดยใช้ค่าเทรชโฮลด์ (Threshold) และการแยกบริเวณโดยใช้ขอบของวัตถุ (Edge-based Segmentation)

5. การคำนวณหาคุณลักษณะของวัตถุ (Feature Extraction) เป็นการคำนวณหาหรือวัดคุณลักษณะต่าง ๆ ของแต่ละบริเวณหรือของวัตถุแต่ละชิ้นที่อยู่ในภาพ ซึ่งจะถูกนำมาเขียนเป็นเวกเตอร์คุณลักษณะ (Feature Vector)

6. การจำแนกวัตถุและแปลความหมาย (Classification and Interpretation) เป็นการจัดกลุ่มให้กับวัตถุที่กำลังพิจารณาอยู่นั้นว่าเป็นวัตถุที่อยู่ในกลุ่มใด โดยอาศัยข้อมูลที่ได้จากการวัดหรือการคำนวณซึ่งเป็นคุณสมบัติของวัตถุนั้น ๆ เปรียบเทียบกับวัตถุตัวอย่างที่อยู่ในแต่ละกลุ่ม โดยก่อนที่ระบบจะสามารถตัดสินใจได้จะต้องมีตัวอย่างของวัตถุในแต่ละกลุ่มเสียก่อน

## 2.9 ภาพดิจิทัล

ภาพดิจิทัล (Digital Image) [29] นั้นเป็นภาพที่แสดงค่าอยู่ในรูปของข้อมูลอาร์เรย์ (Array) หลายมิติ หรือเป็นเมทริกซ์ (Matrix) เชิงตัวเลข ซึ่งภาพดิจิทัลจะแสดงในลักษณะสองมิติ คือจะมีขนาดความกว้างและความสูงของภาพบนแนวแกน X และแกน Y ส่วนจุดที่อยู่บนระนาบนี้ จะเรียกว่า พิกเซล (Pixel) โดยแต่ละพิกเซลจะแสดงค่าความเข้มแสงของภาพเป็นฟังก์ชัน  $f(x,y)$  ดังภาพที่ 2.24



ภาพที่ 2.24 จุดพิกัดบนภาพดิจิทัล

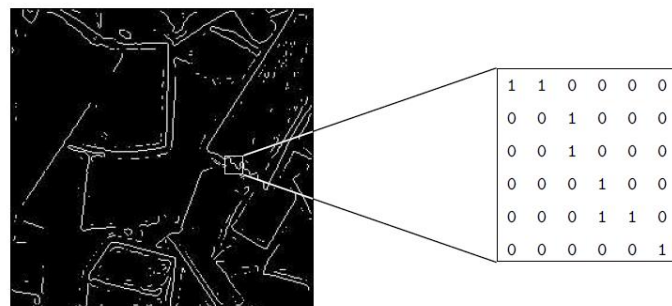
จากที่ได้กล่าวมาแล้วมา สามารถแสดงภาพดิจิทัลซึ่งมีขนาด  $M \times N$  ให้อยู่ในรูปเมทริกซ์ได้ดังภาพที่ 2.25

$$f(x,y) \equiv \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

ภาพที่ 2.25 เมทริกซ์ของภาพดิจิทัล ขนาด  $M \times N$

ประเภทของภาพดิจิทัล (Types of Digital Images) มีการการจำแนกเป็น 4 ประเภท ประกอบด้วย

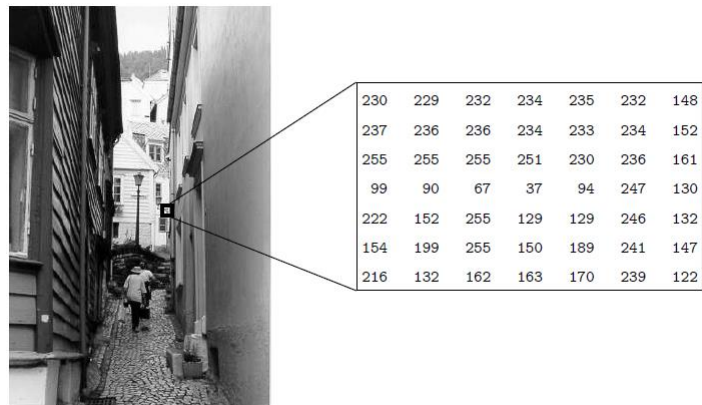
1. ภาพขาวดำ (Black and White Image: B/W) หรือภาพไบนารี (Binary) ซึ่งแต่ละพิกเซลจะมีค่า 1 บิต สามารถแสดงสีได้เพียง 2 สี คือขาวและดำเท่านั้น โดย สีดำแทนด้วย 0 และสีขาวแทนด้วย 1 ดังภาพที่ 2.26



ภาพที่ 2.26 ภาพไบนารี (Binary)

ที่มา : [29]

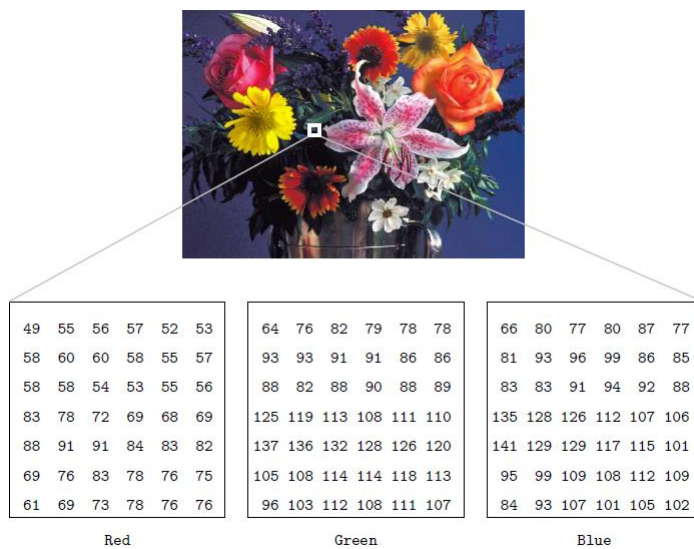
2. ภาพระดับเทา (Grayscale Image) เป็นภาพที่มีระดับความเข้มแต่ละพิกเซลสอดคล้องกับความเข้มแสงปกติในระดับสีเทา ซึ่งแต่ละพิกเซลมีขนาด 8 บิต หรือ 1 ไบต์ สามารถแสดงเฉดสีจาก 0 (สีดำ) ถึง 255 (สีขาว) ดังภาพที่ 2.27



ภาพที่ 2.27 ภาพระดับเทา (Grayscale Image)

ที่มา : [29]

3. ภาพสี (Color Image) หรือภาพสี RGB (RGB Image) เป็นภาพที่สีแต่ละพิกเซลมีขนาด 24 บิต ซึ่งแสดงส่วนประกอบของเวกเตอร์แทนสีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) โดยแต่ละสีจะแสดงระดับของความเข้มแสงในช่วง 0 – 255 ทำให้สามารถแสดงสีได้ทั้งหมด  $256^3$  สี หรือเท่ากับ 16,777,216 สี ดังภาพที่ 2.28



ภาพที่ 2.28 ภาพสี (Color Image)

ที่มา : [29]

4. ภาพแบบดัชนี (Indexed Image) ในแต่ละพิกเซลของภาพจะเก็บค่าดัชนีเป็นตัวเลขจำนวนเต็ม ซึ่งจะถูกนำค่าดัชนีดังกล่าวไปเทียบกับตารางสีที่เป็นตารางแสดงค่าของแสงสีแดง สีเขียว



และสีน้ำเงิน โดยค่าดัชนีนี้จะเป็นตัวบ่งชี้ให้เห็นว่าภาพในแต่ละตำแหน่งพิกเซลใด ๆ มีค่าอัตราส่วนของ สีทั้งสามในอัตราส่วนเท่าใด ดังภาพที่ 2.29



ภาพที่ 2.29 ภาพแบบดัชนี (Indexed Image)

ที่มา : [29]

### 2.10 การหาขอบภาพ

การหาขอบภาพ (Edge Detection Methods) [30] คือ การตรวจสอบว่าเส้นขอบลากผ่านหรือใกล้เคียงกับจุดใด โดยวัดจากการเปลี่ยนแปลงของความเข้มในตำแหน่งที่ใกล้เคียงกับจุดดังกล่าว ซึ่งวิธีการหาขอบภาพนั้น แบ่งออกได้เป็น 2 กลุ่ม ประกอบด้วย Gradient Method และ Laplacian Method

1. Gradient Method เป็นการหาขอบภาพโดยการหาจุดต่ำสุดและจุดสูงสุดในรูปของอนุพันธ์อันดับหนึ่ง (Gradient Operator:  $\nabla$ ) ของภาพ มีหลักการ คือ บริเวณขอบภาพของวัตถุในภาพจะมีเกรเดียนต์ที่สูง การพิจารณาขนาดของเกรเดียนต์ (Gradient Magnitude,  $|\nabla P|$ ) เปรียบเทียบกับค่าอ้างอิง (Threshold) ที่กำหนดขึ้น เมื่อค่าของเกรเดียนต์มีมากกว่าค่าอ้างอิง แสดงว่าจุดดังกล่าวเป็นขอบของวัตถุที่ปรากฏในภาพที่จุด  $P(x,y)$  การค้นหาขอบของวัตถุโดยใช้อนุพันธ์อันดับหนึ่ง เป็นวิธีแยกส่วนประกอบของภาพและเมื่อความไม่ต่อเนื่องของค่าพิกเซลบริเวณรอยต่อระหว่างวัตถุกับพื้นหลังและค่าอนุพันธ์ย่อยที่ไม่ต่อเนื่องตามทิศทางของเกรเดียนต์ของแนวแกน X และแกน Y กำหนดได้ตามสมการที่ 2.1 และ 2.2

$$\nabla_x P(x,y) = P(x,y) - P(x-1,y) \tag{2.1}$$

และ

$$\nabla_y P(x,y) = P(x,y) - P(x,y-1) \tag{2.2}$$

เมื่อ  $P(x,y)$  คือ ฟังก์ชันของภาพ

โดยที่ขนาดของเกรเดียนต์  $P(x,y)$  ได้จากสมการที่ 2.3

$$|\nabla P(x,y)| = \sqrt{(\nabla_x P(x,y))^2 + (\nabla_y P(x,y))^2} \quad 2.3$$

เพื่อให้ง่ายต่อการคำนวณ จึงทำการประมาณค่าขนาดของเกรเดียนต์จากสมการ 2.4

$$|\nabla P(x,y)| = |\nabla_x P(x,y)| + |\nabla_y P(x,y)| \quad 2.4$$

การค้นหาขอบภาพที่มีองค์ประกอบของเส้นตรงเป็นวิธีการแบ่งออกเป็นส่วยย่อยในแต่ละพิกเซล กำหนดรูปแบบการเปลี่ยนแปลงภาพให้มีความเรียบด้วยตัวกรอง Gaussian ก่อนคำนวณหาขนาดและทิศทางของเกรเดียนต์แต่ละค่าของ Mask กำหนดค่าได้ดังสมการที่ 2.5 และ 2.6

$$\text{Mask}(E_x) = \begin{bmatrix} Z_{x1} & Z_{x2} & Z_{x3} \\ Z_{x4} & Z_{x5} & Z_{x6} \\ Z_{x7} & Z_{x8} & Z_{x9} \end{bmatrix} = E_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad 2.5$$

$$\text{Mask}(E_y) = \begin{bmatrix} Z_{y1} & Z_{y2} & Z_{y3} \\ Z_{y4} & Z_{y5} & Z_{y6} \\ Z_{y7} & Z_{y8} & Z_{y9} \end{bmatrix} = E_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad 2.6$$

ให้  $Z$  คือ ตำแหน่งของแต่ละพิกเซลทั้ง 9 จุด

กำหนดให้ขนาดของ Mask เท่ากับ  $3 \times 3$  และมีค่าเท่ากับ  $E_x$  และ  $E_y$  หาค่าอนุพันธ์อันดับ

หนึ่งของส่วน  $\frac{\partial P}{\partial x}$  และส่วนของ  $\frac{\partial P}{\partial y}$  ขนาดและทิศทางของเกรเดียนต์หาได้จากสมการ 2.7 และ 2.8

$$M(i,j) = \sqrt{E_x^2(i,j) + E_y^2(i,j)} \quad 2.7$$

$$\theta(i,j) = \tan^{-1}(E_x^2(i,j) + E_y^2(i,j)) \quad 2.8$$

เมื่อ  $M(i,j)$  คือ ขนาดของ  $\nabla$  และ  $\theta(i,j)$  คือ ทิศทางของ  $\nabla$

การกำหนดค่า Mask ของแนวแกน X และแกน Y นั้น ทำให้ขนาดและทิศทางของเกรเดียนต์เปลี่ยนแปลงและลดขนาดของขอบในแต่ละพิกเซลของขอบภาพลงได้ พิกเซลรอบข้างของทิศทางของเกรเดียนต์ใดที่ไม่ใช่ค่าสูงสุดไม่ถือว่าเป็นขอบ ดังนั้นจะมีเพียงพิกเซลหนึ่งเดียวที่เป็นขอบ ตัวอย่างวิธีการหาขอบของกลุ่มนี้ ได้แก่ Roberts, Prewitt, Sobel และ Canny เป็นต้น

2. Laplacian Method เป็นวิธีหาขอบภาพโดยใช้ออนุพันธ์อันดับสอง (Laplacian Operator:  $\nabla^2 P$ ) บริเวณที่เป็นส่วนขอบจะเด่นชัดขึ้น การประมาณค่าของการหาอนุพันธ์อันดับสองโดยใช้ Mask ของสมการที่ 2.5 และ 2.6 ทำให้ตำแหน่งบริเวณขอบของวัตถุในภาพ คือ จุดผ่านศูนย์ (Zero Crossing) ของการหาอนุพันธ์อันดับสอง ซึ่งสามารถหาได้จากสมการ 2.9

$$\nabla^2 P = \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \quad 2.9$$

การค้นหาขอบภาพด้วยอนุพันธ์อันดับสอง เมื่อทำ  $\nabla^2 P$  บริเวณขอบจะมีตำแหน่งเดียวกับค่าจุดผ่านศูนย์ของค่าจาก  $\nabla^2 P$  พิจารณาได้จากตำแหน่งที่พิกเซลเปลี่ยนแปลงจากค่าที่เป็นบวกเป็นค่าที่เป็นลบหรือจากค่าที่เป็นลบไปเป็นค่าที่เป็นบวก ซึ่งการหาขอบโดยอนุพันธ์อันดับสองนี้ ไม่สนใจทิศทางของภาพในแนวแกน X และแกน Y กำหนดจุดที่ค่า  $y$  เป็นจุดผ่านศูนย์ วิธีนี้จึงใช้เวลาในการคำนวณมากกว่าการค้นหาขอบเขตโดยใช้ออนุพันธ์อันดับหนึ่ง ตัวอย่างวิธีการหาขอบของกลุ่มนี้ ได้แก่ Laplacian of Gaussian และ Marrs-Hildreth เป็นต้น

### 2.11 การหาขอบภาพโดยวิธีโซเบล

การหาขอบภาพด้วยวิธีโซเบล (Sobel Edge Detection) นี้ เป็นการหาขอบที่ไม่เป็นเชิงเส้น โดยใช้เทมเพลตขนาด  $3 \times 3$  จำนวน 2 เทมเพลต โดยเทมเพลตแรก (S) จะใช้หาค่าความแตกต่างในแนวนอน และเทมเพลตที่สอง (T) จะใช้หาค่าความแตกต่างในแนวตั้ง สามารถเปลี่ยนแปลงค่าความไม่ต่อเนื่องได้ตามการปรับปรุงขอบให้เป็นภาพต้นฉบับ และ  $a_0, a_2, a_3, \dots, a_7$  จะแสดงถึงตำแหน่งของแต่ละพิกเซลทั้ง 8 จุด ทวนเข็มนาฬิกา ดังภาพที่ 2.30

$a_3$	$a_2$	$a_1$
$a_4$	$I(i,j)$	$a_0$
$a_5$	$a_6$	$a_7$

ภาพที่ 2.30 แสดงตำแหน่งของตัวแปรด้วยวิธี Sobel

ให้  $u = (a_5 + 2a_6 + a_7) - (a_1 + 2a_2 + a_3)$  และ  $v = (2a_0 + a_1 + a_7) - (a_3 + 2a_4 + a_5)$   
ขนาดขอบของภาพ Sobel เป็น  $m \in R^x$  ให้

$$m(i, j) = \sqrt{u^2 + v^2} \quad 2.10$$

และให้ทิศทางของเกรเดียนต์ของภาพ  $d$  คือ

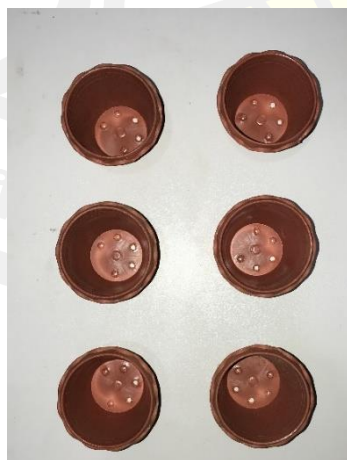
$$d(i, j) = \arctan\left(\frac{u}{v}\right) \quad 2.11$$

ให้  $S$  คือ Mask ของแนวแกน X และ  $T$  คือ Mask ของแนวแกน Y

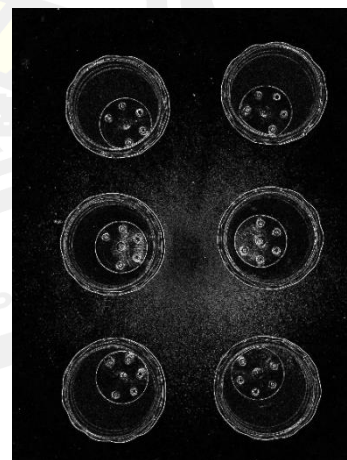
$$S = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \quad 2.12$$

$$T = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad 2.13$$

จากภาพที่ 2.31 แสดงตัวอย่างของการหาขอบภาพของกระถางปลูกพืชด้วยวิธีโซเบล โดยภาพ (a) เป็นภาพต้นฉบับที่ถ่ายจากกล้องเพื่อใช้เป็นภาพอินพุต และ (b) เป็นภาพผลลัพธ์การตรวจจับขอบภาพด้วยวิธีโซเบล



(a) Input Image



(b) Sobel Image

ภาพที่ 2.31 การหาขอบภาพด้วยวิธี Sobel Edge Detection

## 2.12 การหาขอบภาพโดยวิธีแคนนี่

การหาขอบภาพด้วยวิธีแคนนี่ (Canny Edge Detection) นั้น ประกอบด้วย 4 ขั้นตอน โดยเริ่มจากการปรับภาพให้เรียบ (Smoothing) ด้วยตัวกรองเกาส์เซียน (Gaussian Filter) เพื่อกำจัดสัญญาณรบกวน หลังจากนั้นทำการคำนวณค่าขนาด (Magnitude) และทิศทาง (Orientation) ของ Gradient โดยการหาอนุพันธ์อันดับหนึ่ง จากนั้นจึงใช้ Non-maxima Suppression (NMS) กับขนาดของ Magnitude ของเกรเดียนต์ เพื่อให้ได้ขอบที่บางลง และในขั้นตอนสุดท้ายจะใช้ Double Thresholding Algorithm เพื่อหาพิกเซลที่เป็นขอบและทำการเชื่อมต่อขอบ โดยในแต่ละขั้นตอนมีรายละเอียดดังนี้

1. การปรับภาพให้เรียบ (Smoothing) เป็นขั้นตอนแรกของการหาขอบภาพด้วยวิธีแคนนี่ ซึ่งทำการกำจัดสัญญาณรบกวน (Noise) ออกด้วยการใช้ตัวกรองเกาส์เซียน โดยกำหนดกรอบ Mask ที่เป็นเมทริกซ์ขนาด  $3 \times 3$  หรือขนาด 9 พิกเซล ในการกำหนดขนาดของตัวกรองเกาส์เซียนนั้น หากมีขนาดกว้างมากจะทำให้สามารถลดสัญญาณรบกวนได้มาก แต่ถ้าขนาดของกรอบกว้างมากเกินไปจะส่งผลกระทบต่อขอบย่อย ๆ ที่เป็นส่วนของรายละเอียดหายไป ผลของภาพที่ผ่านการปรับภาพให้เรียบด้วยตัวกรองเกาส์เซียนสามารถหาได้จากสมการที่ 2.14

$$S_{(i,y)} = G_{(i,j,\sigma)} \bullet I_{(i,j)} \quad 2.14$$

กำหนดให้	$I_0$	คือ ภาพที่ต้องการหาขอบ
	$G()$	คือ Gaussian Smoothing Filter
	$\sigma$	คือ ควบคุมระดับของการ Smoothing
	$\bullet$	คือ โอเปอเรชันการคูณ

2. การคำนวณค่าของเกรเดียนต์ (Gradient Calculation) ในขั้นตอนแรกของการปรับภาพ  $I_{(i,j)}$  ให้มีความเรียบ จะได้ผลลัพธ์เป็น ค่าของภาพในฟังก์ชัน  $S_{(i,j)}$  ซึ่งต่อมาในขั้นตอนที่สองจะเป็นการหาค่าของเกรเดียนต์ในทิศทางของแกน X และแกน Y และกำหนดขนาดของอนุพันธ์อันดับหนึ่งของ  $Px_{(i,j)}$  และ  $Qy_{(i,j)}$  ตามลำดับ ดังสมการที่ 2.15 และ 2.16

$$Px_{(i,y)} \approx (S_{(i,j+1)} - S_{(i,j)} + S_{(i+1,j+1)} - S_{(i+1,j)}) / 2 \quad 2.15$$

$$Qx_{(i,y)} \approx (S_{(i,j)} - S_{(i+1,j)} + S_{(i,j+1)} - S_{(i+1,j+1)}) / 2 \quad 2.16$$

นำค่า  $P_{x(i,j)}$  และ  $Q_{y(i,j)}$  ที่ผ่านการหาอนุพันธ์อันดับหนึ่งเมื่อคำนวณการเปลี่ยนแปลงรูปแบบจากระนาบของระบบพิกัดฉาก (Rectangular Form) ไปเป็นระนาบพิกัดเชิงขั้ว (Polar Form) เพื่อหาขนาดและทิศทางของเกรเดียนต์ เมื่อแทนค่าตามสมการที่ 2.15 และ 2.16 แล้วจะได้ค่าขนาดเกรเดียนต์

$$M(i,j) = \sqrt{P_x^2(i,j) + Q_y^2(i,j)} \quad 2.17$$

และทิศทางของเกรเดียนต์

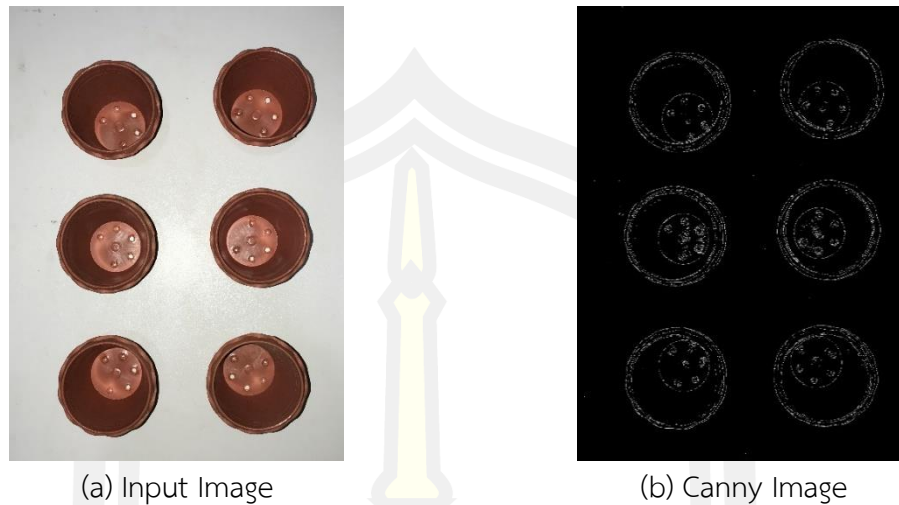
$$\theta(i,j) = \tan^{-1}(Q_y(i,j) + P_x(i,j)) \quad 2.18$$

และสามารถหาค่ามุม  $\theta$  ออกมาได้เมื่อแทนค่าตัวแปรในฟังก์ชัน  $\theta = \tan^{-1}(x, y)$

3. การใช้ Non-maxima Suppression (NMS) ซึ่งการหาขอบภาพด้วยวิธีแคนนี่นั้น จุดที่ถือว่าเป็นขอบภาพได้จะต้องเป็นจุดที่ให้ค่าสูงสุดเฉพาะที่และมีทิศทางเดียวกับเกรเดียนต์ ดังนั้นการหาขอบภาพโดยใช้อนุพันธ์อันดับหนึ่งทำให้ได้ขอบที่บางเพียง 1 พิกเซล ซึ่งภาพที่ได้หลังการทำ Non-maxima Suppression จะให้ค่าเป็นศูนย์ในทุกจุด ยกเว้นจุดที่เป็น Local Maxima Point ที่จะต้องคงค่าเดิมเอาไว้

4. การกำหนดจุดอ้างอิง 2 ระดับ (Double Thresholding) จะถูกนำมาใช้ในขั้นตอนสุดท้าย ถึงแม้ว่าในขั้นตอนแรกนั้นจะผ่านการปรับภาพให้เรียบมาแล้ว แต่เนื่องจากสัญญาณรบกวนหรือลักษณะของวัตถุในภาพมีรายละเอียดมาก ดังนั้นเพื่อลดปัญหาดังกล่าวลงจึงได้กำหนดค่า Threshold ขึ้นมา 2 ค่า คือ High Threshold ( $T_1$ ) และ Low Threshold ( $T_2$ ) โดยพิกเซลที่มีค่ามากกว่า  $T_1$  จะถูกกำหนดเป็น “1” เป็นพิกเซลที่เป็นขอบ แต่ถ้าน้อยกว่า  $T_2$  จะถูกกำหนดเป็น “0” ส่วนค่าที่อยู่ระหว่าง Threshold ทั้งสอง การกำหนดค่าเป็น “0” หรือ “1” นั้น จะขึ้นอยู่กับพิกเซลที่อยู่รอบข้าง หากพบว่าพิกเซลที่อยู่รอบข้างของพิกเซลที่เป็นขอบ (ขอบค่า  $> T_1$ ) มีค่ามากกว่า  $T_2$  แล้ว จะกำหนดค่าพิกเซลดังกล่าวให้มีค่าเป็น “1” และถือว่าเป็นสมาชิกหนึ่งในขอบภาพ ดังนั้นการทำ Threshold จึงเป็นการทำให้ภาพมีขอบภาพที่หนาหรือบางนั่นเอง

จากภาพที่ 2.32 แสดงตัวอย่างของการหาขอบภาพของกระถางปลูกพืชด้วยวิธีแคนนี่ โดยภาพ (a) เป็นภาพต้นฉบับที่ถ่ายจากกล้องเพื่อใช้เป็นภาพอินพุต และ (b) เป็นภาพผลลัพธ์การตรวจจับขอบภาพด้วยวิธีแคนนี่



ภาพที่ 2.32 การหาขอบภาพด้วยวิธี Canny Edge Detection

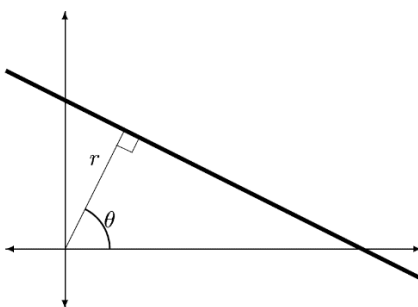
### 2.13 Hough Transform

การหาขอบภาพด้วยเทคนิค Edge Detection ที่กล่าวมา เป็นเพียงการตรวจจับขอบในภาพเท่านั้น แต่ไม่ได้แสดงลักษณะของความชัน (Slope) จุดตัดของเส้น (Intercept of the Line) หรือรัศมีของวงกลม (Radius of a Circle) ซึ่งคุณสมบัติเหล่านี้คำนวณได้จากการใช้ Hough Transform [31] โดยที่วิธีการของ Hough Transform คือ การค้นหาเส้นตรงและวงกลมจากจุดต่างๆ โดยจะมีการโหวตจุดเหล่านั้นในแต่ละจุดว่าอยู่บนเส้นใด เมื่อทุกจุดถูกทำการโหวตแล้ว สมการที่ถูกโหวตมากที่สุดจะเป็นเส้นที่ผ่านจุดมากที่สุด แบ่งออกเป็น

1. Hough Line Transform ซึ่งรูปแบบทั่วไปของเส้นถูกกำหนดโดยสมการ  $y = mx + b$  โดยที่  $m$  แสดงความชันของเส้น และ  $b$  แทนค่าตัดแกน  $y$  แต่ในกรณีของเส้นแนวตั้ง  $m$  จะไม่ถูกนิยาม (Undefined) และไม่มีที่สิ้นสุด (Infinity) ด้วยเหตุนี้ ระนาบของ Accumulator Plane จะมีความยาวไม่สิ้นสุด ส่งผลให้ไม่สามารถทำการโปรแกรมได้บนคอมพิวเตอร์ ดังนั้นจึงต้องใช้พิกัดเชิงขั้ว (Polar Coordinates) ที่มีขอบเขตสำหรับทุกความลาดชันและทุกจุดตัดเพื่อกำหนดลักษณะของเส้น รูปแบบของระบบพิกัดเชิงขั้ว (Polar Form) ของเส้น หรือ Normal Form จะถูกกำหนดโดยสมการที่ 2.19

$$x \cos(\theta) + y \sin(\theta) = r \quad 2.19$$

โดยที่  $r$  มีค่าเป็นบวกและเป็นระยะตั้งฉากระหว่างจุดกำเนิดและเส้นตรง  $\theta$  คือ ความชันของเส้นตรง มีช่วงตั้งแต่  $[0, 180]$  ซึ่งแต่ละจุดในระนาบ  $(x,y)$  หรือที่เรียกว่าระนาบคาร์ทีเซียนนี้ สามารถแปลงเป็นระนาบ  $(r, \theta)$  หรือระนาบแอคคิมูลเตอร์ ดังแสดงในภาพที่ 2.33



ภาพที่ 2.33 เส้นและค่าพารามิเตอร์ของเส้น

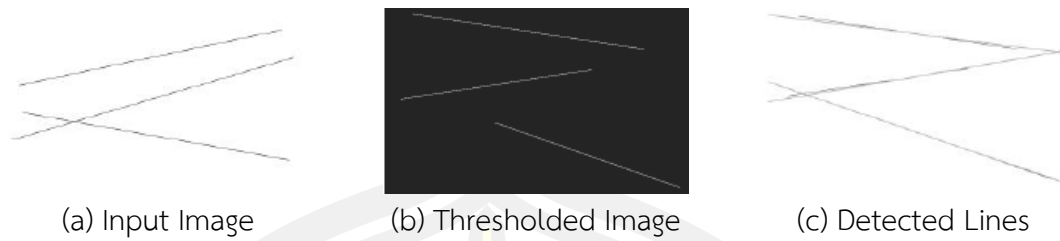
ที่มา : [29]

ภาพที่ถูกแบ่งส่วน (Segmented Image) จะถูกนำไปเป็นอินพุตสำหรับการทำ Hough Line Transform เพื่ออธิบายลักษณะของเส้น โดยระนาบแอคคิมูลเตอร์ 2 มิติ จะถูกสร้างขึ้นสำหรับ  $(r, \theta)$  และแต่ละค่า  $x$  ที่อยู่ในภาพ สอดคล้องกับค่า  $y$  ที่ถูกคำนวณในสมการ โดยสำหรับค่า  $y$  ทุกค่า นั่นคือ Foreground Pixel ตัวอย่างเช่น ค่า  $y$  ที่อยู่บนเส้นนั้น จำทำการเพิ่มค่า 1 ลงไปบน  $(r, \theta)$  ในระนาบแอคคิมูลเตอร์ แล้วทำกระบวนการนี้ซ้ำกับค่าทั้งหมดของ  $(r, \theta)$  ผลลัพธ์ของระนาบแอคคิมูลเตอร์จะมีความเข้มสูงสุดที่จุดซึ่งสอดคล้องกับเส้น จากนั้น  $(r, \theta)$  ที่สอดคล้องกับจุดสูงสุดในแต่ละพื้นที่จะให้ค่าพารามิเตอร์ของเส้นในภาพต้นฉบับ

จากตัวอย่างในภาพที่ 2.34 โดยภาพ (a) เป็นภาพอินพุตที่ถูกแปลงเป็นภาพระดับเทา จากนั้นใช้ Otsu's Method เพื่อแปลงให้ภาพไบนารีดังแสดงในภาพ (b) จากนั้นภาพ (c) จะแสดงผลลัพธ์ของการใช้ Hough Line Transform เพื่อตรวจจับเส้น ซึ่งเส้นหนาที่แสดงในภาพจะเป็นเส้นที่ถูกตรวจพบโดย Hough Line Transform

ถ้าภาพอินพุตมีขนาด  $N \times N$  จำนวนของค่า  $r$  คือ  $M$  และจำนวนจุดใน  $\theta$  คือ  $K$  แล้ว เวลาในการคำนวณสำหรับอาร์เรย์ของแอคคิมูลเตอร์ คือ  $O(KMN^2)$  ดังนั้น Hough Line Transform จึงเป็นกระบวนการที่หนัก (Intensive Process) ถ้า  $\theta$  มีช่วง  $[0, 180]$  และมีขนาดของ Step Size = 1 ดังนั้น  $K = 180$  ตามแนวแกน  $\theta$  หากรู้ช่วงของ  $\theta$  และมีขนาดเล็กกว่า  $[0, 180]$  ค่าของ  $K$  ก็จะลดลง ส่งผลให้เวลาในการคำนวณเร็วขึ้น เช่นเดียวกับปัจจัยอื่น เช่น การลดค่าของ  $M$  และ  $N$  ลง ก็ส่งผลให้สามารถลดเวลาในการคำนวณได้เช่นเดียวกัน





ภาพที่ 2.34 ตัวอย่างของ Hough Line Transform

ที่มา : [31]

2. Hough Circle Transform ซึ่งรูปแบบทั่วไปของวงกลมถูกกำหนดโดยสมการ

$$(x - a)^2 + (y - b)^2 = R^2 \quad 2.20$$

โดยที่  $(a, b)$  เป็นจุดศูนย์กลางของวงกลม และ  $R$  คือ รัศมีของวงกลม และสมการนี้สามารถเขียนใหม่ในรูปของ

$$y = b \pm \sqrt{R^2 - (x - a)^2} \quad 2.21$$

และอีกวิธีหนึ่ง สามารถเขียนในพิกัดเชิงขั้วได้เป็น

$$\begin{cases} x = a + R \cos(\theta) \\ y = b + R \sin(\theta) \end{cases} \quad 2.22$$

โดย  $\theta$  มีช่วง  $[0, 360]$  จากสมการที่ 2.22 แสดงให้เห็นว่า แต่ละจุดในระนาบ  $(x, y)$  สามารถแปลงให้อยู่ในรูป  $(a, b, R)$  ของระนาบไฮเปอร์โพลาร์หรือระนาบแอคคิวมูเลเตอร์ โดยค่าของ  $R$ ,  $a$  และ  $b$  จะถูกสร้างขึ้นในระนาบแอคคิวมูเลเตอร์ 3 มิติ เพื่อใช้ในการอธิบายคุณลักษณะของวงกลมนั้น โดยที่ค่าของ  $x$  และ  $y$  จะถูกคำนวณโดยใช้สมการที่ 2.22 เพื่อกำหนด  $(R, a, b)$  สำหรับแต่ละค่าของ  $\theta$  ซึ่งทุกค่าของ  $x$  และ  $y$  เป็นพิกเซลส่วนหน้า (Foreground Pixel) นั่นคือ จุด  $(x, y)$  ซึ่งอยู่บนวงกลม จะถูกเพิ่มค่าของ 1 เข้าไปใน  $(R, a, b)$  ในระนาบแอคคิวมูเลเตอร์ และขบวนการดังกล่าวนี้จะถูกทำซ้ำกับค่าทั้งหมดของ  $(R, a, b)$  ผลลัพธ์ของระนาบแอคคิวมูเลเตอร์ไฮเปอร์เพลนจะมีความเข้มสูงที่จุดที่สอดคล้องกับวงกลม จากนั้น  $(R, a, b)$  ที่สอดคล้องกับจุดสูงสุดในพื้นที่ที่จะกำหนดค่าพารามิเตอร์ของวงกลมในภาพต้นฉบับให้

ภาพที่ 2.35 คือ ภาพ Computerized Tomography Scan (CT Image) ที่ประกอบด้วย บริเวณวงกลมสว่างสีขาว 2 วง ที่หลุดเลือดซึ่งมีแตกต่างชัดเจนระหว่างสีดำและขาว โดยภาพ (a) ซึ่งใช้เป็นภาพอินพุต ภาพ (b) เป็นภาพที่ผ่านการกรองด้วย Median Filter และภาพ (c) เป็นภาพเอาต์พุตที่ถูกตรวจจับและวาดวงกลมด้วย Hough Circle Transform



(a) Input Image

(b) Image After Performing  
Median Filter(c) Output with min-radius  
= 10 max-radius = 30

ภาพที่ 2.35 ตัวอย่างของ Hough Circle Transform

ที่มา : [31]

ถ้าภาพอินพุตมีขนาด  $N \times N$  จำนวนของค่า  $a$  และ  $b$  คือ  $M$  และจำนวนจุดใน  $R$  คือ  $K$  แล้ว เวลาที่ใช้ในการคำนวณ คือ  $O(KM^2N^2)$  ดังนั้น Hough Circle Transform จึงเป็นกระบวนการที่มีการประมวลผลหนักมากกว่า Hough Line Transform ถ้าหากช่วงของรัศมีที่นำมาทดสอบมีขนาดเล็กลง  $K$  ก็จะเล็กลงและส่งผลให้ลดเวลาในการคำนวณตามไปด้วย ซึ่งถ้าหากทราบตำแหน่งโดยประมาณของวงกลมก็จะช่วยลดช่วงของ  $a$  และ  $b$  ลง จากผลลัพธ์ดังกล่าวจะลดค่าของ  $M$  ลง ทำให้สามารถคำนวณได้เร็วยิ่งขึ้น

## 2.14 ระบบสมองกลฝังตัว

ระบบสมองกลฝังตัว (Embedded System) [32] คือ ระบบคอมพิวเตอร์ขนาดเล็กที่ถูกฝังไว้ในอุปกรณ์อิเล็กทรอนิกส์ และเครื่องใช้ไฟฟ้าต่าง ๆ เพื่อทำงานเฉพาะทาง เพิ่มความฉลาดในการรับรู้หรือตัดสินใจ ตลอดจนความสามารถอื่น ๆ ให้กับอุปกรณ์เหล่านั้น อาจกล่าวได้ว่าระบบสมองกลฝังตัว เป็นระบบอิเล็กทรอนิกส์ที่ใช้สำหรับงานควบคุม รวมถึงการแสดงผลการทำงานต่าง ๆ โดยที่ระบบเหล่านี้ถูกใช้เป็นส่วนหนึ่งของระบบและอุปกรณ์ควบคุมเครื่องมือเครื่องจักรต่าง ๆ การที่ใช้คำว่า “ระบบแบบฝังตัว หรือ ระบบสมองกลฝังตัว” เนื่องจากระบบเหล่านี้เป็นส่วนหนึ่งของระบบใหญ่

ในหลายกรณีที่ใช้ทั่วไปไม่อาจทราบว่าคุณสมบัติควบคุมเครื่องมือ เครื่องจักร รวมถึงระบบใดที่ใช้งานเป็นประจำเหล่านั้นเป็นระบบแบบฝังตัว

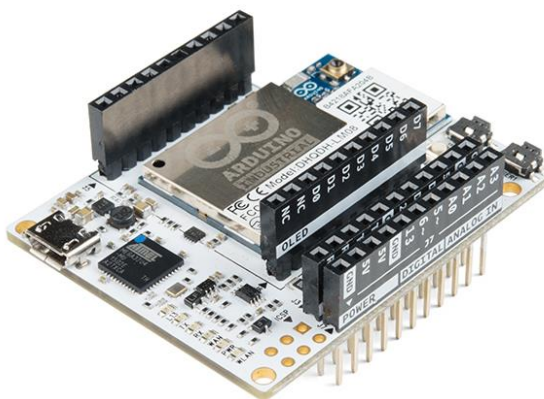
ระบบสมองกลฝังตัวนี้ แม้ไม่ใช่เครื่องคอมพิวเตอร์ แต่ก็มีระบบคอมพิวเตอร์บรรจุอยู่ภายใน อาจจะเป็นเพียงไมโครโพรเซสเซอร์ (Microprocessor) หรือชิป (Chip) หรือโพรเซสเซอร์ (Processor) ที่ประกอบด้วยชิป (Chip) ที่มีวงจรซับซ้อน โดยจะมีหลักการทำงาน คือ มีสัญญาณข้อมูลเข้า (Input) จากอุปกรณ์เซ็นเซอร์ (Sensor) เข้าสู่ระบบ และมีสัญญาณผลลัพธ์ (Output) ของระบบไปควบคุมบังคับ (Actuator) สวิตช์ของเครื่องควบคุมต่าง ๆ เช่น สวิตช์เครื่องจักร หรือ วาล์วควบคุมทิศทางการไหลของท่อต่าง ๆ ในส่วนของไมโครโพรเซสเซอร์ (Microprocessor) หรือ ชิป (Chip) จะมีทั้งที่ออกแบบมาเพื่อให้สามารถเพิ่มโปรแกรมเข้าไปในระบบฝังตัวได้และเพิ่มโปรแกรมไม่ได้ แต่ที่พบเห็นโดยทั่วไปเป็นแบบที่ควบคุมด้วยคำสั่งโปรแกรมซึ่งจะไม่สามารถแก้ไขได้อีก ถ้าหากต้องการแก้ไขก็ต้องเอาชิปดังกล่าวออกและเปลี่ยนชิปตัวใหม่ที่บรรจุคำสั่งโปรแกรมที่แก้ไขแล้ว

นอกจากนี้ แบบและรุ่นของสมองกลฝังตัวมีหลากหลายประเภท ทั้งระบบที่เป็นแบบง่าย ๆ การทำงานไม่ซับซ้อน ตลอดจนแบบระบบที่ซับซ้อน ซึ่งขึ้นอยู่กับประเภทและจำนวนไมโครโพรเซสเซอร์รวมถึงงานโปรแกรมควบคุมในระบบ ซึ่งประเภทของระบบสมองกลฝังตัวสามารถแบ่งได้เป็น 2 ประเภท ได้แก่

1. แบบไมโครโพรเซสเซอร์เดี่ยว เป็นระบบซึ่งใช้อยู่ในอุปกรณ์ขนาดเล็ก เช่น อุปกรณ์ไฟฟ้า วงจรไฟฟ้าต่าง ๆ และเครื่องตรวจจับต่าง ๆ
2. แบบไมโครโพรเซสเซอร์หลายตัวรวมกันในวงจร เป็นระบบซึ่งใช้อยู่ในอุปกรณ์ควบคุมที่ซับซ้อน เช่น อุปกรณ์ควบคุมการไหลของแก๊ส ของเหลว กระแสไฟฟ้า อุปกรณ์ขยายสัญญาณต่าง ๆ อุปกรณ์เปิดวาล์ว เครื่องควบคุมเครื่องจักรในโรงงาน ซึ่งจะมีทั้งที่ไม่ทำงานเกี่ยวกับเวลาและทำหน้าที่เกี่ยวกับเวลา

ทั้งนี้แนวโน้มของระบบสมองกลฝังตัวจะมีความซับซ้อน (Complexity) เพิ่มมากขึ้น เนื่องจากหน้าที่การทำงานที่พัฒนามากขึ้นบนชิป ความต้องการที่จะลดต้นทุนในอุตสาหกรรม อิเล็กทรอนิกส์ลง และความจำเป็นสำหรับระบบอัตโนมัติและการเชื่อมโยงระบบต่าง ๆ เข้าด้วยกัน เป็นเครือข่าย ปัจจุบันเทคโนโลยีทางด้านระบบสมองกลฝังตัวได้พัฒนาอย่างต่อเนื่อง ทั้งทางด้าน ฮาร์ดแวร์ (Hardware) และซอฟต์แวร์ (Software) โดยระบบสมองกลฝังตัวนี้ถูกนำมาใช้งานในชีวิตประจำวัน โดยติดตั้งอยู่ภายใน เครื่องปรับอากาศ หม้อหุงข้าว เต้าไมโครเวฟ เครื่องรับโทรทัศน์ ยานยนต์ เครื่องมือสื่อสาร ตลอดจนเครื่องใช้ไฟฟ้าอื่น ๆ ซึ่งระบบสมองกลฝังตัวนี้ นับวันจะถูกไปใช้ในนวัตกรรมและสิ่งประดิษฐ์มากขึ้น ไม่ว่าจะเป็น หุ่นยนต์ เครื่องจักรกลในโรงงาน หรือยูทิลิตี้กรรณ เป็นต้น เนื่องจากความสามารถที่หลากหลาย ความยืดหยุ่นในการใช้งาน ตลอดจนต้นทุนในการพัฒนาที่ต่ำ ปัจจุบันการนำระบบสมองกลฝังตัวไปใช้งานนั้น สามารถทำได้ง่ายและสะดวกมาก เพียง

แก่นักพัฒนานำบอร์ดระบบสมองกลฝังตัวกับอุปกรณ์อิเล็กทรอนิกส์เพียงไม่กี่ชิ้นก็สามารถนำไปประยุกต์ใช้งานได้หลากหลาย โดยบอร์ดระบบสมองกลฝังตัวมีการพัฒนาขึ้นในหลากหลายแพลตฟอร์ม ซึ่งขึ้นอยู่กับวัตถุประสงค์ของการนำไปใช้ เช่น Arduino ดังแสดงในภาพที่ 2.36 เป็นบอร์ดแบบ Open Source Hardware ขนาดเล็ก ที่สามารถเรียนรู้ได้ง่าย มีโครงสร้างระบบ Software และ Hardware เป็นแบบ Open Source เหมาะสำหรับงานที่มีการคำนวณน้อยและระบบไม่ซับซ้อน หากต้องการทำงานที่มีความซับซ้อนมากยิ่งขึ้น สามารถเลือกใช้เป็น Raspberry Pi หรือ Beaglebone Black ซึ่งบอร์ดเหล่านี้ มีความสามารถเทียบเท่ากับคอมพิวเตอร์ส่วนบุคคล รวมทั้งมี Input Port และ Output Port ที่สามารถรองรับกับการเชื่อมต่อจากอุปกรณ์ภายนอกได้อย่างมากมาย และมีระบบซอฟต์แวร์ในการพัฒนาระบบเป็นแบบ Open Source เช่นเดียวกัน



ภาพที่ 2.36 อุปกรณ์ Open Source Hardware สำหรับใช้พัฒนาระบบสมองกลฝังตัว

ที่มา: [33]

## 2.15 เซ็นเซอร์

เซ็นเซอร์ (Sensor) [34] หมายถึง ชุดอุปกรณ์ ระบบ หรือวงจร มีทำหน้าที่ในการตรวจวัด เพื่อเพิ่มประสิทธิภาพการรับรู้ของมนุษย์ และตรวจจับการเปลี่ยนแปลงคุณสมบัติ หรือลักษณะของสารที่เป็นเป้าหมายในการวิเคราะห์ (Analytical Target) และแสดงผลในลักษณะของสัญญาณที่สามารถตรวจวัดในเชิงปริมาณได้ ทั้งสัญญาณไฟฟ้า สัญญาณกลศาสตร์ และสัญญาณเชิงแสง ตัวอย่างของอุปกรณ์ประเภทเซ็นเซอร์ดังแสดงในภาพที่ 2.37 โดยอุปกรณ์เซ็นเซอร์สามารถแบ่งได้เป็น 3 ประเภท ตามคุณสมบัติในการตรวจวัด ประกอบด้วย

1. เซ็นเซอร์ด้านกายภาพ (Physical Sensor) คือ เซ็นเซอร์ที่ใช้ในการตรวจวัดคุณสมบัติทางกายภาพ เช่น เซ็นเซอร์ในการจับภาพ เซ็นเซอร์วัดอุณหภูมิและความชื้น

2. เซ็นเซอร์ด้านเคมี (Chemical Sensor) คือ เซ็นเซอร์ที่ใช้ในการตรวจวัดสารเคมีต่าง ๆ โดยอาศัยปฏิกิริยาจำเพาะทางเคมี และมีการแปลงเป็นข้อมูลหรือสัญญาณที่สามารถอ่านวิเคราะห์ได้ เช่น เซ็นเซอร์ตรวจวัดสารเคมีปนเปื้อนในสิ่งแวดล้อมหรือดินและน้ำ

3. เซ็นเซอร์ทางชีวภาพ (Biosensor) คือ เซ็นเซอร์ที่อาศัยเทคนิคการนำสารชีวภาพ (Biological Recognition Material) มาเป็นตัวทำปฏิกิริยาจำเพาะกับสารเป้าหมาย เช่น เซ็นเซอร์ที่ใช้ในการตรวจวัดระดับน้ำตาลในเลือด

ทิศทางการพัฒนาอุปกรณ์ในการตรวจวิเคราะห์ทดสอบมีแนวโน้มจะประยุกต์เป็นอุปกรณ์ เซ็นเซอร์ที่เป็นเครื่องมืออิเล็กทรอนิกส์ที่สามารถอ่านผลได้ง่าย แสดงผลเป็นระบบดิจิทัลหรือตัวเลข โดยไม่จำเป็นต้องอาศัยผู้เชี่ยวชาญในการตรวจวิเคราะห์และอ่านผล ผู้ใช้สามารถใช้อุปกรณ์ดังกล่าวได้ด้วยตัวเอง (Point-of-Care: PoC) ซึ่งจากลักษณะเฉพาะของเซ็นเซอร์ที่สามารถใช้เป็นอุปกรณ์ในการตรวจวิเคราะห์ที่มีขนาดพกพาสะดวกและใช้งานง่ายนั้น จึงถูกนำมาประยุกต์ใช้ในอุตสาหกรรมต่าง ๆ ดังนี้

1) เซ็นเซอร์ด้านการเกษตรและอาหาร ถูกนำมาใช้เพื่อมุ่งสร้างผลผลิตด้านการเกษตรเพิ่มมากขึ้น และมีบทบาทสำคัญสำหรับเทคโนโลยีการเกษตรอัจฉริยะ (Smart Farming) ที่มุ่งเน้นการบริหารจัดการทรัพยากรที่มีอยู่จำกัดให้เกิดประโยชน์สูงสุดรวมถึงการใช้เทคโนโลยีเพื่อการวิเคราะห์ปัจจัยแวดล้อมที่ผลกระทบต่อผลผลิตที่เกิดขึ้น ยิ่งไปกว่านั้น เมื่อพิจารณาจากกระบวนการแปรรูปวัตถุดิบทางการเกษตรเพื่อผลิตเป็นอาหารแล้วนั้น ระดับความปลอดภัยและคุณภาพของอาหารเป็นปัจจัยสนับสนุนที่สำคัญ ดังนั้นอุปกรณ์ด้านเซ็นเซอร์ที่ใช้ในการตรวจวิเคราะห์จึงมีบทบาทที่สำคัญ ตั้งแต่การตรวจวัดค่าปนเปื้อนและความปลอดภัยในวัตถุดิบทางการเกษตร ก่อนที่จะนำมาเข้าสู่กระบวนการแปรรูปเป็นอาหาร จนถึงการจำหน่ายและแจกจ่ายสู่ผู้บริโภค

2) เซ็นเซอร์ในอุตสาหกรรมยานยนต์และระบบโลจิสติกส์ มีแนวโน้มการพัฒนาเทคโนโลยีของโลกที่มุ่งสู่โลกของระบบยานยนต์อัจฉริยะ (Smart Cars) และระบบขนส่งอัจฉริยะ (Intelligent Transport System) ที่ตอบสนองความต้องการของผู้ใช้ การใส่ใจในสิ่งแวดล้อม การประหยัดพลังงานและความปลอดภัยของผู้ขับขี่ นำไปสู่การใช้อุปกรณ์และวัสดุที่มีประสิทธิภาพ นาโนเทคโนโลยีจึงเข้ามามีบทบาทสำคัญ รวมถึงการพัฒนาให้เกิดระบบอัจฉริยะของยานยนต์ที่สามารถติดต่อสื่อสารและเชื่อมโยงด้วยระบบไร้สาย เกิดการสื่อสารกันระหว่างยานพาหนะกับยานพาหนะ (Vehicle-to-Vehicle: V2V) และยานพาหนะกับระบบโครงสร้างพื้นฐานต่าง ๆ (Vehicle-to-Infrastructure: V2I) เพื่ออำนวยความสะดวกแก่ผู้บริโภคเพิ่มมากขึ้น

3) เซ็นเซอร์ด้านสุขภาพและการแพทย์ รวมถึงชุดตรวจวิเคราะห์ด้านสุขภาพและการแพทย์นั้น ใช้เพื่อการตรวจวิเคราะห์สารชีวโมเลกุลขนาดเล็กเปลี่ยนวิถีการตรวจรักษาที่ปกติ ตรวจวัดที่โรงพยาบาลเป็นการตรวจวัดที่บ้านได้ เช่น เครื่องตรวจวัดน้ำตาลในเลือดสำหรับผู้ป่วย

โรคเบาหวาน เครื่องตรวจวัดความดันโลหิต อัตราการเต้นของหัวใจ ซึ่งส่วนใหญ่เป็นเซ็นเซอร์ที่ใช้เพื่อตรวจวัดลักษณะภายนอก และเซ็นเซอร์ที่ใช้เพื่อการตรวจวัดสารเคมีต่าง ๆ โดยการใช้ชื่อโมเลกุลที่มีความสามารถในการจดจำ เป็นตัวทำปฏิกิริยาในรูปแบบใดรูปแบบหนึ่ง ได้แก่ เอนไซม์ ดีเอ็นเอแอนติบอดี และโปรตีน เป็นต้น และมีการแปลงสัญญาณเพื่อการวิเคราะห์ค่าต่าง ๆ อย่างไรก็ตาม ถึงแม้จะมีการนำระบบไฟฟ้าอิเล็กทรอนิกส์เข้ามามีส่วนช่วยในการพัฒนาเซ็นเซอร์พบว่า การนำไฟฟ้าหรือการส่งผ่านอิเล็กตรอนนำมาใช้กับสารตรวจวัดทางชีวภาพไม่เท่าที่ควร จึงได้มีการพัฒนาและปรับปรุงโดยการนำนาโนเทคโนโลยีเข้ามาเป็นตัวช่วยให้มีประสิทธิภาพมากยิ่งขึ้น

4) เซ็นเซอร์ด้านสิ่งแวดล้อมและที่อยู่อาศัย ซึ่งการพัฒนาองค์ประกอบต่าง ๆ ให้เข้าสู่ระบบบ้านอัจฉริยะ (Smart Home) เพื่อเสริมสร้างคุณภาพชีวิตของประชากร ทั้งด้านสุขภาพ ความปลอดภัย การประหยัดพลังงาน อาทิ การใช้ระบบเซ็นเซอร์ควบคุมระบบปรับอากาศ ระบบส่องสว่าง และการควบคุมพลังงาน ระบบตรวจและติดตามสิ่งแวดล้อม การแจ้งเตือนภัย การควบคุมการปิดเปิดประตูและหน้าต่าง นอกจากนี้ การพัฒนาเซ็นเซอร์ที่ใช้ในการตรวจวิเคราะห์สภาพแวดล้อมต่าง ๆ ในสังคมมีส่วนในการยกระดับคุณภาพชีวิตของประชากรอีกด้วย ได้แก่ เซ็นเซอร์ที่ใช้ในการตรวจวัดคุณภาพของน้ำ อากาศ ของเสียและขยะมูลฝอย การตรวจวัดระดับของเสียง แสง อุณหภูมิตามสถานที่ต่าง ๆ เป็นต้น



(a) Physical Sensor

(b) Chemical Sensor

(c) Biosensor

ภาพที่ 2.37 ประเภทอุปกรณ์เซ็นเซอร์ตามคุณสมบัติในการตรวจวัด

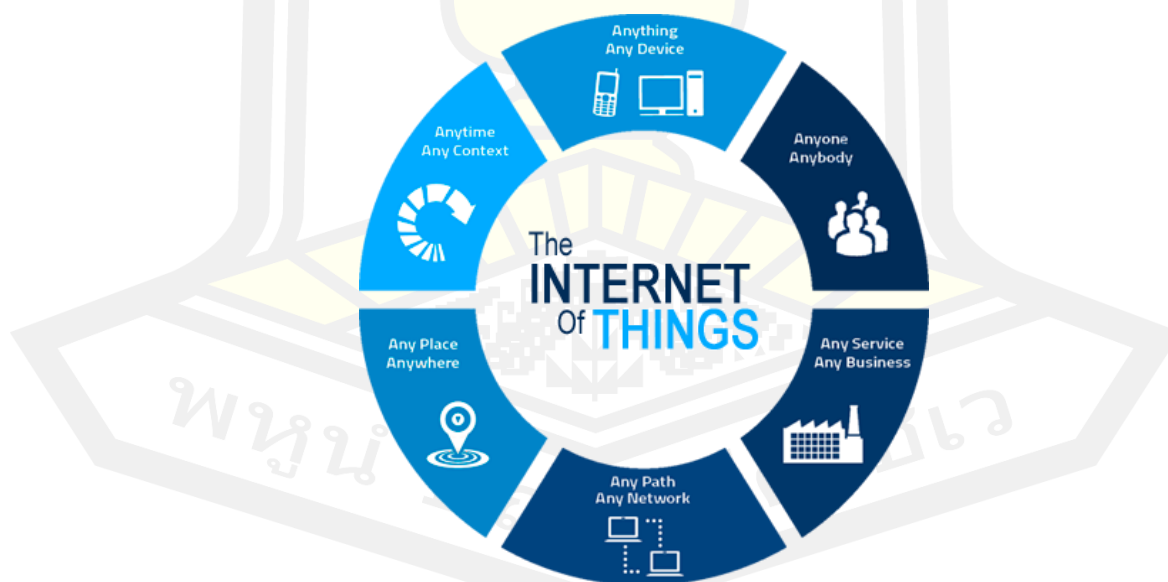
ที่มา: [35]

## 2.16 อินเทอร์เน็ตของสรรพสิ่ง

อินเทอร์เน็ตของสรรพสิ่ง (Internet of Thing หรือ IoT) [36] แนวคิดนี้ถูกเสนอโดย Kevin Ashton ในปี ค.ศ. 1999 ซึ่งเริ่มต้นจากโครงการ Auto-ID Center ของมหาวิทยาลัย

Massachusetts Institute of Technology หรือ MIT จากเทคโนโลยีทางด้านอาร์เอฟไอดี (RFID) ที่จะทำให้เป็นมาตรฐานระดับโลกสำหรับอาร์เอฟไอดี เช่น เซอร์ต่าง ๆ ที่จะเชื่อมต่อกันได้ ต่อมาในปี ค.ศ. 2000 โลกมีการพัฒนาอุปกรณ์อิเล็กทรอนิกส์ออกมาเป็นจำนวนมากและมีการใช้คำว่า Smart ซึ่งหมายถึง Smart Device, Smart Grid, Smart Home, Smart Network, Smart Intelligent Transportation ต่าง ๆ เหล่านี้ ที่ล้วนมีโครงสร้างพื้นฐานที่สามารถเชื่อมต่อกับโลกอินเทอร์เน็ตได้ ซึ่งการเชื่อมต่อเหล่านี้เองจึงกลาย มาเป็นแนวคิดที่ว่าอุปกรณ์เหล่านั้นสามารถสื่อสารกันได้ด้วย เช่นกัน ดังแนวความคิดที่แสดงไว้ในภาพที่ 2.38 โดยอาศัยตัวเซ็นเซอร์ในการสื่อสารถึงกัน นั้นแปลว่า นอกจาก Smart Devices ต่าง ๆ จะเชื่อมต่ออินเทอร์เน็ตได้แล้วมันยังสามารถเชื่อมต่อไปยังอุปกรณ์ตัวอื่นได้ด้วย โดยนิยามไว้ว่าเป็น “Internet-Link” หรืออุปกรณ์อิเล็กทรอนิกส์สามารถสื่อสารพูดคุยกันเองได้ ซึ่งศัพท์คำว่า “Things” จะแทนอุปกรณ์อิเล็กทรอนิกส์ที่กล่าวมาก่อนหน้านั้นเอง

ตัวแปรสำคัญสำหรับ Internet of Things ที่ใช้ในการสื่อสารนั้นไม่เพียงแต่อินเทอร์เน็ตหรือเน็ตเวิร์คเพียงเท่านั้น แต่ยังมีตัวแปรอื่นเข้ามาเกี่ยวข้องอีกนั่นคือ โหนดเซ็นเซอร์ต่าง ๆ จำนวนมากที่ทำให้เกิดเครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Network: WSN) ให้กับอุปกรณ์ต่าง ๆ สามารถเชื่อมต่อเข้ามาได้ ซึ่งเครือข่ายเซ็นเซอร์ไร้สายนี้เอง สามารถตรวจจับปรากฏการณ์ต่าง ๆ (Physical Phenomena) ในเครือข่ายได้ ยกตัวอย่างเช่น แสง อุณหภูมิ ความชื้นสัมพัทธ์ ความดัน เป็นต้น เพื่อส่งค่าไปยังอุปกรณ์อื่น ๆ ในระบบให้ทำงาน หรือสั่งงานอุปกรณ์อื่น ๆ ต่อไป

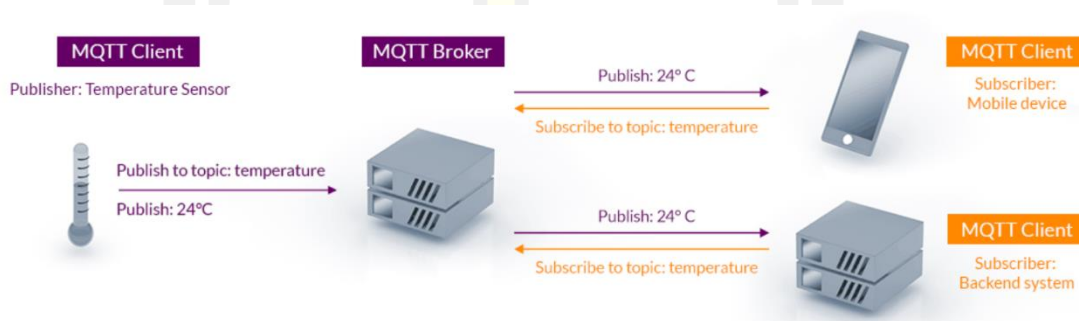


ภาพที่ 2.38 แนวความคิดของเทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง

ที่มา: [37]

## 2.17 Message Queuing Telemetry Transport

Message Queuing Telemetry Transport (MQTT) [38, 39] เป็นโพรโตคอลสำหรับใช้ในสื่อสารข้อมูลระหว่าง Machine to Machine (M2M) ถูกพัฒนาขึ้นโดย Dr. Andy Stanford-Clark จาก IBM และ Arlen Nipper จาก Arcom (Eurotech) ในปี ค.ศ. 1999 โดยใช้โมเดลเน็ตเวิร์คแบบ Publish-Subscribe ที่มีความแตกต่างจากโพรโตคอลอื่นซึ่งใช้โมเดล Server-Client ในการรับส่งข้อมูล MQTT รันอยู่บนเทคโนโลยี TCP/IP จึงทำให้การส่งข้อมูลนั้นไม่เกิดการ Loss ซึ่งออกแบบมาเพื่อใช้สื่อสารในระบบเครือข่ายที่อยู่ห่างไกล มีทรัพยากรค่อนข้างจำกัด ใช้งาน Bandwidth ต่ำ



ภาพที่ 2.39 MQTT Publish / Subscribe Architecture

ที่มา : [38, 39]

MQTT ประกอบด้วย 2 ส่วน คือ

1. MQTT Client ทำหน้าที่ Publish ข้อมูลต่าง ๆ ไปยัง MQTT Broker และสามารถ Subscribe ข้อมูลจาก MQTT Broker ผ่านทาง TCP/IP Protocol เช่นเดียวกัน
2. MQTT Broker หรือ MQTT Server เป็นซอฟต์แวร์ซึ่งทำหน้าที่รับข้อมูลจาก MQTT Client ที่ได้ Publish เข้ามา และสามารถ Publish ข้อมูลจาก MQTT Broker ไปยัง MQTT Client ที่ได้ Subscribe ข้อมูลไว้ได้

ตัวอย่างของรูปการในการสื่อสารของ MQTT ในกรณีการ Monitor ค่าของอุณหภูมิจากชุดเซ็นเซอร์ที่ติดตั้งอยู่บนอุปกรณ์ IoT ผ่านสมาร์ตโฟน มีขั้นตอนดังนี้

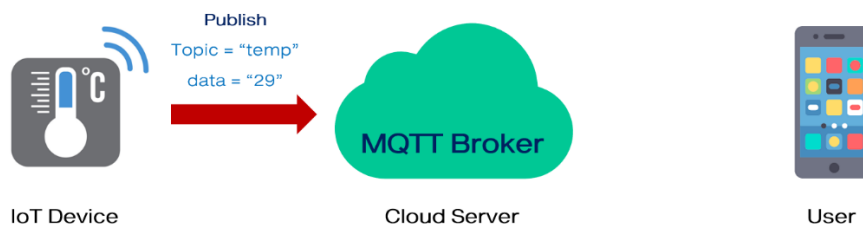
1. กำหนดให้ MQTT Client ทำการ Subscribe ไปยัง MQTT Broker ตาม Topic ที่ต้องการ ในตัวอย่างนี้ คือ ต้องการให้สมาร์ตโฟนติดตาม Topic ชื่อ “temp” เพื่อคอยรับค่าของอุณหภูมิจากอุปกรณ์ IoT ดังภาพที่ 2.40





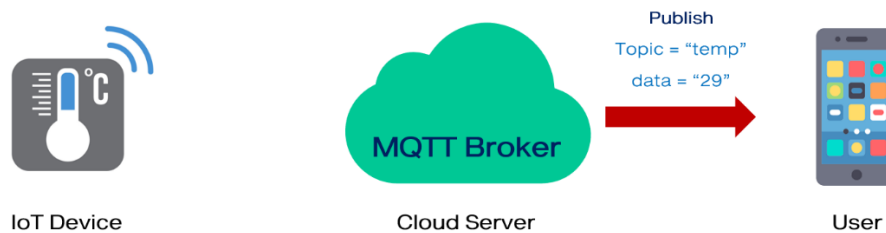
ภาพที่ 2.40 ขั้นตอนการ Subscribe ของ MQTT Client

2. กำหนดให้ MQTT Client ในตัวอย่างนี้หมายถึง Device ที่ต่อกับเซ็นเซอร์วัดอุณหภูมิ ทำการ Publish ค่าของ Topic ชื่อ “temp” ซึ่งมี Data คือ “29” ไปยัง MQTT Broker ดังภาพที่ 2.41



ภาพที่ 2.41 ขั้นตอนการ Publish ของ MQTT Client

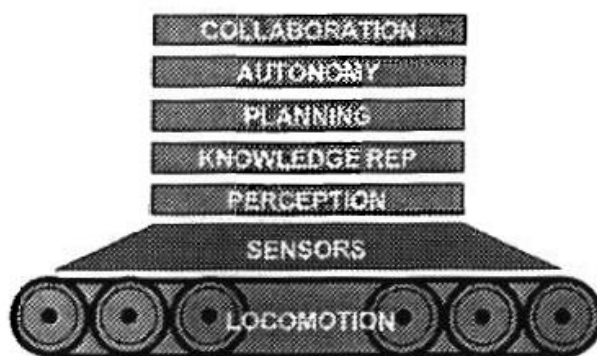
3. จากนั้น MQTT Broker จะทำการ Publish ค่าของ Topic ชื่อ “temp” ซึ่งมี Data คือ “29” ไปยัง MQTT Client ที่เป็นสมาร์ทโฟน ดังนั้นในตัวอย่างนี้ สมาร์ทโฟนจึงได้รับข้อมูลอุณหภูมิจากเซ็นเซอร์ที่อยู่บนอุปกรณ์ IoT ผ่านทางการสื่อสารของ MQTT Broker เรียบร้อยแล้ว ดังภาพที่ 2.42



ภาพที่ 2.42 ขั้นตอนการ Publish ของ MQTT Broker

## 2.18 การประเมินประสิทธิภาพ

เทคนิควิธีในการประเมินประสิทธิภาพการทำงาน (Performance Evaluation Techniques) ของหุ่นยนต์ [40] ในการวิจัยครั้งนี้ มีวัตถุประสงค์หลักในการประเมินประสิทธิภาพ คือ การวัดและการทดสอบความสามารถ (Capabilities) รวมถึงพฤติกรรม (Behaviors) ของหุ่นยนต์ ที่ถูกพัฒนาขึ้น ซึ่งมีองค์ประกอบต่าง ๆ ที่ทำงานร่วมกันแบบอัตโนมัติ บนแพลตฟอร์มของหุ่นยนต์ ประเภท Mobile Robot โดยองค์ประกอบของหุ่นยนต์ที่มีการเคลื่อนที่แบบอิสระ (Autonomy Robot) เมื่อไม่คำนึงถึงโดเมนทางด้านแอปพลิเคชันแล้ว จะมีองค์ประกอบพื้นฐานดังภาพที่ 2.43



ภาพที่ 2.43 Elements of Mobile Robot Autonomy

ที่มา : [40]

จากภาพดังกล่าว ในระดับต่ำสุด คือ ความสามารถในการเคลื่อนที่ (Locomotion) ของแพลตฟอร์มทางกายภาพของหุ่นยนต์ ระดับต่อมาเป็นการรับรู้ผ่านทางประสาทสัมผัส (Sensory Perception) ของหุ่นผ่านทางเซ็นเซอร์และโมดูลการรับรู้ต่าง ๆ เพื่อให้หุ่นยนต์เข้าใจในสภาพแวดล้อมที่อยู่รอบ ๆ ซึ่งนำไปสู่กระบวนการในการตรวจจับ การระบุตำแหน่ง การเคลื่อนที่ หรือการระบุเป้าหมายของหุ่นยนต์ ถัดมานั้นเป็นระดับของการนำเสนอความรู้ (Knowledge Representation) ที่เป็นส่วนในความสามารถของการสร้างแบบจำลองของหุ่นยนต์ ซึ่งมีทั้งการใช้ข้อมูลความรู้เดิม (Prior Information) และข้อมูลที่ได้มาใหม่ (Acquired Information) จากการตรวจวัดสภาพแวดล้อมผ่านทางเซ็นเซอร์ ต่อมาระดับของการวางแผน (Planning) หรือการสร้างพฤติกรรมของหุ่นยนต์ จะถูกสร้างขึ้นโดยอาศัยข้อมูลจากสองระดับที่อยู่ก่อนหน้านี้ ประกอบด้วยข้อมูลของการรับรู้ผ่านทางประสาทสัมผัสและการนำเสนอข้อมูล ถูกนำไปใช้ในการนำทางหุ่นยนต์ให้เคลื่อนที่หรือหลบหลีกอุปสรรค เพื่อเดินทางไปยังเป้าหมายในภารกิจ ซึ่งในขั้นตอนี้จะมีผลกระทบต่อเวลาในการทำงานของหุ่นยนต์ ต่อมาระดับของความอิสระ (Autonomy) ของหุ่นยนต์ เป็น

องค์ประกอบถัดมาที่จะได้รับการประเมิน เนื่องจากหุ่นยนต์ได้ถูกออกแบบให้ทำงานแทนมนุษย์ ดังนั้นการโต้ตอบหรือการมีปฏิสัมพันธ์ของหุ่นยนต์อาจมีความแตกต่างกันไปตามนัยสำคัญของการออกแบบ ความสามารถ สถานการณ์ ตลอดจนปัจจัยอื่น ๆ โดยบางกรณี เพื่อให้เกิดประโยชน์สูงสุด อาจจะมีการสื่อสารกันระหว่างหุ่นยนต์และมนุษย์ ในการตัดสินใจร่วมกัน นอกเหนือจากการที่กำหนดเงื่อนไขให้หุ่นยนต์สามารถตัดสินใจและดำเนินการได้แบบอัตโนมัติด้วยตัวเอง และองค์ประกอบสุดท้าย คือ การประเมินความสามารถในการทำงานร่วมกัน (Collaboration) ของหุ่นยนต์ เพื่อร่วมกันทำภารกิจให้สำเร็จ

ในการเปรียบเทียบประสิทธิภาพของหุ่นยนต์ที่มีความแตกต่างกันทางลักษณะกายภาพ [41] อัลกอริทึมการทำงาน อุปกรณ์เซ็นเซอร์ในการรับรู้ ตลอดจนความสามารถในการเคลื่อนไหวที่แตกต่างกัน แต่เมื่อถูกนำมาทำงานประเภทเดียวกันโดยมีการกำหนดสภาพแวดล้อมที่คล้ายกันนั้น จำเป็นต้องมีการกำหนดตัวชี้วัดสำหรับการประเมินผลขึ้น ซึ่งเป็นการวัดความสามารถทั่วไปของหุ่นยนต์ที่ประกอบด้วยคุณสมบัติ การสร้างแผนที่ (Mapping) การระบุตำแหน่ง (Localization) เส้นทางการเคลื่อนที่ (Trajectory) การวางแผนการเคลื่อนที่ (Planning) การหลีกเลี่ยงอุปสรรค (Obstacle Avoidance) และการกำกับการทำงานทั่วไป (General Supervision) ดังนั้น เพื่อให้การประเมินมีประสิทธิภาพจึงต้องกำหนดตัวชี้วัด (Metrics) และตัวบ่งชี้ (Indicators) ขึ้น ซึ่งเป็นสมรรถนะที่สามารถวัดได้จากหุ่นยนต์จากบางพารามิเตอร์ที่มีความเกี่ยวข้องกัน โดยก่อนอื่นจะต้องกำหนดพารามิเตอร์ที่มีความเกี่ยวข้องในการประเมินประสิทธิภาพของหุ่นยนต์เสียก่อน ที่เรียกว่า ประสิทธิภาพของงาน (Task Performance)

1. สภาพแวดล้อม (Environment) ถูกกำหนดโดยขนาดและค่าความซับซ้อน ซึ่งขนาดมีผลต่อการใช้พลังงานและระยะเวลาของภารกิจ ส่วนความซับซ้อนของสภาพแวดล้อมมีผลต่อปฏิริยาและความทนทานของหุ่นยนต์

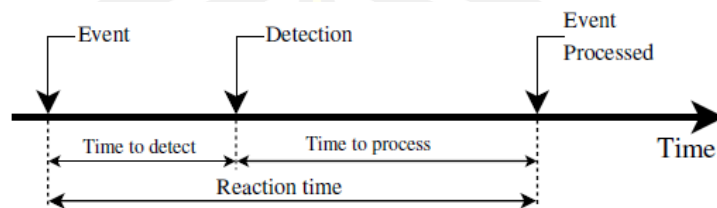
2. ปริมาณของข้อมูลเกี่ยวข้องกับสิ่งแวดล้อม ข้อมูลที่มีนี้จะมีผลต่อประสิทธิภาพของการทำงาน หากสภาพแวดล้อมเป็นที่รู้จักอย่างสมบูรณ์ ย่อมสามารถคาดหวังถึงประสิทธิภาพของหุ่นยนต์ที่ดีขึ้น เนื่องจากลดการใช้พลังงานลง รวมถึงการใช้เวลาในการทำภารกิจที่สั้นลงด้วย นอกจากนี้ ข้อมูลยังมีความเกี่ยวข้องกับความเข้าใจของหุ่นยนต์ที่มีต่อสภาพแวดล้อม เพื่อให้รู้ถึงสถานะและตำแหน่งของตนซึ่งอยู่ในสภาพแวดล้อมนั้น ที่มีผลต่อการระบุถึงตำแหน่งที่แม่นยำ

3. การใช้พลังงาน (Energy Consumption) ของหุ่นยนต์ในการทำภารกิจ

4. กำหนดเวลา (Time Limit) ในการทำภารกิจ

5. ปฏิริยาของหุ่นยนต์ (Robot Reactivity) เป็นเวลาที่ระบบต้องการในการตอบสนองต่อเหตุการณ์ ประกอบด้วย 2 ส่วน คือ เวลาสำหรับการตรวจจับเหตุการณ์ และเวลาในการประมวลผลเหตุการณ์ ดังภาพที่ 2.44 ตัวอย่างเช่น การหลีกเลี่ยงอุปสรรค เซ็นเซอร์ของหุ่นยนต์มีช่วงเวลาที่ไม่

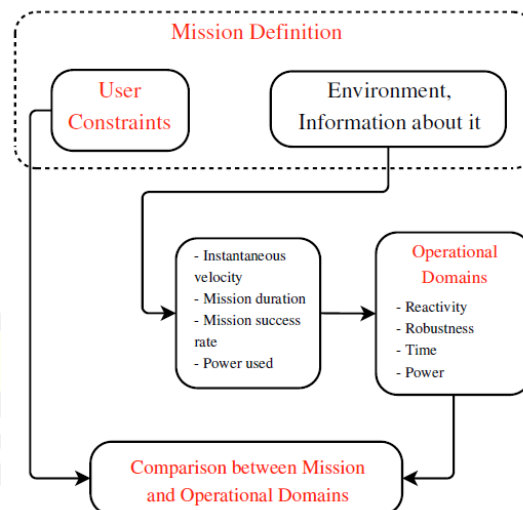
สามารถรับรู้ได้ เวลาจะเริ่มต้นเมื่อสิ่งกีดขวางทะลุผ่านบริเวณที่สังเกตได้ และหยุดเมื่อพิจารณาสิ่งกีดขวาง จากนั้นเวลาในการประมวลผล คือ เวลาที่จำเป็นในการหลีกเลี่ยงอุปสรรคนี้ได้อย่างมีประสิทธิภาพ



ภาพที่ 2.44 เวลา (Time) และ ปฏิกริยา (Reactivity) ของหุ่นยนต์

ที่มา : [41]

6. ความทนทานของหุ่นยนต์ (Robot Robustness) เป็นความสามารถของหุ่นยนต์ในการปรับตัวให้เข้ากับการเปลี่ยนแปลงด้านสิ่งแวดล้อม ซึ่งทำการวัดจากประสิทธิภาพของหุ่นยนต์และความซับซ้อนของสภาพแวดล้อม



ภาพที่ 2.45 วิธีการพิจารณาประสิทธิภาพของหุ่นยนต์ที่คาดหวัง

ที่มา : [41]

จากภาพที่ 2.45 พิจารณาภารกิจการนำทางของหุ่นยนต์ โดยทำการทดสอบภารกิจง่าย ๆ เช่น การนำทางเบื้องต้น (การเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่ง) ในพื้นที่ที่มีขนาดแน่นอน สำหรับในแต่ละภารกิจจะมีคู่ของความแตกต่างของความซับซ้อน (Complexity) และข้อมูล (Information)

และทำการวัดความเร็วของหุ่นยนต์ ระยะเวลาในการทำภารกิจ การใช้พลังงาน และตรวจสอบว่าภารกิจเหล่านั้นสำเร็จหรือไม่ ซึ่งจะเป็นฟังก์ชันเพื่ออธิบายประสิทธิภาพของหุ่นยนต์ตามความซับซ้อนและข้อมูลของพื้นที่ ในตอนนี้จะใช้การอธิบายสภาพแวดล้อมและข้อมูลความรู้เดิมเพื่อแยกแยะและทำนายสิ่งที่เรียกว่า โดเมนปฏิบัติการของหุ่นยนต์สำหรับภารกิจนั้น นั่นหมายถึงขอบเขตที่ยอมรับได้เกี่ยวกับปฏิกิริยา ความทนทาน เวลา และการใช้พลังงานของหุ่นยนต์ ขั้นสุดท้ายจะเปรียบเทียบข้อจำกัดของผู้ใช้กับโดเมนในการดำเนินงาน ซึ่งจะนำมาคิดเป็นเปอร์เซ็นต์ของข้อจำกัดในภารกิจที่คาดหวังจากตัวหุ่นยนต์

ดังที่กล่าวไปแล้วว่า หุ่นยนต์ต้องการตัวชี้วัด (Metrics) เชิงปฏิบัติเพื่อทำการวัดปริมาณที่ต้องการศึกษา โดยมีตัวชี้วัด 2 ประเภท สำหรับการวัดเพื่อประเมินหุ่นยนต์

1) การดำเนินการ (Performances) เช่น ในภารกิจการนำทางแบบง่าย จะมีตัวเลือกระหว่างตัวชี้วัดหลายอย่าง เช่น

- 1.1) ความเร็วในทันที (Instantaneous Velocity)
- 1.2) ความเร็วเฉลี่ย (Average Velocity)
- 1.3) ระยะในการเดินทาง (Traveled Distance)
- 1.4) ระยะเวลาในการทำภารกิจ (Mission Duration)
- 1.5) อัตราความสำเร็จของภารกิจ (Mission Success Rate)

ถึงแม้ว่า เวลา ระยะทาง และความเร็วจะดูเหมือนว่ามีความสัมพันธ์ซึ่งกันและกัน แต่ค่าเหล่านี้ไม่ได้มีความสำคัญเหมือนกันในทุกภารกิจของหุ่นยนต์ สำหรับภารกิจหนึ่งเวลาอาจมีความสำคัญ แต่ในอีกภารกิจหนึ่งที่แตกต่างออกไประยะทางอาจมีความสำคัญที่มากกว่า สิ่งนี้อาจมีผลต่อเส้นทางในการเคลื่อนที่ของหุ่นยนต์ นอกจากนี้เมื่อหุ่นยนต์วางแผนเส้นทางในการเคลื่อนที่แล้ว จะส่งผลให้ระยะทางถูกกำหนดตายตัว ในขณะที่เวลาจะถูกเปลี่ยนแปลง จากตัวชี้วัดที่ได้กล่าวมานั้นสามารถแบ่งตัวบ่งชี้ได้ 2 ชนิด คือ ตัวบ่งชี้แบบ Global Indicators ที่ให้ข้อมูลเกี่ยวกับภารกิจทั้งหมด เช่น ค่าความเร็วเฉลี่ยของหุ่นยนต์ กับตัวบ่งชี้แบบ Local Indicators ซึ่งเป็นเหมือน Snapshot ของภารกิจ เช่น ความเร็วชั่วขณะของหุ่นยนต์

2) ความซับซ้อนของพื้นที่ (World Complexity) สำหรับตัวชี้วัดประเภทนี้ เป็นการวัดความซับซ้อนของสภาพแวดล้อม ซึ่งสามารถกำหนดตัวชี้วัดได้ทั้งในระดับ Local และระดับ Global ที่จะมีตัวบ่งชี้ 2 ตัว โดยตัวแรกคำนึงถึงสภาพแวดล้อมทั้งหมด ในขณะที่ตัวที่สองมุ่งเน้นไปที่บริเวณใกล้เคียงหุ่นยนต์

## 2.19 งานวิจัยที่เกี่ยวข้อง

ปัจจุบันมีงานวิจัยที่เกี่ยวข้องกับการพัฒนาหุ่นยนต์ที่ใช้ในภาคการเกษตรอย่างกว้างขวาง ซึ่งมักพบว่า ส่วนใหญ่มุ่งเน้นแก้ไขปัญหาแรงงานในภาคอุตสาหกรรมการเกษตร ทดแทนแรงงานมนุษย์ ลดเวลา ลดความยุ่งยากในภาระงานที่ต้องทำเป็นประจำของเกษตรกรลง เพื่อให้สามารถควบคุมการผลิตหรือการเก็บเกี่ยวได้อย่างแม่นยำ โดยพบว่าในการวิจัยเหล่านี้เป็นสหวิทยาการเชิงประยุกต์ ที่นำวิชาพื้นฐานหลักกว่าด้วย วิศวกรรมเครื่องกล วิศวกรรมอิเล็กทรอนิกส์ วิศวกรรมการควบคุมอัตโนมัติ วิทยาการคอมพิวเตอร์ และเทคโนโลยีสารสนเทศ มาบูรณาการเข้าด้วยกันเพื่อการออกแบบและพัฒนาหุ่นยนต์ โดยการนำเทคโนโลยีในหลากหลายด้าน เช่น ระบบสมองกลฝังตัว (Embedded System) การประมวลผลภาพ (Image Processing) ปัญญาประดิษฐ์ (Artificial Intelligence) และอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) เข้ามาทำงานร่วมกันอย่างเป็นระบบ เพื่อให้เกิดประสิทธิภาพในการทำงานของหุ่นยนต์ตามวัตถุประสงค์ที่ต้องการ ซึ่งในงานวิจัยนี้จะเกี่ยวข้องกับการออกแบบและพัฒนาหุ่นยนต์ที่ใช้ในการเพาะปลูกพืชภายในโรงเรือน โดยมีงานวิจัยที่เกี่ยวข้อง ดังนี้

### 1. งานวิจัยที่เกี่ยวข้องกับระบบฟาร์มอัจฉริยะและเกษตรแบบแม่นยำ

Kaloxilos และคณะ [2] ได้ศึกษาการปรับปรุงการทำการเกษตรในรูปแบบของระบบฟาร์มอัจฉริยะและเกษตรแบบแม่นยำ (Smart/Precision Farming Systems) เพื่อปรับปรุงกิจกรรมทางการเกษตรในช่วงที่ผ่านมา ด้วยระบบการจัดการฟาร์มรูปแบบใหม่ เพื่อแทนที่การเกษตรรูปแบบเดิมที่ซับซ้อนและล้าสมัย โดยนำเอาเทคโนโลยีสมัยใหม่เข้ามาช่วยบริหารจัดการระบบการเกษตรผ่านทางเครือข่ายอินเทอร์เน็ต แต่ยังคงพบว่าการดำเนินการภายใต้เครือข่ายอินเทอร์เน็ตในปัจจุบันยังมีข้อจำกัด อยู่ เช่น การจัดการกับจำนวนของอุปกรณ์ทางด้านเครือข่ายที่เชื่อมต่อกันเป็นระบบโครงข่ายการทำงาน อย่างเช่น Internet of Things (IoT) ที่มีจำนวนมาก หรือการยอมให้สามารถผสานการทำงานของระบบหรือบริการที่ถูกพัฒนาโดยนักพัฒนาระบบรายอื่น ๆ สามารถทำได้โดยง่าย ในการศึกษานี้ได้อธิบายถึงฟังก์ชันการทำงานของสถาปัตยกรรม FMS ขั้นสูง ที่จะถูกนำมาใช้กับคุณลักษณะอินเทอร์เน็ตในอนาคต ซึ่งคุณลักษณะหลัก ได้แก่ การสนับสนุนขั้นตอนของการทำการเกษตรในรูปแบบทั่วไป การสนับสนุนและผสานการทำงานของผู้ที่มีส่วนเกี่ยวข้องกับการก่อสร้างพื้นฐานทางด้านระบบเครือข่าย การนำระบบอัตโนมัติและระบบการเรียนรู้เพื่อให้เข้าใจในกระบวนการของการจัดการในองค์กรรวม ซึ่งสถาปัตยกรรมนี้เป็นผลมาจากการวิเคราะห์กรณีศึกษาจำนวนมากโดยนักวิจัยที่มีความเชี่ยวชาญในหลากหลายสาขา ซึ่งจะช่วยให้เกษตรกรก้าวเข้าสู่ความเป็นโหนดทางด้านการเกษตรของเครือข่ายระดับโลก (Node in an Agricultural World Wide Web) โดยวิธีการเพิ่มประสิทธิภาพในการบริหารจัดการงานภายในนี้ คือ การบุกเบิกการให้บริการ

ทางด้านเครือข่ายอินเทอร์เน็ตสำหรับการป้อนให้กับตลาดทั่วโลก ง่ายต่อการเข้าถึงข้อมูลและการรับคำปรึกษา ความสะดวกในการติดต่อสื่อสารกับผู้มีส่วนร่วมทั้งหมดในวงจรห่วงโซ่อาหาร การผสมผสานและการจัดการข้อมูลที่มีความหลากหลายซึ่งมาจากแหล่งที่แตกต่างอย่างมีประสิทธิภาพในการศึกษาขั้นต่อไปจะเป็นวางแผนดำเนินการและทดสอบสถาปัตยกรรมน้อยอย่างน้อยสองโครงการนำร่อง ประกอบด้วย โครงการแรกจะเกี่ยวข้องกับการบริหารจัดการโรงเรือน (Greenhouse) และโครงการที่สองจะเกี่ยวข้องกับระบบการประสานงานและจัดการแบบอัตโนมัติของเครื่องจักรในพื้นที่เปิดโล่งภายใต้สถานการณ์ที่แตกต่างกัน ซึ่งกรณีเหล่านี้จะช่วยให้การทดสอบในสภาพแวดล้อมจริงมีความเหมาะสมกับเทคโนโลยีที่มีอยู่เช่นเดียวกับความจำเป็นในการใช้วิธีการที่ได้รับการพัฒนาขึ้นมาใหม่ โดยเฉพาะอย่างยิ่งภายใต้ขอบเขตความสามารถของระบบอินเทอร์เน็ตในอนาคต

ธีรพงษ์ มังคะวัฒน์ [3] ได้ศึกษาถึงแนวคิดและกระบวนการพัฒนาเทคโนโลยีการเกษตรแม่นยำสูง (Precision Farming) โดยมีกรอบความคิดว่าการเกษตรกรรมเป็นเรื่องใกล้ตัว เนื่องจากปัจจัยทางด้านอาหาร เครื่องนุ่งห่ม ยารักษาโรค และที่อยู่อาศัย ล้วนเป็นผลผลิตจากภาคการเกษตรทั้งสิ้น อีกทั้งปัจจุบันเทคโนโลยีมีความก้าวหน้าเป็นลำดับ จึงเริ่มส่งผลให้เกิดการปรับเปลี่ยนทางด้านรูปแบบของการเกษตรกรรมที่เปลี่ยนแปลงไป เพื่อให้ผลผลิตสูงขึ้น มีคุณภาพ และสามารถควบคุมปัจจัยต่าง ๆ ที่เกี่ยวข้องซึ่งส่งผลต่อการผลิตได้ จากแต่เดิมที่เกษตรกรไทยมักทำการเกษตรแบบพึ่งพาธรรมชาติ เปลี่ยนรูปแบบมาเป็นการใช้เทคโนโลยีเข้ามามีส่วนร่วมในการบริหารจัดการในรูปแบบฟาร์มเกษตรอัจฉริยะ (Smart Farm) ซึ่งประกอบด้วยสิ่งสำคัญสามส่วน คือ สารสนเทศ เทคโนโลยี และการบริหารจัดการ ในการศึกษาพบว่าระบบ Smart Farm มี 5 องค์ประกอบ ซึ่งได้แก่ 1) Data Collection เป็นขั้นตอนการเก็บข้อมูลของดิน น้ำ แสง ภูมิอากาศ ผลผลิต เป็นต้น ด้วยวิธีการและเทคโนโลยีอย่าง เครือข่ายเซ็นเซอร์ 2) Diagnostics เป็นขั้นตอนในการวินิจฉัยข้อมูล สร้าง กรอง และเก็บข้อมูลเข้าสู่ฐานข้อมูล 3) Analysis เป็นขั้นตอนในการวิเคราะห์ข้อมูล การทำนาย รวมถึงการจัดวางแผนการจัดการข้อมูล 4) Precision Field Operation เป็นขั้นตอนการปฏิบัติการตามแผนที่วางเอาไว้ และ 5) Evaluation เป็นขั้นตอนในการประเมินผลการปฏิบัติการ นอกจากองค์ประกอบเหล่านี้แล้วระบบยังมีการนำเอาเทคโนโลยีอื่น ๆ ที่ทันสมัยเข้ามามีบทบาทสำคัญเพื่อช่วยในการบริหารจัดการ Smart Farm ด้วย เช่น ระบบเทคโนโลยีในการระบุพิกัด หรือ Global Positioning System (GPS) ระบบเทคโนโลยีภูมิสารสนเทศ หรือ Geographic Information System (GIS) ระบบเทคโนโลยีรับรู้ระยะไกล (Remote Sensing) ระบบเทคโนโลยีรับรู้ระยะใกล้ (Proximal Sensing) ระบบเทคโนโลยีการให้ปุ๋ย/น้ำ/ยาฆ่าแมลง หรือ Variable Rate Technology (VTR) และระบบการทำนายและตัดสินใจ หรือ Crop Model and Decision Support System (DSS) โดยมีไร่ไวน์กรานมอนเต้ เขาใหญ่ เป็นตัวอย่างของการนำระบบดังกล่าวมาใช้เพื่อเฝ้าติดตามความเป็นไปภายในไร่ผ่านทางเครือข่ายอินเทอร์เน็ต และอุปกรณ์สมาร์ตโฟน โดยอาศัยเทคโนโลยี Multi-

functional and Multi-dimensional Sensors ซึ่งจะคอยตรวจสอบสภาพอากาศ ดิน ความเร็วและทิศทางลม ปริมาณน้ำฝน พลังงานแสง และค่าพารามิเตอร์อื่น ๆ ที่สำคัญ รวมถึงการนำเทคโนโลยี RFID มาใช้ในการดูแลกิจกรรมในไร่ นอกจากนี้ยังทำการการพัฒนาซอฟต์แวร์ช่วยตัดสินใจ Decision Support System การบูรณาการข้อมูลภูมิสารสนเทศ และอุตุนิยมวิทยาทั้งระดับไร่ และระดับภูมิภาค โดยมีเป้าหมายทำให้เกิด ไร่นาอัจฉริยะ (Smart Vineyard)

Kanjilal และคณะ [42] ได้ทำการศึกษาเพื่อพัฒนาต้นแบบของฟาร์มอัจฉริยะ (Smart Farm) ภายใต้แนวคิดของการเพิ่มประสิทธิภาพในการผลิตทางภาคการเกษตรโดยอาศัยเทคโนโลยีและการทำงานในรูปแบบของระบบอัตโนมัติ เพื่อลดการใช้แรงงาน และทรัพยากรต่าง ๆ ลง ซึ่งปัจจัยในการควบคุมและตรวจวัด ประกอบไปด้วย 1) การควบคุมแสงในโรงเรือน 2) การควบคุมอุณหภูมิและความชื้นภายในโรงเรือน 3) การตรวจสอบการเกิดเพลิงใหม่และควันไฟ 4) การเปิด-ปิดประตูของโรงเรือน 5) ความชื้นสัมพัทธ์และความชื้นของดิน 6) การตั้งเวลาการทำงานของระบบล่วงหน้า 7) การควบคุมการให้น้ำ และ 8) การควบคุมการทำงานของระบบจากระยะไกล ซึ่งการศึกษาในครั้งนี้จึงเป็นการพยายามในการสร้างระบบการจัดการฟาร์มที่มีประสิทธิภาพ โดยการรวบรวมเอาการตรวจวัดค่าพารามิเตอร์ที่สำคัญต่าง ๆ ด้วยเซ็นเซอร์ เชื่อมต่อเข้ากับระบบไมโครคอนโทรลเลอร์เพื่อการตรวจสอบและตัดสินใจในการแก้ไขปัญหาด้วยระบบเองโดยอัตโนมัติ อีกทั้งยังสามารถทำการตั้งโปรแกรมการทำงานล่วงหน้า และควบคุมได้จากระยะทางไกลผ่านทางเครือข่าย GSM อีกด้วย ซึ่งผลการทดสอบการนำไปใช้งานจริงของคณะผู้ศึกษา พบว่าการออกแบบและพัฒนาระบบฟาร์มอัจฉริยะนี้ช่วยลดกิจกรรมของการใช้แรงงานลงอย่างเห็นได้ชัด เนื่องจากการควบคุมปัจจัยต่าง ๆ ของฟาร์มต้นแบบนั้น มีการตรวจสอบและจัดการได้เองโดยอัตโนมัติ ไม่ต้องอาศัยการใช้แรงงานของมนุษย์อย่างที่เคยทำมาก่อน ที่จะต้องมีช่วงเวลาในการรอและไม่สามารถจัดการกับปัญหาที่เกิดขึ้นในทันที รวมถึงการที่ระบบสามารถตั้งโปรแกรมการทำงานและเงื่อนไขของการควบคุมภายในได้ จึงทำให้ผลผลิตมีแนวโน้มที่เพิ่มสูงขึ้น มีคุณภาพขึ้น ในขณะที่มีการลดการใช้แรงงาน การใช้พลังงาน และการใช้งานทรัพยากรอื่น ๆ ลง ทั้งนี้ในการทำงานของระบบยังมีความปลอดภัยเพิ่มขึ้น เนื่องจากมีระบบการตรวจสอบการเปิด-ปิดประตูเพื่อป้องกันการบุกรุก และการตรวจสอบการเกิดอัคคีภัย อันอาจจะสร้างความสูญเสียในระหว่างการผลิตได้ นอกจากนี้ระบบที่พัฒนาขึ้นยังเชื่อมต่อกับโมดูล GSM ซึ่งเป็นโมดูลที่คอยเชื่อมต่อสื่อสารของระบบฟาร์มอัตโนมัติกับผู้ใช้ เพื่อคอยอำนวยความสะดวกให้กับเจ้าของฟาร์มในการเข้าถึงระบบและควบคุมการทำงานได้จากระยะไกลผ่านทางสมาร์ตโฟนของตน

Roham และคณะ [43] ได้ศึกษาการประยุกต์ใช้เครือข่ายเซ็นเซอร์ไร้สาย (Wireless Sensor Network: WSN) มาใช้ในการทำงานด้านเกษตรกรรมของเกษตรกรชาวอินเดีย เพื่อพัฒนาต้นแบบของระบบควบคุมและจัดการในเชิงของการเกษตรแม่นยำ โดยนำอุปกรณ์เซ็นเซอร์ประเภทต่าง ๆ มาใช้ในการวัดสภาพภูมิอากาศ และค่าพารามิเตอร์ที่สำคัญ ในการตรวจสอบและการควบคุม



การผลิตพืชภายใต้ระบบโรงเรือนที่จำลองขึ้นเพื่อเป็น Testbed โดยที่เครือข่ายเซ็นเซอร์ไร้สายจะทำงานต่าง ๆ นี้ โดยอัตโนมัติ มีการวิเคราะห์ค่าพารามิเตอร์ต่าง ๆ เพื่อตัดสินใจ รวมถึงแสดงผลของการตรวจวัดออกมาให้ทราบ ซึ่งทางคณะผู้วิจัยได้ทำการพัฒนาให้สามารถใช้งานระบบได้บนระบบเว็บแอปพลิเคชัน และสมาร์ตโฟนแอปพลิเคชัน ซึ่งการควบคุมการทำงานระบบนี้ จะใช้ Beagle Bone ทำหน้าที่เป็นบอร์ดหลักในการควบคุมการทำงาน ซึ่งจะทำการสื่อสารกับเครือข่ายเซ็นเซอร์ไร้สายที่อยู่ในโรงเรือนด้วย ZigBee ซึ่งการศึกษานี้ค้นพบว่าอุณหภูมิ ความชื้น และคาร์บอนไดออกไซด์ เป็นพารามิเตอร์สำคัญที่มีความจำเป็นต่อการเจริญเติบโตของพืชภายในโรงเรือน ปัจจุบันเกษตรกรชาวอินเดียไม่ยังมีระบบใดที่จะใช้ในการตรวจวัดและแสดงผลของพารามิเตอร์ที่สำคัญทั้งสามนี้ ในลักษณะ Real-time แม้แต่เกษตรกร ที่มีประสบการณ์ก็ไม่อาจทราบได้ว่า เมื่อใดที่มีอุณหภูมิ ความชื้น หรือระดับของคาร์บอนไดออกไซด์ ที่เพิ่มหรือลดลงในโรงเรือนของเขา จนกว่าการผลิตพืชนั้นจะได้รับผลกระทบเสียก่อน ซึ่งระบบที่ถูกพัฒนาขึ้นนี้จะไปตรวจสอบพารามิเตอร์ที่สำคัญเหล่านี้ว่ามีการเปลี่ยนแปลงอย่างไรเป็นระยะ ๆ เพื่อบันทึกค่า และดำเนินการควบคุมโดยอัตโนมัติ รวมถึงแสดงค่าที่ตรวจวัดได้เป็นข้อมูลแบบสตรีมมิ่งแบบ Real-time ซึ่งต่อไปเกษตรกรจะสามารถติดตามและควบคุมการทำงานของอุปกรณ์ได้จากกระยะไกลโดยใช้สมาร์ตโฟน

## 2. งานวิจัยที่เกี่ยวข้องกับอินเทอร์เน็ตของสรรพสิ่ง

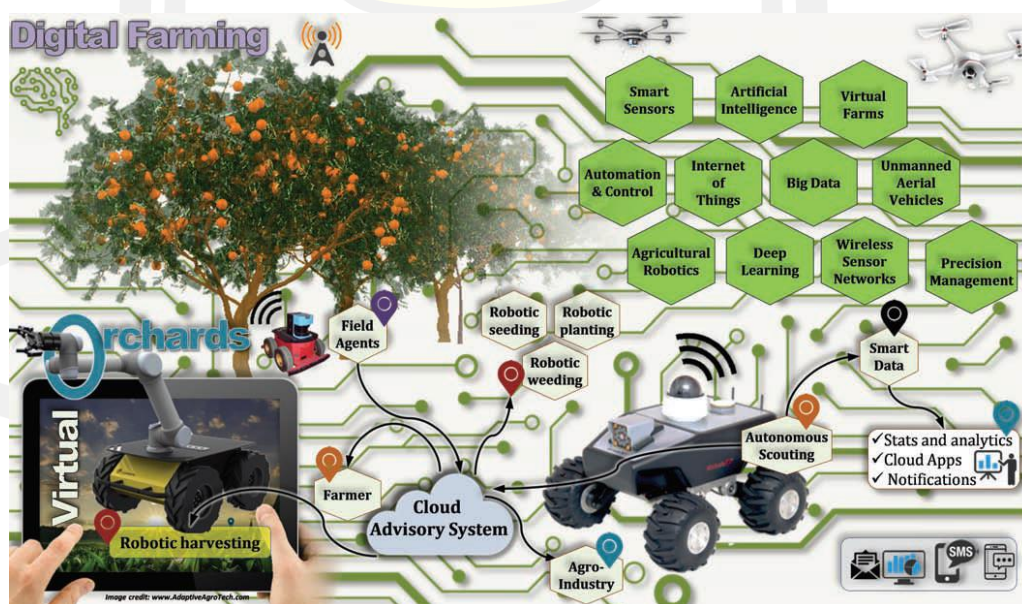
Boonchieng และคณะ [44] ได้พัฒนาต้นแบบฟาร์มเห็ดหลินจืออัจฉริยะที่เป็น Best Practice ของไทย เพื่อส่งเสริมการทำฟาร์มเห็ดหลินจือซึ่งเป็นพืชอุตสาหกรรมชนิดใหม่ โดยการประยุกต์ใช้ NodeMCU ในการวัดค่าและส่งค่าจากโรงเรือน ด้วย NETPIE ของ NECTEC ทั้งในการรับ-ส่งค่า การ Monitor ค่า ตลอดจนควบคุมการทำงานของระบบ เช่น การควบคุมการเปิด-ปิดของปั้มน้ำ และส่งข้อความแจ้งเตือนโดย Line API เครื่องมือที่ใช้ในการพัฒนาฮาร์ดแวร์ในการควบคุมและตรวจสอบค่าของระบบ คือ NodeMCU, Temperature Sensor, Humidity Sensor, Real Time Clock Module, Relay Module และปั้มน้ำ พัฒนาซอฟต์แวร์ของระบบ ด้วยภาษา C++ และ Node.JS และใช้ NETPIE ทำหน้าที่เป็น Protocol ทางด้านฝั่ง IoT โดยมีปัจจัยที่ถูกควบคุมในการวิจัย คือ แสง อุณหภูมิ ความชื้นสัมพัทธ์ และการถ่ายเทของอากาศภายในโรงเรือน ซึ่งใช้ค่าความชื้นสัมพัทธ์ของโรงเรือนเป็นเงื่อนไขในการควบคุมการเปิด-ปิดปั้มน้ำของระบบรดน้ำและพ่นหมอก ใช้ NodeMCU ในการเชื่อมต่อกับ NETPIE ผ่าน Wi-Fi เพื่อรับ-ส่ง บันทึกค่าบน Cloud และแสดงผล รวมถึงแจ้งเตือนผ่าน Line API ทั้งนี้การสร้างต้นแบบของฟาร์มเห็ดหลินจือจะต้องสามารถควบคุมความชื้นสัมพัทธ์ในสภาพแวดล้อมได้ (90-95%) ด้วยการควบคุมการเปิด-ปิดระบบน้ำ เพื่อที่จะประหยัดค่าใช้จ่ายและดูแลรักษาได้ง่าย มีการส่งข้อมูลไปเก็บบน Cloud พร้อมทั้งสามารถดูข้อมูลได้บนคอมพิวเตอร์และมือถือ ในการตรวจสอบการทำงานของระบบจะใช้ Thermometer แบบกระเปาะเปียกและกระเปาะแห้งวัดค่าเปรียบเทียบกับค่าที่อ่านได้จาก IoT กล้อง CCTV จะ

นำมาใช้ในการตรวจสอบการเปิดปิดของปั๊ม เทียบกับการแสดงผลที่ NETPIE FREEBOARD ผลการวิจัยและพัฒนาพบว่า 1) ทางด้านเศรษฐกิจ: ระบบควบคุมโรงเรือน (IoT Systems) ที่พัฒนาขึ้น มีต้นทุนและค่าใช้จ่ายในการบริหารจัดการต่ำกว่า เมื่อเปรียบเทียบกับการใช้แรงงานคน (Manpower) ถึง 10 เท่า 2) ทางด้าน Services: NETPIE FEED สามารถใช้งานได้ฟรี แต่มีข้อจำกัดคือ Writing Rate ต้องมีความเร็วมากกว่า 15 วินาที/ครั้ง และมี Reading Rate ไม่เกิน 5 ครั้ง ใน 4 วินาที และข้อมูลจะถูกเก็บไว้เพียง 1 ปี 3) ค่าความชื้นสัมพัทธ์ที่วัดได้จากระบบ IoT เป็นตัวเลขที่มีความละเอียด แตกต่างจากการวัดค่าด้วยเทอร์โมมิเตอร์แบบประปรายเป็ยก กระเปาะแห้งโดยคนที่อาจ error ขึ้น 4) การเปรียบเทียบการวัดค่า: มีการนำเอา Mean Absolute Percentage Error มาใช้คำนวณ ซึ่งมีการนำค่าที่อ่านได้โดยระบบ IoT และมนุษย์มาทำการเปรียบเทียบกัน โดยค่าเฉลี่ยที่คำนวณได้ มีค่า 2.2% ซึ่งมีความน่าเชื่อถือที่  $\alpha = .05$  ( Sampling Data ของค่าความชื้นสัมพัทธ์ จะมี 4 period ใน 1 วัน ตามช่วงเวลา) และงานวิจัยในอนาคตจะเป็นการปรับปรุงระบบ IoT ที่ใช้กับโรงเรือน 1) ใช้ SD Card ในการบันทึกข้อมูล เมื่อไม่ได้เชื่อมต่ออินเทอร์เน็ต 2) ใช้ Solar Cell มาใช้เพื่อให้มีความสะดวกและใช้งานง่ายเมื่อแยกโรงเรือนออกจากกัน 3) ใช้ระบบการสื่อสาร GSM เมื่อไม่สามารถเชื่อมต่อ Wi-Fi 4) ทำการทดสอบเปรียบเทียบระบบโรงเรือน IoT กับโรงเรือนเห็ดหลินจือที่ทำการเพาะปลูกแบบปกติ

### 3. งานวิจัยที่เกี่ยวข้องกับการออกแบบและพัฒนาหุ่นยนต์

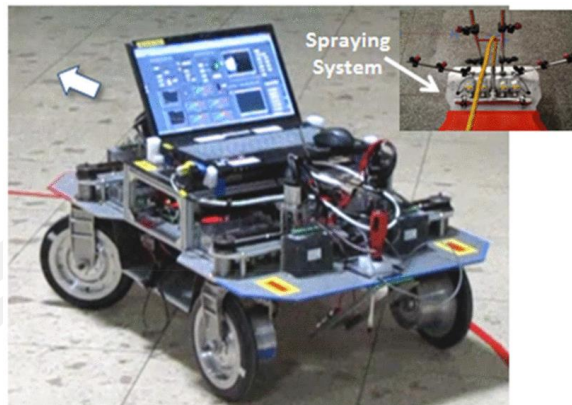
Shamshiri และคณะ [27] ทำการศึกษาการทำฟาร์มดิจิทัล (Digital Farming) ซึ่งเป็นการใช้เทคโนโลยีสมัยใหม่อย่าง เซ็นเซอร์ หุ่นยนต์ และการวิเคราะห์ข้อมูล เพื่อปรับเปลี่ยนกระบวนการที่ยุ่งยากในภาคการเกษตร ไปเป็นกระบวนการอัตโนมัติที่มีความต่อเนื่อง เพื่อให้ได้ผลผลิตที่มากขึ้น มีคุณภาพสูง ลดค่าใช้จ่าย และลดการใช้แรงงานมนุษย์ลง โดยได้อธิบายถึงความสำเร็จบางส่วนในการวิจัยและพัฒนาเพื่อนำหุ่นยนต์ไปใช้ในด้านเกษตรกรรม (Agricultural Robotics) โดยเฉพาะอย่างยิ่งหุ่นยนต์การเกษตรที่ใช้สำหรับ การควบคุมวัชพืช (Weed Control Robot) การสำรวจพื้นที่ภาคสนาม (Field Scouting Robo) และการเก็บเกี่ยว (Harvesting Robot) ซึ่งในการ Implement ของ Digital Farming และ Site-Specific Precision นั้น พบว่า Main Point ของการศึกษาชี้ว่า สิ่งนี้ไม่ได้ขึ้นอยู่กับเทคโนโลยีของเซ็นเซอร์เท่านั้น แต่ยังรวมไปถึงข้อมูลที่ถูกต้องแม่นยำซึ่งได้จากภาคสนามอย่างต่อเนื่อง ซึ่งสามารถทำได้โดยการใช้ประโยชน์จากหุ่นยนต์การเกษตร ซึ่ง การระบุวัตถุ (Object Identification) อัลกอริทึมในการวางแผนงาน (Task Planning Algorithms) การแปลงกระบวนการทำงานไปเป็นข้อมูลดิจิทัล (Digitalization) ตลอดจนการเพิ่มประสิทธิภาพของเซ็นเซอร์ (Sensor Optimization) จะถูกนำไปเป็นส่วนหนึ่งของความท้าทายที่ในบริบทของการเพาะปลูกแบบฟาร์มดิจิทัล นอกจากนี้ การศึกษายังพบแนวคิดของการทำงานของหุ่นยนต์หลายตัว การทำงานร่วมกันของมนุษย์กับหุ่นยนต์ การฟื้นฟูสภาพแวดล้อมจากภาพทาง

อากาศ และเซ็นเซอร์บนพื้นดินเพื่อสร้างฟาร์มเสมือนถูกเน้นเป็นเกตเวย์ของการทำฟาร์มดิจิทัล แสดงให้เห็นว่าหนึ่งในแนวโน้มและทิศทางการวิจัยที่มุ่งเน้นไปที่หุ่นยนต์ภาคสนามทางการเกษตร นั่นคือการสร้างกลุ่มของหุ่นยนต์ขนาดเล็กและโดรน ที่ทำงานร่วมกัน เพื่อเพิ่มประสิทธิภาพการผลิตของเกษตรกรและเปิดเผยข้อมูลที่ถูกละเลยหรือปกปิดจากการเกษตรในรูปแบบดั้งเดิม ส่วนกรณีการศึกษาการเก็บเกี่ยวด้วยหุ่นยนต์นั้น พบว่ามีโครงสร้างที่เป็นลักษณะแบบอิสระ (Autonomous Framework) ที่มีตัวจัดการแบบเรียบง่ายหลายตัว ซึ่งสามารถทำงานได้รวดเร็วและมีประสิทธิภาพมากกว่าการจ้างแรงงานที่มีประสบการณ์และมีค่าตอบแทนที่สูง ถึงแม้ว่าในขณะที่หุ่นยนต์การเกษตรกำลังกลายเป็นส่วนหนึ่งของฟาร์มยุคใหม่ (Modern Farms) จนไม่สามารถแยกออกจากกันได้นั้น อย่างไรก็ตาม ข้อสรุปของการศึกษานี้กลับชี้ว่า การคาดการณ์ตามสมมติฐานจะไม่ใช่ไปตามความเป็นที่จริงว่า ระบบของฟาร์มที่ใช้หุ่นยนต์การเกษตรนั้น จะสามารถทำงานได้อย่างอัตโนมัติทั้งระบบในอนาคต เนื่องจากการทำเกษตรกรรมในเชิงอุตสาหกรรมนั้น ต้องการความเป็นระบบอัตโนมัติที่ทำงานได้ด้วยตนเอง (Full Automation) ทั้งระบบ ตั้งแต่เริ่มการเพาะปลูก ดูแล กำจัดศัตรูพืช จนถึงกระบวนการเกี่ยวเกี่ยวผลผลิต ซึ่งจากการงานวิจัยและพัฒนาที่การศึกษานี้ได้นำเสนอต่างอยู่ในช่วงของการพัฒนาต้นแบบ (Prototype Phase) ซึ่งการพัฒนานี้ถูกออกแบบมาให้ทำงานต่าง ๆ ในฟาร์ม แต่ก็ยังเป็นเพียงการทำงานเฉพาะด้าน เพียงด้านใดด้านหนึ่งเท่านั้น ไม่ได้ครบถ้วนตลอดทั้งกระบวนการ อีกทั้ง ยังมีความจำเป็นในบางงานที่ยังต้องใช้การมีปฏิสัมพันธ์ระหว่างหุ่นยนต์กับมนุษย์เพื่อควบคุมการทำงานให้มีประสิทธิภาพสูงสุดอีกด้วย ดังภาพที่ 2.46



ภาพที่ 2.46 แนวความคิดเกี่ยวกับการเกษตรแบบดิจิทัลโดยเน้นบทบาทของหุ่นยนต์การเกษตร

Ko และคณะ [45] ได้นำเสนอแนวทางระบบของการออกแบบและพัฒนาแพลตฟอร์มหุ่นยนต์เคลื่อนที่แบบ 4WD ดังภาพที่ 2.47 ซึ่งมีการรับรู้พารของเส้นทางโดยใช้ระบบกล้องเดี่ยวและการประมวลผลภาพ สำหรับการประยุกต์ใช้ในงานด้านการเกษตร เช่นเดียวกับหุ่นยนต์ภาคสนามอื่น ๆ แพลตฟอร์มของหุ่นยนต์เคลื่อนที่สำหรับการใช้งานทางด้านการเกษตรต้องใช้ชุดแอตทริบิวต์การควบคุมที่แตกต่างซึ่งถูกกำหนดไว้ล่วงหน้า คณะผู้วิจัยได้ออกแบบ พัฒนา และทำการทดสอบเพื่อแสดงให้เห็นถึงแพลตฟอร์มหุ่นยนต์เคลื่อนที่สำหรับการประยุกต์ใช้ในการฉีดพ่นสารเคมีกำจัดศัตรูพืช การออกแบบและการพัฒนารวมถึงการผสมผสานการทำงานของระบบกลไก เซ็นเซอร์ การควบคุม อุปกรณ์อิเล็กทรอนิกส์ และซอฟต์แวร์ โดยภายในงานวิจัยนี้ประกอบด้วย 1) ข้อกำหนดทางกลไกของหุ่นยนต์ที่ประกอบด้วยการออกแบบโดยใช้คอมพิวเตอร์และการเพิ่มประสิทธิภาพของโครงสร้างทางกลและล้อ 2) การกำหนดสูตรของรูปแบบพารในการเคลื่อนที่ 3) การพัฒนาอัลกอริทึมระบบนำทางจากข้อมูลที่ได้จากการสังเกต/การศึกษา 4) การพัฒนาอัลกอริทึมที่ใช้ตัวกรอง Kalman Filter เพื่อขจัด Noises 5) การรวมการทำงานของเซ็นเซอร์และตัวกระตุ้นโดยใช้ DCS (Distributed Control Scheme) โดยการพัฒนาโค้ดสำหรับจัดการเซ็นเซอร์และตัวกระตุ้น 6) บูรณาการการทำงานของซอฟต์แวร์และการออกแบบส่วนติดต่อผู้ใช้ และ 7) การพัฒนาสถานการณ์ในการทดสอบสำหรับการทดลองและการดำเนินงานขั้นสุดท้าย ซึ่งการพัฒนาระบบการนำทางแบบอัตโนมัติในครั้งนี้ประสบความสำเร็จใน 3 ขั้นตอน ได้แก่ ขั้นตอนการเรียนรู้ (Learning State) ขั้นตอนการฝึก (Training State) และ ขั้นตอนการทดสอบ (Testing State) โดยในขั้นตอนการเรียนรู้ จะกำหนดรูปแบบเส้นทาง (Path Patterns) ศึกษาและบันทึกพฤติกรรมของยานพาหนะในสภาพแวดล้อมจริง ในขั้นตอนการฝึกนั้น อัลกอริทึมต่าง ๆ สำหรับระบบการขับเคลื่อนสี่ล้อ จะได้รับการพัฒนาและแก้ไขข้อผิดพลาดโดยการใช้เครื่องมือและวิธีการขั้นสูงด้วยระบบการประมวลผลภาพซึ่งใช้ระบบกล้องเดี่ยวนำไปปรับอัลกอริทึมเพื่อควบคุมจลนศาสตร์ของการเคลื่อนที่ในหุ่นยนต์ ดังภาพที่ 2.48 ผ่านปัจจัยที่มีความสำคัญ คือ มุมองศาของกล้อง (Angle of the Camera) ความเร็วในการเคลื่อนที่ (Driving Speed: SV) และ มุมของการเลี้ยวต่อพิกเซล (Steering Angle per Pixel) ซึ่งถูกจัดการอย่างเหมาะสม เพื่อขับเคลื่อนหุ่นยนต์โดยอัตโนมัติบนเส้นทางที่มีการกำหนดไว้ และในขั้นตอนการทดสอบที่นำแพลตฟอร์มหุ่นยนต์ทดสอบในรูปแบบเส้นทางที่ไม่ได้มีการกำหนดรูปแบบเอาไว้ และผลการทดลองแสดงให้เห็นว่า หุ่นยนต์สามารถเดินทางในแบบอัตโนมัติได้สำเร็จด้วยตัวเอง ซึ่งคณะผู้วิจัยกล่าวว่า หุ่นยนต์เคลื่อนที่ขนาดกลางนี้จะสามารถนำไปพัฒนาในเชิงพาณิชย์เพื่อใช้สำหรับการเกษตรในโรงเรือนได้อย่างมีประสิทธิภาพต่อไป



ภาพที่ 2.47 ต้นแบบการพัฒนาแพลตฟอร์มหุ่นยนต์เคลื่อนที่แบบ 4WD

ที่มา: [45]

SCHEMATIC DRAWING	RELATION BETWEEN A RADIUS OF ROTATION AND STEERING ANGLE
	$\sin \theta_F = L/R, \theta_F = \sin^{-1}(L/R) \quad (1)$ $\cos \theta_F = r/R \quad (2)$ $r = R \cos \theta_F = R \cos(\sin^{-1}(L/R)) \quad (3)$ $\tan \theta_{F\_R} = 1/(r - T/2) \quad (4)$ $\tan \theta_{F\_L} = 1/(r + T/2) \quad (5)$ $\theta_{F\_R} = \tan^{-1}\left(\frac{1}{r - (T/2)}\right) \quad (6)$ $\theta_{F\_L} = \tan^{-1}\left(\frac{1}{r + (T/2)}\right) \quad (7)$
	$\tan \theta_{R\_R} = 1/(R - T/2) \quad (8)$ $\tan \theta_{R\_L} \cong \theta_{R\_R} \quad (9)$ $\tan \theta_{R\_L} = 1/(R + T/2) \quad (10)$ $\tan \theta_{R\_R} \cong \theta_{R\_L} \quad (11)$ $\theta_{F\_R} = \tan^{-1}\left(\frac{1}{R - (T/2)}\right) \quad (12)$ $\theta_{F\_L} = \tan^{-1}\left(\frac{1}{R + (T/2)}\right) \quad (13)$
	$\sin \theta_F = L/2R \quad (14)$ $\theta_F = \sin^{-1}(L/2R) \quad (15)$ $\theta_F = \theta_R \quad (16)$ $\cos \theta_F = r \quad (17)$ $r = R \cos(\sin^{-1}(L/2R)) \quad (18)$ $\tan \theta_{F\_R} = \frac{L/2}{(r - T/2)} \quad (19)$ $\tan \theta_{F\_L} = \frac{L/2}{(r + T/2)} \quad (20)$
$\tan \theta_{F\_R} = \frac{L/2}{R \cos(\sin^{-1}(L/2R)) - T/2} \quad (22)$ $\tan \theta_{F\_L} = \frac{L/2}{R \cos(\sin^{-1}(L/2R)) + T/2} \quad (23)$ $\theta_{F\_R} = \theta_{R\_R}, \theta_{F\_L} = \theta_{R\_L} \quad (24)$	$\theta_{F\_R} = \tan^{-1}\left(\frac{L/2}{R \cos(\sin^{-1}(L/2R)) - T/2}\right) \quad (22)$ $\theta_{F\_L} = \tan^{-1}\left(\frac{L/2}{R \cos(\sin^{-1}(L/2R)) + T/2}\right) \quad (23)$

ภาพที่ 2.48 จลนศาสตร์ของกลไกการเคลื่อนที่ของหุ่นยนต์ 4WD

ที่มา: [45]

#### 4. งานวิจัยที่เกี่ยวข้องกับหุ่นยนต์เพาะปลูกพืช

Neha และคณะ [46] ได้ทำการสร้างและทดสอบการทำงานของระบบหุ่นยนต์หยอดเมล็ด Agribot จากบอร์ดไมโครคอนโทรลเลอร์ Arm 7 โดยใช้หลักการออกแบบบน Open Loop Control System โดยหุ่นยนต์จะถูกติดตั้ง DC Motor จำนวน 4 ชุด สำหรับใช้เป็นล้อในการขับเคลื่อน มี DC Motor อีก 2 ชุด เพื่อใช้ในการควบคุมการเคลื่อนที่ของ V-Shaped Arm ชุดเซ็นเซอร์ IR เพื่อนำทางในแปลงทดสอบ พร้อมทั้งติดตั้งปั้มน้ำและตัวหยอดเมล็ด ต้นแบบหุ่นยนต์นี้ใช้ในการเพาะพันธุ์พืชแบบอัตโนมัติใน 4 ประเภท ได้แก่ ฝ้าย ข้าวโพด ถั่วเหลือง และข้าวสาลี ซึ่งได้ค้นพบว่าวิธีดังกล่าว มีประสิทธิภาพและมีความถูกต้องแม่นยำสูงกว่ารูปแบบวิธีการหว่านเมล็ดพันธุ์แบบดั้งเดิมโดยมนุษย์ ซึ่งระบบมีปัจจัยที่สำคัญซึ่งเป็นระยะของการหยอดเมล็ดที่ใช้ในการศึกษา มีทั้งหมดอยู่ 3 ข้อ ประกอบด้วย ระยะในการหว่านเมล็ด ความลึกในการหยอดเมล็ด และระยะทางระหว่างแถวของพืช (ระยะของแถวกับคอคลัมน์) ซึ่งทดสอบบนพื้นดินเปียก โดยกำหนดระยะทางการเคลื่อนที่ของหุ่นยนต์เอาไว้ล่วงหน้า โดยเริ่มจากระยะ 4 เซนติเมตร ไป 8 เซนติเมตร และจาก 2 เซนติเมตร ไป 3 เซนติเมตร ตามลำดับ ตามค่าทางทฤษฎี ซึ่งจะต้องใช้ระยะห่างแถวและคอคลัมน์สำหรับพืชชนิดนี้ในระบบด้วย เนื่องจากระยะห่างเพียงไม่กี่เซนติเมตรในระยะที่กำหนดหุ่นยนต์จะหยอดเมล็ดครอบคลุมระยะระหว่างพืชผลได้อย่างมีประสิทธิภาพ ผลของระยะในการหยอดเมล็ดพันธุ์ที่ถูกกระทำโดย Agribot จะจัดเรียงเป็นในลักษณะของตาราง นอกจากนี้ Agribot ยังสามารถตรวจวัดค่าพารามิเตอร์ที่อยู่บนพื้นที่เพาะปลูกได้อีกด้วย ซึ่งการทำงานของหุ่นยนต์มีความแม่นยำในการทำงาน ใช้งานง่าย มีความซับซ้อนน้อยกว่าในการออกแบบทางกลและทางเทคนิค ด้วยการออกแบบระบบนำทางที่เรียบง่ายทำให้หุ่นยนต์มีต้นทุนต่ำและมีขนาดเล็กลง เมื่อเทียบกับรถแทรกเตอร์ธรรมดา ต้นแบบนี้อาจจะยังมีข้อบกพร่องอยู่บ้าง แต่ก็สามารถจะแก้ไขได้ด้วยการออกแบบระบบ Mechatronic และเพิ่มระบบการควบคุมเพิ่มเติมใหม่

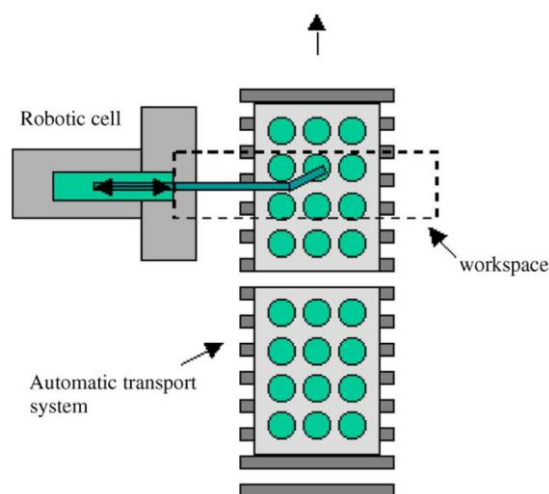
Belforte และคณะ [47] ได้พัฒนาหุ่นยนต์อัตโนมัติที่ใช้ฉีดพ่นสารเคมีในการทำการเกษตรภายในโรงเรือน ถูกสร้างขึ้นภายใต้กรอบความร่วมมือของทีมนักวิจัยชาวอิตาลี จาก Dipartimento di Economia e Ingegneria Agraria Forestale e Ambientale ของมหาวิทยาลัย Universita' degli Studi di Torino, ห้องปฏิบัติการ IMAMOTER ของ Consiglio Nazionale delle Ricerche และ Dipartimento di Automatica e Informatica ของ Politecnico di Torino โดยสร้างหุ่นยนต์ต้นแบบให้มีความเรียบง่าย ด้วยวัสดุที่มีมาตรฐานในท้องตลาด เพื่อลดต้นทุน แต่ยังคงคุณสมบัติเพียงพอที่จะสนับสนุนการวิจัยได้ การออกแบบกลไกและจลนศาสตร์ของหุ่นยนต์ จึงประกอบด้วย 3 องศาอิสระ (Degrees of Freedom: DOF) ที่เป็นเสปคขั้นต่ำของข้อกำหนดสำหรับการทดสอบ ให้สามารถติดตั้ง End-Effectors หรือ Tools อื่นที่แตกต่างกันได้ เพื่อเพิ่ม DOF ของระบบ ดังภาพที่ 2.49 ประกอบด้วย 3 ข้อต่อ ที่ขับเคลื่อนด้วยมอเตอร์กระแสตรง

ขนาด 12 โวลต์ ควบคุมด้วย Pulse Width Modulation (PWD) หน่วยควบคุมใช้บอร์ด PXI-8170 และ PXI-7344 ของบริษัท National Instrument โดยใช้ LabView 6 ร่วมกับ IMAQ และ Flexmotion Libraries เพื่อพัฒนาโค้ดควบคุม มีระบบ JAI S3200 ที่ติดตั้งบนข้อต่อตัวที่ 2 เป็นระบบการมองเห็น (Simple Vision System) เพื่อความถูกต้องในการทำงานภายใต้สภาพแวดล้อม ซึ่งหน้าที่ตรวจสอบและการตรวจนับวัตถุ ปริภูมิของพื้นที่ทำงานที่สามารถเข้าถึงได้ จะประมาณพื้นที่ด้วยกรอบรูปสี่เหลี่ยม ดังภาพที่ 2.50 โดยหุ่นยนต์จะไม่มีเคลื่อนที่ เนื่องจากไม่มีความจำเป็นต่อการทดลองและเพิ่มเติมความสามารถได้ในภายหลัง แต่แขนหุ่นยนต์จะมีการเคลื่อนที่ตามแนวตั้งฉากกับการเคลื่อนที่ทิศทางของพาเลทที่จัดวางกระถางต้นไม้ ในการวิจัยได้กำหนดการทดลองออกเป็น 2 งาน ประกอบด้วย 1) การฉีดยุ่อย่างแม่นยำ เพราะต้องฉีดยุ่ใต้ใบ ซึ่งเป็นที่อาศัยของศัตรูพืช ต้องมีการวางตำแหน่งของหัวฉีดยุ่ให้ถูกต้องและฉีดยุ่สารเคมีให้เป็นละอองฝอยเพื่อให้เกิดประสิทธิภาพ 2) การใส่ปุ๋ยอย่างแม่นยำลงในกระถาง มีการกำหนดปริมาณเม็ดปุ๋ยไว้ล่วงหน้า เพื่อให้เหมาะสมกับความต้องการเฉพาะของพืช ซึ่งการวิจัยได้ดำเนินการที่โรงเรียนของ Centro Regionale di Sperimentazione e Assistenza Agricola (CeRSAA) ตั้งอยู่ที่ Albenga ประเทศอิตาลี ซึ่งพบว่า หุ่นยนต์ต้นแบบ มีความเร็วในการฉีดยุ่สารเคมีที่ 7 ต้น/นาที และอัตราการใส่ปุ๋ยที่ 8 กระถาง/นาที โดยเฉลี่ยนั้น หุ่นยนต์ต้นแบบสามารถทำความเร็วได้ประมาณ 400 – 500 ต้น/ชั่วโมง (ยังไม่เพียงพอต่อการผลิตในเชิงพาณิชย์) ซึ่งสามารถเพิ่มประสิทธิภาพได้โดยการพัฒนาหุ่นยนต์ในทางวิศวกรรม



ภาพที่ 2.49 หุ่นยนต์ต้นแบบ CeRSAA กำลังทำงานภายในโรงเรือน

ที่มา: [47]



ภาพที่ 2.50 เลย์เอาท์และปฏิภูมิการทำงานของหุ่นยนต์

ที่มา: [47]

Batista และคณะ [48] ได้ทำการพัฒนาหุ่นยนต์ 3-dimensional แบบพิกัดคาร์ทีเซียน (Cartesian Robot) ต้นทุนต่ำ ดังภาพที่ 2.51 ประกอบด้วยชุดเซ็นเซอร์และไมโครคอนโทรลเลอร์ Arduino Mega 2560 ซึ่งพัฒนาบนภาษา Python มีโครงสร้างหลักเป็นโลหะ ขนาด 50 x 20 มิลลิเมตรหนา 1.20 มิลลิเมตร ส่วนควบคุมการเคลื่อนไหวในแนวแกน  $x$ ,  $y$ ,  $z$  ประกอบด้วย DC Motor ที่เชื่อมต่อกับเพลา เฟืองตัวหนอน และชุดเกียร์พลาสติก การเคลื่อนไหวถูกควบคุมโดยอาศัยข้อมูลจากเซ็นเซอร์และอุปกรณ์อิเล็กทรอนิกส์ มีการเก็บข้อมูลลงในหน่วยความจำ Secure Digital Card (SD Card) เมื่อหุ่นยนต์เคลื่อนที่ไปถึงจุดที่กำหนดไว้ก็จะทำการปล่อยน้ำออกมาเพื่อรดน้ำต้นไม้ โดยการวิจัยครั้งนี้มีจุดประสงค์เพื่อพัฒนาหุ่นยนต์ต้นทุนต่ำใช้ภายในโรงเรือนของเกษตรกรที่อยู่ในชนบทของประเทศบราซิล เนื่องจากปัญหาด้านต้นทุนเครื่องจักรกลทางการเกษตรที่มีราคาสูงมาก จนเกษตรกรรายย่อยไม่สามารถซื้อมาใช้ได้ ประกอบกับเกิดการอพยพย้ายถิ่นของแรงงานในภาคการเกษตร อันเนื่องมาจากงานด้านการเกษตรมีความยากลำบากและไม่น่าสนใจ ดังนั้นในการแก้ไขปัญหา คณะผู้วิจัยจึงได้นำเทคโนโลยีที่มีคุณภาพ ทันสมัย แต่มีต้นทุนในการพัฒนาที่ต่ำและสามารถหาวัสดุอุปกรณ์ได้ภายในท้องถิ่น เพื่อมาช่วยเพิ่มประสิทธิภาพในการผลิตของการเกษตรทั้งด้านของปริมาณและคุณภาพ โดยต้นแบบนั้นถูกติดตั้งที่ศูนย์ Agricultural Science Center ของมหาวิทยาลัย Universidade Federal do Ceará ใน Fortaleza ประเทศบราซิล ในการขยายผลไปสู่เกษตรกรนั้น คณะผู้วิจัยได้ทำการฝึกอบรมเกี่ยวกับการใช้งาน การดูแลรักษา และการแก้ไขปัญหาความผิดพลาดเบื้องต้นของหุ่นยนต์ โดยเกษตรกรไม่จำเป็นจะต้องมีความรู้ทางด้านเทคนิคขั้นสูง ซึ่งพบว่าหุ่นยนต์ที่ได้พัฒนาขึ้นนี้มีต้นทุนในการสร้างที่ต่ำ (US\$ 538) สามารถหาอุปกรณ์ได้ง่ายสามารถทำงานได้จริง เข้ามาช่วยในงานที่ต้องทำซ้ำเป็นประจำ ช่วยลดความเครียดของเกษตรกร อีก



ทั้งนี้มีผลกระทบทางสังคมด้านบวก เนื่องจากช่วยลดปัญหาแรงงานขาดแคลนและลดงานที่ทำให้เกิดปัญหาทางสุขภาพของเกษตรกรลง



ภาพที่ 2.51 หุ่นยนต์ RIRRIG ของ Universidade Federal do Ceará

ที่มา: [48]

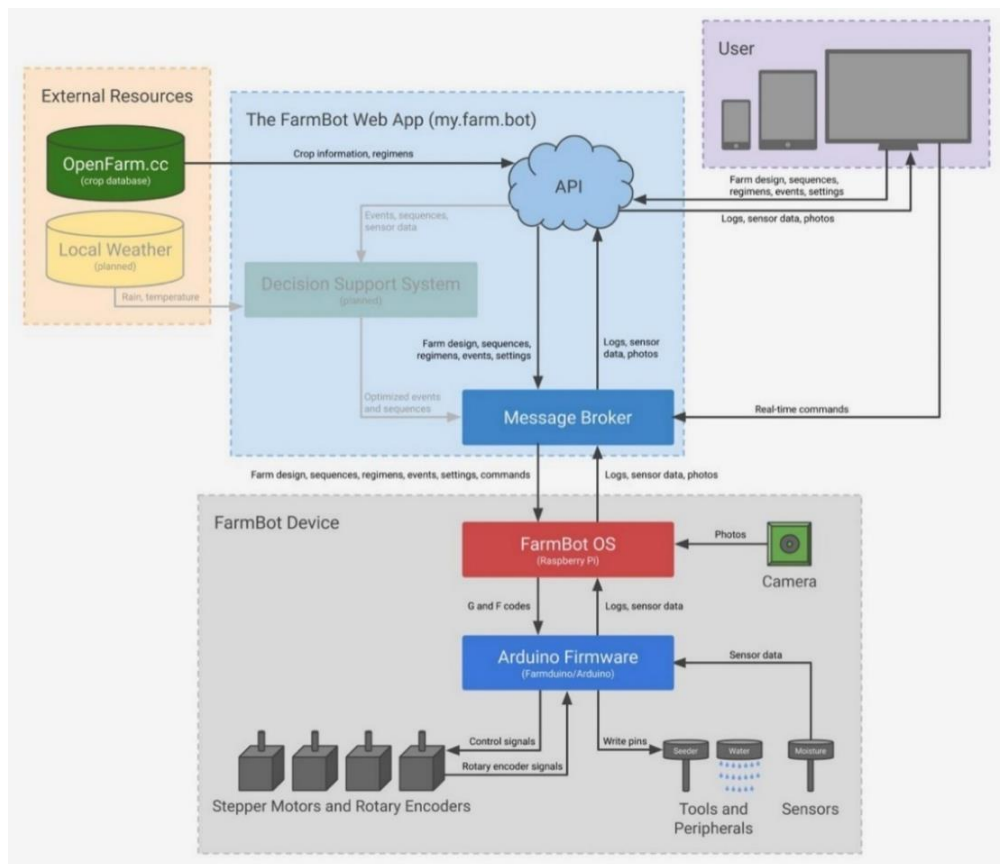
Aronson [49] ได้ทำการพัฒนา FarmBot ดังภาพที่ 2.52 เพื่อตอบสนองความต้องการในการทำฟาร์มอัตโนมัติขนาดเล็ก ที่สามารถทำการวางแผนการเพาะปลูกได้ล่วงหน้า ในรูปแบบ Open Source Farming Platform ซึ่งทำงานในลักษณะ Computer Numerical Controlled (CNC) สามารถเคลื่อนที่ได้ทั้งแกน  $x$ ,  $y$  และ  $z$  เพื่อควบคุมการปลูก รดน้ำ กำจัดวัชพืช และตรวจสอบพื้นที่ โดยอาศัยการเคลื่อนที่เชิงเส้นแบบ Linear Motion และ Motor Encoder ในการควบคุม Navigation และ Localization ในขณะที่เคลื่อนที่อยู่บนราง (Tracks) และโครง (Gantry) สำหรับการทำงานของ FarmBot นั้น ทางด้านฮาร์ดแวร์จะมี Raspberry Pi เป็นชุดทำงานหลัก ซึ่งติดตั้งระบบปฏิบัติการ FarmBot OS เพื่อทำการประมวลผลและตัดสินใจ ตลอดจนควบคุมการทำงานของชุดเซ็นเซอร์และมอเตอร์ผ่านบอร์ด Arduino ที่มีการเขียน Firmware ขึ้นมาทำงาน โดยเฉพาะดังแสดงในภาพที่ 2.53 ต้นแบบของการพัฒนานี้ใช้ชื่อว่า “FarmBot Genesis” มีปริภูมิการทำงานขนาด  $1.5 \times 3$  เมตร ซึ่งกำหนดเป็นพื้นที่เพาะปลูกได้สูงสุด 4.5 ตารางเมตร รองรับความสูงของต้นพืชได้ไม่เกิน 1 เมตร โครงสร้างหลัก FarmBot ถูกประกอบจากอะลูมิเนียมโพรไฟล์และพลาสติกขึ้นรูปโดยเครื่องพิมพ์สามมิติ ติดตั้งบนแปลงปลูกที่ถูกติดกรอบด้วยแผ่นไม้ให้มีลักษณะเป็นสี่เหลี่ยมผืนผ้าบนพื้นราบ โดยให้ระดับของดินจะอยู่ต่ำกว่ารางของ FarmBot เล็กน้อย ซึ่งอุปกรณ์ส่วนปลายที่ติดตั้งบน End-Effectors ใช้เครื่องมือ 3 ชนิด เพื่อใช้ในการทำงานที่แตกต่างกัน

ประกอบด้วย ชุดหัวปลูก (Seed Injector) ชุดหัวรดน้ำ (Watering Nozzle) และชุดหัวกำจัดวัชพืช (Cutter/Shredder) ส่วนด้านซอฟต์แวร์นั้น มีการทำงานแบบ Web Application ที่รันบน Cloud Service ถูกพัฒนาขึ้นให้ช่วยในการควบคุมและปรับแต่งค่าการทำงานของ FarmBot ผ่านทาง Web Browser ซึ่งมี Interface แบบ Drag-and-Drop ในการเลือกเมล็ดพืชและการจัดตำแหน่งการปลูก ตลอดจนควบคุมการทำงานอื่น ๆ ซึ่งทำงานบน Graphic User Interface (GUI) ปัจจุบัน FarmBot ถูกพัฒนาไปใช้งานได้จริงในเชิงการค้า โดยจัดจำหน่ายในชื่อ “Humanity ‘s Open-Source CNC Farming Machine” ถูกปรับปรุงให้มีปริภูมิการทำงานที่ใหญ่ขึ้นเป็น 3 x 6 เมตร รองรับพื้นที่การเพาะปลูกพืชได้ถึง 18 ตารางเมตร ที่ระดับความสูงของพืชไม่เกิน 1 เมตร ซึ่งแม้ว่า FarmBot จะมีการทำงานที่มีประสิทธิภาพและสามารถทำหน้าที่ตั้งแต่การปลูก รดน้ำ และดูแลพืช แต่อย่างไรก็ตามยังพบว่ามีข้อจำกัดในด้านเวลาและการสูญเสียพลังงานที่เกิดขึ้นในระหว่างการทำงาน โดยเกิดจากการที่จะต้องเคลื่อนย้ายส่วนของ End Effector ไปที่ Tools Head Bay เพื่อทำการเปลี่ยนเครื่องมือในการทำงานทุกครั้งที่สลับหน้าที่ของอุปกรณ์ส่วนปลาย ซึ่งเป็นการเคลื่อนที่โดยไม่จำเป็น เพราะจะต้องเคลื่อนย้ายกลับมายังจุดอ้างอิงเริ่มต้นอีกครั้งแทนที่จะสามารถทำงานต่อเนื่องไปได้เลย นอกจากนี้ยังมีข้อจำกัดด้านการขยายตัวของระบบ (Scalability) เมื่อมีการเปลี่ยนแปลงขนาดของฟาร์ม รวมถึงยังต้องกำหนดจุดของ Motion Guides ในพื้นที่ของฟาร์มเพื่อใช้ในการควบคุมการทำงาน ด้วยเหตุนี้ในการติดตั้งระบบจึงต้องการพื้นที่ที่มีความเรียบและได้ระดับ เพื่อวางโครงสร้าง Motion Guides ให้มีการทำงานที่ถูกต้อง แม่นยำ และมีประสิทธิภาพ



ภาพที่ 2.52 ต้นแบบของ FarmBot

ที่มา : [50]

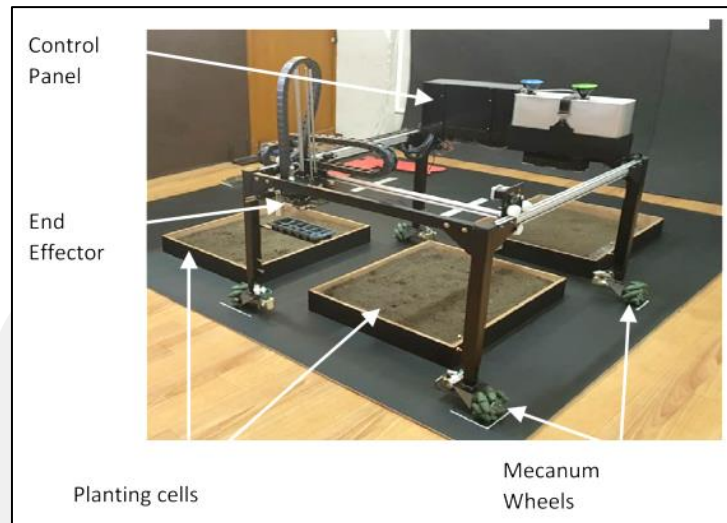


ภาพที่ 2.53 ภาพรวมของ FarmBot Software ในระดับ High Level

ที่มา : [51]

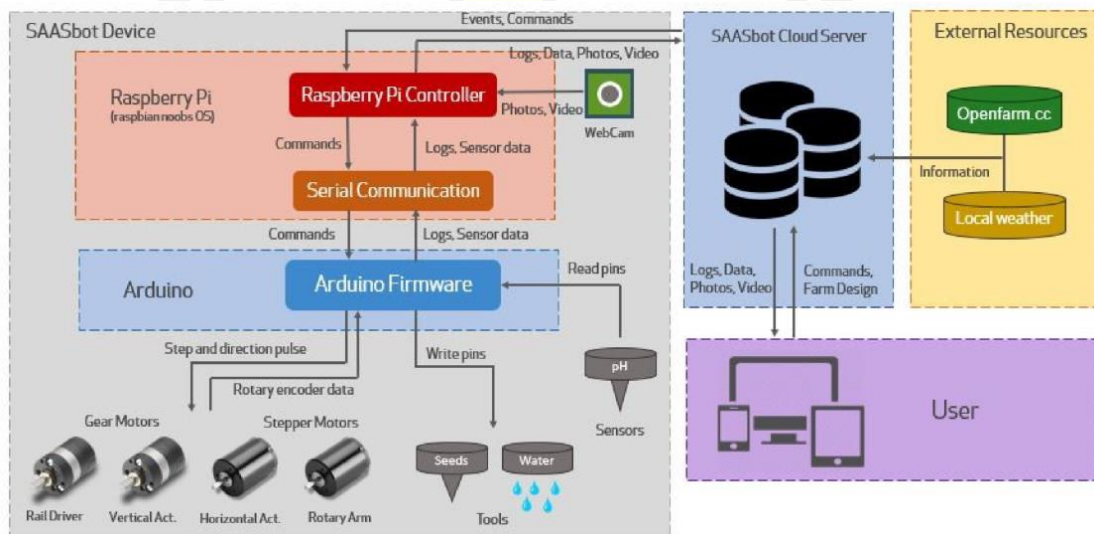
Athukorala และคณะ [52] ได้ทำการศึกษาปัญหาที่เกิดขึ้นกับภาคการเกษตรในประเทศไทยในการทำฟาร์มขนาดเล็กและกลางที่ลดลง เนื่องจากการขาดแรงงานในภาคอุตสาหกรรม การเกษตรที่มาจากความกังวลในอันตรายของสารตกค้างและยาฆ่าแมลง เกิดเป็นแนวคิดในการแก้ไขปัญหาด้วยการออกแบบและพัฒนาหุ่นยนต์อัตโนมัติ SAASbot ในการทำฟาร์มขึ้น โดยมีวัตถุประสงค์ในการแก้ไขปัญหาการทำฟาร์มขนาดเล็กอย่างสวนครัวจนไปถึงฟาร์มขนาดกลาง ถูกออกแบบให้สามารถปรับการทำงานตามขนาดปริภูมิของฟาร์มที่เปลี่ยนแปลงได้ ในขณะเดียวกันก็มีฟังก์ชันการทำงานทั้งหมดในการจัดการฟาร์ม ทำหน้าที่ในการเพาะเมล็ด รดน้ำ ใส่ปุ๋ย และกำจัดวัชพืช ซึ่ง SAASbot นำเสนอวิธีแก้ปัญหาค่าที่คุ้มค่า (Cost Effective) ที่ไม่ต้องการพื้นที่ทำงานซึ่งมีความจำเป็นต้องปรับปรุงพื้นที่เพื่อรองรับการทำงานของหุ่นยนต์ให้ยุ่งยากและซับซ้อน SAASbot มีเครื่องมือติดอยู่ที่ส่วนปลายของ End Effector ที่สามารถหมุนได้ เพื่อสลับหน้าที่ในการทำงาน ช่วยลดการเคลื่อนที่ของหุ่นยนต์โดยไม่จำเป็นลง ซึ่งส่งผลให้ลดการใช้พลังงานและเพิ่มความเร็วในการทำงาน โครงสร้างของ SAASbot ประกอบด้วย CNC Mechanism แบบ 3 แกน ถูกพัฒนาให้มี

แพลตฟอร์มแบบ Mobile Robot ใช้การขับเคลื่อนด้วยล้อ Mecanum Wheel ซึ่งหุ่นยนต์ต้นแบบนี้มีขนาดความกว้าง 92 เซนติเมตร ยาว 106 เซนติเมตร สูง 108 เซนติเมตร และมีน้ำหนัก 24.8 กิโลกรัม โดยโครงสร้างของ CNC จะถูกนำมาใช้ในการทำกิจกรรมด้านการเกษตร (Farming Operations) ภายใต้ปฏิภูมิการทำงานของหุ่นยนต์ที่เรียกว่า “Cell” การทำงานนี้จะดำเนินการในลักษณะของ Cellular Manager คือ เมื่อหุ่นยนต์มีตำแหน่งอยู่ใน Cell แล้ว กลไกของระบบ CNC จะถูกใช้ในการทำกิจกรรมต่าง ๆ บน Cell ที่หุ่นยนต์กำลังอยู่ในขณะนั้น ดังภาพที่ 2.54 ซึ่ง CNC จะช่วยให้การเคลื่อนไหวของ End Effector มีความแม่นยำและรวดเร็วขึ้น เนื่องจากหุ่นยนต์มีการเก็บเมล็ด น้ำ และปุ๋ยไว้บนตัวของหุ่นยนต์เอง ส่วนแพลตฟอร์มของ Mobile Robot จะเป็นกลไกที่ใช้ในการเคลื่อนย้ายหุ่นยนต์ไปมาในระหว่าง Cell ทางด้านซอฟต์แวร์นั้น SAASbot สามารถควบคุมการทำงานผ่านทาง Mobile Application ในการควบคุมการปลูกเมล็ด รดน้ำ หรือใส่ปุ๋ย ซึ่งผู้ใช้สามารถสร้างโปรไฟล์สำหรับพืชแต่ละประเภทแยกออกจากกันได้ ส่วนทางด้านฮาร์ดแวร์ SAASbot สามารถเคลื่อนที่ไปในทิศทางใดก็ได้ในแนวระนาบ เพื่อเคลื่อนย้ายชุดส่วนปลาย หรือ End Effector ไปตามแนวแกน X และ Y ด้วย VEX Mecanum Wheel จำนวน 4 ล้อ โดยใช้ชุดขับเคลื่อนเป็น DC Geared Motor แบบแปรงถ่าน และมี Encoder ช่วยในการควบคุมความแม่นยำในการเคลื่อนที่ ในขณะที่ Actuator แนวตั้งจะใช้ในการเคลื่อนย้าย End Effector ไปตามแนวแกน Z ดังสถาปัตยกรรมระบบโดยรวมในภาพที่ 2.55 แต่อย่างไรก็ตามพบว่า SAASbot มีข้อจำกัดซึ่งเกิดจาก Mecanum Wheel ไม่สามารถใช้งานได้อย่างมีประสิทธิภาพบนพื้นดินปกติ ทำให้ SAASbot ต้องออกแบบระบบรางยาง (Rubber Tracks) มาเพื่อประกอบเป็นตารางกริดขึ้น ครอบ Cell ลงไป เพื่อกำหนดปฏิภูมิการทำงานของหุ่นยนต์ ซึ่งการวิจัยนี้ใช้ได้นำ SAASbot มาทำการทดลองปลูกต้นมะเขือเทศ โดยปลูกลงบน Cell ขนาด 3,575 ตารางเซนติเมตร (55 เซนติเมตร x 65 เซนติเมตร) ซึ่งจากการทดสอบพบว่าหุ่นยนต์สามารถมี Running Time สูงสุดเฉลี่ย 1 ชั่วโมง ที่การใช้งานสูงสุดบนแบตเตอรี่ขนาด 2,200 mAh มีความเร็วในการปลูก 90 ต้นต่อชั่วโมง ส่วนความคลาดเคลื่อนนั้นพบว่า ทางด้านแกน X มีความผิดพลาด 3.1 มิลลิเมตร แกน Y มีความผิดพลาด 4.5 มิลลิเมตร และความผิดพลาดในการกำหนดตำแหน่ง 0.26 องศา



ภาพที่ 2.54 การติดตั้งระบบของ SAASbot

ที่มา : [52]



ภาพที่ 2.55 สถาปัตยกรรมระบบโดยรวมของ SAASbot

ที่มา : [52]

ตารางที่ 2.1 การเปรียบเทียบข้อดี-ข้อเสียของหุ่นยนต์ปลูกพืชในงานวิจัยที่ผ่านมา

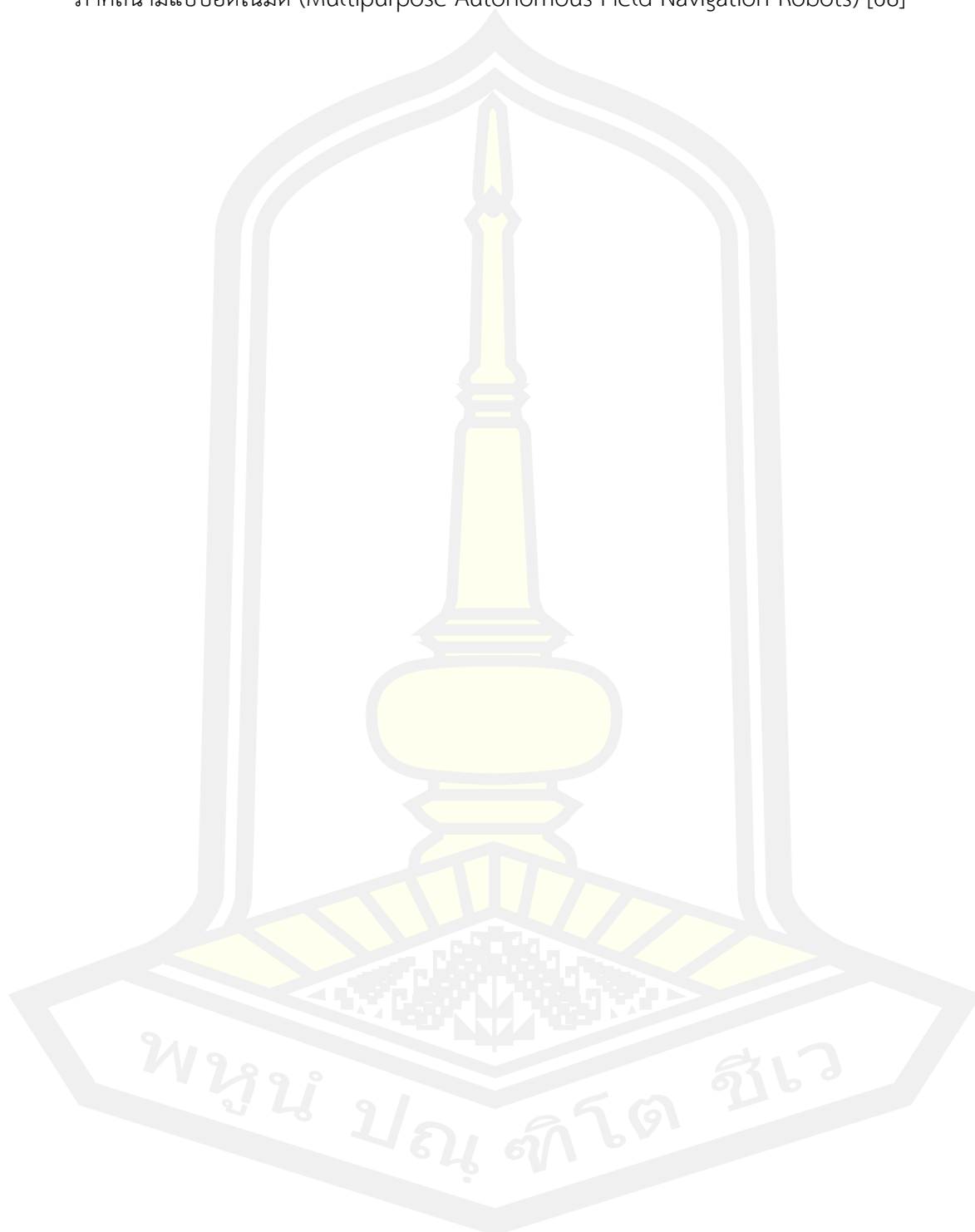
งานวิจัย	ข้อดี	ข้อเสีย
Lady Bug	<ul style="list-style-type: none"> <li>- มีต้นทุนในการพัฒนาที่ต่ำ</li> <li>- สามารถปรับขนาดของ Workspace ได้</li> <li>- สามารถจัดวางตำแหน่งของกระถางได้</li> <li>- อิสระภายใน Workspace ของ Cell</li> <li>- สามารถเคลื่อนที่ได้อิสระ และทำงานได้ในพื้นที่จำกัด</li> <li>- มีความเร็วในการเคลื่อนที่ 360 องศา/ชั่วโมง</li> <li>- Seed Injector มีกระบอกสำหรับเก็บเมล็ด สามารถหยอดเมล็ดได้ทันที ลดเวลาในการทำงานไม่ต้องกลับไปที่ถาดเก็บเมล็ด</li> <li>- มีการทำงานแบบอัตโนมัติ (Automatic)</li> <li>- มีความแม่นยำเนื่องจากใช้กล้องในการตรวจจับและระบุตำแหน่ง</li> <li>- Monitor พารามิเตอร์ และควบคุมการทำงานได้ด้วย IoT ผ่าน MQTT Client และ Dashboard</li> <li>- ต้นแบบมีแค่ฟังก์ชันการปลูก แต่สามารถเพิ่มฟังก์ชันการทำงานได้ในอนาคต</li> </ul>	<ul style="list-style-type: none"> <li>- ไม่สามารถทำงานในพื้นที่ขรุขระ ลาดเอียง มีหลุม และน้ำขัง</li> <li>- กระถางจะต้องถูกวางอยู่ใน Workspace <math>\geq 50\%</math> ระบบจึงจะสามารถตรวจพบ</li> <li>- ขนาดกระถางที่ถูกวางไว้ใน Workspace หากมีขนาด (Size) แตกต่างกันมากเกินไป จะส่งผลให้การตรวจจับใช้เวลานาน และทำให้ประสิทธิภาพของการตรวจจับลดลง จนอาจได้ผลที่เป็น False Positive จำนวนมาก</li> </ul>
CeRSAA [47]	<ul style="list-style-type: none"> <li>- สามารถทำงานได้ในหลายฟังก์ชัน</li> <li>- มีความแม่นยำเนื่องจากใช้กล้องในการตรวจจับและระบุตำแหน่ง</li> <li>- มีความรวดเร็วในการทำงาน เนื่องจากใช้การเคลื่อนที่ของพาเลท กับ End Effector ทำงานร่วมกัน</li> </ul>	<ul style="list-style-type: none"> <li>- มีต้นทุนการพัฒนาที่สูง</li> <li>- ระบบต้นแบบมีขนาดใหญ่ ไม่สามารถเคลื่อนย้ายได้</li> <li>- ใช้พลังงานไฟฟ้าในการทำงานมาก</li> </ul>
RIRRIG [48]	<ul style="list-style-type: none"> <li>- มีต้นทุนการพัฒนาที่ต่ำ</li> <li>- ใช้เซ็นเซอร์พื้นฐานทั่วไป</li> <li>- ระบบไม่มีความยุ่งยากซับซ้อน</li> <li>- การบำรุงรักษาระบบกลไกง่าย ไม่ซับซ้อน</li> </ul>	<ul style="list-style-type: none"> <li>- ทำงานในพื้นที่จำกัด ไม่สามารถขยายได้</li> <li>- ถูกกำหนดการทำงานไว้แล้ว การทำงานไม่สามารถเปลี่ยนแปลงได้ภายหลัง</li> <li>- มีความเร็วในการทำงานที่ค่อนข้างช้า</li> </ul>

ตารางที่ 2.1 (ต่อ) การเปรียบเทียบข้อดี-ข้อเสียของหุ่นยนต์ปลูกพืชในงานวิจัยที่ผ่านมา

งานวิจัย	ข้อดี	ข้อเสีย
FarmBot [49]	<ul style="list-style-type: none"> <li>- เป็นแพลตฟอร์มแบบ Open Source</li> <li>- สามารถทำงานได้ในหลายฟังก์ชัน</li> <li>- มีความแม่นยำเนื่องจากใช้กล้องในการตรวจจับและระบุตำแหน่ง</li> <li>- สามารถควบคุมการทำงานและตั้งการทำงานล่วงหน้าได้</li> <li>- ซอฟต์แวร์ทำงานบน Cloud Service</li> </ul>	<ul style="list-style-type: none"> <li>- ทำงานในพื้นที่จำกัด ไม่สามารถขยายได้</li> <li>- ขณะทำงานมีการเคลื่อนที่ซึ่งไม่จำเป็น ทำให้มีการสูญเสียเวลาและพลังงาน</li> <li>- ความเร็วในการทำงานของระบบจะลดลงหากสั่งให้ปลูกด้วยเมล็ดพันธุ์ต่างชนิดกัน</li> <li>- ต้นทุนในการพัฒนาระบบค่อนข้างสูง</li> </ul>
SAASbot [52]	<ul style="list-style-type: none"> <li>- สามารถปรับขนาดของพื้นที่ฟาร์มได้</li> <li>- สามารถทำงานได้ในหลายฟังก์ชัน</li> <li>- มีความแม่นยำเนื่องจากใช้กล้องในการตรวจจับและระบุตำแหน่ง</li> <li>- ใช้ Mobile Application ในการควบคุม</li> <li>- การสลับอุปกรณ์ที่ปลาย End Effector ทำให้ลดเวลาและการใช้พลังงานลง</li> <li>- เป็นแพลตฟอร์มแบบ Mobile Robot มีการเคลื่อนที่อิสระในแนวระนาบ</li> </ul>	<ul style="list-style-type: none"> <li>- จะต้องใช้รางยางปูเพื่อให้ Mecanum Wheel ทำงานได้อย่างมีประสิทธิภาพ</li> <li>- แม้จะขยายขนาดของฟาร์มได้ แต่การทำงานก็ยังจำกัดในพื้นที่ของ Cell ที่เป็นปริภูมิของหุ่นยนต์ ไม่สามารถขยายได้</li> </ul>

จากการทบทวนวรรณกรรมจึงพบว่า งานวิจัยและพัฒนาทางด้านหุ่นยนต์ในภาคการเกษตร หรือ Agriculture Robotics ส่วนใหญ่นั้น ต่างมุ่งประเด็นการศึกษาไปทางด้าน การควบคุมด้วยวิสัยทัศน์ (Vision-based Control) การประมวลผลภาพขั้นสูง (Advanced Image Processing Techniques) และ การออกแบบมือจับสำหรับการเก็บเกี่ยวผลไม้แบบอัตโนมัติ (Gripper Design for Automated Harvesting of Valuable Fruits) โดยงานวิจัยที่ผ่านมาเหล่านี้ มีการประยุกต์ใช้กับพันธุ์พืชที่หลากหลาย เช่น พริกหวาน [53] ปาล์มน้ำมัน [54] มะม่วง [55] แตงกวา [56] อัลมอนต์ [57] แอปเปิล [58] สตรอเบอร์รี่ [59] เชอร์รี่ [60] มะนาว [61] ไร่องุ่น [62] และมะเขือเทศ [63] นอกจากนี้การศึกษายังโฟกัสไปทางด้านอัลกอริทึมในการนำทาง (Navigation Algorithms) และระบบแมชชีนวิชั่น (Machine Vision Systems) ที่มีประสิทธิภาพ เพื่อนำมาใช้งานในด้านการประมาณค่าผลผลิต (Yield Estimation) [64] การกำจัดวัชพืชและพ่นสารเคมี (Weeding and Targeted Spraying) [65] การเพาะปลูกต้นกล้า (Seedling and Transplanting) [66] การจัดการ

พันธุ์ไม้ที่เปราะบาง (Delicate Handling of Sensitive Flowers) [67] และหุ่นยนต์นำร่อง  
ภาคสนามแบบอัตโนมัติ (Multipurpose Autonomous Field Navigation Robots) [68]

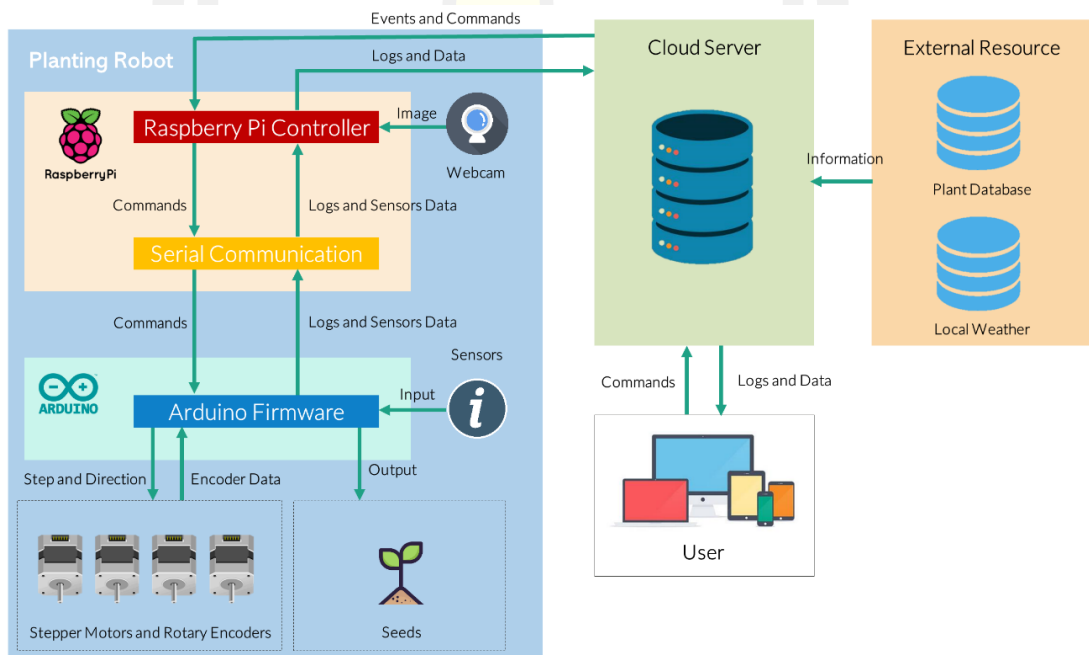




### บทที่ 3

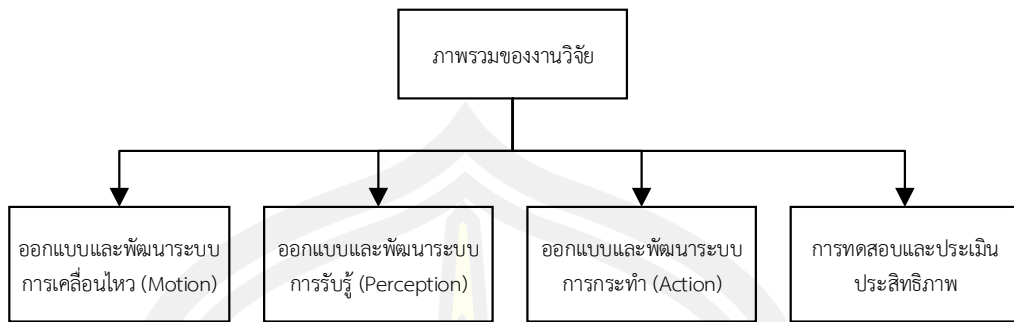
### วิธีดำเนินการวิจัย

งานวิจัยนี้เป็นงานวิจัยเชิงประยุกต์ (Applied Research) มีวัตถุประสงค์ เพื่อพัฒนาต้นแบบหุ่นยนต์ที่ใช้สำหรับทำการเพาะปลูกพืชบนกระถางที่ถูกจัดเรียงไว้ภายในโรงเรือน ซึ่งเป็นการประยุกต์การทำงานของหุ่นยนต์ (Robotics) เข้ากับอินเทอร์เน็ตของสรรพสิ่ง (Internet of Things) การประมวลผลภาพ (Image Processing) และปัญญาประดิษฐ์ (Artificial Intelligent) ตามแนวความคิดของสถาปัตยกรรมระบบ (Overall System Architecture) ดังภาพที่ 3.1 ให้มีความเหมาะสมกับภาคการเกษตร โดยใช้ต้นทุนต่ำ เกษตรกรสามารถลงทุนได้ ทดแทนแรงงานมนุษย์ ลดเวลา ลดความยุ่งยากในภาระงานที่ต้องทำเป็นประจำ รวมถึงการออกแบบขั้นตอนการประเมินผลที่รัดกุมเพื่อนำผลของการวิจัยที่ได้ไปใช้ในการพัฒนาหุ่นยนต์ให้สามารถควบคุมคุณภาพการผลิตได้อย่างแม่นยำยิ่งขึ้น



ภาพที่ 3.1 สถาปัตยกรรมระบบโดยรวมของหุ่นยนต์ปลูกพืชแบบอัตโนมัติ

โดยบทนี้จะกล่าวถึงกระบวนการในการออกแบบและพัฒนาหุ่นยนต์เพาะปลูกพืชอัตโนมัติ เครื่องมือที่ใช้ในการพัฒนาต้นแบบ สภาพแวดล้อมในการทดสอบ ตลอดจนเครื่องมือที่ใช้ในการทดสอบประสิทธิภาพในการทำงาน

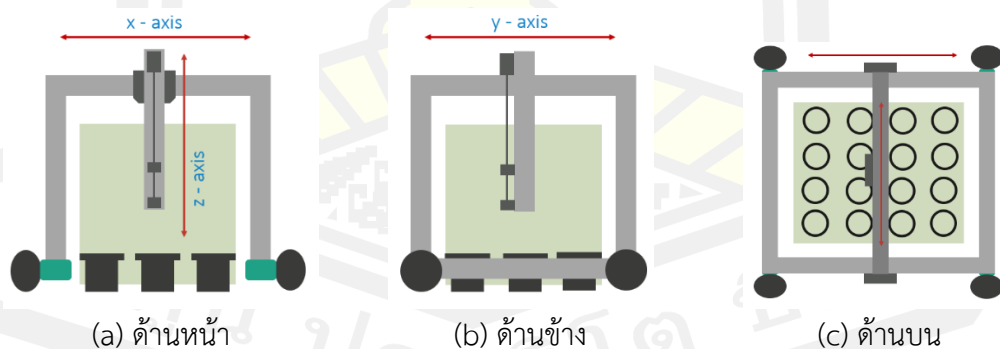


ภาพที่ 3.2 ภาพรวมการดำเนินการวิจัย

ภาพที่ 3.2 แสดงภาพรวมในการดำเนินงานวิจัย ซึ่งใช้ระเบียบวิธีวิจัยเชิงพัฒนาในการวิจัย โดยมีรายละเอียดและขั้นตอนการดำเนินการวิจัยดังต่อไปนี้

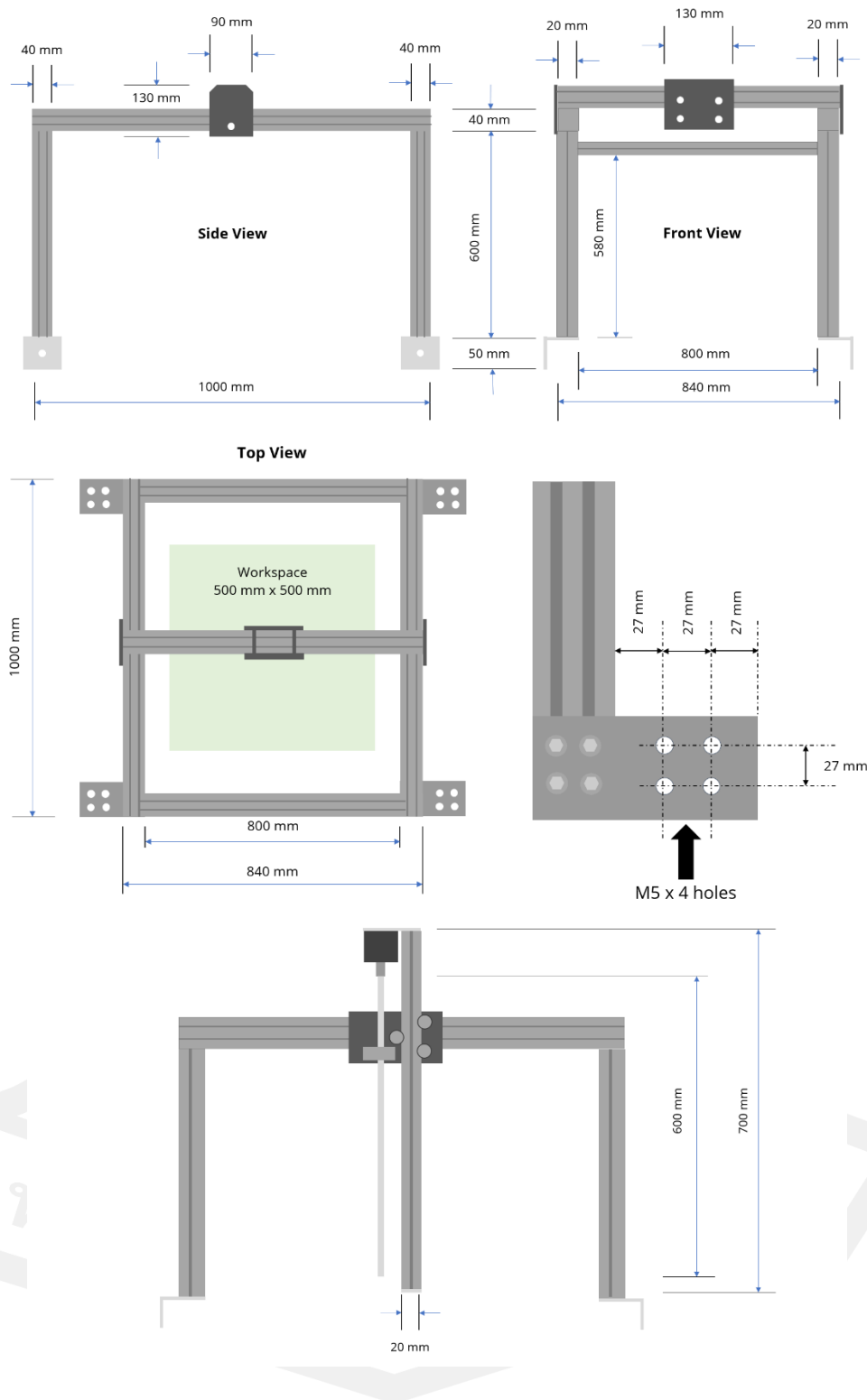
### 3.1 การออกแบบและพัฒนาหุ่นยนต์

หุ่นยนต์เพาะปลูกพืชอัตโนมัติในงานวิจัยนี้ เป็นการออกแบบหุ่นยนต์โดยการประยุกต์การทำงานของหุ่นยนต์ในแพลตฟอร์ม Manipulator Robot ซึ่งมีหน้าที่ในการควบคุมอุปกรณ์ส่วนปลาย (End-Effector) ที่ใช้ในการปลูก ให้ทำงานร่วมกับ Mobile Robot ซึ่งมีหน้าที่ควบคุมการเคลื่อนที่ของหุ่นยนต์ภายในแปลงปลูก ดังแนวคิดของการออกแบบหุ่นยนต์ในภาพที่ 3.3 ซึ่งขั้นตอนนี้เป็นการออกแบบและพัฒนาหุ่นยนต์ตามกรอบโครงสร้างของการพัฒนาหุ่นยนต์ (Robot Frameworks) [69, 70] โดยแบ่งออกเป็น 3 ส่วน ประกอบด้วย การออกแบบและพัฒนาระบบการเคลื่อนที่ (Motion) การออกแบบและพัฒนาระบบการรับรู้ (Perception) และการออกแบบและพัฒนาระบบการกระทำ (Action)



ภาพที่ 3.3 แนวคิดการออกแบบหุ่นยนต์ปลูกพืชแบบอัตโนมัติ

โครงสร้างหลักของหุ่นยนต์จะถูกประกอบขึ้นจาก Aluminum Profile ซึ่งเป็นวัสดุที่มีความแข็งแรง น้ำหนักเบา และสามารถนำมาตัด ต่อ ประกอบขึ้นโครงสร้างเป็นรูปร่างต่าง ๆ ได้ง่าย



ภาพที่ 3.4 แบบโครงสร้างของหุ่นยนต์ปลุกพีช

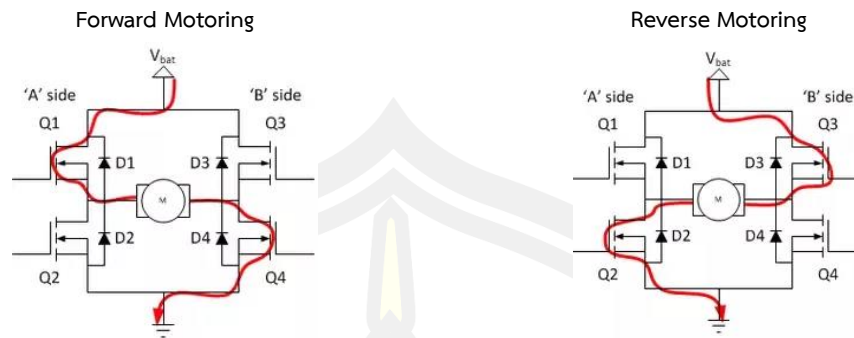


ภาพที่ 3.5 โครงสร้างของหุ่นยนต์ปลุกพีช

#### 1. การออกแบบและพัฒนาระบบการเคลื่อนที่ (Motion)

ในขั้นตอนนี้เป็นการศึกษาการเคลื่อนที่ (Locomotion) ของหุ่นยนต์ และงานวิจัยก่อนหน้านี้ของหุ่นยนต์ที่มีแพลตฟอร์มแบบ Mobile Robot ซึ่งประกอบด้วย การเคลื่อนที่แบบ Holonomic Locomotion และ Non-Holonomic Locomotion [71] โดยการสำรวจและนำผลวิเคราะห์เพื่อเป็นข้อมูลในการออกแบบและพัฒนาระบบการเคลื่อนที่ของหุ่นยนต์เพาะปลูกพืชอัตโนมัติ โดยมีรายละเอียดดังนี้

1) ออกแบบระบบการเคลื่อนที่ของหุ่นยนต์แบบ Holonomic Locomotion ซึ่งใช้มอเตอร์ไฟฟ้ากระแสตรง (DC Motor) ที่ควบคุมความเร็วของมอเตอร์ด้วยการจ่ายไฟฟ้าเป็นช่วงเวลา และมีทิศทางการหมุนขึ้นอยู่กับการจ่ายแรงดันไฟฟ้าที่ป้อนให้กับมอเตอร์ โดยการควบคุมทิศทางการหมุนของมอเตอร์ใช้หลักการขับเคลื่อนแบบเอชบริดจ์ (H-Bridge) โดยมีวงจรดังภาพที่ 3.6 ร่วมกับชุดล้อ Mecanum Wheel [72] ซึ่งอาศัยการทำงานของโรลเลอร์ (Roller) ที่ติดตั้งทำมุม 45 องศา เรียงรายอยู่บน Mecanum Wheel เพื่อรองรับการหมุนรอบข้าง ดังภาพที่ 3.7 เมื่อรวมกับการหมุนของล้อทั้ง 4 ข้าง จะทำให้เกิดการเคลื่อนที่ได้รอบทิศทาง ซึ่งล้อ Mecanum Wheel ที่ถูกติดตั้งบนตัวหุ่นยนต์นั้น จะต้องมีการส่งการด้วยสัญญาณไฟฟ้าแบบแยกอิสระ 4 ล้อ เนื่องจากการทำงานจะใช้ความสัมพันธ์ในการหมุนของล้อทั้ง 4 เป็นตัวควบคุม โดยมีลักษณะของการทำงาน 3 รูปแบบ คือ



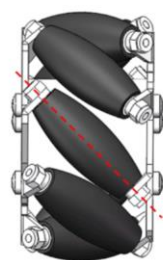
ภาพที่ 3.6 ไดอะแกรมของวงจร H-Bridge Circuit

ที่มา : [73]

รูปแบบที่ 1 การทำงานที่สัมพันธ์ทั้ง 4 ล้อ โดยมีทิศทางการหมุนของล้อไปด้านเดียวกัน ส่งผลให้หุ่นยนต์เคลื่อนที่ไปข้างหน้าและถอยหลัง เหมือนล้อยกปกติที่ใช้กัน

รูปแบบที่ 2 การทำงานสัมพันธ์กันแบบทแยงมุม โดยมีทิศทางการหมุนของล้อที่เหมือนกัน 2 ชุด เช่น ล้อซ้ายบนหมุนไปทิศทางเดียวกับล้อขวาล่าง และล้อขวาบนหมุนไปทิศทางเดียวกับล้อซ้ายล่าง จะทำให้หุ่นยนต์เคลื่อนที่ไปด้านข้าง ด้วยการเคลื่อนที่ลักษณะดังกล่าวนี้ Roller ที่ติดตั้งอยู่บน Mecanum Wheel จะทำหน้าที่ในการหมุนเคลื่อนที่ได้โดยไม่เสียสมดุล

รูปแบบที่ 3 การหมุนรอบตัวเอง โดยใช้การหมุนที่สัมพันธ์กันของล้อ Mecanum Wheel เช่น ล้อด้านซ้ายทั้งล้อบนและล้อล่างจะหมุนไปในทิศทางเดียวกัน พร้อมกับล้อด้านขวาทั้งล้อบนและล้อล่างที่หมุนไปในทิศทางตรงข้ามกับล้อด้านซ้าย จะส่งผลให้หุ่นยนต์หมุนรอบตัวเองอย่างสมดุลและราบรื่น



Left Mecanum Wheel



Right Mecanum Wheel

ภาพที่ 3.7 รูปแบบโครงสร้างของล้อ Mecanum Wheel

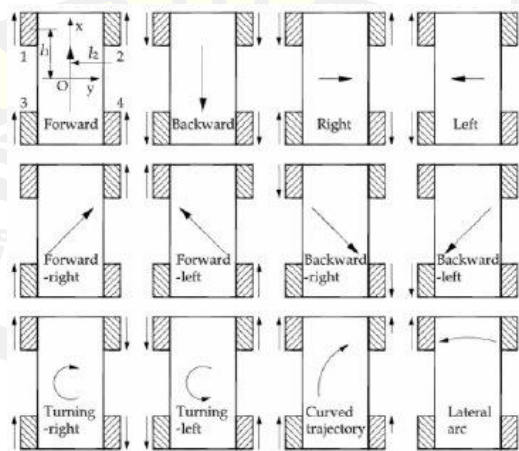
ที่มา : [74]

2) ทดสอบการเคลื่อนที่ของ Mecanum Wheel โดยสั่งการด้วยสัญญาณไฟฟ้าไปยังบอร์ด Motor Driver ที่ทำหน้าที่ขับ Gear Motor ทั้ง 4 ชุด ด้วย PWM ตามเงื่อนไขดังตารางที่ 3.1

ตารางที่ 3.1 รูปแบบที่ใช้ในการทดสอบการเคลื่อนที่ของล้อ Mecanum Wheel

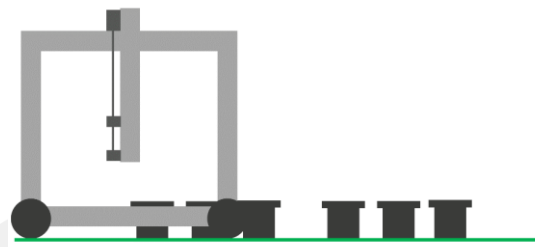
Movement Pattern	Mecanum Wheel			
	FL	FR	BL	BR
Forward	Forward	Forward	Forward	Forward
Backward	Backward	Backward	Backward	Backward
Right	Backward	Forward	Forward	Backward
Left	Forward	Backward	Backward	Forward
Forward Right	Stop	Forward	Forward	Stop
Forward Left	Forward	Stop	Stop	Forward
Backward Right	Backward	Stop	Stop	Backward
Backward Left	Stop	Backward	Backward	Stop
Turning Right	Forward	Backward	Forward	Backward
Turning Left	Backward	Forward	Backward	Forward

หมายเหตุ FL = ล้อหน้าซ้าย FR = ล้อหน้าขวา BL = ล้อหลังซ้าย และ BR = ล้อหลังขวา

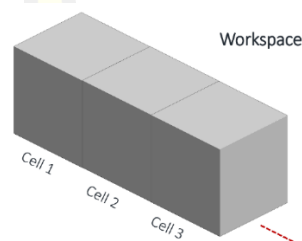


ภาพที่ 3.8 การเคลื่อนที่ของหุ่นยนต์ตามทิศทางและความเร็วเชิงมุมของล้อ

ที่มา : [75]



ภาพที่ 3.9 แสดงการเคลื่อนที่ของหุ่นยนต์ภายในแปลงปลูก



ภาพที่ 3.10 ปริภูมิการทำงานของหุ่นยนต์

## 2. การออกแบบและพัฒนาระบบการรับรู้ (Perception)

ในขั้นตอนนี้เป็นการศึกษาระบบการรับรู้ของหุ่นยนต์ และงานวิจัยก่อนหน้านี้ เพื่อออกแบบและพัฒนาระบบการรับรู้ของหุ่นยนต์เพาะปลูกพืชอัตโนมัติ ซึ่งความสามารถในการรับรู้สภาพแวดล้อมที่อยู่บริเวณรอบ ๆ ตัวของหุ่นยนต์นั้น เกิดจากการประมวลผลข้อมูลที่ได้รับจากอุปกรณ์สำคัญ 2 ชุด คือ กล้อง และชุดเซ็นเซอร์ที่ถูกติดตั้งบนตัวของหุ่นยนต์ ซึ่งเป็นกระบวนการของการประมวลผลภาพ (Image Processing) เพื่อนำข้อมูลที่ได้รับนั้นมาใช้ในการตัดสินใจของหุ่นยนต์ เพื่อให้หุ่นยนต์เคลื่อนที่ หยิบจับ หรือทำงานอื่น ๆ ตามที่ถูกออกแบบการทำงานเอาไว้



ภาพที่ 3.11 Raspberry Pi และ Camera Module

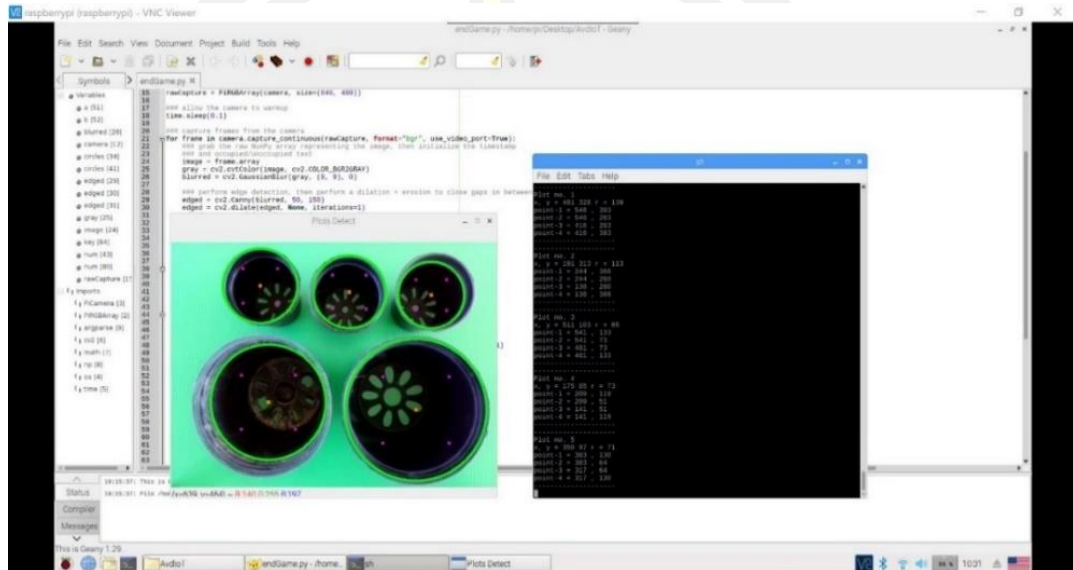
ในการวิจัยนี้จะใช้ Raspberry Pi ร่วมกับ Raspberry Pi Camera Module ดังภาพที่ 3.11 โดยทำงานร่วมกันในการประมวลผลภาพ เพื่อนำผลลัพธ์ที่ได้มาควบคุมและสั่งการไปยัง Arduino ผ่านทาง Serial Communication ซึ่งมีรายละเอียดดังนี้

1) ออกแบบระบบการรับรู้ของหุ่นยนต์โดยใช้การประมวลผลภาพ (Image Processing) ร่วมกับปัญญาประดิษฐ์ (Artificial Intelligent) เพื่อทำการตรวจจับกระถางแบบวงกลมที่ถูกจัดวางไว้ภายในแปลงปลูก ด้วยเทคนิค Hough Circle Transform (HCT) ซึ่งเขียนโค้ดด้วยภาษา Python โดยการใช้ HoughCircles() ซึ่งเป็นฟังก์ชันใน OpenCV เพื่อตรวจจับวงกลมที่อยู่ภายในภาพ ฟังก์ชัน HoughCircles() นี้จะใช้ค้นหาวงกลมที่อยู่ในภาพระดับสีเทา (Grayscale Image) แล้วจะดึงเอาคุณลักษณะ (Extract Features) ออกจากรูปภาพ จากนั้นจึงทำการลงคะแนน หรือการ Vote เพื่อกำหนดรูปร่างของวัตถุที่อยู่ในภาพ

ในการใช้งานฟังก์ชันการตรวจจับวงกลมให้มีประสิทธิภาพนั้น จำเป็นจะต้องระบุค่าของอาร์กิวเมนต์ (Argument) ให้กับ HoughCircles() เพื่อปรับแต่งการทำงาน โดยอาร์กิวเมนต์แรกของ HoughCircles() คือ ภาพที่ต้องการตรวจจับวงกลม ซึ่งจะต้องผ่านการแปลงภาพให้อยู่ในระดับสีเทาก่อน อาร์กิวเมนต์ที่สอง คือ วิธีการที่ใช้ในการตรวจจับวงกลม โดยแบ่งเป็น cv2.HOUGH\_STANDARD สำหรับการตรวจจับแบบคลาสสิกหรือมาตรฐาน, cv2.HOUGH\_PROBABILISTIC เป็นการตรวจจับในรูปแบบ Probabilistic Hough Transform ใช้เมื่อมีส่วนเชิงเส้นยาวอยู่ในรูปภาพ, cv2.HOUGH\_MULTI\_SCALE ซึ่งประกอบด้วยตัวแปรหลายระดับใน Hough Transform แบบคลาสสิก, cv2.HOUGH\_GRADIENT และ cv2.GRADIENT\_ALT เป็นต้น อาร์กิวเมนต์ที่สาม คือ อัตราส่วนของความละเอียดสะสม (Accumulator Resolution) และความละเอียดของภาพ (Image Resolution) หากอัตราส่วนเป็น 1 ความละเอียดสะสมและความละเอียดของภาพจะเท่ากัน และหากอัตราส่วนเป็น 2 ความละเอียดสะสมของภาพในส่วนของความกว้างและความสูงจะกลายเป็นครึ่งหนึ่ง (อัตราส่วนที่แนะนำสำหรับเมธอด cv2.HOUGH\_GRADIENT\_ALT คือ 1.5) อาร์กิวเมนต์ที่สี่ คือ ระยะห่างขั้นต่ำระหว่างจุดศูนย์กลางของวงกลมสองวง อาร์กิวเมนต์ที่ห้า คือ พารามิเตอร์เฉพาะสำหรับวิธีแรกในกรณีของ cv2.HOUGH\_GRADIENT และ cv2.HOUGH\_GRADIENT\_ALT จะถูกใช้เป็นเกณฑ์ (Threshold) สำหรับการตรวจจับขอบภาพด้วยเทคนิค Canny อาร์กิวเมนต์ที่หกเป็นพารามิเตอร์เฉพาะสำหรับวิธีที่สองในกรณีของ cv2.HOUGH\_GRADIENT จะถูกใช้เป็นเกณฑ์สำหรับคำนวณหาจุดศูนย์กลางของวงกลม สำหรับในกรณีของ cv2.HOUGH\_GRADIENT\_ALT นั้น อาร์กิวเมนต์ที่หกจะถูกใช้เป็นค่าความสมบูรณ์แบบ (Perfectness Value) สำหรับวงกลมแทน อาร์กิวเมนต์ที่เจ็ด คือ ค่าต่ำสุดของรัศมีวงกลม และอาร์กิวเมนต์ที่แปด คือ รัศมีสูงสุดของวงกลมที่ทำการตรวจจับ

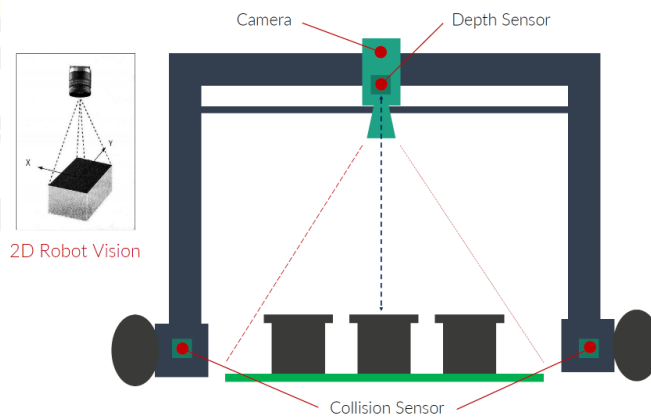


ในภาพที่ 3.12 จะเป็นการทดสอบการตรวจจับกระถางปลูกพืชแบบวงกลม โดยใช้ฟังก์ชัน HoughCircles() และวาดวงกลม (Contours) ล้อมรอบตำแหน่งที่ตรวจพบกระถาง พร้อมทั้งระบุตำแหน่งของจุดศูนย์กลางบนรูปภาพต้นฉบับ โดยใช้ฟังก์ชัน circle() ของ OpenCV ซึ่งเขียนด้วย Python บน Linux ใน Raspberry Pi



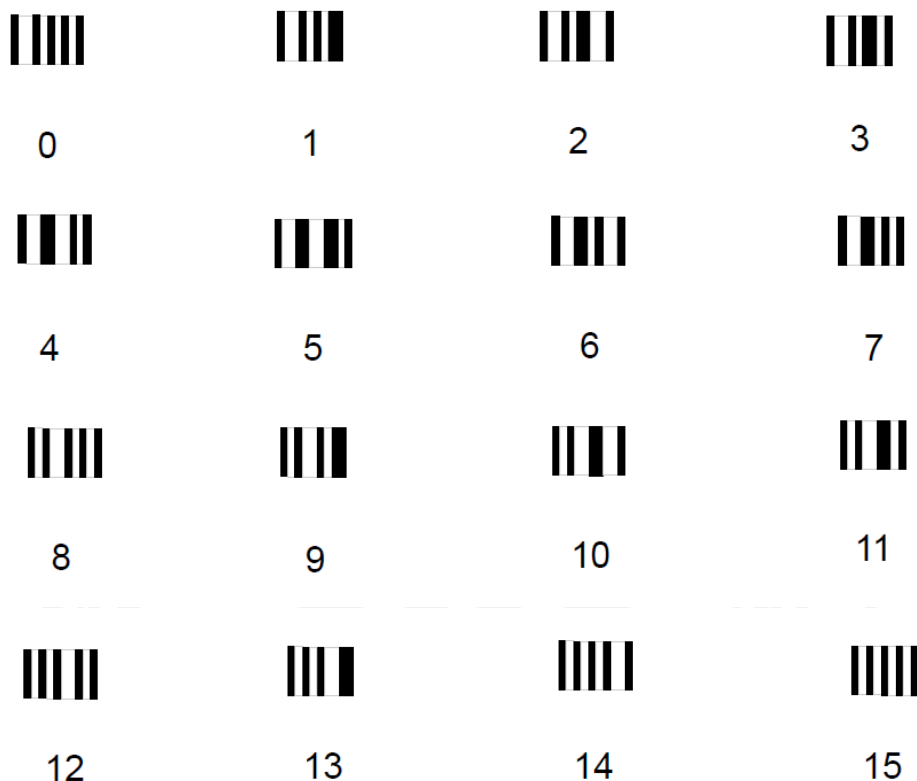
ภาพที่ 3.12 การตรวจจับกระถางและหาค่าจุด Centroids ด้วย HCT

2) ออกแบบระบบการรับรู้ของหุ่นยนต์โดยใช้ Ultrasonic Distance Sensor เพื่อวัดระยะห่างระหว่างกระถาง กับ Seed Injector โดยทำงานร่วมกับกล้องในการหาตำแหน่งบนแกน  $x$ ,  $y$  และ  $z$  เพื่อใช้เป็นจุดอ้างอิงในการหยอดเมล็ดลงจุดกึ่งกลางของกระถางปลูก



ภาพที่ 3.13 แนวคิดการออกแบบระบบการรับรู้ของหุ่นยนต์

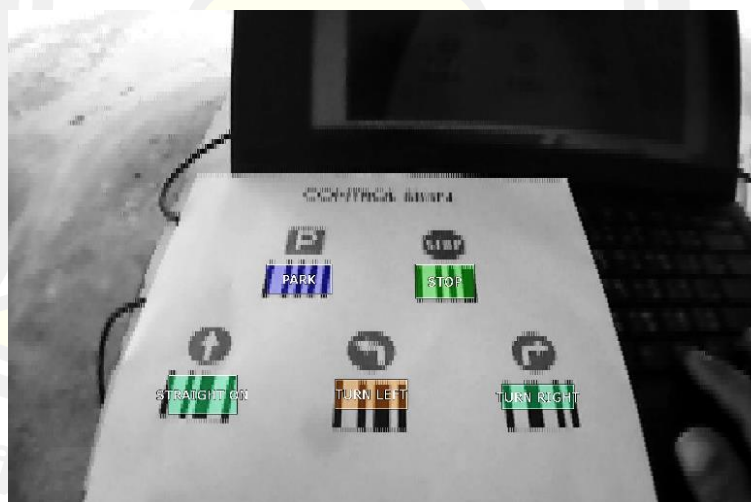
3) ออกแบบระบบการรับรู้ของหุ่นยนต์โดยใช้ชุด MCU-Cam [76] เพื่อตรวจจับสัญลักษณ์ (Tag) ซึ่งใช้ในการควบคุมการเคลื่อนที่และตัดสินใจในการทำงาน โดยมีอัตรา Frame Rate ที่ 60 เฟรมต่อวินาที ซึ่งทำให้มีความไวในการประมวลผลเฟรมภาพทั้งหมด 1/60 วินาที หรือ 16.7 ms เพียงพอต่อการตรวจจับและอ่านค่าจากสัญลักษณ์ของหุ่นยนต์ ในขณะที่เคลื่อนที่ด้วยความเร็วไม่เกิน 40 ไมล์ต่อชั่วโมง นอกจากการตรวจจับเส้น (Lines) และทางแยก (Intersections) แล้ว MCU-Cam ยังสามารถตรวจจับบาร์โค้ด (Barcode) ขนาดเล็กได้ โดยนำมาประยุกต์ใช้บอกหุ่นยนต์ว่าต้องการทำงานอะไรที่บาร์โค้ดนี้ แต่ในการตรวจจับนั้นไม่สามารถทำได้กับบาร์โค้ดทุกรูปแบบ ซึ่ง MCU-Cam มีข้อจำกัดในการตรวจจับรูปแบบของบาร์โค้ดขนาดเล็ก ที่มีความแตกต่างกันได้ 16 รหัส ดังภาพที่ 3.14 โดยจะทำการตรวจจับแล้ว Return ค่า 0 – 15 ซึ่งเป็นรหัสประจำตัวของบาร์โค้ด แล้วรายงานผ่านการสื่อสารแบบ Serial Communication ไปยัง Raspberry Pi ทันทีเมื่อตรวจพบ โดยระบบการตัดสินใจของหุ่นยนต์จะสามารถเลือกได้ว่าจะตอบสนองต่อบาร์โค้ดแต่ละรหัสอย่างไร ซึ่งในการวิจัยครั้งนี้ จะกำหนดรูปแบบการทำงานไว้ดังตารางที่ 3.2



ภาพที่ 3.14 รูปแบบบาร์โค้ดของ MCU-Cam ทั้ง 16 รหัส

ตารางที่ 3.2 การกำหนดเงื่อนไขการทำงานของหุ่นยนต์ด้วย Barcode

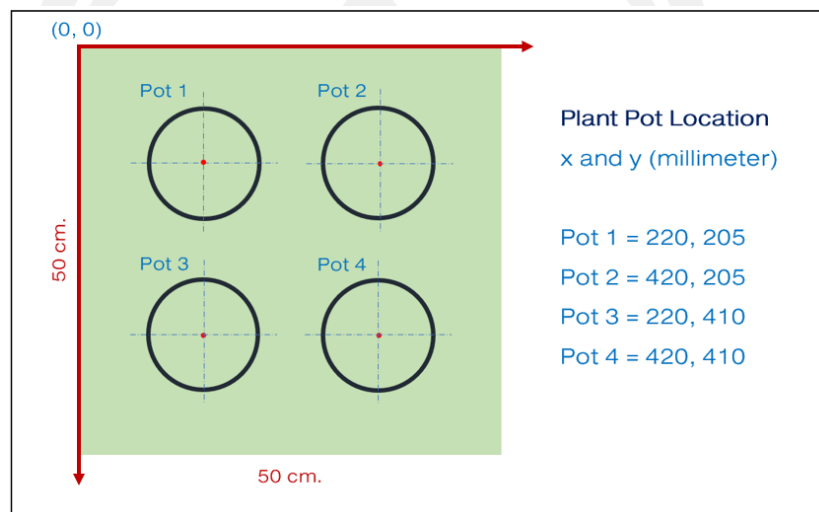
Symbol	Task	Meaning
	Start + Init + Forward	เริ่มการทำงานของหุ่นยนต์ ตั้งค่าเริ่มต้นของระบบ แล้วเคลื่อนที่ไปข้างหน้า
	Stop + Seeding + Forward	หยุด ทำการตรวจจับกระถาง หยอดเมล็ด จากนั้นเคลื่อนที่ไปข้างหน้า
	Stop + Right	หยุด แล้วเคลื่อนที่ไปด้านขวา
	Stop + Backward	หยุด แล้วเคลื่อนที่ไปด้านหลัง
	Stop + Seeding + Backward	หยุด ทำการตรวจจับกระถาง หยอดเมล็ด จากนั้นเคลื่อนที่ไปด้านหลัง
	Stop + Init + End	หยุด คินค่าเป็นค่าเริ่มต้นแล้วปิดการทำงานของระบบ เสร็จสิ้นการทำงาน



ภาพที่ 3.15 การทดสอบอ่านค่า Symbol ที่ใช้ในการควบคุมหุ่นยนต์

4) ทดสอบการทำงานฮาร์ดแวร์และซอฟต์แวร์ของระบบการรับรู้ในการตรวจจับกระถาง โดยทำการวางกระถางปลูกพืชแบบกลม สีดำ ขนาด 5 นิ้ว จำนวน 4 ใบ บนพื้นที่ซึ่งถูกกำหนดตำแหน่งเอาไว้แล้ว (Fixed Position) ดังภาพที่ 3.16 จากนั้นให้ Raspberry Pi Camera

Module ถ่ายภาพ Snapshot แบบสี่เหลี่ยม (Square) 1 : 1 ที่ความละเอียด (Resolution) ขนาด 1,024 x 1,024 Pixels แล้วนำข้อมูลภาพไปทำการประมวลผลใน HoughCircles() เพื่อตรวจจับ กระจ่าง วาด Contours หาจำนวนกระจ่าง และหาตำแหน่งพิกัดของ x และ y ที่เป็นจุดศูนย์กลาง ซึ่งจะใช้สำหรับการหยอดเมล็ด แล้วแปลงจากพิกัด Pixel ของภาพให้มีหน่วยเป็นมิลลิเมตรใน ตำแหน่งจริงของกระจ่าง



ภาพที่ 3.16 ตำแหน่งการจัดวางกระจ่างเพื่อใช้ในการทดสอบระบบการรับรู้

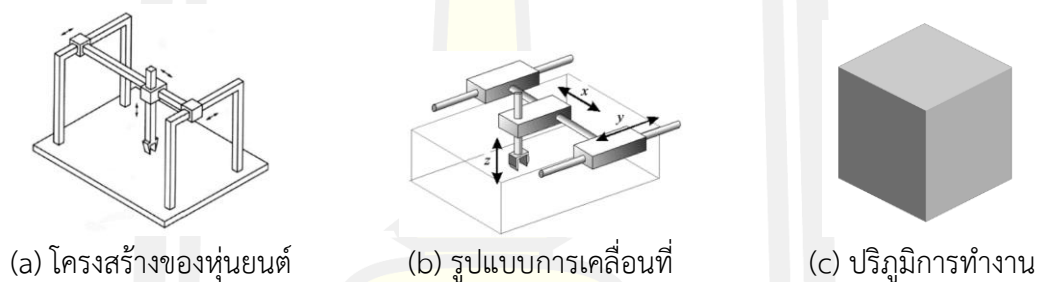


ภาพที่ 3.17 การทดสอบระบบการตรวจจับกระจ่างของหุ่นยนต์

### 3. การออกแบบและพัฒนาระบบการกระทำ (Action)

ในขั้นตอนนี้เป็นการศึกษาระบบการกระทำของหุ่นยนต์ และงานวิจัยก่อนหน้านี้ เพื่อเป็นข้อมูลในการออกแบบและพัฒนาระบบการกระทำของหุ่นยนต์เพาะปลูกพืชอัตโนมัติ โดยมีรายละเอียดดังนี้

1) ออกแบบระบบการควบคุมและระบบการเคลื่อนที่ของส่วนการกระทำของหุ่นยนต์ โดยการประยุกต์ใช้รูปแบบการทำงานของ Gantry Robot ซึ่งมีลักษณะทางโครงสร้างแบบ Overhead Crane มีการเคลื่อนที่แบบเชิงเส้น (Prismatic) บนแกนตั้งฉาก สามารถทำงานได้บน 3 แกน คือ  $x$ ,  $y$  และ  $z$  โดยหุ่นยนต์ประเภทนี้มีโครงสร้างแข็งแรงตลอดแนวการเคลื่อนที่ เคลื่อนที่ได้รวดเร็วและมีความแม่นยำสูง ใช้พื้นที่ในการทำงานน้อย จึงมีความเหมาะสมในการนำมาใช้งานในพื้นที่ที่จำกัด ดังโครงสร้างที่แสดงในภาพที่ 3.18



ภาพที่ 3.18 หุ่นยนต์ประเภท Gantry Robot

2) ออกแบบระบบควบคุมและระบบการเคลื่อนที่ของชุดอุปกรณ์ส่วนปลาย (End Effector) ตามแนวคิดของการออกแบบในภาพที่ 3.19 ซึ่งใช้ในการเคลื่อนที่ไปยังตำแหน่งระนาบแกน  $x$ ,  $y$  และ  $z$  ที่ถูกกำหนดบนกระถางในแปลงปลูก โดยการนำ Aluminum Profile มาขึ้นรูปเป็นโครงสร้าง Area Gantry Robot หรือ AGR เนื่องจากแข็งแรง มีน้ำหนักเบา และมีอุปกรณ์ที่สามารถนำมาติดตั้งบนโครงสร้างได้หลากหลายตามความเหมาะสมของของงาน ส่วนชุดแกนเลื่อน Actuator จะถูกขับเคลื่อนด้วย Stepper Motor ซึ่งเป็นมอเตอร์ไฟฟ้าที่ขับเคลื่อนด้วยสัญญาณพัลส์ (Pulse) โดยการหมุนรอบแกน 360 องศา แบบไม่ต่อเนื่อง และจะเคลื่อนเป็น Step โดยแต่ละ Step จะขับเคลื่อนได้ 1, 1.5, 1.8 หรือ 2 องศา ขึ้นอยู่กับโครงสร้างของมอเตอร์ เพื่อให้ตำแหน่งการเคลื่อนที่มีความแม่นยำสูง ซึ่งใช้ Stepper Motor จำนวน 3 ชุด แยกควบคุมการเคลื่อนที่ในแต่ละแกน แต่ในส่วนของแกน  $z$  นั้น จำเป็นจะต้องใช้ชุด Lead Screw ดังภาพที่ 3.20 เข้ามาควบคุมการเคลื่อนที่ เนื่องจากในแนวแกนนี้เป็นการทำงานของหัวปลูก ที่นอกจากจะต้องการความแม่นยำในการเคลื่อนที่แล้ว จำเป็นจะต้องมีความต้านทานต่อแรงต้านที่เกิดจากหัวปลูกกระทบกับดินภายในกระถาง ซึ่งอาจสร้างความเสียหายให้อุปกรณ์และความคาคเคลื่อนที่ของตำแหน่งแกน  $z$  ในระหว่างการทำงานได้

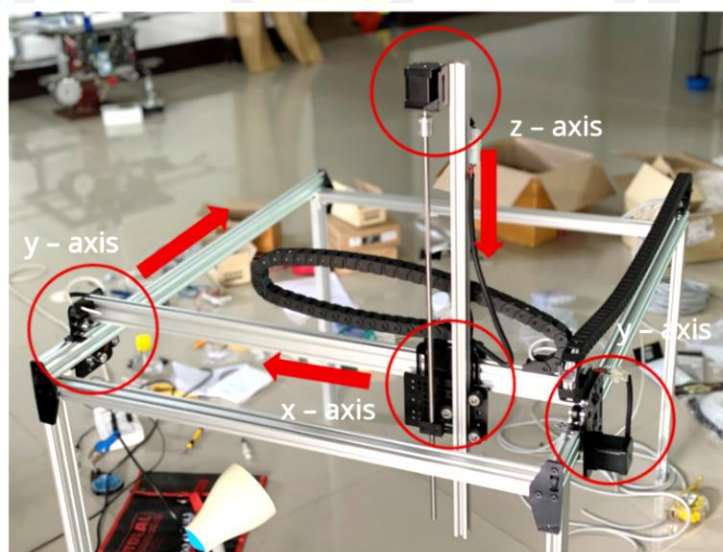


ภาพที่ 3.19 แนวคิดในการออกแบบโครงสร้างระบบการเคลื่อนที่ในระนาบ  $x-y$  ของหุ่นยนต์

ที่มา : [78]

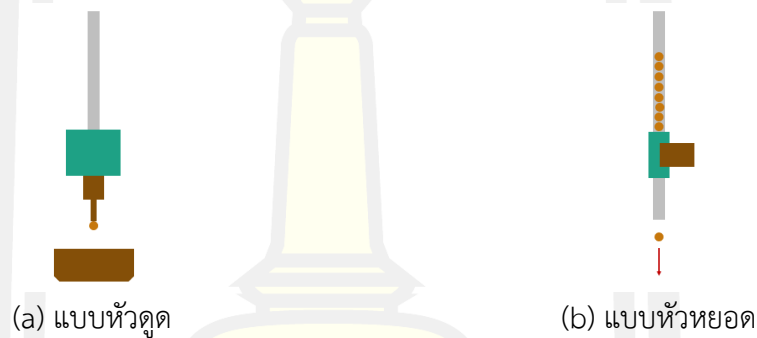


ภาพที่ 3.20 ตัวอย่างการประยุกต์ใช้ Lead Screw ในแนวแกน  $z$

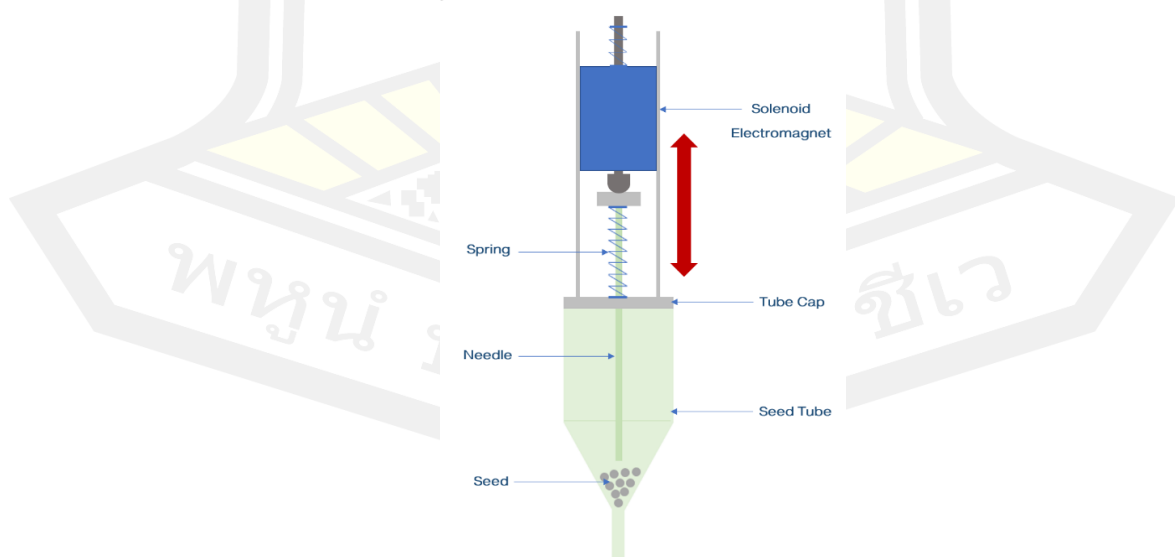


ภาพที่ 3.21 การประกอบโครงสร้างการเคลื่อนที่เชิงเส้นของหุ่นยนต์

3) ออกแบบหัวปลูก (Seed Injector) ของหุ่นยนต์ซึ่งใช้ในการหยอดเมล็ดพืชลงในกระถาง โดยมีแนวคิดของการออกแบบดังภาพที่ 3.22 ประกอบด้วยหัวปลูกใน 2 ลักษณะ คือ หัวปลูกแบบดูด ซึ่งใช้แรงลมดูดเมล็ดจากถาดเก็บเมล็ดบนโครงสร้างของหุ่นยนต์ จากนั้นหัวปลูกจะเคลื่อนที่ไปยังตำแหน่งกึ่งกลางของกระถางที่กำหนดไว้ แล้วทำการแทงหัวปลูกลงไปในดินที่อยู่ในกระถางเพื่อนำเมล็ดไปหยอดลงไป จากนั้นหัวปลูกก็จะเคลื่อนที่กลับไปยังตำแหน่งถาดเก็บเมล็ดเพื่อดูดเอาเมล็ดไปปลูกอีกครั้งในรอบถัดไป และหัวปลูกแบบหัวหยอด ซึ่งจะใช้วิธีเก็บเมล็ดไว้ที่หลอดเก็บเมล็ดที่หัวปลูก ส่งผลให้ลดการสูญเสียเวลาในการปลูกลงเนื่องจากไม่ต้องเคลื่อนที่กลับไปยังถาดเก็บเมล็ด โดยหัวปลูกแบบนี้จะเคลื่อนที่ไปยังตำแหน่งกึ่งกลางของกระถางที่กำหนดไว้ แล้วทำการแทงหัวปลูกลงไปในดินที่อยู่ในกระถาง จากนั้นจะเมล็ดจะถูกปล่อยออกจากหลอดเก็บเมล็ดหยอดลงไป

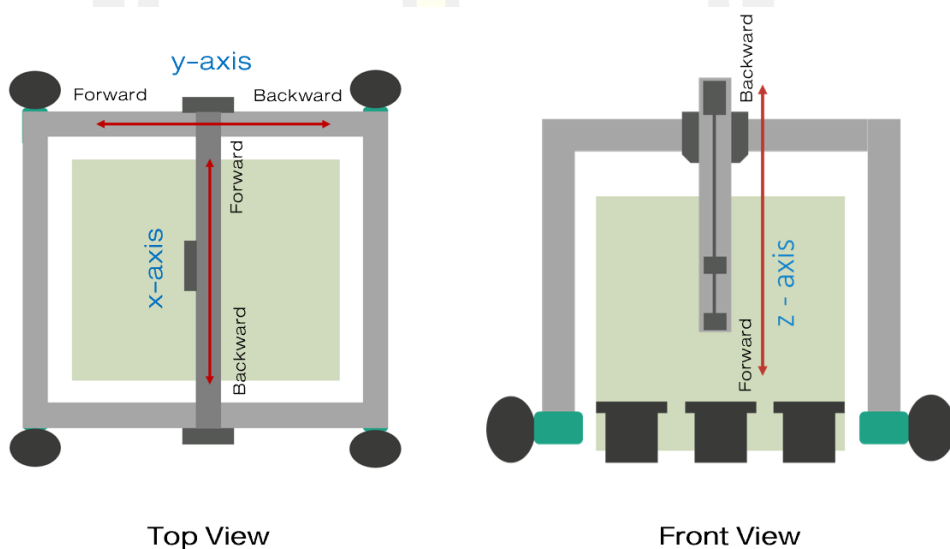


ภาพที่ 3.22 แนวคิดการออกแบบหัวปลูก

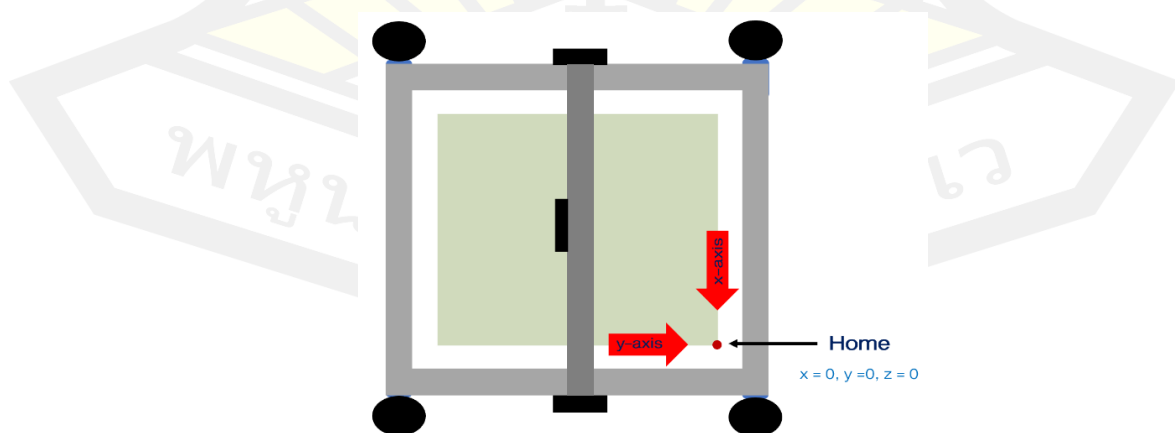


ภาพที่ 3.23 Seed Injector ที่ใช้สำหรับหยอดเมล็ด

4) ทดสอบการควบคุมระบบการเคลื่อนที่ของชุดอุปกรณ์ส่วนปลาย โดยใช้ Open Builds Control [79] ของ Open Builds Team ซึ่งเป็นซอฟต์แวร์สำหรับงานด้าน CNC โดยเฉพาะเพื่อทดสอบการเคลื่อนที่ของระบบขับเคลื่อนทั้ง 3 แกน โดยการ Jog หรือการสั่งให้เคลื่อนที่เป็นระยะทางสั้น ๆ ตามที่กำหนดค่าไว้ในซอฟต์แวร์ ไปในทิศทางด้านหน้า (Forward) และถอยหลัง (Backward) ดังภาพที่ 3.24 โดยกำหนดระยะที่ใช้ทดสอบการเคลื่อนที่ในแต่ละครั้ง เท่ากับ 100 มิลลิเมตร แล้วนำระยะการเคลื่อนที่จริงของระบบขับเคลื่อนในแต่ละแกนมาเปรียบเทียบกับค่าอ้างอิง พร้อมทั้งทดสอบการเคลื่อนที่กลับไปยังตำแหน่ง Home ของแต่ละแกน โดยตำแหน่ง Home คือตำแหน่งที่แกน x, y และ z มีระยะเป็น 0 มิลลิเมตร ดังภาพที่ 3.25



ภาพที่ 3.24 รูปแบบการทดสอบระบบการเคลื่อนที่ของชุดอุปกรณ์ส่วนปลาย



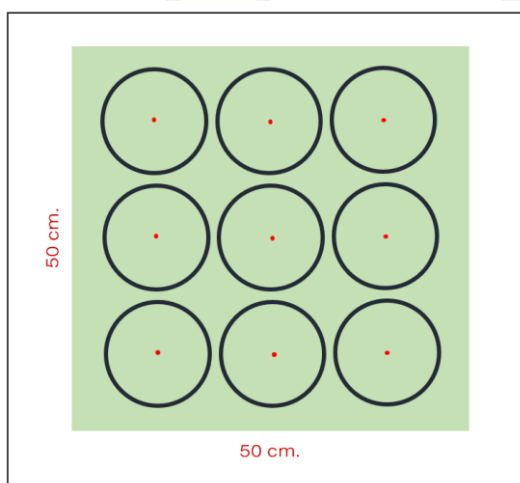
ภาพที่ 3.25 การเคลื่อนที่กลับไปยังตำแหน่ง Home



5) ทดสอบระบบการกระทำของหุ่นยนต์ในการเคลื่อนที่ไปยังตำแหน่งของกระถางและการหยอดเมล็ดลงในกระถางภายในแปลงปลูก โดยทำการวางกระถางปลูกพืชแบบกลม สีดำ ขนาด 5 นิ้ว จำนวน 9 ใบ บนพื้นที่ซึ่งถูกกำหนดตำแหน่งเอาไว้เป็นแปลงปลูก จากนั้นให้หุ่นยนต์ทำการตรวจจับกระถางเพื่อทำขั้นตอนของการหยอดเมล็ด ซึ่งมีเกณฑ์การประเมิน คือ

- การหยอดเมล็ดสำเร็จ หมายถึง การที่หัวปลูกของหุ่นยนต์เคลื่อนที่ไปยังตำแหน่งของกระถาง แล้วสามารถหยอดเมล็ดพันธุ์ลงในกระถางได้ โดยมีความคลาดเคลื่อน  $\pm 20\%$  จากจุดกึ่งกลางของกระถางที่ตำแหน่ง 70 มิลลิเมตร (กระถางขนาด 5 นิ้ว มีเส้นผ่านศูนย์กลางภายในขนาด 140 มิลลิเมตร)

- การหยอดเมล็ดไม่สำเร็จ หมายถึง การที่หัวปลูกของหุ่นยนต์เคลื่อนที่ผ่านกระถางไปโดยไม่ทำการหยอดเมล็ด หรือมีการเคลื่อนที่ไปยังตำแหน่งของกระถาง แต่หยอดเมล็ดออกนอกกระถาง หรือกรณีที่หัวปลูกทำงานถูกต้องแล้ว แต่ไม่มีเมล็ดถูกหยอดลงกระถาง หรือหยอดลงกระถางในตำแหน่งที่มีความคลาดเคลื่อนเกิน  $\pm 20\%$



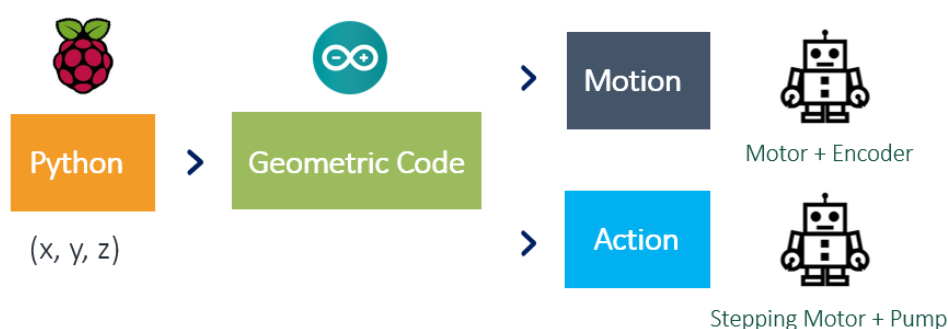
ภาพที่ 3.26 ตำแหน่งการจัดวางกระถางเพื่อใช้ในการทดสอบระบบการกระทำ

### 3.2 การออกแบบซอฟต์แวร์ควบคุมการทำงานของหุ่นยนต์

ในระหว่างการพัฒนาโครงสร้างหุ่นยนต์ต้นแบบนั้น จะทำการออกแบบและพัฒนาซอฟต์แวร์ที่ใช้ในการควบคุมการทำงานไปในลักษณะคู่ขนาน โดยการพัฒนาทางด้านซอฟต์แวร์จะแบ่งออกเป็น 2 แพลตฟอร์ม ดังภาพที่ 3.27 ประกอบด้วย ซอฟต์แวร์ที่ทำงานบนบอร์ด Raspberry Pi และซอฟต์แวร์ที่ทำงานบนบอร์ด Arduino

1. ซอฟต์แวร์บนบอร์ด Raspberry Pi นั้น จะถูกเขียนโค้ดขึ้นเพื่อใช้ในการถ่ายภาพกระถางบนแปลงปลูก จากนั้นจึงนำภาพที่ได้มาทำการประมวลผลภาพ เพื่อตรวจจับกระถาง แล้วค้นหาตำแหน่งของจุดศูนย์กลาง (Centroid) ของกระถางด้วยเทคนิค Hough Circle Transform ด้วย OpenCV เพื่อส่งค่าตำแหน่งในรูปแบบของค่า  $x$ ,  $y$  และ  $z$  สื่อสารผ่าน Serial Communication ไปสั่งการบอร์ด Arduino รวมถึงควบคุมการเคลื่อนที่ของหุ่นยนต์ในแปลงปลูกและการสื่อสารในส่วน of Internet of Things

2. ซอฟต์แวร์บนบอร์ด Arduino จะถูกเขียนโค้ดขึ้นเพื่อใช้ในการควบคุมการทำงานของ Stepping Motor ของทั้ง 3 แกน ประกอบด้วยแกน  $x$ ,  $y$  และ  $z$  ที่ทำหน้าที่ควบคุมการเคลื่อนที่ของอุปกรณ์ส่วนปลายซึ่งใช้ในการหยอดเมล็ด ด้วย Geometric Code (G-Code) รวมถึงการตรวจวัดค่าพารามิเตอร์อื่น ๆ ที่อยู่ในสิ่งแวดล้อมผ่านเซ็นเซอร์ที่ถูกติดตั้งไว้บนโครงสร้างหุ่นยนต์

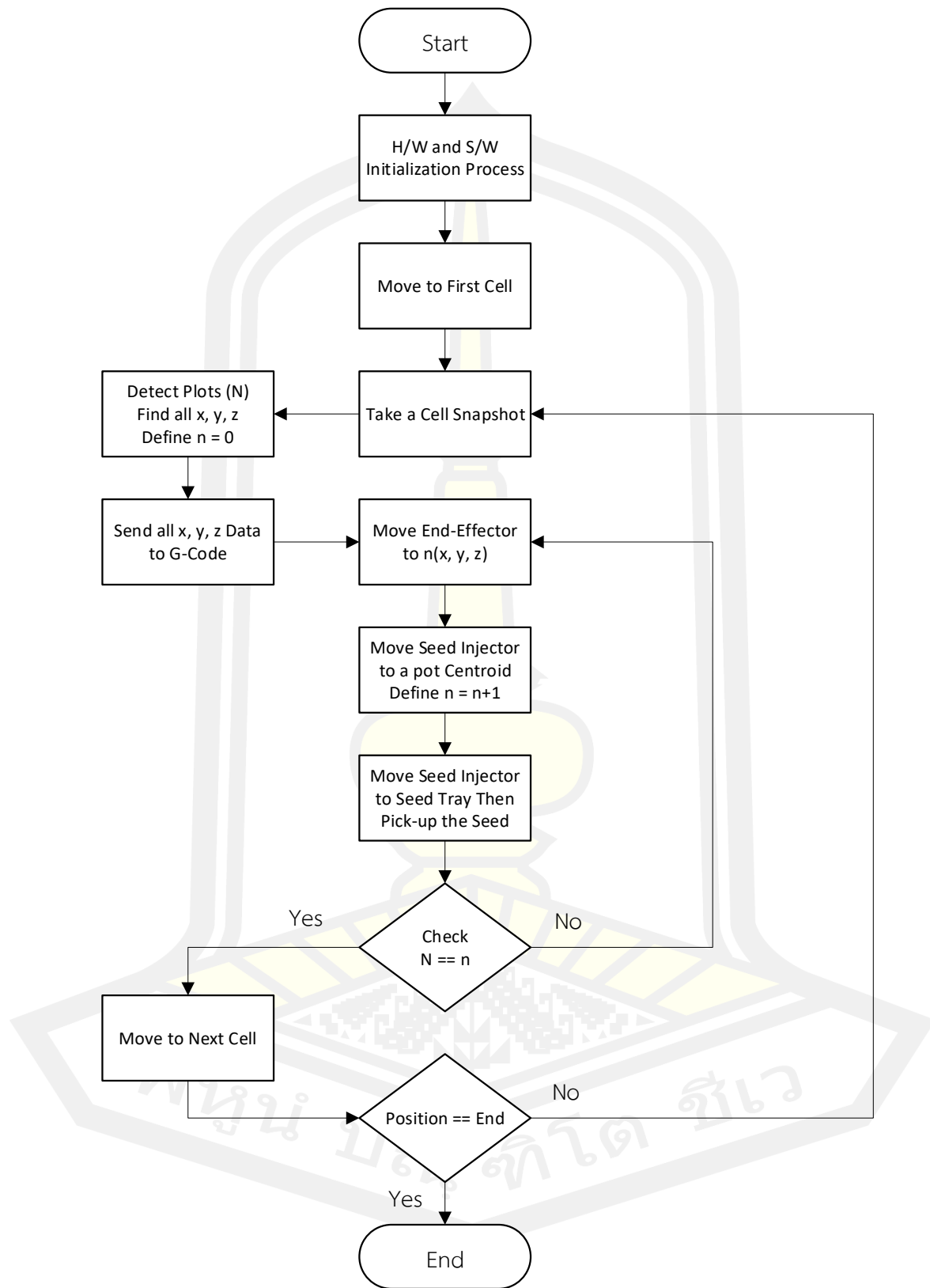


ภาพที่ 3.27 การออกแบบและพัฒนาซอฟต์แวร์ควบคุมการทำงาน

การพัฒนาอัลกอริทึมการทำงานของหุ่นยนต์ มีแนวคิดในการออกแบบดัง Flow Diagram ในภาพที่ 3.28 โดยด้านซอฟต์แวร์จะเริ่มตั้งแต่ขั้นตอนการนำหุ่นยนต์ไปติดตั้งที่ตำแหน่งจุดเริ่มต้น (Start Point) ในสภาพแวดล้อมที่ใช้ในการทดสอบ จากนั้นเมื่อเปิดการทำงานของระบบ หุ่นยนต์จะทำการ Initialization ระบบ โดยทำการกำหนดค่าตัวแปรต่างของซอฟต์แวร์ให้เป็นค่าเริ่มต้น ทดสอบการเชื่อมต่อการสื่อสารของ Serial Communication ระหว่างบอร์ด Raspberry Pi กับ Arduino และ Raspberry Pi กับ Motor Driver จากนั้นบอร์ดที่ควบคุมระบบอินเทอร์เนตของสรรพสิ่งจะเริ่มทำการ Initialization ระบบ แล้วเชื่อมต่อเข้ากับเครือข่าย GSM 3G ของ AIS เพื่อติดต่อไปยัง MQTT Broker หากการเชื่อมต่อไม่สำเร็จ ระบบก็จะ Loop เพื่อวนรอบกลับมาทำการเชื่อมต่ออีกครั้ง เมื่อการเชื่อมต่อเสร็จสิ้น ค่าของพารามิเตอร์ต่าง ๆ ในสภาพแวดล้อม ซึ่งอ่านได้จากชุด Sensor Unit ที่เชื่อมต่อกับระบบอินเทอร์เนตของสรรพสิ่งจะเริ่มทำการ Publish ข้อมูลไปยัง MQTT Server

พร้อมทั้งรอการส่งสัญญาณกลับมาควบคุมหุ่นยนต์ผ่านการ Subscribe จาก MQTT Broker ด้วยเช่นเดียวกัน

ด้านฮาร์ดแวร์ ระบบควบคุมจะทำการตรวจสอบ Limit Switch ของระบบขับเคลื่อนทั้ง 3 แกนว่า ชุดขับเคลื่อนของ End-Effector อยู่ในตำแหน่ง Home ( $x = 0, y = 0$  และ  $z = 0$ ) แล้วหรือไม่ หากมีแกนใดที่ยังไม่อยู่ในตำแหน่งดังกล่าว ระบบจะสั่งงานให้ชุดขับเคลื่อนของ End-Effector เคลื่อนที่กลับไปยังตำแหน่งของ Home เสียก่อน ซึ่งเป็นจุดที่ใช้อ้างอิงการทำงานของระบบ จากนั้นหุ่นยนต์จะเคลื่อนที่ตรงไปข้างหน้า (Forward) เพื่อเข้าสู่แปลงปลูก Cell แรก เมื่อเคลื่อนที่มาถึง Cell และปริภูมิของการทำงานของหุ่นยนต์ครอบคลุมพื้นที่ Cell แรกแล้ว หุ่นยนต์จะทำการถ่ายภาพ Snapshot ของ Cell เพื่อนำภาพที่ได้ขึ้นไปทำการประมวลผลภาพ ซึ่งจะเป็นการตรวจจับกระถางที่ถูกวางอยู่ใน Cell นั้น เพื่อดึงเอาค่าพารามิเตอร์ต่าง ๆ ออกมา ซึ่งประกอบด้วย จำนวนของกระถางทั้งหมด เก็บไว้ในตัวแปร  $N$  และพิกัดของจุดกึ่งกลาง (Centroid) ของกระถาง ที่เก็บเป็นค่า  $n(x, y, z)$  โดยที่  $n(x, y, z)$  หมายถึง ค่าพิกัด  $(x, y, z)$  ของกระถางใบที่  $n$  รวมถึงการกำหนดค่า  $n$  เริ่มต้นให้มีค่าเป็น 0 จากนั้นค่าทั้งหมดจะถูกส่งต่อไปยังส่วนของ G-Code เพื่อควบคุมการหมุนของ Stepping Motor 2 ชุด ที่ควบคุมการเคลื่อนที่ในระนาบ  $x$  และ  $y$  ให้นำ End-Effector เคลื่อนที่ไปยังตำแหน่ง  $n(x, y, z)$  ที่มี  $n = 0$  ในครั้งแรก ต่อมาจึงสั่งให้ Stepping Motor ที่ควบคุมแนวแกน  $z$  นำ Seed Injector เคลื่อนที่ลงมาตามแนวแกน  $z$  เพื่อหยอดเมล็ดพันธุ์ลงบนจุดกึ่งกลางของกระถาง แล้วทำการเพิ่มค่าของ  $n$  โดย กำหนดให้  $n = n+1$  ก่อนที่จะเคลื่อน End-Effector ไปยังตำแหน่งต่อไป จากนั้นจะทำการเปรียบเทียบค่า  $N$  กับ  $n$  ว่าเท่ากันหรือไม่ เพื่อตรวจสอบว่าหุ่นยนต์ได้ทำการปลูกครบทุกกระถางภายใน Cell นั้นเสร็จสิ้นแล้วหรือยัง โดยถ้า  $N == n$  เป็นเท็จ แสดงว่าหุ่นยนต์ยังทำการปลูกไม่ครบจะต้องกลับไปทำกระบวนการปลูกซ้ำอีกรอบ แล้ววนกลับมาตรวจสอบอีกครั้งจนกว่าเงื่อนไขจะเป็นจริง แต่ถ้า  $N == n$  เป็นจริง หุ่นยนต์จะทำการตรวจสอบทิศทางการเคลื่อนที่จากเซ็นเซอร์ ก่อนที่จะตัดสินใจว่าจะต้องเคลื่อนที่ต่อไปในทิศทางใด เมื่อหุ่นยนต์เคลื่อนที่ไปยัง Cell ที่อยู่ถัดไปเรียบร้อยแล้ว จะทำการตรวจสอบว่าตำแหน่งดังกล่าวเป็นจุดสิ้นสุดการทำงาน (End Point) ใช่หรือไม่ ถ้าพบว่าไม่ใช่จุดสิ้นสุดการทำงาน ระบบจะวนกลับไปเริ่มกระบวนการทำงานตั้งแต่การถ่ายภาพ Snapshot ของ Cell ใหม่อีกครั้ง แต่ถ้าตรวจสอบพบว่าเป็นจุดสิ้นสุดการทำงาน ระบบควบคุมจะสั่งให้หุ่นยนต์จะหยุดการทำงานในที่สุด



ภาพที่ 3.28 Flow Diagram แสดงโครงสร้างการทำงานของหุ่นยนต์ปลูกพืชอัตโนมัติ

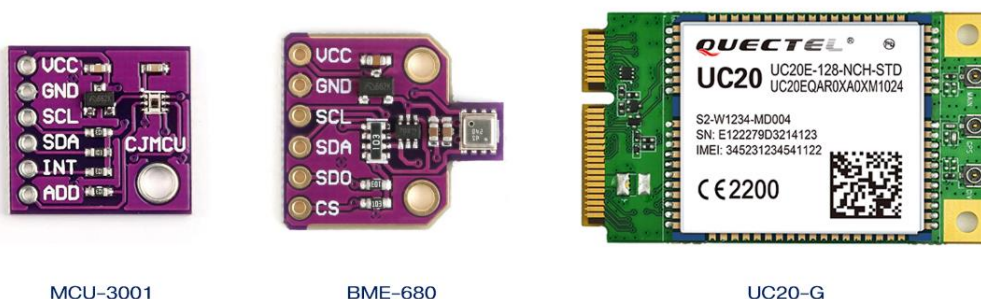
### 3.3 การออกแบบและพัฒนาระบบอินเทอร์เน็ตของสรรพสิ่ง

1. การออกแบบและพัฒนาระบบอินเทอร์เน็ตของสรรพสิ่ง เพื่อตรวจวัดค่าพารามิเตอร์ของสภาพแวดล้อมของโรงเรือนในขณะที่หุ่นยนต์ปลูกพืชอัตโนมัติกำลังทำงานนั้น ใช้ชุดของ Sensor Units จำนวน 3 Modules ซึ่งประกอบด้วย

1) MCU-3001 (OPT3001) Ambient Light Sensor เป็นเซ็นเซอร์วัดความเข้มแสง มีหน่วยวัดเป็น LUX สามารถวัดความเข้มของแสงได้ช่วงกว้างตั้งแต่ 0.01 lux ถึง 83,000 lux

2) BME-680 Low Power Gas, Pressure, Temperature และ Humidity Sensor เป็นเซ็นเซอร์ที่มีความสามารถในการตรวจวัดค่าจากสภาพอากาศได้ 4 ประเภท ประกอบด้วย อุณหภูมิ (หน่วยเป็นองศาเซลเซียส) ความชื้น (หน่วยเป็นเปอร์เซ็นต์ความชื้นในอากาศ) ความดันอากาศ (หน่วยเป็น hPa) และคุณภาพของอากาศ (หน่วยเป็น  $m\Omega$ )

3) Quectel UC20-G เป็นอุปกรณ์ที่ได้รับการออกแบบให้สามารถใช้งานเทคโนโลยี HSPA+/WCDMA และรองรับการสื่อสารแบบ GSM/GPRS/EDGE ในระบบ 3G ดั้งเดิม นอกจากนี้ยังมีโมดูล Global Positioning System (GNSS Receiver Module) เพื่อใช้ระบุตำแหน่งจากระบบดาวเทียม GPS ในตัว



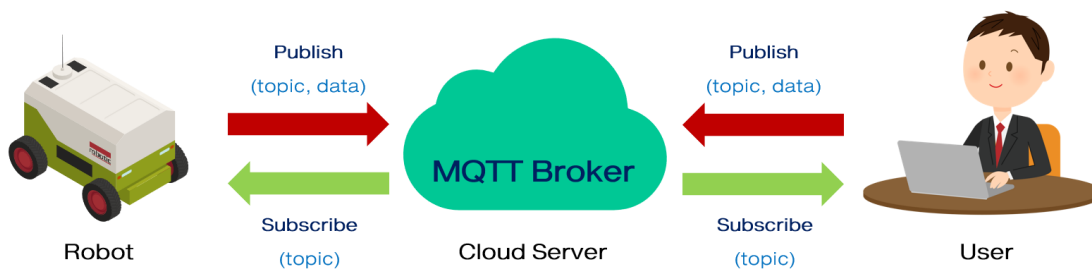
MCU-3001

BME-680

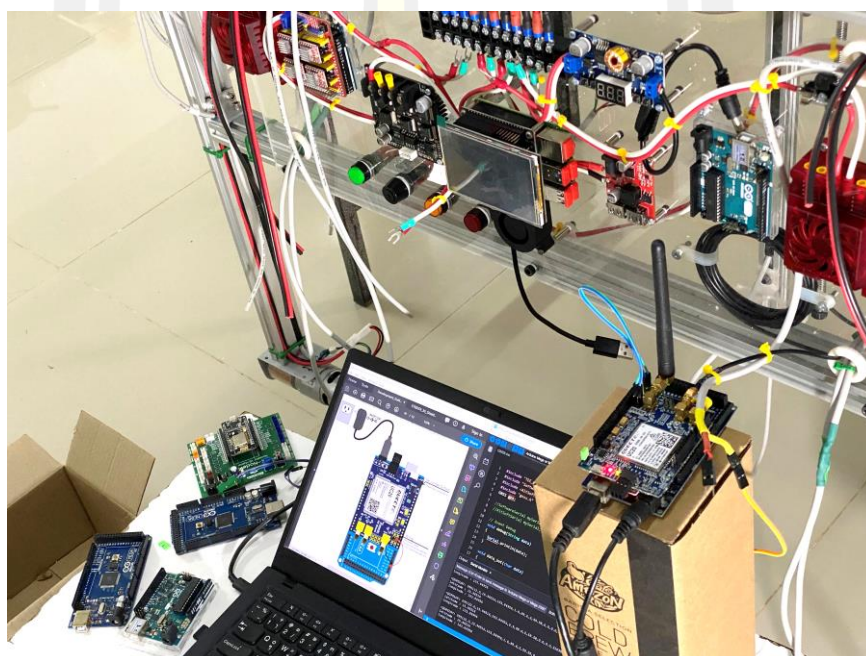
UC20-G

ภาพที่ 3.29 ชุด Sensor Units ที่ใช้ในงานวิจัย

โดยข้อมูลของสภาพแวดล้อมที่ได้รับจากทั้ง 3 Module นี้ จะถูกนำมาสื่อสารในแบบ Real Time ด้วย MQTT Protocol ของ Internet of Things ผ่านเครือข่ายอินเทอร์เน็ตโดยใช้ HSPA+ บนเครือข่ายโทรศัพท์เคลื่อนที่ เพื่อทำการ Publish และ Subscribe ข้อมูลของ Topics ที่ MQTT Broker บนระบบ Cloud Computing ก่อนถูกนำไปควบคุมการทำงานของหุ่นยนต์ และแสดงผลข้อมูลดังกล่าวต่อบนแพลตฟอร์มอื่น เช่น คอมพิวเตอร์ แท็บเล็ต หรือสมาร์ทโฟน ต่อไป



ภาพที่ 3.30 รูปแบบการทำงานของ MQTT Protocol



ภาพที่ 3.31 การติดตั้งระบบ IoT เพื่อรับส่งข้อมูลผ่าน HSPA+ บนเครือข่าย 3G

2. การทดสอบระบบอินเทอร์เน็ตของสรรพสิ่ง ซึ่งในการทดสอบมี 2 รูปแบบ ประกอบด้วยรูปแบบที่ 1 จะทำการ Publish ข้อมูลจาก MCU-3001 BME-680 และ UC20-G ที่ถูกติดตั้งไว้บนหุ่นยนต์ ด้วย MQTT Protocol ผ่านเครือข่ายอินเทอร์เน็ตบนระบบ HSPA+ (3G) ของ Advanced Info Service (AIS) ไปยัง MQTT Broker ของผู้ให้บริการ CloudMQTT เพื่อส่งข้อมูลพารามิเตอร์ที่วัดได้จากสภาพแวดล้อมของเซ็นเซอร์แต่ละตัวไปยัง Topics ที่ถูกกำหนดไว้ และรูปแบบที่ 2 หุ่นยนต์ จะทำการ Subscribe ข้อมูลของ Topic ที่ชื่อ “status” ด้วย MQTT Protocol ผ่านเครือข่ายอินเทอร์เน็ตบนระบบ HSPA+ (3G) ของ Advanced Info Service (AIS) จาก MQTT Broker ของผู้ให้บริการ CloudMQTT เพื่อนำมาใช้ในการควบคุมการทำงานของ Actuators

### ตารางที่ 3.3 เงื่อนไขในการประเมินการทำงานของระบบอินเทอร์เน็ตของสรรพสิ่ง

Number of topics	9
Number of publishers	3
Number of subscribers	3
Payload	16 Bytes

โดยมีรายละเอียดของเงื่อนไขในการประเมินดังต่อไปนี้

- 1) Topic ที่ใช้ในการ Publish มีจำนวน 8 Topics คือ temp (อุณหภูมิ), humid (ความชื้นสัมพัทธ์), pressure (ความกดอากาศ), altimeter (ความสูงจากระดับน้ำทะเล), voc (คุณภาพของอากาศ), light (ความเข้มแสง), lat (พิกัด GPS-Latitude) และ long (พิกัด GPS-Longitude)
- 2) Topic ในการ Subscribe จำนวน 1 Topic คือ status (สถานะ)
- 3) Publisher จำนวน 3 Units ประกอบด้วย หุ่นยนต์เพาะปลูกพืช คอมพิวเตอร์ และสมาร์ตโฟน
- 4) Subscriber จำนวน 3 Units ประกอบด้วย หุ่นยนต์เพาะปลูกพืช คอมพิวเตอร์ และสมาร์ตโฟน
- 5) Payload มีขนาด 16 Bytes

The screenshot shows the Arduino IDE interface with the MQTTCloud beta 1.2.10 code loaded. The code includes headers for hardware and MQTT, and defines the APN as 'internet'. The serial monitor window displays the following data:

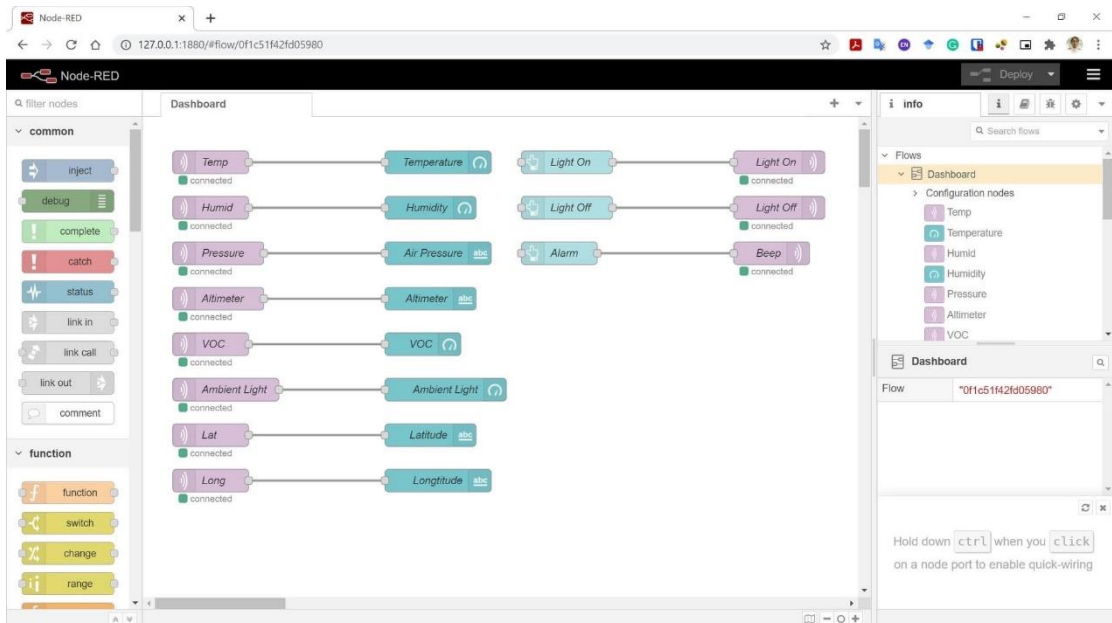
```

- Initializing OPT3001 Ambient Light Sensor
- OPT3001 Manufacturer: 1021577
- OPT3001 Device: 1022255
BoP3001 Current Config:
-----
Conversion ready (R/W):0
Conversion time (R/W):0
Fault count field (R/W):0
Flag high field (R-only):0
Flag low field (B-only):0
Latch field (R/W):1
Mask exponent field (R/W):0
Mode of conversion operation (R/W):3
Polarity field (R/W):0
Overflow flag (R-only):0
Range number (R/W):0
-----
- Initializing BMX680 sensor
- Setting 16x oversampling for all sensors
- Setting IIR filter to a value of 4 samples
- Setting gas measurement to 320°C for 150ms

Loop Temp°C Humid% Press hPa Alt m Air mC lux Lat Long
1 29.13 67.685 997.78 129.60 252.53 38.73 15.96511 103.84954
2 29.38 67.764 997.74 129.54 324.69 38.58 15.96513 103.84954
3 29.32 67.646 997.76 129.77 309.35 38.50 15.96516 103.84955
4 29.27 67.739 997.80 129.43 311.60 38.50 15.96516 103.84955
5 29.23 67.746 997.80 129.43 317.31 38.58 15.96516 103.84955
6 29.20 67.768 997.80 129.43 322.44 38.50 15.96516 103.84955
7 29.19 67.723 997.82 129.26 324.20 38.50 15.96517 103.84956
8 29.18 67.690 997.78 129.60 328.50 38.50 15.96518 103.84958
9 29.18 67.669 997.80 129.43 330.31 38.50 15.96518 103.84958

Loop Temp°C Humid% Press hPa Alt m Air mC lux Lat Long
10 29.17 67.611 997.82 129.26 332.94 38.50 15.96519 103.84957
11 29.17 67.684 997.82 129.26 334.80 38.50 15.96520 103.84957
12 29.16 67.684 997.80 129.43 334.80 38.50 15.96520 103.84957
13 29.16 67.684 997.80 129.43 336.41 38.50 15.96520 103.84957
  
```

ภาพที่ 3.32 การเขียนโค้ดเพื่อ Publish และ Subscribe ไปยัง MQTT Broker



ภาพที่ 3.33 การพัฒนา Dashboard ของระบบด้วย Node-RED

ขั้นตอนการประเมินการทำงานของระบบอินเทอร์เน็ตของสรรพสิ่งมีรายละเอียดของการประเมิน ประกอบด้วย

1) ทำการเชื่อมต่อหุ่นยนต์เพาะปลูกเข้ากับ Server ของผู้ให้บริการ MQTT Broker (driver.cloudmqtt.com) ด้วยระบบ HSPA+ (3G) จากนั้นจึงทำการเชื่อมต่อคอมพิวเตอร์กับ MQTT Broker ผ่านเครือข่าย Wi-Fi และสมาร์ตโฟนผ่านเครือข่าย LTE (4G)

2) ทดสอบการ Publish ข้อมูลของค่าพารามิเตอร์ในสภาพแวดล้อมที่วัดโดยชุดเซ็นเซอร์ MCU-3001 BME-680 และ UC20-G ที่ถูกติดตั้งบนหุ่นยนต์เพาะปลูกพืช โดยทำการ Publish ข้อมูลจำนวน 8 Topics ที่มีขนาดของ Payload 16 Bytes (ขนาดของ Payload อาจจะมีขนาดเล็กกว่าที่กำหนด ซึ่งขึ้นอยู่กับขนาดของข้อมูลที่ได้รับจาก Sensor Units แต่ละตัว) กำหนดอัตรา Refresh Rate ทุก 5 วินาที จากนั้นประเมินผลว่าการ Publish ในแต่ละครั้งส่งข้อมูลได้ครบถ้วนถูกต้องและสำเร็จหรือไม่

3) ทดสอบการ Publish ข้อมูลสถานะใน Topic ชื่อ “status” จากคอมพิวเตอร์และสมาร์ตโฟน ไปยัง driver.cloudmqtt.com บน Amazon Web Services โดยทางด้านของหุ่นยนต์เพาะปลูกจะถูกกำหนดให้ทำการ Subscribe ใน Topic ชื่อ “status” เพื่อคอยรับ Payload จาก MQTT Broker มาประมวลผลและนำไปควบคุม Actuators ต่อ ซึ่งมีรูปแบบในการควบคุมการทำงานของหุ่นยนต์เพาะปลูกดังตารางที่ 3.4



ตารางที่ 3.4 รูปแบบในการควบคุมการทำงานของ Topic “status”

Payload	Action
on	สั่งงานให้หุ่นยนต์เปิดหลอดไฟ LED
off	สั่งงานให้หุ่นยนต์ปิดหลอดไฟ LED
beep	สั่งงานให้หุ่นยนต์ส่งสัญญาณเสียงผ่าน Active Buzzer

4) สรุปผลการทดสอบการ Publish และ Subscribe ข้อมูลไปยัง MQTT Broker บนระบบ Cloud Computing

โดยเกณฑ์การประเมินที่ถูกใช้เพื่อพิจารณาผลลัพธ์ที่เกิดจากการ Publish และ Subscribe ด้วย Protocol MQTT ผ่านเครือข่ายอินเทอร์เน็ตโดยใช้ HSPA+ บนเครือข่ายโทรศัพท์เคลื่อนที่ไปยัง MQTT Broker มีเกณฑ์การประเมินดังต่อไปนี้

1) การการส่งข้อมูลสำเร็จ หมายถึง การที่ MQTT Client (หุ่นยนต์เพาะปลูกพืช คอมพิวเตอร์ และสมาร์ทโฟน) สามารถทำการ Publish ข้อความจาก Sensor Units ทั้ง 8 Topics ไปยัง MQTT Broker ได้ มีความถูกต้องตรงกันของข้อมูลทั้งบน MQTT Client และ MQTT Broker โดยพิจารณาผลลัพธ์ที่ปรากฏบน MQTT Terminal และ WebSocket UI

2) การการส่งข้อมูลไม่สำเร็จ หมายถึง การที่ MQTT Client ไม่สามารถทำการ Publish ข้อความจาก Sensor Units ทั้ง 8 Topics ไปยัง MQTT Broker ได้ หรือส่งข้อมูลได้แต่ไม่ครบถ้วน รวมถึงข้อมูลบน MQTT Client และ MQTT Broker ไม่ถูกต้องตรงกัน โดยพิจารณาผลลัพธ์ที่ปรากฏบน MQTT Terminal และ WebSocket UI

3) การรับข้อมูลสำเร็จ หมายถึง การที่ MQTT Client สามารถทำการ Subscribe ข้อความจาก Sensor Units ทั้ง 8 Topics ได้ มีความถูกต้องตรงกันของข้อมูลทั้งบน MQTT Client และ MQTT Broker โดยพิจารณาผลลัพธ์ที่ปรากฏบน MQTT Terminal และ Dashboard UI

4) การรับข้อมูลไม่สำเร็จ หมายถึง การที่ MQTT Client ไม่สามารถทำการ Subscribe ข้อความจาก Sensor Units ทั้ง 8 Topics ได้ หรือรับข้อมูลได้แต่ไม่ครบถ้วน รวมถึงข้อมูลบน MQTT Client และ MQTT Broker ไม่ถูกต้องตรงกัน โดยพิจารณาผลลัพธ์ที่ปรากฏบน MQTT Terminal และ Dashboard UI

5) การควบคุมการทำงานสำเร็จ หมายถึง การที่ MQTT Client (เฉพาะหุ่นยนต์เพาะปลูกพืช) สามารถทำการ Subscribe ใน Topic ชื่อ “status” จาก MQTT Broker โดยมี MQTT Client อื่น (คอมพิวเตอร์และสมาร์ทโฟน) ทำหน้าที่ Publish ข้อมูลคำสั่งในการควบคุม

หุ่นยนต์ไปยัง MQTT Broker และเมื่อหุ่นยนต์เพาะปลูกได้รับ Payload ซึ่งเป็นคำสั่งควบคุมจาก MQTT Broker แล้วสามารถทำงานได้ถูกต้องตามตารางที่ 3.4

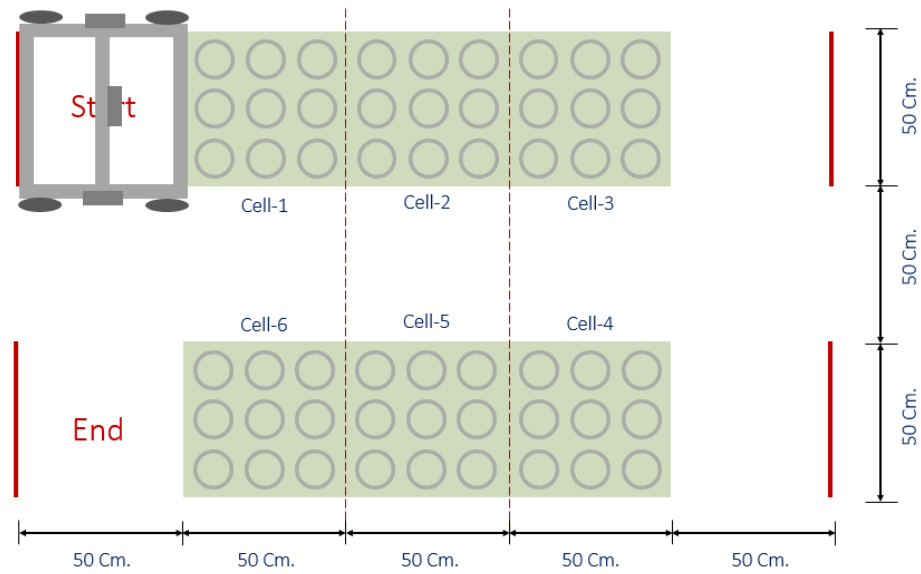
6) การควบคุมการทำงานล้มเหลว หมายถึง การที่ MQTT Client (เฉพาะหุ่นยนต์เพาะปลูกพืช) ไม่สามารถทำการ Subscribe ใน Topic ชื่อ “status” จาก MQTT Broker หรือสามารถทำการ Subscribe แต่เมื่อได้รับ Payload แล้วไม่สามารถทำงานได้ถูกต้อง

7) เวลาในการตอบสนอง (Latency Time) หมายถึง การวัดระยะเวลาที่ระบบใช้ในการส่งข้อความจาก Publisher ไปยัง Subscriber มีหน่วยเป็นมิลลิวินาที ในทดสอบจะจำลองสถานการณ์ของ MQTT Client (หุ่นยนต์เพาะปลูกพืช คอมพิวเตอร์ และสมาร์ทโฟน) ที่ทำการ Publish ข้อความไปยัง MQTT Broker แล้วทำการวัดค่าเวลาที่ใช้ในการส่งข้อมูลจาก MQTT Client ไปยัง MQTT Broker ซึ่งรวมถึงเวลาที่ตอบกลับมาด้วย

### 3.4 สภาพแวดล้อมที่ใช้ในการทดสอบ

หุ่นยนต์ต้นแบบจะถูกทดสอบการทำงานบนสภาพแวดล้อมจำลอง ดังแสดงในภาพที่ 3.34 จำนวน 20 ครั้ง เพื่อทำการประเมินประสิทธิภาพ โดยสภาพแวดล้อมในการทดสอบ (Testbed) นี้จะประกอบด้วยกระถางปลูกแบบกลม สีดำ ขนาด 5 นิ้ว ดังภาพที่ 3.35 ซึ่งถูกจัดวางไว้ในแปลงปลูกพืชขนาด ความกว้าง 50 เซนติเมตร ความยาว 150 เซนติเมตร จำนวน 2 แปลง ติดกัน โดยมีระยะห่างระหว่างแปลง 50 เซนติเมตร ซึ่งแต่ละแปลงจะถูกแบ่งออกเป็น Cell ย่อย ๆ จำนวน 3 Cell ขนาด ความกว้าง 50 เซนติเมตร ความยาว 50 เซนติเมตร และ Cell

ในการทดสอบการทำงาน หุ่นยนต์จะถูกติดตั้งในจุดเริ่มต้น (Start Point) ซึ่งตั้งอยู่บริเวณด้านซ้ายมือของ Cell-1 จากนั้นเมื่อทำการเปิดการทำงานของระบบ หุ่นยนต์จะเคลื่อนที่ไปยัง Cell-1 เพื่อทำการตรวจจับกระถางที่ถูกวางไว้บน Cell ค้นหาจุดกึ่งกลางของกระถาง แล้วทำการหยอดเมล็ดลงในกระถาง เมื่อหุ่นยนต์หยอดเมล็ดลงในกระถางครบแล้วก็จะเคลื่อนที่ต่อไปยัง Cell-2 ซึ่งอยู่ถัดไป พร้อมทั้งทำการตรวจจับกระถาง หาจุดกึ่งกลาง และหยอดเมล็ดลงในกระถางเหมือนเดิมทำงานแบบนี้วนไปจนกว่าจะถึง Cell-6 และเมื่อเซ็นเซอร์ตรวจจับได้ว่าหุ่นยนต์เคลื่อนที่มาถึงจุดสิ้นสุด (End Point) ระบบจึงจะสั่งให้หุ่นยนต์หยุดการทำงาน



ภาพที่ 3.34 สภาพแวดล้อมที่ใช้ในการทดสอบ



ภาพที่ 3.35 ภาชนะพลาสติกที่ใช้ในการทดสอบ

### 3.5 การประเมินประสิทธิภาพ

การประเมินประสิทธิภาพ (Performance Evaluation) ในการวิจัยครั้งนี้จะทำการประเมินประสิทธิภาพ 2 ด้าน ประกอบด้วย การวัดและการทดสอบความสามารถ (Capabilities) และพฤติกรรม (Behaviors) ของหุ่นยนต์ที่ถูกพัฒนาขึ้น บน Test-bed ที่ถูกจำลองมาจากสภาพแวดล้อมการทำงาน (Real Environment) ของเงื่อนไขจริง ซึ่งการทดสอบในลักษณะดังกล่าว นั้น มีความต้องการวัสดุ อุปกรณ์ โค้ด เงื่อนไขและเวลาที่ใช้ในการทำการทดลองจริง โดยมีการเฝ้าระวัง (Monitor) เพื่อสังเกตผลการทดลองเป็น Key Point สำคัญ ซึ่งจะได้ผลลัพธ์ที่เหมือนจริงจากการประเมินประสิทธิภาพของหุ่นยนต์ต้นแบบที่ถูกนำมาทำการทดลอง ดังนั้น Prototype จึงมีความสำคัญในการนำมาใช้ทดสอบ ซึ่งการประเมินประสิทธิภาพ ประกอบด้วยตัวชี้วัด (Metrics) ดังนี้

1. ความเร็ว (Velocity) อัตราความเร็วในการเคลื่อนจากจุดหนึ่งไปยังอีกจุดหนึ่ง ณ ขณะเวลานั้นของหุ่นยนต์ต้นแบบ

2. ความแม่นยำ (Accuracy) อัตราความแม่นยำเฉลี่ยในการปลูก โดยประเมินจากความแม่นยำของการหยอดเมล็ดพันธุ์ลงในกระถางปลูกบนจุด Centroid ที่กำหนด ภายใต้การกำหนด Error Rate ที่ยอมรับได้ของหุ่นยนต์ต้นแบบ

3. ระยะเวลาในการทำภารกิจ (Mission Duration) ระยะเวลาในการทำภารกิจตั้งแต่เริ่มเคลื่อนที่จากจุดเริ่มต้นเข้าไปยังแปลงปลูก ตรวจสอบกระถางในแปลง ประมวลผลข้อมูล ดำเนินการหยอดเมล็ดพันธุ์ลงในกระถาง ทำซ้ำกระบวนการหยอดเมล็ดจนเสร็จสิ้นครบตามจำนวนกระถางที่ตรวจสอบได้ และเคลื่อนที่ไปหยุดนิ่งที่จุดสิ้นสุดของหุ่นยนต์ต้นแบบ

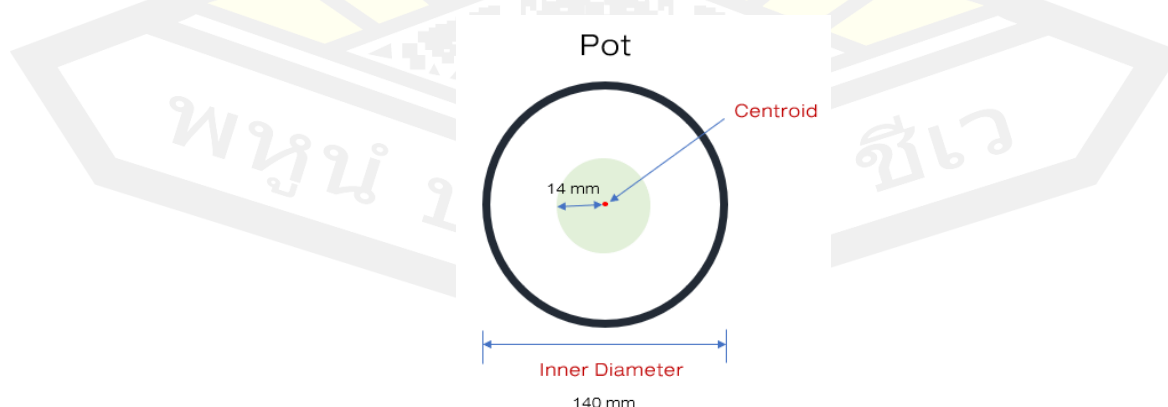
4. อัตราความสำเร็จของภารกิจ (Mission Success Rate) อัตราส่วนของจำนวนกระถางที่หุ่นยนต์ต้นแบบทำการหยอดเมล็ดพันธุ์กับจำนวนกระถางที่มีอยู่จริงในแปลงปลูก

### 3.6 เกณฑ์ในการประเมิน

เกณฑ์การประเมินที่ถูกใช้เพื่อพิจารณาผลลัพธ์ที่เกิดจากการทดสอบหุ่นยนต์ต้นแบบ มีเกณฑ์การประเมินดังต่อไปนี้

1) การหยอดเมล็ดสำเร็จ หมายถึง การที่หัวปลูกของหุ่นยนต์เคลื่อนที่ไปยังตำแหน่งของกระถางและสามารถหยอดเมล็ดพันธุ์ลงในกระถางได้ถูกต้อง โดยมีความคลาดเคลื่อน  $\pm 20\%$  จากจุดกึ่งกลางของกระถางที่ตำแหน่ง 70 มิลลิเมตร ภายในพื้นที่สีเขียวดังภาพที่ 3.36

2) การหยอดเมล็ดไม่สำเร็จ หมายถึง การที่หัวปลูกของหุ่นยนต์เคลื่อนที่ผ่านกระถางไปโดยไม่ทำการหยอดเมล็ด หรือมีการเคลื่อนที่ไปยังตำแหน่งของกระถางแต่หยอดเมล็ดนอกกระถาง หรือมีความคลาดเคลื่อนมากกว่า  $\pm 20\%$  จากจุดกึ่งกลางของกระถาง



ภาพที่ 3.36 ตำแหน่งของการหยอดเมล็ดบนกระถางตามเงื่อนไขในการทดสอบ

3) การทำภารกิจสำเร็จ หมายถึง การที่หุ่นยนต์ต้นแบบสามารถเคลื่อนที่ไปทำภารกิจในสภาพแวดล้อมที่ถูกกำหนดไว้และทำการหยุดเมสึดได้สำเร็จตามเงื่อนไข

4) การทำภารกิจไม่สำเร็จ หมายถึง การที่หุ่นยนต์ต้นแบบไม่สามารถเคลื่อนที่ หรือเคลื่อนที่ผิดพลาดจนหลุดออกจากเส้นทางในการทำภารกิจ หรือทำภารกิจในสภาพแวดล้อมที่ถูกกำหนดไว้ไม่สำเร็จครบถ้วนตามเงื่อนไข

### 3.7 เครื่องมือที่ใช้ในการวิจัย

เครื่องมือที่ใช้ในการวิจัย แบ่งออกเป็น 2 ส่วน ประกอบด้วย

#### 1. ฮาร์ดแวร์ (Hardware)

- 1) บอร์ด Raspberry Pi 4 Model B (RAM 8 GB)
- 2) บอร์ด Camera Module Version 2
- 3) บอร์ด MCU-Cam
- 3) บอร์ด Arduino Mega 2560
- 4) บอร์ด Motor Drive Module
- 5) DC Gear Motor with Encoder
- 6) Stepping Motor
- 7) Sensors และ Actuator
- 8) DC Battery
- 9) Mechanical Parts
- 10) Aluminum Profile

#### 2. ซอฟต์แวร์ (Software)

- 1) Raspbian (Kernel version 4.14)
- 2) Python (Version 3.7)
- 3) OpenCV (Version 3.4.6)
- 4) Arduino IDE 1.8.10
- 5) Geometric Code
- 6) Node.js (Version: 16.13.1)
- 7) Node-RED (Version: 2.1.4)
- 8) Autodesk Fusion 360
- 9) Ultimaker Cura (Version 4.11.0)
- 10) Open Builds Control (Version 1.0319)

## บทที่ 4

### ผลการวิจัย

งานวิจัยนี้มีจุดมุ่งหมายเพื่อพัฒนาต้นแบบหุ่นยนต์ที่ใช้สำหรับทำการเพาะปลูกพืชบนกระถางที่ถูกจัดเรียงไว้ในโรงเรือน ซึ่งประยุกต์การทำงานของหุ่นยนต์เข้ากับการประมวลผลภาพและปัญญาประดิษฐ์ รวมถึงการสื่อสารข้อมูลด้วยเทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง ให้สามารถทำงานทดแทนแรงงานมนุษย์ ลดเวลา ลดความยุ่งยากในภาระงานที่ต้องทำเป็นประจำได้

ในบทนี้จะกล่าวถึงผลการประเมิน ผลการออกแบบ และผลการพัฒนาต้นแบบ โดยผลการออกแบบและพัฒนาจะประกอบด้วย ระบบการเคลื่อนที่ ระบบการรับรู้ ระบบการกระทำ ระบบอินเทอร์เน็ตของสรรพสิ่ง และซอฟต์แวร์ควบคุมการทำงานของหุ่นยนต์ โดยมีรายละเอียดดังนี้

#### 4.1 ผลการออกแบบและพัฒนาระบบการเคลื่อนที่

1) การออกแบบและพัฒนาระบบควบคุมการทำงานของหุ่นยนต์ในแนวระนาบ  
การออกแบบและพัฒนาระบบควบคุมการทำงานของหุ่นยนต์ในแนวระนาบ ซึ่งต้นแบบของการพัฒนานั้น จะทำการออกแบบและสร้างหุ่นยนต์คาร์ทีเซียน ที่มีการเคลื่อนที่ตามแนวแกน  $x$



ภาพที่ 4.1 โครงสร้าง Aluminum Profile ของหุ่นยนต์

โดยในระยะแรกนี้ ผู้วิจัยทำการพัฒนาหุ่นยนต์ให้สามารถเคลื่อนที่ตามความยาวของแปลงพืชภายในโรงเรือนก่อน ระยะต่อไปจึงเป็นการเคลื่อนที่ตามความกว้าง และเคลื่อนที่ขึ้นลงตามแนวตั้ง โดยเลือกใช้วัสดุเป็น Aluminum Profile มาเป็นโครงสร้างต้นแบบก่อน เนื่องจากเป็นวัสดุที่สามารถนำมาตัดแต่ง ขึ้นรูปได้ง่าย สามารถนำมาประยุกต์ใช้ได้หลากหลาย มีน้ำหนักเบา ไม่เป็นสนิม และประกอบได้ง่าย

เมื่อทำการประกอบโครงสร้าง Aluminum Profile ของหุ่นยนต์ต้นแบบเสร็จเรียบร้อยแล้ว จึงทำการเจาะรูเฟรมทั้ง 4 มุม เพื่อทำการยึดชุดเกียร์มอเตอร์และล้อ ดังภาพที่ 4.2 และภาพที่ 4.3 แล้วทำการตรวจสอบ Balance โดยการวางบนพื้นราบแล้วตรวจสอบความเรียบร้อยด้วยเครื่องมือวัดระดับน้ำ



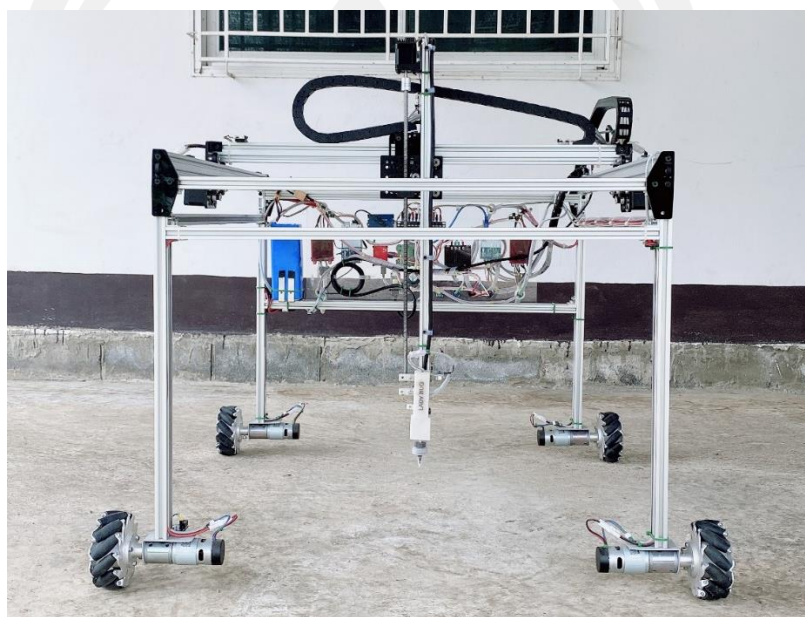
ภาพที่ 4.2 การติดตั้งเฟรมสำหรับยึดเกียร์มอเตอร์



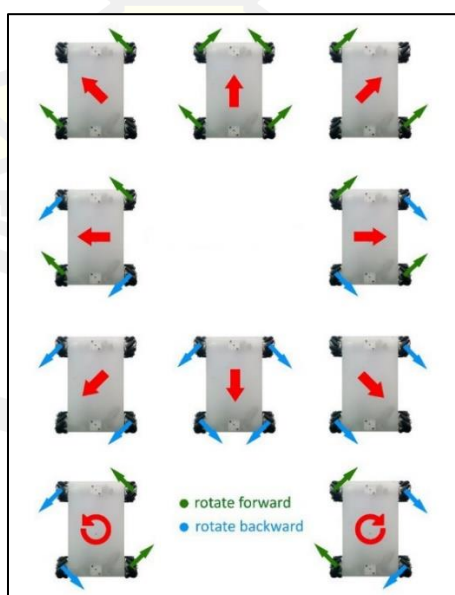
ภาพที่ 4.3 การติดตั้งชุดขับเคลื่อนเข้ากับเฟรมของหุ่นยนต์

หลังจากที่ได้ทำการประกอบต้นแบบหุ่นยนต์และทำการทดสอบการเคลื่อนที่พบว่า โครงสร้างที่ออกแบบนั้น มีความแข็งแรง สามารถรับน้ำหนักได้ดี เพราะเป็น Aluminum Profile ที่ผลิตจากวัสดุ 6063-T5 ที่สามารถรับน้ำหนักแนวตั้งได้สูงสุด 120 กิโลกรัมต่อเมตร และรับน้ำหนักแนวนอนได้สูงสุด 30 กิโลกรัมต่อเมตร จึงไม่มีปัญหาการทรึงตัวเมื่อต้องรับน้ำหนักที่จะกระทบต่อความแม่นยำในการควบคุมตำแหน่งได้ ซึ่งมีแนวคิดในการออกแบบโครงสร้างด้วยการนำ Gantry

Robot มาทำงานร่วมกับ Mobile Robot ดังภาพที่ 4.4 จากนั้นนำต้นแบบของหุ่นยนต์มาทำการทดสอบการเคลื่อนที่ในรูปแบบต่าง ๆ ด้วยการ Simulation การหมุนของเกียร์มอเตอร์ทั้ง 4 ชุด ตามเงื่อนไขในภาพที่ 4.5 เสียก่อน ว่ามีความถูกต้องหรือไม่ก่อนการนำไปทดสอบเคลื่อนที่บนพื้นผิวจริง เพื่อลดความผิดพลาดและความเสียหายของโครงสร้างหุ่นยนต์หากเกิดอุบัติเหตุจากการเคลื่อนที่ที่ไม่สามารถควบคุมได้

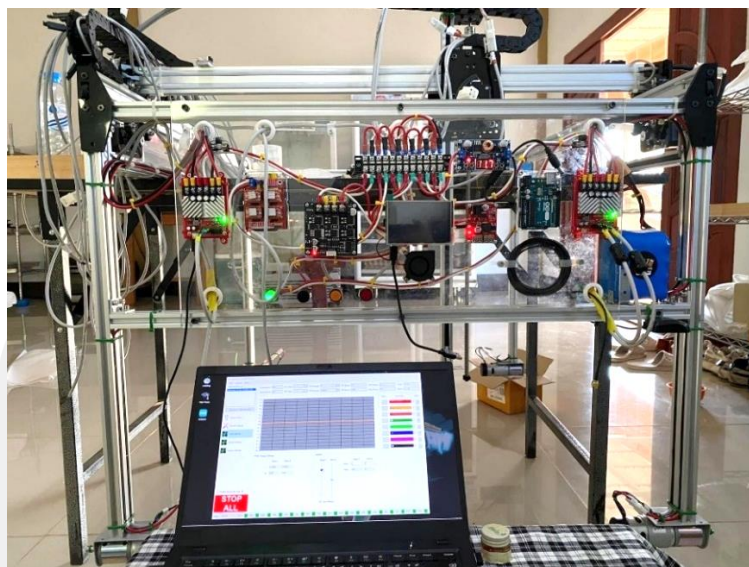


ภาพที่ 4.4 ต้นแบบหุ่นยนต์เพื่อใช้ในการเพาะปลูกแบบอัตโนมัติ



ภาพที่ 4.5 รูปแบบเงื่อนไขในการทดสอบการเคลื่อนที่ของหุ่นยนต์





ภาพที่ 4.6 ทดสอบการเคลื่อนที่ของหุ่นยนต์ด้วยการ Simulation บน Trestles

ตารางที่ 4.1 ผลการทดสอบการเคลื่อนที่ของหุ่นยนต์

Movement Pattern	Mecanum Wheel Movement
Forward	Pass
Backward	Pass
Right	Pass
Left	Pass
Forward Right	Pass
Forward Left	Pass
Backward Right	Pass
Backward Left	Pass
Turning Right	Pass
Turning Left	Pass

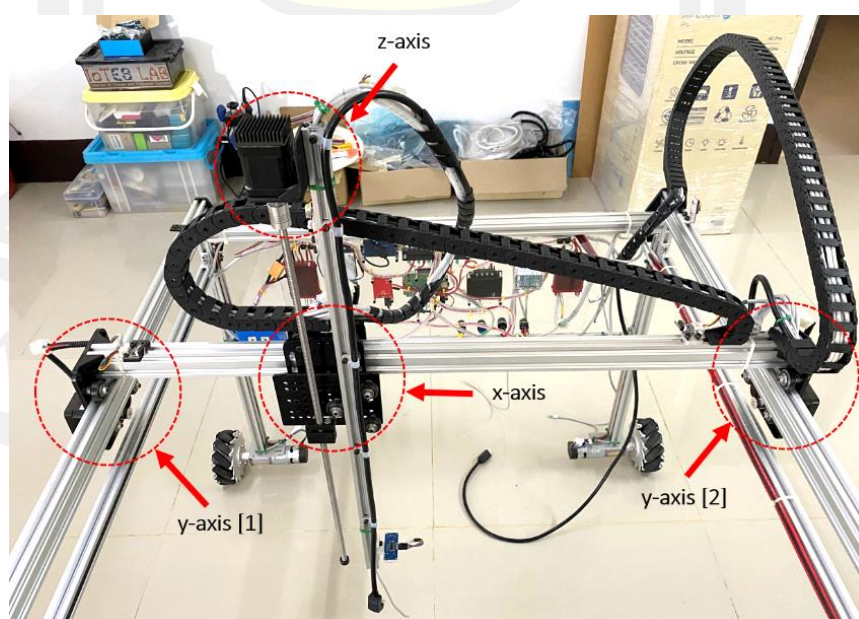
Pass = สามารถเคลื่อนที่ได้ถูกต้อง Not Pass = ไม่สามารถเคลื่อนที่ได้ถูกต้อง

จากผลการทดลองในตารางที่ 4.1 พบว่าการเคลื่อนที่ของหุ่นยนต์ที่ใช้ล้อแบบ Mecanum Wheel เป็นระบบขับเคลื่อนนั้น มีความพิเศษเฉพาะตัว สามารถเคลื่อนที่รอบทิศทางได้อย่างอิสระ โดยอาศัยชุด Roller ที่ติดตั้งทำมุม 45 องศา ซึ่งอยู่รอบ ๆ ชุดเฟรมล้อ เพื่อรองรับการ

หมุนด้านข้าง ซึ่งในการควบคุมการเคลื่อนที่ที่จะต้องมีการควบคุมการทำงานแยกกันในแต่ละล้อ เพราะในการเคลื่อนที่ไปในทิศทางใดใดจะต้องใช้ความสัมพันธ์ในการหมุนของล้อทั้งสี่ล้อเอง ซึ่งในระหว่างการทดสอบนั้นพบว่า ล้อแต่ละข้างของ Mecanum Wheel นั้น Roller จะมีการเอียงสลับไปทางด้านซ้ายและทางด้านขวา โดยมีเพลลาที่ส่วนบนของล้อขนานกับแนวทแยงของโครงสร้างของหุ่นยนต์ จึงตั้งฉากกับแนวทแยงเมื่อส่วนล่างของล้อสัมผัสกับพื้น ด้วยหลักการดังกล่าว แต่ละล้อจะสร้างแรงขับเคลื่อนกับเส้นทแยงมุมของเฟรมที่สอดคล้องกัน โดยมีการเปลี่ยนแปลงความเร็วในการหมุนและทิศทางของล้อ ที่เป็นผลรวมของเวกเตอร์แรงจากแต่ละล้อ จึงส่งผลให้เกิดการเคลื่อนที่หาเชิงเส้นรวมถึงการหมุนของโครงสร้างหุ่นยนต์

## 2) การออกแบบและพัฒนาระบบควบคุมการทำงานของหุ่นยนต์ในแนวตั้ง

โดยการประยุกต์ใช้รูปแบบการทำงานของ Gantry Robot ซึ่งมีลักษณะทางโครงสร้างแบบ Overhead Crane มีการเคลื่อนที่เชิงเส้นบนแกนตั้งฉาก มีการทำงานบน 3 แกน คือ แกน  $x$  แกน  $y$  และแกน  $z$  โดยหุ่นยนต์ต้นแบบนี้จะใช้ Stepper Motor จำนวน 4 ชุด ประกอบด้วย ชุดขับเคลื่อนแกน  $x$  จำนวน 1 ชุด ชุดขับเคลื่อนแกน  $y$  จำนวน 2 ชุด และชุดขับเคลื่อนแกน  $z$  จำนวน 1 ชุด ดังภาพที่ 4.7 นอกจากนี้ยังติดตั้ง Limit Switch จำนวน 3 ชุด เพื่อทำหน้าที่เป็น End Stop ของแกน  $x$ ,  $y$  และ  $z$  เพื่อใช้ในการกำหนดจุด Home ในการเริ่มต้นระบบ และเพื่อตัดการทำงานเชิงกล ป้องกันไม่ให้ Stepper Motor ทำงานผิดพลาดจนระบบกลไกชำรุดเสียหาย

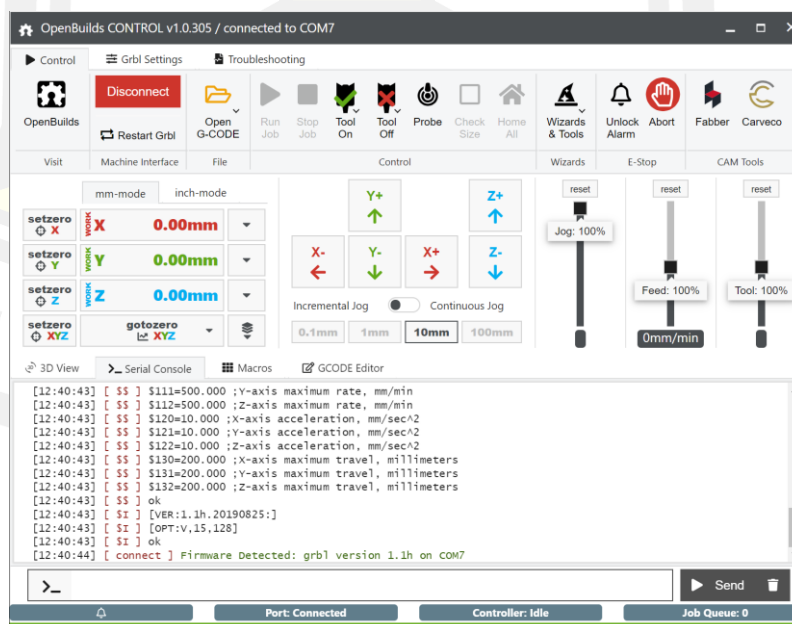


ภาพที่ 4.7 ชุดควบคุมการเคลื่อนที่เชิงเส้นของหุ่นยนต์ต้นแบบ

กำหนดค่า Step/mm (จำนวน Step ของ Stepper Motor ที่ใช้ในการเคลื่อนที่ 1 มิลลิเมตร) ของชุดขับเคลื่อนให้กับระบบ ซึ่งแกน x และแกน y มีการขับเคลื่อนในรูปแบบของ Belt Driven System ดังนั้น Step/mm จึงมีค่าเท่ากับ 80.00 ส่วนทางด้านแกน z นั้น ใช้การขับเคลื่อนแบบ Lead Screw Driven System ค่าของ Step/mm จึงเท่ากับ 400.00



ภาพที่ 4.8 การทดสอบการเคลื่อนที่เชิงเส้นทั้ง 3 แกนของหุ่นยนต์



ภาพที่ 4.9 Open Builds Control ซอฟต์แวร์ที่ใช้ในการ Simulation

ตารางที่ 4.2 ผลการทดสอบการเคลื่อนที่เชิงเส้นทั้ง 3 แกนของหุ่นยนต์

Axis	Forward (mm)									
	1	2	3	4	5	6	7	8	9	10
x-axis	103	103	104	103	103	103	104	104	103	103
y-axis	105	105	105	106	105	105	105	106	107	105
z-axis	110	109	112	112	110	111	111	109	112	111
Axis	Forward (mm)									
	11	12	13	14	15	16	17	18	19	20
x-axis	104	104	102	103	102	103	103	104	103	103
y-axis	105	105	105	104	105	105	105	105	106	106
z-axis	109	111	111	112	110	111	111	111	112	112
Axis	Backward (mm)									
	1	2	3	4	5	6	7	8	9	10
x-axis	104	104	103	104	104	103	103	102	102	103
y-axis	106	106	105	105	105	105	106	105	106	105
z-axis	109	112	111	111	112	111	112	111	112	112
Axis	Backward (mm)									
	11	12	13	14	15	16	17	18	19	20
x-axis	103	103	103	104	103	103	103	104	103	103
y-axis	106	105	105	105	106	106	106	106	105	105
z-axis	110	111	111	111	110	111	110	110	111	111

ตารางที่ 4.3 สรุปผลการทดสอบการเคลื่อนที่เชิงเส้นทั้ง 3 แกนของหุ่นยนต์

Axis	Forward	Backward	Home	Average	Precision
x-axis	Pass	Pass	Pass	103.20 mm	0.968
y-axis	Pass	Pass	Pass	105.35 mm	0.946
z-axis	Pass	Pass	Pass	110.90 mm	0.891

Pass = มีทิศทางการเคลื่อนที่ถูกต้อง    Not Pass = มีทิศทางการเคลื่อนที่ไม่ถูกต้อง

จากตารางที่ 4.2 การทดลองจะใช้ซอฟต์แวร์ Open Builds Control เพื่อทดสอบการเคลื่อนที่ของระบบขับเคลื่อนทั้ง 3 แกน โดยการสั่งให้เคลื่อนที่เป็นระยะทางสั้น ๆ (Jog) ไปในทิศทาง Forward และ Backward โดยใช้ระยะทาง 100 มิลลิเมตร ในการ Jog แต่ละครั้ง และทดสอบการเคลื่อนที่ไปยังตำแหน่ง Home (ตำแหน่งที่  $x = 0 \text{ mm.}$ ,  $y = 0 \text{ mm.}$  และ  $z = 0 \text{ mm.}$ ) ซึ่งผลการทดลองพบว่าระบบขับเคลื่อนของทั้งแกน  $x$ ,  $y$  และ  $z$  สามารถเคลื่อนที่ไปในทิศทางที่ต้องการ การเคลื่อนที่ทั้งสามแกนมีความนิ่งไม่มีอาการสั่นไหว รวมถึงสามารถเคลื่อนที่กลับไปยังจุด Home ได้อย่างถูกต้อง โดยในระหว่างการทดสอบมีข้อสังเกตว่า ความแม่นยำของชุดแกน  $x$  และแกน  $y$  ถึงแม้จะเป็นระบบ Belt Driven System เช่นเดียวกัน แต่ก็ยังมีปัจจัยที่ส่งผลให้เกิดความคลาดเคลื่อนที่แตกต่างได้ เนื่องจากการที่แกน  $x$  มี Stepper Motor ขับเคลื่อนสายพานเพียงตัวเดียว ในขณะที่แกน  $y$  ต้องใช้ Stepper Motor ถึง 2 ตัวในการขับสายพานแยกกัน 2 ด้าน ซึ่งความผิดพลาดจากความตึง-หย่อนของสายพานที่แตกต่างกันของ Stepper Motor ก็มีผลสำคัญที่ทำให้เกิดความคลาดเคลื่อนขึ้นได้ ส่วนทางด้านแกน  $z$  นั้น การที่ใช้ระบบ Lead Screw Driven System ที่มีต้นทุนต่ำกว่าแทน Ball Screw Driven System ซึ่งมีราคาสูงมาก ก็มีข้อจำกัดในประเด็นของการเกิด Backlash หรือระยะฟรีขณะเกิดการหมุนของเพลาเกลียว ที่ส่งผลให้การเคลื่อนที่ของชุดแกน  $z$  ไม่ถูกต้องในระหว่างการเคลื่อนที่ไปและเคลื่อนที่กลับ ส่งผลให้ระยะการเคลื่อนที่จริงไม่ตรงกับค่าที่กำหนด ซึ่งสามารถปรับปรุงข้อผิดพลาดดังกล่าวให้ดีขึ้นได้ด้วยการใช้ Anti-Backlash Nut Block เข้ามาช่วยป้องกันการ Slip ของ Lead Screw ในระหว่างการทำงาน

#### 4.2 ผลการออกแบบและพัฒนาระบบการรับรู้

การวิจัยในส่วนนี้จะเป็นการออกแบบระบบการรับรู้ของหุ่นยนต์โดยใช้การประมวลผลภาพเพื่อตรวจจับกระถางภายในแปลงปลูก โดยใช้การถ่ายภาพแบบ Snap Shot ของแปลงปลูกที่ได้มาจากชุด Raspberry Pi Camera Module ดังภาพที่ 4.10 แทนการนำภาพวิดีโอแบบ Real Time ซึ่งต้องใช้ทรัพยากรของระบบที่สูง รวมถึงมีระยะเวลาการประมวลผลที่ช้าเมื่อต้องทำงานอยู่บนแพลตฟอร์มของ Raspberry Pi มาเป็นการถ่ายและบันทึกภาพเป็นไฟล์ Jpeg เพื่อนำไปผ่านกระบวนการทางการประมวลผลภาพต่อ ใช้หาจุดที่เป็นกระถาง จากนั้นทำการวาด Contour รอบตรงจุดที่เป็นกระถางทุกใบที่ระบบตรวจพบ พร้อมทั้งนับจำนวนของกระถาง แล้วนำค่าได้จากการประมวลผลภาพมาทำการคำนวณหาค่าตำแหน่งบนแกน  $x$  และแกน  $y$  ของจุดที่จะใช้ในการหยอดเมล็ด ซึ่งจะต้องแปลงค่าจากขนาดของจุดพิกัดตำแหน่งในระบบ Pixel ของภาพ ให้เป็นตำแหน่งที่ใช้เป็นจุดอ้างอิงในการหยอดเมล็ดลงจุดกึ่งกลางของกระถางปลูกจริง ซึ่งมีหน่วยเป็นมิลลิเมตร ด้วยเทคนิค Hough Circle Transform ซึ่งเขียนด้วยภาษา Python ร่วมกับ OpenCV

ตารางที่ 4.4 ผลการทดสอบระบบการตรวจจับกระถางของหุ่นยนต์

No.	Detect Pots	Pot 1 (x, y)		Pot 2 (x, y)		Pot 3 (x, y)		Pot 4 (x, y)	
		220	205	420	205	220	410	420	410
1	4	218	209	420	198	204	408	426	410
2	4	229	204	416	206	205	412	420	415
3	4	209	199	413	195	204	407	419	412
4	4	215	203	423	198	204	401	421	408
5	4	223	201	414	201	204	400	425	409
6	4	207	211	424	209	204	405	419	410
7	4	222	198	427	193	204	404	425	412
8	4	229	208	420	199	205	407	418	411
9	4	215	207	415	202	204	401	426	405
10	4	223	208	411	209	204	408	418	403
11	4	208	207	411	202	205	413	416	418
12	4	217	209	420	198	204	401	427	401
13	4	216	211	420	200	206	408	413	400
14	4	225	207	413	198	205	414	419	416
15	4	207	202	413	206	205	411	425	413
16	4	218	201	410	191	205	400	429	403
17	4	227	206	419	204	204	411	431	408
18	4	217	208	415	198	204	402	426	416
19	4	205	211	420	196	205	408	418	407
20	4	228	206	414	207	204	407	417	410
% Error		0.96	0.39	0.74	2.24	7.58	0.89	0.45	0.15
Precision		0.990	0.996	0.993	0.998	0.997	0.991	0.993	0.998

จากตารางที่ 4.4 การทดสอบระบบการตรวจจับกระถาง พบว่าระบบการประมวลผลภาพสามารถตรวจจับกระถางได้ครบทั้ง 4 ใบในการทดลองทุกครั้ง เมื่อนำตำแหน่งอ้างอิงในการทดลองมาเปรียบเทียบกับผลการตรวจจับที่ได้จะพบว่า จุดที่เป็นตำแหน่งของการหยุดเมสส์ดีจะมีความ

คลาดเคลื่อน (Error) ของพิกัด  $x$  และ  $y$  ในแต่ละครั้งที่ทำการทดสอบ เฉลี่ยเท่ากับ 1.68 เพอร์เซ็นต์ ซึ่งเกิดจากการ Calibration และ Sensitivity ของ Hough Circle Transform ซึ่งขึ้นอยู่กับสภาพแสงในสิ่งแวดล้อมของการทดสอบ ที่มีผลทำให้การตรวจจับขอบภาพนั้นเพิ่มหรือลดขนาดลงหรือขยับตำแหน่งไป โดยเห็นได้จาก Contour ในผลการตรวจจับที่มีขนาดเพิ่มหรือลด รวมถึงการขยับเคลื่อนตำแหน่งไป ดังภาพที่ 4.11 ที่มีผลของการตรวจจับคลาดเคลื่อนเล็กน้อย แม้จะเป็นกระถางชุดเดิมที่ถูกตั้งในตำแหน่งและสถานที่เดียวกัน แต่ทำการตรวจจับในช่วงเวลาที่ต่างกันไม่กี่นาที

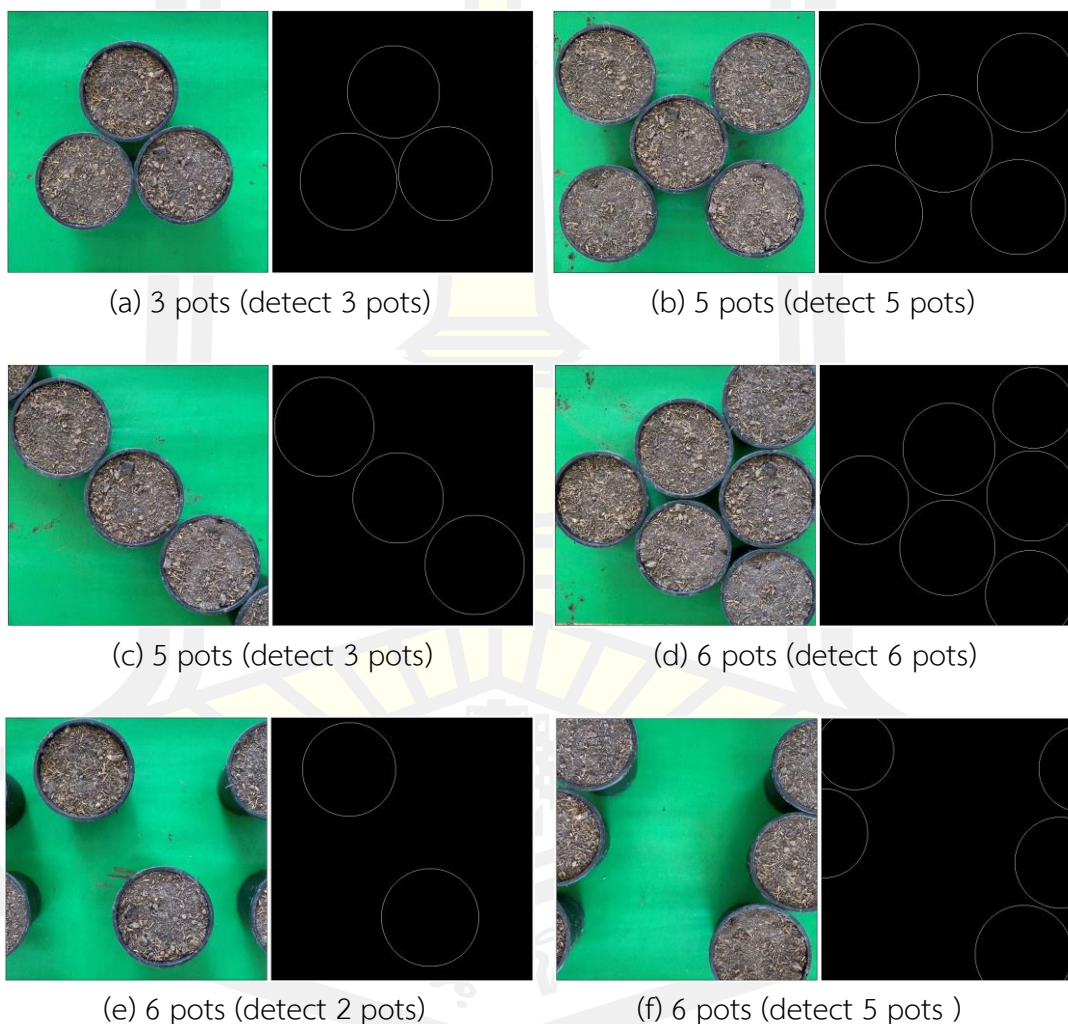


ภาพที่ 4.10 การติดตั้งชุด Sensors ของ Perception Part บนแกน z



ภาพที่ 4.11 ความคลาดเคลื่อนขณะทดสอบการตรวจจับกระถาง

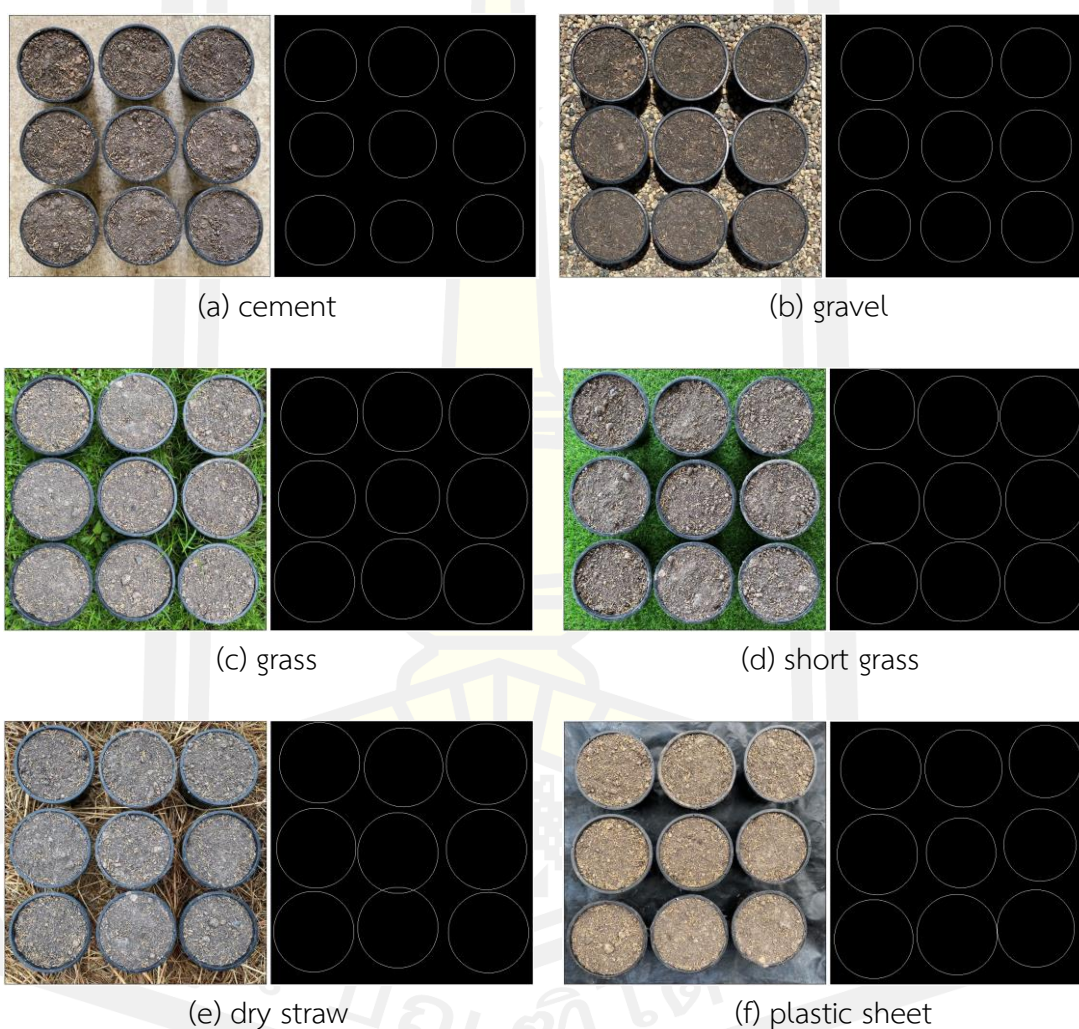
นอกจากนี้เพื่อค้นหาข้อจำกัดในการตรวจจับกระถางของหุ่นยนต์ต้นแบบ จึงได้ทำการทดสอบเพิ่มเติมโดยการจัดวางกระถางในรูปแบบอื่น ๆ เพื่อประเมินประสิทธิภาพและความถูกต้องของการตรวจจับกระถางดังแสดงในภาพที่ 4.12 ซึ่งการทดลองพบว่า ตำแหน่งของการจัดวางกระถางนั้น มีผลต่อการตรวจจับที่ถูกต้องของระบบการรับรู้ หากการวางกระถางนั้นอยู่ในตำแหน่งที่กล้องไม่สามารถจับภาพองค์ประกอบของกระถางได้ครบถ้วน จะส่งผลให้ระบบการรับรู้ตรวจไม่พบการมีอยู่ของกระถางในภาพที่กำลังทำการตรวจสอบ ซึ่งองค์ประกอบของกระถางชิ้นต่ำสุดที่ระบบจะสามารถตรวจสอบได้ จะต้องมีพื้นที่ภาพหน้าตัดของกระถาง  $\geq 50$  เปอร์เซ็นต์ โดยที่มีจุดกึ่งกลางของกระถางปรากฏอยู่ในภาพ และสามารถลากเส้นรัศมี  $R$  จากจุดกึ่งกลางไปยังขอบกระถางในภาพได้ด้วย



ภาพที่ 4.12 ผลการตรวจจับกระถางเมื่อจัดวางในรูปแบบอื่น















ในส่วนของพื้นที่ที่กระถางถูกวางอยู่ ก็ส่งผลกระทบต่อการทำงานของระบบการรับรู้ เช่นเดียวกัน เมื่อพิจารณาการทดสอบในภาพที่ 4.13 จะพบว่าถ้าพื้นที่ที่ใช้ในการวางกระถางนั้น หากมีความต่างของสี หรือ Contrast (การมองเห็นจุดที่อยู่ข้างกันมีความแตกต่างกัน) ของกระถางกับพื้นที่ที่จัดวางมาก จะส่งผลให้การตรวจจับกระถางทำได้อย่างมีประสิทธิภาพ คือ ตรวจจับจำนวนของกระถางได้ถูกต้อง มีขนาดและตำแหน่งของการจัดวางที่คลาดเคลื่อนจากค่าจริงเพียงเล็กน้อย ซึ่งความต่างนี้จะสัมพันธ์กับความเข้มแสงในแต่ละ Pixel นั้นเอง



ภาพที่ 4.13 ผลการตรวจจับกระถางเมื่อจัดวางบนพื้นผิวที่แตกต่าง

นอกจากชุดของ Raspberry Pi Camera Module ที่ถูกใช้ในการตรวจจับกระถางเพื่อหาตำแหน่งในการหยอดเมล็ดแล้ว ในการวิจัยนี้ยังได้ออกแบบระบบการรับรู้ของหุ่นยนต์โดยใช้ Ultrasonic sensor ดังภาพที่ 4.15 เพื่อวัดระยะห่างจากกระถางถึง Seed Injector และชุด MCU-Cam Module ดังภาพที่ 4.16 เพื่อช่วยในการตรวจจับ Barcode ในสภาพแวดล้อมการทำงาน เพื่อนำมาใช้ในประมวลผลและตัดสินใจในการควบคุมการเคลื่อนที่ของหุ่นยนต์

ตารางที่ 4.5 ผลทดสอบการทำงานของหุ่นยนต์ด้วยสัญลักษณ์ Barcode

Symbol	Task	Experimental Results									
		1	2	3	4	5	6	7	8	9	10
	Start + Forward	P	P	P	P	P	P	P	P	P	P
	Stop + Forward	P	P	P	P	P	P	P	P	P	P
	Stop + Right	P	P	P	P	P	P	P	P	P	P
	Stop + Backward	P	P	P	P	P	P	P	P	P	P
	Stop + Backward	P	P	P	P	P	P	P	P	P	P
	Stop + End	P	P	P	P	P	P	P	P	P	P
Symbol	Task	Experimental Results									
		11	12	13	14	15	16	17	18	19	20
	Start + Forward	P	P	P	P	P	P	P	P	P	P
	Stop + Forward	P	P	P	P	P	P	P	P	P	P
	Stop + Right	P	P	P	P	P	P	P	P	P	P
	Stop + Backward	P	P	P	P	P	P	P	P	P	P
	Stop + Backward	P	P	P	P	P	P	P	P	P	P
	Stop + End	P	P	P	P	P	P	P	P	P	P

P = Pass (ทำงานได้ถูกต้อง)    NP = Not Pass (ทำงานไม่ถูกต้อง)

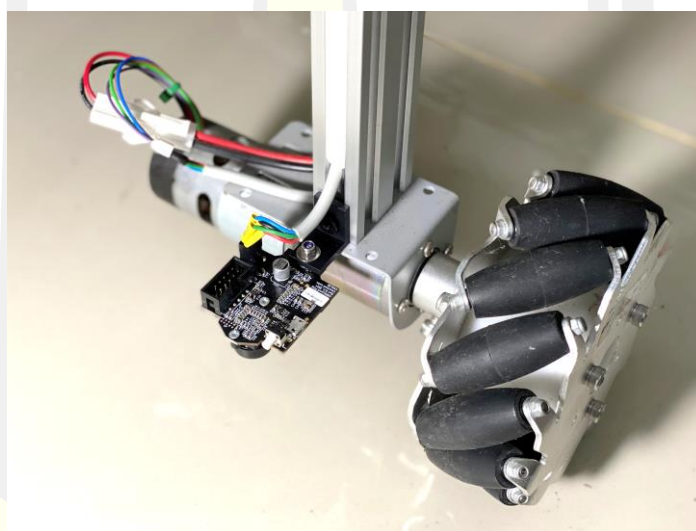
จากผลทดสอบในตารางที่ 4.5 การทำงานของหุ่นยนต์ด้วยสัญลักษณ์ Barcode โดยการใช้ MCU-Cam Module เพื่อตรวจจับ Tag ที่ถูกติดไว้บนพื้นนั้น พบว่ามีการทำงานที่ถูกต้อง หุ่นยนต์สามารถตรวจจับ Tag ของ Barcode และประมวลผลเพื่อปฏิบัติตามชุดคำสั่งในการทำงานได้อย่างราบรื่น แต่อย่างไรก็ตาม ในการปรับแต่งอุปกรณ์ระหว่างการทดลองพบว่า ระดับความสูงของการติดตั้งชุด MCU-Cam Module นั้น ส่งผลกระทบต่อประสิทธิภาพในการตรวจจับ กล่าวคือ หากติดตั้งไว้ในระดับที่สูงขึ้นจะมีระยะของการตรวจจับได้กว้างและรวดเร็วกว่าการติดตั้งในระยะที่ต่ำกว่านั่นเอง ซึ่งหากติดตั้งไว้ในระดับที่ต่ำเกินไป อาจส่งผลให้การตรวจจับ Tag มีความผิดพลาดได้ เพราะระยะในการมองเห็นของ MCU-Cam Module จะแคบ ทำให้การตรวจจับนั้นไม่ครอบคลุมตัว Tag ทั้งหมด จะตรวจพบ Tag แคบบางส่วนเท่านั้น การประมวลผลจึงเกิดความผิดพลาด เพราะข้อมูลที่ได้รับมานั้น เมื่อนำมาเปรียบเทียบจะไม่พบว่าอยู่ในเงื่อนไขการทำงานใดของหุ่นยนต์ ซึ่งในทางตรงกันข้าม หากติดตั้งไว้ในระดับที่สูงเกินไป ชุด MCU-Cam Module ก็จะสามารถตรวจพบ Tag ได้อย่างรวดเร็ว เพราะมีมุมมองของการตรวจจับที่กว้างมาก จึงส่งผลให้การประมวลผลและการควบคุมการทำงานของหุ่นยนต์ผิดพลาดตามไปด้วย เพราะระบบควบคุมประมวลผลแล้วคิดว่า หุ่นยนต์เคลื่อนที่มาถึงจุดที่กำหนดแล้ว แต่ในความเป็นจริงหุ่นยนต์ยังเคลื่อนที่ไปไม่ถึงจุดอ้างอิงที่แท้จริง แต่ชุด MCU-Cam Module กลับตรวจจับ Tag ได้ก่อน



ภาพที่ 4.14 การทดสอบการทำงานของหุ่นยนต์ด้วยสัญลักษณ์ Barcode



ภาพที่ 4.15 การติดตั้ง Ultrasonic distance sensor เพื่อใช้หาระยะของความสูง

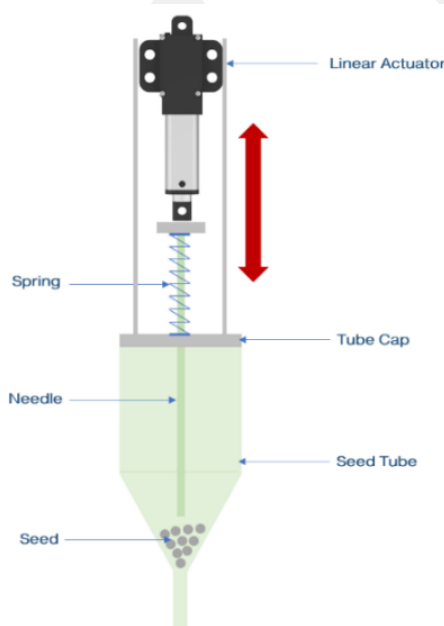


ภาพที่ 4.16 การติดตั้ง MCU-Cam เพื่อใช้ในการตัดสินใจเคลื่อนที่ของหุ่นยนต์

#### 4.3 ผลการออกแบบและพัฒนาระบบการกระทำ

การออกแบบหัวปลูก หรือ Seed Injector ของหุ่นยนต์ซึ่งใช้ในการหยอดเมล็ดพืชลงบนกระถาง มีแนวคิดของการออกแบบดังภาพที่ 4.17 โดยหัวปลูกเป็นแบบหัวหยอด ซึ่งจะใช้วิธีเก็บเมล็ดพันธุ์ไว้ในหลอดเก็บเมล็ดที่หัวปลูก เมื่อหัวปลูกเคลื่อนที่มาถึงจุดหยอดเมล็ด ชุดก้านของ Linear Actuator จะเคลื่อนไปข้างหน้าเพื่อดันให้ก้านของหัวหยอด ดันให้เมล็ดที่ถูกเก็บอยู่ข้างในหยอดตกลง

ในกระถาง และเมื่อการหยอดเมล็ดเสร็จสิ้น ชุด Linear Actuator ก็จะเคลื่อนที่ถอยหลังกลับไปยังตำแหน่งเดิม พร้อมกับชุดสปริงที่อยู่บริเวณตอนท้ายของก้านของหัวหยอดก็จะดันให้ก้านหัวหยอดกลับสู่ตำแหน่งเดิมเช่นเดียวกัน เพื่อไม่ให้เมล็ดที่อยู่ภายในหลอดไหลออกมาได้ ซึ่งวิธีการในลักษณะนี้จะส่งผลให้ลดการสูญเสียเวลาในการปลูกเนื่องจากไม่ต้องเคลื่อนที่กลับไปกลับมาอย่างจุดที่ถาดเก็บเมล็ดอยู่



ภาพที่ 4.17 การออกแบบชุดหัวปลูกแบบ Seed Injector

การทำงานของ Seed Injector ในภาพที่ 4.18 มี 4 ขั้นตอน คือ

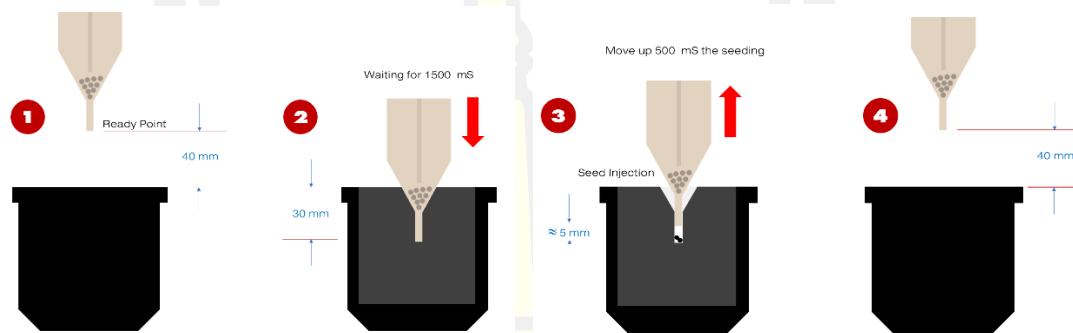
1. Seed Injector เคลื่อนที่ไปยังตำแหน่งพิกัด  $x$  และ  $y$  ซึ่งเป็นจุดกึ่งกลางของกระถาง จากนั้นแกน  $z$  จะเคลื่อนที่ลงไปไปยังตำแหน่งเตรียมหยอดเมล็ด (Ready Point) ซึ่งมีความสูงเหนือระดับปากกระถางปลูก 40 มิลลิเมตร โดยอ้างอิงตำแหน่งจาก Ultrasonic distance sensor

2. Seed Injector บนแกนแนวแกน  $z$  เคลื่อนที่ลงไปเป็นระยะทาง 65 มิลลิเมตร ซึ่งเป็นระยะของ Ready Point + ปลายของ Seed Injector จากนั้นทำการหน่วง Delay Time ไว้เป็นระยะเวลา 1,500 มิลลิวินาที (1,500 mS) เพื่อให้ดินรอบ ๆ ปลายของ Seed Injector มีการคงตัวชั่วคราว

3. ปลายของ Seed Injector ถูกยกขึ้นในแนวแกน  $z$  จากกระถาง ในขณะเดียวกันระบบจะทำการหน่วงเวลาไว้ 500 mS โดยที่ปลายของ Seed Injector จะยกตัวขึ้นอยู่เหนือดินเล็กน้อย แต่ยังไม่พ้นปากรูที่ตกลงไปในกระถาง จากนั้นระบบจะสั่งงานให้ Linear Actuator ทำงาน

โดยเคลื่อนตัวก้านโลหะไปทางด้าน Forward เป็นระยะ 10 มิลลิเมตร ดันแป้นของก้านหยอดเมล็ดลง เพื่อปล่อยเมล็ดพืชลงในรูที่ถูกสร้างขึ้น จากนั้น Linear Actuator ก็จะเคลื่อนตัวกลับไปทางด้าน Backward ทันที เพื่อปล่อยให้สปริงของก้านหยอดเมล็ดดันก้านหยอดเมล็ดกลับสู่ตำแหน่งเดิม ป้องกันไม่ให้เมล็ดไหลออกมา

4. Seed Injector เคลื่อนที่กลับขึ้นไปรอยระดับความสูงของตำแหน่งเตรียมหยอดเมล็ดอีกครั้ง รอคำสั่งให้เคลื่อนที่ไปพิกัด  $x$  และ  $y$  เพื่อหยอดเมล็ดในกระถางใบต่อไป



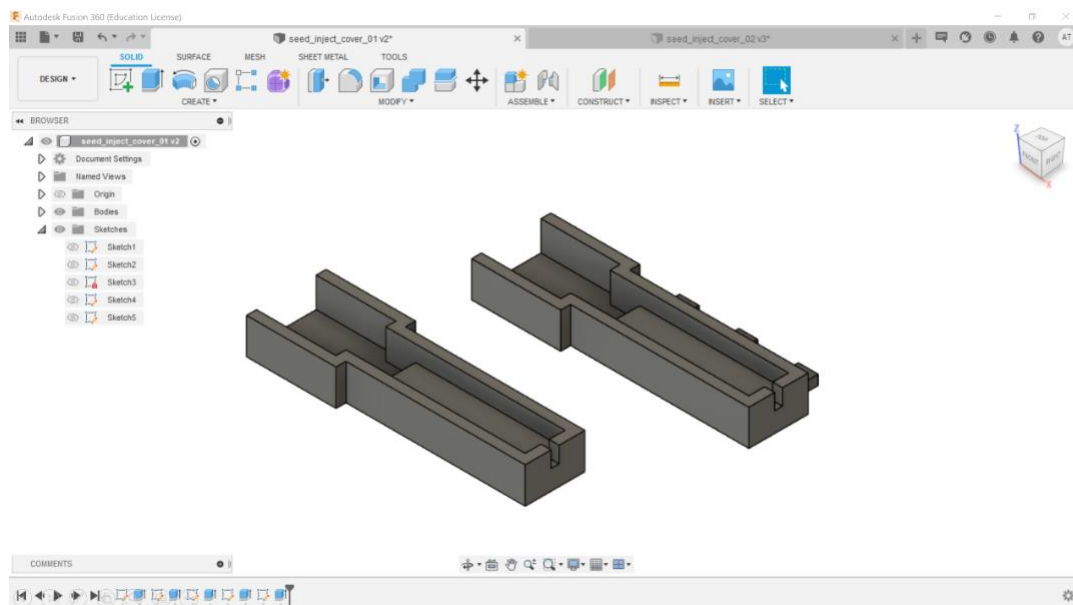
ภาพที่ 4.18 ขั้นตอนการทำงานของ Seed Injector



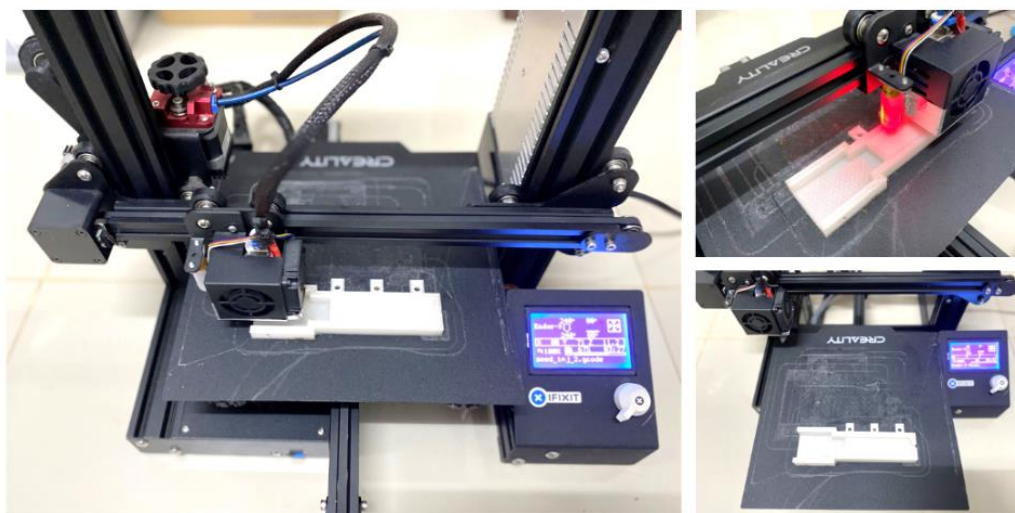
(a) ชิ้นส่วนหัวปลุก ครอบ และก้านปลุก

(b) ก้านปลุก

ภาพที่ 4.19 โมลด์พลาสติกของ Seed Injector



ภาพที่ 4.20 การออกแบบเคสของ Seed Injector ด้วยซอฟต์แวร์ Autodesk Fusion 360



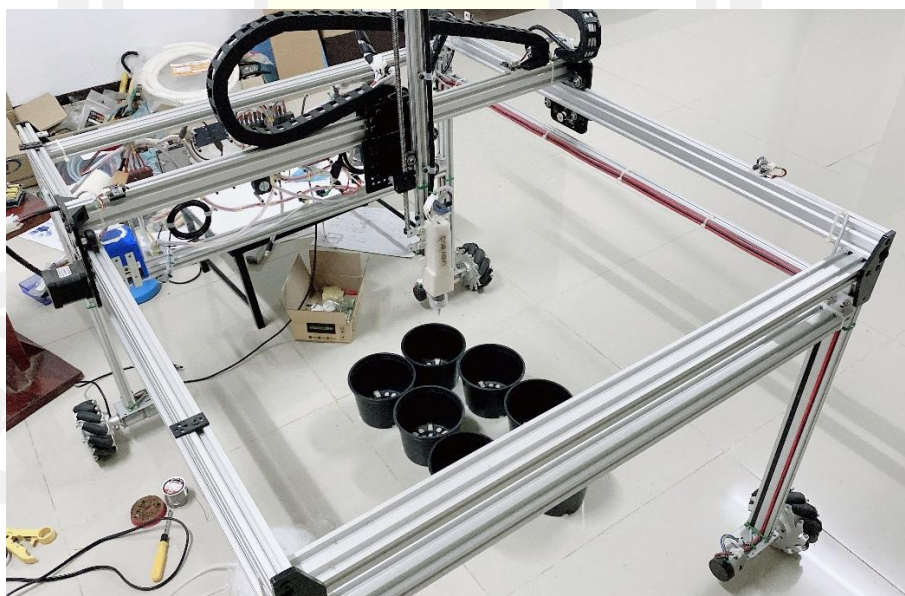
ภาพที่ 4.21 เคสของ Seed Injector ที่พิมพ์ด้วยเครื่อง 3D Printer

ในการพัฒนา Seed Injector นั้นจะแบ่งออกเป็น 2 ส่วน ประกอบด้วย ครอบกหยอดเมล็ด ดังภาพที่ 4.19 ผลิตโดยการขึ้นรูปโพลด์พลาสติก มีชิ้นส่วนสำคัญ 3 ชิ้น คือ หัวปลุก ครอบก และก้านปลุก ซึ่งจะใช้สปริงสวมไว้ที่ท้ายก้านปลุก เพื่อทำหน้าที่ดันก้านปลุกให้กลับสู่ตำแหน่งเดิมเมื่อไม่มีแรงกดที่ปลายด้านท้าย และส่วนที่เป็นเคสของ Seed Injector เพื่อใช้ห่อหุ้มครอบกหยอดเมล็ด กับ Linear Actuator เข้าด้วยกัน โดยการขึ้นรูปเคสด้วย 3D Printer ดังภาพที่ 4.21 และใช้วัสดุเส้น

พลาสติกประเภท ABS Filament ในการพิมพ์ชิ้นงาน เนื่องจากมีความทนทานต่อแสงแดดและความร้อนในสภาพแวดล้อมได้ดีกว่าวัสดุ PLA Filament ที่ใช้งานทั่วไปในการพิมพ์ โดยภาพที่ 4.22 และภาพที่ 4.23 จะแสดงถึงการประกอบและติดตั้งอุปกรณ์ในชุด Seed Injector ของหุ่นยนต์ต้นแบบ



ภาพที่ 4.22 ต้นแบบของ Seed Injector



ภาพที่ 4.23 การติดตั้งชุด Seed Injector บนแกน z

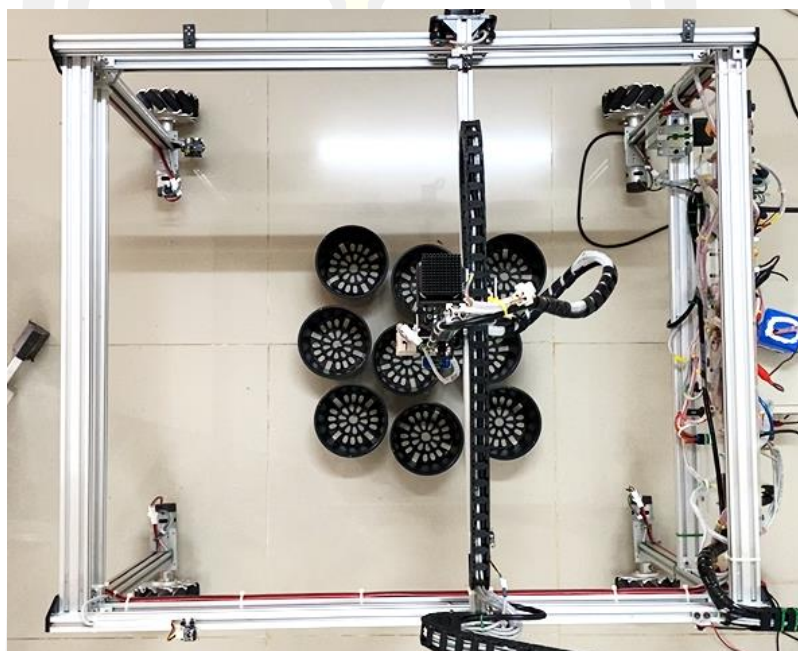


ตารางที่ 4.6 ผลการทดสอบระบบควบคุมการหยอดเมล็ด

No.	Number of Seeds								
	Pot 1	Pot 2	Pot 3	Pot 4	Pot 5	Pot 6	Pot 7	Pot 8	Pot 9
1	3	2	1	2	3	2	2	3	2
2	3	2	2	1	2	2	1	2	3
3	2	3	2	2	2	1	2	4	2
4	2	2	4	2	2	2	2	3	2
5	2	2	3	2	2	2	3	2	2
6	1	2	3	2	2	3	4	2	2
7	2	2	3	2	1	2	2	2	4
8	2	2	2	4	2	2	2	2	2
9	3	1	2	3	2	2	2	1	2
10	2	2	3	3	4	2	2	2	2
11	2	3	2	4	3	2	2	2	2
12	4	2	2	1	2	3	2	4	1
13	2	2	3	2	2	2	2	2	2
14	2	2	2	2	1	2	2	1	2
15	2	2	2	2	2	4	2	2	2
16	1	1	2	2	2	2	2	2	1
17	2	2	1	2	4	2	2	2	2
18	2	2	3	3	1	4	2	2	3
19	2	3	3	2	2	2	3	2	3
20	3	4	2	2	2	2	2	1	3
Average	2.2	2.15	2.35	2.25	2.15	2.25	2.15	2.15	2.2

จากผลการทดสอบระบบควบคุมการหยอดเมล็ดในตารางที่ 4.6 พบว่า ในการทดลองนั้น หุ่นยนต์มีการหยอดเมล็ดลงในกระถางทุกใบ ซึ่งปริมาณของการหยอดเมล็ดในแต่ละครั้งไม่คงที่ โดยปริมาณการหยอด คือ 1 - 4 เมล็ดต่อกระถาง โดยมีค่าเฉลี่ยอยู่ที่ 2.20 หรือประมาณ 2 เมล็ดต่อกระถาง ซึ่งเป็นอัตราของการหยอดเมล็ดพันธุ์ที่เหมาะสม [80] หากอัตราการงอกของเมล็ดอยู่ในระดับ 90 เปอร์เซ็นต์ขึ้นไป สำหรับความไม่สม่ำเสมอของปริมาณเมล็ดพันธุ์ในการหยอดนั้น เป็นผล

มาจากก้านปลูกในภาพที่ 4.19 (b) ที่ใช้ในการลำเลียงเมล็ด เพื่อดันออกไปทางส่วนปลายของ Seed Injector นั้น ถูกออกแบบมาให้ใช้กับเมล็ดขนาดเส้นผ่านศูนย์กลาง 2 มิลลิเมตร ซึ่งเป็นค่าเฉลี่ยขนาดของเมล็ดทั่วไป แต่เมื่อนำเมล็ดพืชที่ถูกจำหน่ายในท้องตลาดมาใช้ในการทดลอง จะพบว่ารูปร่างตลอดจนขนาดไม่เท่ากัน ดังภาพที่ 4.25 ส่งผลให้ปริมาณของเมล็ดที่ถูกหยอดลงบนกระถางไม่คงที่ ซึ่งเป็นข้อจำกัดที่เกิดจากปัจจัยทางธรรมชาติ ไม่สามารถควบคุมได้



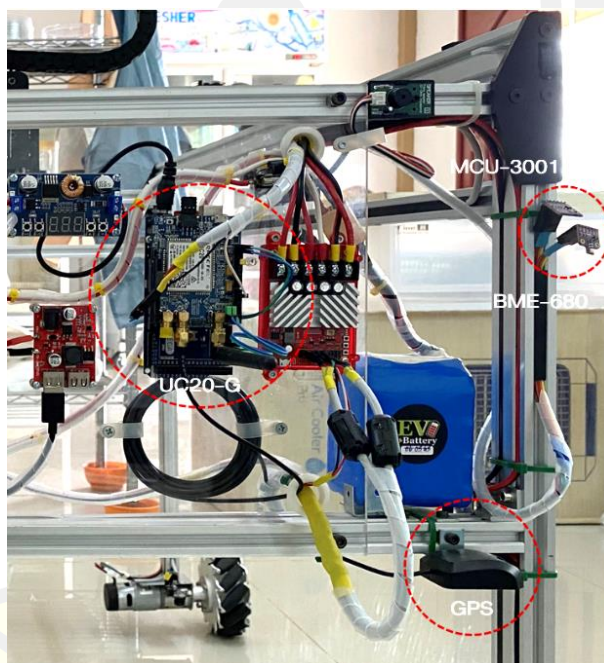
ภาพที่ 4.24 การทดสอบระบบควบคุมการหยอดเมล็ด



ภาพที่ 4.25 ความแตกต่างของขนาดของเมล็ดพันธุ์ที่ถูกนำมาทดสอบ

#### 4.4 ผลการออกแบบและพัฒนาระบบอินเทอร์เน็ตของสรรพสิ่ง

ในส่วนของการออกแบบและระบบอินเทอร์เน็ตของสรรพสิ่งนั้น จะเป็นการทำงานที่เกี่ยวข้องกับระบบตรวจวัดค่าพารามิเตอร์ของสภาพแวดล้อมของโรงงานในขณะที่หุ่นยนต์ปลุกพืชอัตโนมัติกำลังทำงานอยู่ โดยใช้ชุดของ Sensor Units จำนวน 3 Modules ดังภาพที่ 4.26 ซึ่งประกอบด้วย MCU-3001 (Ambient Light Sensor) ใช้สำหรับวัดความเข้มแสง BME-680 (Low Power Gas, Pressure, Temperature and Humidity Sensor) ใช้สำหรับตรวจวัดค่าจากสภาพอากาศ 4 ประเภท คือ อุณหภูมิ ความชื้น ความดันอากาศ และคุณภาพของอากาศ พร้อมทั้งมีชุด Quectel UC20-G ซึ่งเป็นอุปกรณ์ที่ใช้ HSPA+/WCDMA/GSM/GPRS/EDGE ในการสื่อสารและมี GPS แบบ GNSS Receiver Moduleในตัว เพื่อใช้ระบุตำแหน่งจากระบบดาวเทียม GPS



ภาพที่ 4.26 ชุด Sensor Units ของระบบอินเทอร์เน็ตของสรรพสิ่ง

ขั้นตอนของการทำงานจะเริ่มจากข้อมูลของสภาพแวดล้อมที่ได้รับจากทั้ง 3 Module นี้ ซึ่งถูกกำหนดอัตรา Refresh Rate ทุก 5 วินาที จะถูกนำมาสื่อสารในแบบ Real Time ผ่าน MQTT Protocol ส่งไปในเครือข่ายอินเทอร์เน็ตโดยใช้ HSPA+ บนเครือข่ายโทรศัพท์เคลื่อนที่ GSM 3G เพื่อทำการ Publish และ Subscribe ข้อมูลของ Topics ที่ MQTT Broker ดังภาพที่ 4.27 ซึ่งถูกติดตั้งไว้บนระบบ Cloud Computing ที่ <https://www.cloudmqtt.com> โดยทำการ Publish ข้อมูลสถานะใน Topic ชื่อ “status” จากคอมพิวเตอร์และสมาร์ตโฟน ดังภาพที่ 4.29 ไปยังเครื่อง MQTT Broker

Server ชื่อ driver.cloudmqtt.com บน Amazon Web Services และอีกฝั่งนั้น ด้านของหุ่นยนต์ เพาะปลูกจะถูกกำหนดให้ทำการ Subscribe ใน Topic ชื่อ “status” เพื่อคอยรับ Payload จาก MQTT Broker มาประมวลผลและนำไปควบคุม Actuators ต่อ ซึ่งจะใช้เป็นรูปแบบในการควบคุมการทำงานของหุ่นยนต์เพาะปลูก และนำไปใช้แสดงผลข้อมูลภาพรวมใน Dashboard ดังภาพที่ 4.30

ตารางที่ 4.7 ผลการทดสอบ Publish และ Subscribe

No.	Publish Topics (Robot)								Subscribe (MQTT Client)							
	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
1	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
2	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
3	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
4	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
5	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
6	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
7	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
8	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
9	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
10	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
11	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
12	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
13	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
14	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
15	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
16	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
17	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
18	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
19	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
20	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P

P = Pass (ทำงานได้ถูกต้อง)    NP = Not Pass (ทำงานไม่ถูกต้อง)

จากผลการทดสอบในตารางที่ 4.7 การ Publish และ Subscribe ข้อมูลของ Topic แบบ Real Time ด้วย MQTT Protocol บนระบบ WebSocket UI ของ CloudMQTT ดังแสดงในภาพที่ 4.27 โดยทดสอบการทำงานบน Web Browser และ MQTT Client ดังภาพที่ 4.29 ซึ่งเป็น Mobile Application บนระบบปฏิบัติการ IOS ของเครื่อง Apple iPhone 11 Pro Max พบว่า มีจำนวนครั้งของการ Publish และ Subscribe หัวข้อ Topic ที่ถูกต้อง นอกจากนี้ ข้อมูลที่อ่านได้จากสภาพแวดล้อมโดยชุดเซ็นเซอร์ของหุ่นยนต์ก็มีความถูกต้องตรงกันกับข้อมูลที่ถูก Publish และ Subscribe ถึงแม้หุ่นยนต์จะทำงานอยู่บนเครือข่าย GSM 3G ในพื้นที่ซึ่งตั้งอยู่ห่างไกลจากสถานีทวนสัญญาณ (Cell Site) ประมาณ 3 กิโลเมตรทางอากาศ โดยการรับ-ส่งข้อมูลในระหว่างการทดสอบ แม้จะมีความเร็วที่ต่ำเพียง 384 Kbps – 3 Mbps ก็ไม่พบความผิดพลาดของการทำงาน เนื่องจาก MQTT Protocol มีความรวดเร็วในการรับ-ส่งข้อมูล เพราะ Header มีขนาดเล็ก ใช้ทรัพยากรและพลังงานในการทำงานที่ต่ำ รวมถึงการที่เป็นระบบแบบ Event-Driven จึงสามารถรับข้อมูลได้ตลอดเวลาที่ Topic เหล่านั้นถูก Publish

The screenshot shows the CloudMQTT console interface for a Websocket. The left sidebar contains navigation options: DETAILS, SETTINGS, CERTIFICATES, USERS & ACL, BRIDGES, AMAZON KINESIS STREAM, WEBSOCKET UI (selected), STATISTICS, CONNECTIONS, LOG, and MAINTENANCE. The main content area is titled 'Websocket' and includes a 'Send message' section with a 'Topic' dropdown menu, a 'Message' input field, and a 'Send' button. Below this is a 'Clear session' section with a 'Client ID' input field. On the right, the 'Received messages' section displays a table of incoming data:

Topic	Message
temp	29.14
humid	67.543
pressure	997.76
altimeter	129.77
voc	377.29
a_light	38.58
lat	15.96518
lat	103.84956
temp	29.14
humid	67.591

ภาพที่ 4.27 การแสดงผลข้อมูลที่ได้รับจากหุ่นยนต์ผ่าน WebSocket UI

หลังจากการทดสอบ Subscribe และ Publish ของ Topic แล้ว จะเป็นขั้นตอนการทดสอบการควบคุมหุ่นยนต์ผ่านระบบอินเทอร์เน็ตของสรรพสิ่ง โดยการควบคุมผ่าน MQTT Client ดังภาพที่ 4.29 ซึ่งสร้างปุ่มกด เพื่อใช้ควบคุมการทำงาน จำนวน 3 ฟังก์ชัน คือ

1. ปุ่มเปิดเสียง Beep ใช้เพื่อสั่งงานให้หุ่นยนต์ส่งสัญญาณ Alarm แฉ่งเตือน
2. ปุ่มเปิดไฟ LED ใช้เพื่อสั่งงานให้หุ่นยนต์เปิดไฟส่องสว่าง LED ที่ถูกติดตั้งไว้
3. ปุ่มปิดไฟ LED ใช้เพื่อสั่งงานให้หุ่นยนต์ปิดไฟส่องสว่าง LED ที่ถูกติดตั้งไว้

ในการทดสอบนั้นจะทำการ Monitor ผลของการควบคุมผ่านทาง Serial Console ด้วย Putty ดังภาพที่ 4.28 และการสังเกตการทำงานของหุ่นยนต์หลังจากที่ทำการกดปุ่มใน MQTT Client

```

COM11 - PuTTY
+CPIN: READY

+QUSIM: 1

+QIND: SMS DONE

+QIND: PB DONE

MC20 Ready...

Close Echo
OK
SetOperator --> "AIS",4
SignalQuality --> 10
OK
GPS Start
- Initializing OPT3001 Ambient Light Sensor
- OPT3001 Manufacturer ID21577
- OPT3001 Device ID12289
OPT3001 Current Config:
-----
Conversion ready (R):0
Conversion time (R/W):0
Fault count field (R/W):0
Flag high field (R-only):0
Flag low field (R-only):0
Latch field (R/W):1
Mask exponent field (R/W):0
Mode of conversion operation (R/W):3
Polarity field (R/W):0
Overflow flag (R-only):0
Range number (R/W):C
-----
High-Limit83865.54Low-Limit0.00-----
- Initializing BME680 sensor
- Setting 16x oversampling for all sensors
- Setting IIR filter to a value of 4 samples
- Setting gas measurement to 320°C for 150ms

Loop Temp°C Humid% Press hPa Alt m Air mΩ lux Lat Long
=====
1 28.87 62.272 997.43 132.99 302.21 48.04
2 29.17 62.292 997.43 132.99 370.95 48.04 15.96530 103.84947
3 29.05 61.924 997.43 132.99 310.27 48.04 15.96531 103.84947
4 28.94 62.031 997.43 132.99 302.21 48.04 15.96531 103.84947
5 28.87 62.020 997.43 132.99 298.34 47.74 15.96531 103.84947

Loop Temp°C Humid% Press hPa Alt m Air mΩ lux Lat Long
=====
6 28.81 61.996 997.45 132.82 297.07 47.74 15.96531 103.84947

```

ภาพที่ 4.28 การตรวจสอบการทำงานของระบบ IoT ผ่าน Serial Console

ตารางที่ 4.8 ผลการทดสอบการควบคุมหุ่นยนต์ผ่านระบบอินเทอร์เน็ตของสรรพสิ่ง

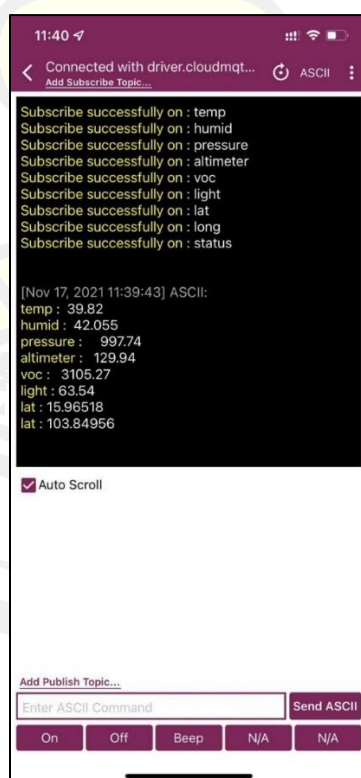
No.	Buzzer	LED	
		“On”	“Off”
1	Pass	Pass	Pass
2	Pass	Pass	Pass
3	Pass	Pass	Pass
4	Pass	Pass	Pass
5	Pass	Pass	Pass
6	Pass	Pass	Pass
7	Not Pass	Pass *	Pass
8	Pass	Pass	Pass
9	Pass	Pass	Pass
10	Pass	Pass	Pass
11	Pass	Pass	Pass
12	Pass	Pass	Pass
13	Not Pass	Pass *	Pass *
14	Pass	Pass	Pass
15	Pass	Pass	Pass
16	Pass	Pass	Pass
17	Pass	Pass	Pass
18	Pass	Pass	Pass
19	Pass	Pass	Pass
20	Pass	Pass	Pass

Pass = ทำงานได้ถูกต้อง    Not Pass = ทำงานไม่ถูกต้อง    \* = เกิด Delay Time

จากผลการทดสอบการควบคุมหุ่นยนต์ผ่านระบบอินเทอร์เน็ตของสรรพสิ่งในตารางที่ 4.8 พบว่า ในการทดลองนั้นสามารถสั่งงานให้หุ่นยนต์ทำงานได้ โดยมีค่า Accuracy เท่ากับ 0.96 โดยมีความล้มเหลวในการควบคุมเกิดขึ้น 2 ครั้ง และส่งผลให้การกดปุ่มเพื่อควบคุมหุ่นยนต์ครั้งถัดไปหลังจากการควบคุมล้มเหลวนั้น เกิด Delay Time ขึ้นชั่วคราว แต่ก็สามารถทำงานได้ จากการสังเกตการทำงานจากระบบด้วย Serial Console ทำให้ทราบว่า สาเหตุของการควบคุมล้มเหลวดังกล่าว เกิดจากการที่ Quectel UC20-G มีสถานะ Connection Loss หลุดจากการเชื่อมต่อเครือข่าย GSM 3G และพยายามทำ

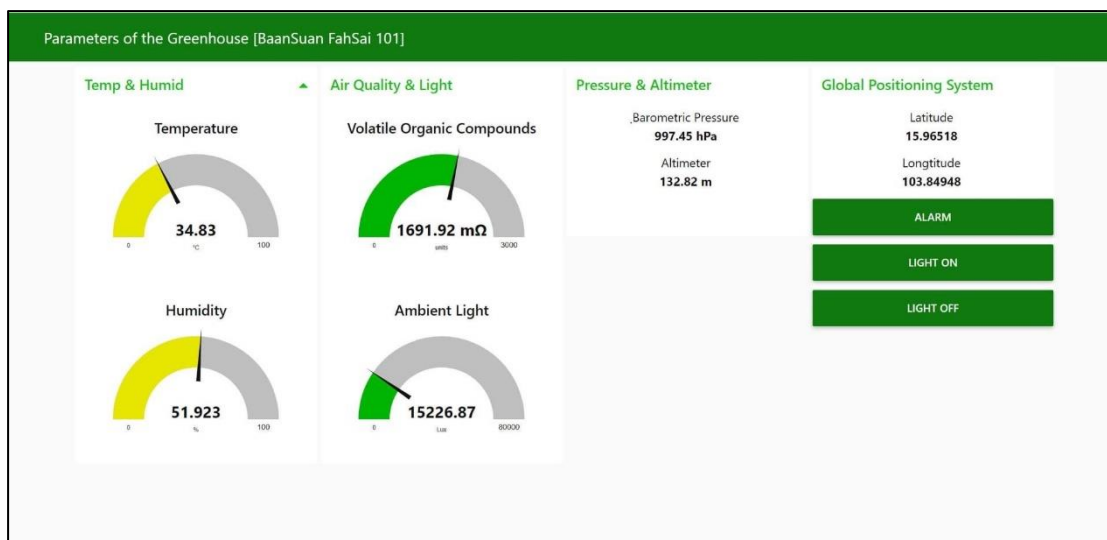
การ Reconnect ใหม่อีกครั้ง ซึ่งในระหว่างช่วงเวลาของการเชื่อมต่อเครือข่ายใหม่ดังกล่าวนี้ จะส่งผลให้หุ่นยนต์ไม่สามารถรับข้อมูลของ Topic ที่ Publish มาจากการกดปุ่มควบคุมบน MQTT Client ได้ และเมื่อเชื่อมต่อได้สำเร็จ ในการส่งงานครั้งต่อไปก็เกิด Delay Time ขึ้นจากผลของการพยายามเชื่อมต่อเครือข่าย

เมื่อพิจารณาผลการทดสอบ Publish และ Subscribe ของ Topic ในตารางที่ 4.7 จะพบว่ามีการทำงานที่ถูกต้องแม่นยำ และมีประสิทธิภาพสูงกว่า หากเปรียบเทียบกับผลการทดสอบการควบคุมหุ่นยนต์ผ่านระบบอินเทอร์เน็ตของสรรพสิ่ง ในตารางที่ 4.8 เนื่องจากการทดสอบ Publish และ Subscribe ของ Topic นั้น ทางด้าน Hardware ของหุ่นยนต์จะถูกกำหนดให้ทำการตรวจสอบการเชื่อมต่อกับเครือข่าย GSM 3G ในทุกครั้งก่อนทำการทำงาน หากเกิด Connection Loss ขึ้น Hardware ก็จะทำให้การ Reconnect ใหม่ให้สำเร็จก่อน จึงจะทำงานชุดคำสั่งต่อไป ในทางตรงกันข้ามนั้น การทดสอบการควบคุมหุ่นยนต์ผ่านระบบอินเทอร์เน็ตของสรรพสิ่ง ซึ่งเป็นการสั่งงานจาก MQTT Client บน Smartphone ไม่ใช่การสั่งการของ Hardware บนตัวหุ่นยนต์ และเนื่องจากการสั่งการนี้อยู่ในพื้นที่ใดก็ได้ที่มีการเชื่อมต่อกับเครือข่ายอินเทอร์เน็ตอยู่ จึงส่งผลให้ MQTT Client ไม่ทราบถึงสถานะการเชื่อมต่อของ Quectel UC20-G บนตัวหุ่นยนต์ในขณะนั้น ว่ามีการเชื่อมต่อปกติ หรือการเชื่อมต่อที่ล้มเหลว ด้วยเหตุนี้ MQTT Client จึงทำหน้าที่เป็นเพียงตัวกลางในการส่งคำสั่งจากผู้ใช้งานไปควบคุมหุ่นยนต์เท่านั้น



ภาพที่ 4.29 การทดสอบ Subscribe และ Publish ข้อมูลผ่าน Mobile Application





ภาพที่ 4.30 Dashboard แสดงผลข้อมูลที่ได้รับจากหุ่นยนต์

ระบบ Dashboard ที่ถูกพัฒนาขึ้น ดังภาพที่ 4.30 จะใช้สำหรับแสดงผลข้อมูลของสิ่งแวดล้อมที่ได้รับจากหุ่นยนต์ ซึ่งประกอบด้วยมาตรวัดหรือเกจ (Gauge) จำนวน 4 ชุด ที่ใช้แสดงค่าข้อมูลของอุณหภูมิ มีหน่วยเป็นองศาเซลเซียส ค่าข้อมูลความชื้นสัมพัทธ์ แสดงผลเป็นหน่วยเปอร์เซ็นต์ ค่าคุณภาพของอากาศ มีหน่วยเป็นเมกกะโอห์ม (mΩ) และค่าข้อมูลความเข้มของแสง มีหน่วยเป็น Lux พร้อมทั้งข้อมูลตัวเลข จำนวน 3 ชุด ที่แสดงค่าของความกดอากาศ มีหน่วยเป็น hPa ค่าข้อมูลความสูงจากระดับน้ำทะเล ที่มีหน่วยเป็นเมตร และค่าข้อมูลพิกัดตำแหน่งที่ระบุด้วยระบบดาวเทียม GPS ซึ่งแสดงค่าในพิกัดของ Latitude และ Longitude โดยค่าทั้งหมดที่แสดงบน Dashboard จะถูก Refresh อัตโนมัติทันที เมื่อหุ่นยนต์มีการ Publish ค่าของ Topics ไปยัง MQTT Broker นอกจากนี้ยังมีปุ่ม (Button) จำนวน 3 ปุ่ม ที่ใช้สำหรับควบคุม Actuator ที่ติดตั้งอยู่บนหุ่นยนต์ โดยมีการทำงาน 3 ฟังก์ชัน คือ การส่งสัญญาณเตือน การส่งเปิดไฟ และการส่งปิดไฟ การกดปุ่มดังกล่าวนี้ Dashboard จะทำการ Publish การสั่งงานไปยัง MQTT Broker เพื่อส่งต่อไปควบคุมที่หุ่นยนต์อีกครั้งหนึ่ง ผ่านการ Subscribe หัวข้อ Topic ที่ชื่อ “status” ของหุ่นยนต์

#### 4.5 ผลการออกแบบและพัฒนาซอฟต์แวร์ควบคุมการทำงานของหุ่นยนต์

ในการออกแบบและพัฒนาซอฟต์แวร์ที่ใช้ในการควบคุมการทำงานของหุ่นยนต์ จะแบ่งออกเป็น 2 แพลตฟอร์ม ประกอบด้วย ซอฟต์แวร์ที่ทำงานบนบอร์ด Raspberry Pi และซอฟต์แวร์ที่ทำงานบนบอร์ด Arduino

1. บอร์ด Raspberry Pi จะใช้ควบคุมการทำงานของหุ่นยนต์ 2 ระบบ ประกอบด้วย

1) ระบบการตรวจจับกระถางและสิ่งแวดล้อม โดยเขียนโค้ดขึ้นเพื่อใช้ในการถ่ายภาพกระถางในแปลงปลูก จากนั้นจึงนำภาพมาทำการประมวลผลภาพ เพื่อตรวจจับกระถาง แล้วค้นหาตำแหน่งของจุดศูนย์กลางของกระถางด้วย Hough Circle Transform เพื่อส่งค่าตำแหน่งในรูปแบบของค่า  $x$ ,  $y$  และ  $z$  สื่อสารผ่าน Serial Communication ไปสั่งการบอร์ด Arduino

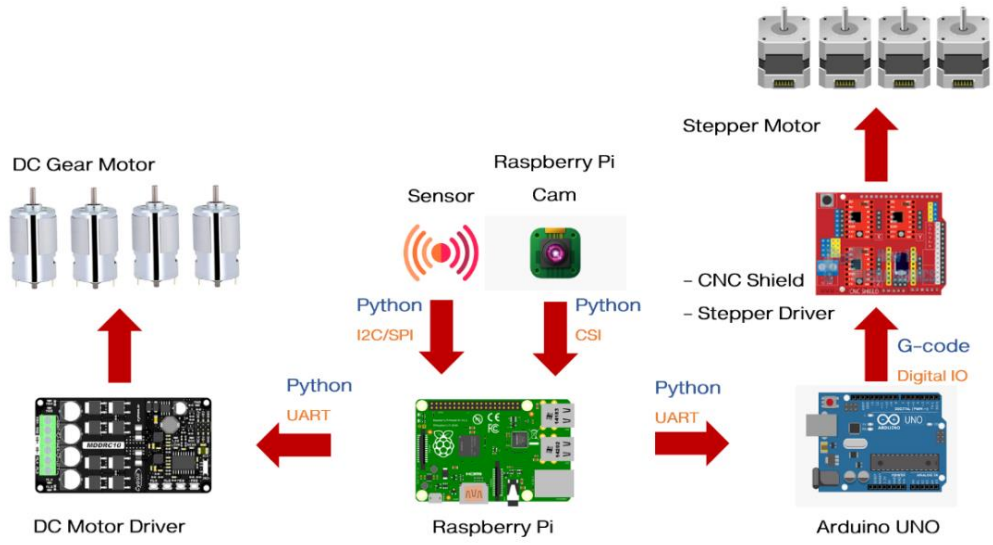
2) ระบบการควบคุมการเคลื่อนที่ของหุ่นยนต์ โดยเขียนโค้ดขึ้นเพื่อใช้ในการส่งสัญญาณไปควบคุมบอร์ด Motor Driver ซึ่งควบคุมทิศทางการหมุนของ Gear Motor ที่ติดตั้ง Mecanum Wheel ไว้ทั้ง 4 ชุด ให้สามารถเคลื่อนที่ได้อิสระตามเงื่อนไขของการควบคุม

2. บนบอร์ด Arduino จะใช้ควบคุมการทำงานของหุ่นยนต์ 2 ระบบ ประกอบด้วย

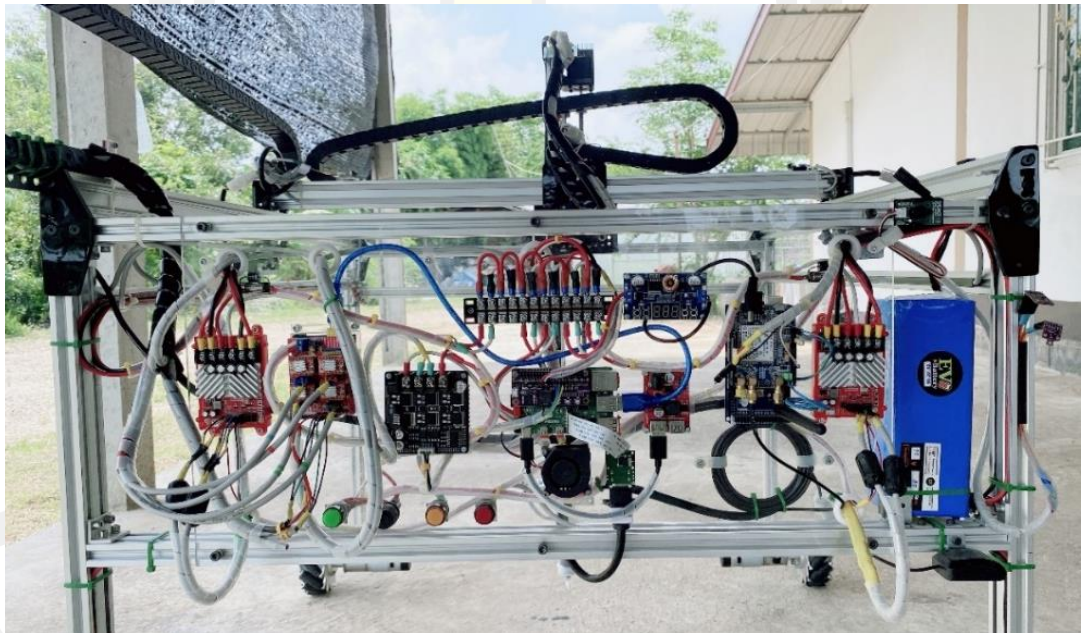
1) ระบบควบคุมการหยุดเมล็ด จะถูกเขียนโค้ดขึ้นเพื่อใช้ในการควบคุมการทำงานของ Stepping Motor จำนวน 4 ชุด ที่ควบคุมการทำงานของทั้ง 3 แกน ประกอบด้วยแกน  $x$ ,  $y$  และ  $z$  ที่ทำหน้าที่ควบคุมการเคลื่อนที่ของ Seed Injector ซึ่งใช้ในการหยุดเมล็ด ด้วยการรับค่าพารามิเตอร์ของตำแหน่งที่ส่งมาจากบอร์ด Raspberry Pi

2) ระบบอินเทอร์เน็ตของสรรพสิ่ง จะถูกเขียนโค้ดขึ้นเพื่อทำการเชื่อมต่อไปยังเครือข่าย GSM 3G เพื่อใช้ทำการ Subscribe และ Publish ข้อมูลของ Topic ไปยัง MQTT Broker รวมถึงควบคุมหุ่นยนต์ผ่านระบบอินเทอร์เน็ต โดยการควบคุมผ่าน MQTT Client

นอกจากนี้ยังออกแบบพัฒนา Dashboard จาก Node-RED บนแพลตฟอร์ม PC บนระบบปฏิบัติการ Windows เพื่อใช้ทำการ Publish และ Subscribe ข้อมูลของ Topic กับ MQTT Broker เพื่อแสดงผลข้อมูลของพารามิเตอร์ในสิ่งแวดล้อมที่ได้รับจากหุ่นยนต์ รวมถึงควบคุมการทำงานของ Actuator บนหุ่นยนต์



ภาพที่ 4.31 โมเดลการทำงานร่วมกันของระบบ Hardware



ภาพที่ 4.32 ชุดแผงควบคุมการทำงานของหุ่นยนต์ต้นแบบ

#### 4.6 ผลการประเมินประสิทธิภาพของหุ่นยนต์

หุ่นยนต์ต้นแบบจะถูกทดสอบการทำงานบนสภาพแวดล้อมจำลอง เพื่อทำการประเมินประสิทธิภาพ โดยสภาพแวดล้อมของ Testbed นี้ จะประกอบด้วยกระถางแบบกลม สีดำ ขนาด 5 นิ้ว จำนวน 54 ใบ ซึ่งวางไว้ในแปลงปลูก จำนวน 6 Cell ใน 2 แปลง แปลงละ 3 Cell ซึ่งในการทดสอบการทำงาน หุ่นยนต์จะถูกติดตั้งในจุดที่เริ่มต้น ซึ่งอยู่ด้านซ้ายมือของ Cell-1 จากนั้นเมื่อทำการเปิดการทำงานของระบบ หุ่นยนต์จะเคลื่อนที่ไปยัง Cell-1 เพื่อทำการตรวจจับกระถางที่ถูกวางไว้บน Cell-1 คำนวณหาจุดกึ่งกลางของกระถาง แล้วทำการหยอดเมล็ดลงในกระถาง เมื่อหุ่นยนต์หยอดเมล็ดลงในกระถางครบแล้วก็จะเคลื่อนที่ต่อไปยัง Cell-2 ซึ่งอยู่ถัดไป พร้อมทั้งทำการตรวจจับกระถาง หาจุดกึ่งกลาง และหยอดเมล็ดลงในกระถาง ทำงานแบบนี้วนไปจนกว่าจะถึง Cell-6 และเมื่อเซ็นเซอร์ตรวจจับได้ว่าหุ่นยนต์เคลื่อนที่มาถึงจุดสิ้นสุด (End Point) ระบบจึงจะสั่งให้หุ่นยนต์หยุดการทำงาน

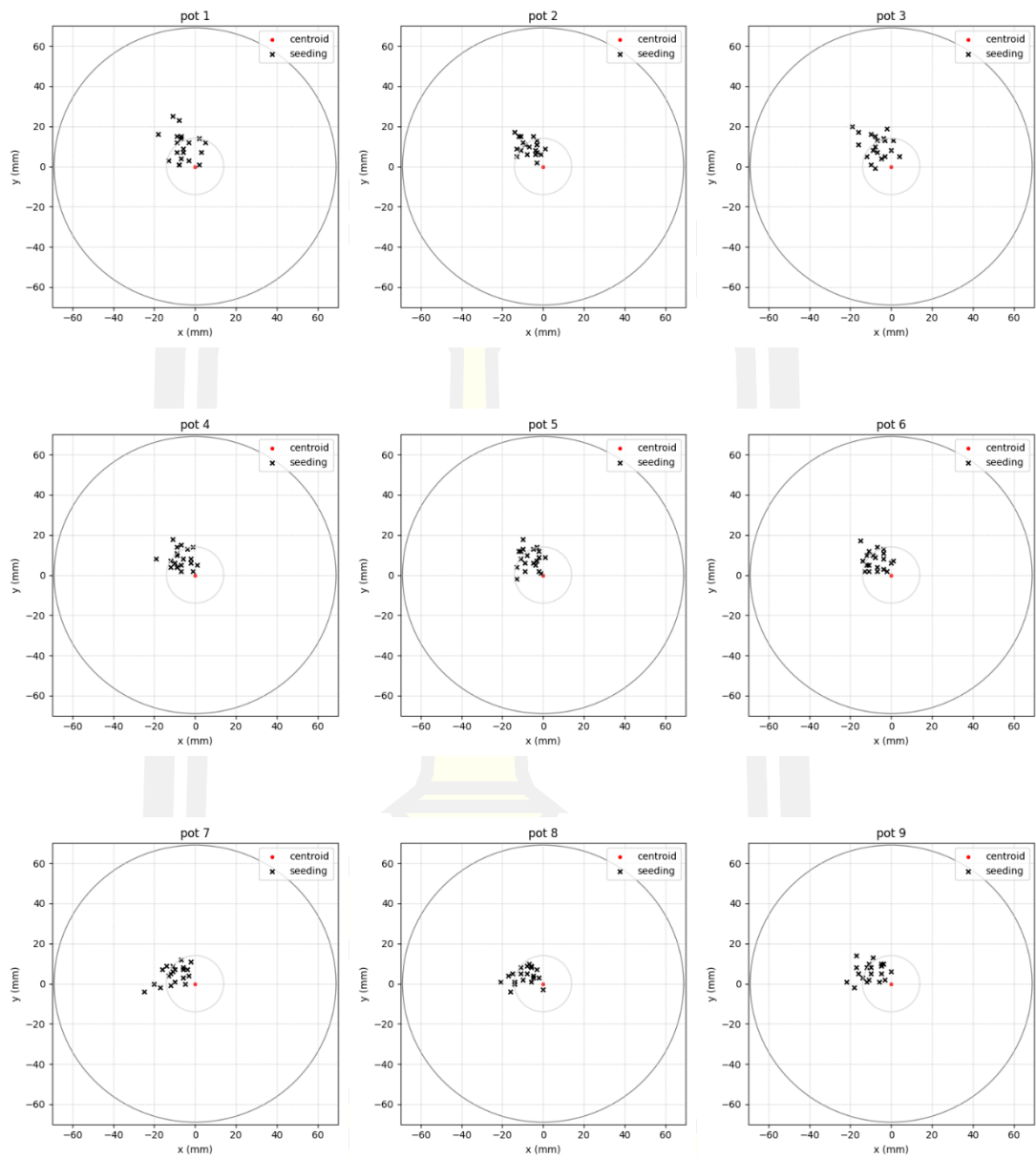


ภาพที่ 4.33 การทดสอบการหยอดเมล็ดของหุ่นยนต์ในแปลงปลูกจำลอง

ตารางที่ 4.9 ตำแหน่งการหยอดเมล็ดในกระถางของหุ่นยนต์

No.	Seeding Position (x, y) from Centroid (mm.)								
	Pot 1	Pot 2	Pot 3	Pot 4	Pot 5	Pot 6	Pot 7	Pot 8	Pot 9
1	-6, 7	1, 9	-5, 4	-1, 2	-4, 5	-4, 3	-17,-2	-5, 4	-3, 2
2	2, 1	-3, 11	-7, 7	-2, 6	-2, 9	-4, 8	-6, 3	-6, 1	0, 6
3	-8, 1	-1, 6	-4, 14	-6, 8	-9, 2	-7, 4	-13, 4	0, 3	-5, 5
4	-6, 9	-4, 8	-10, 16	-9, 10	-9, 6	-2, 2	-10, 7	-5, 3	-10, 8
5	2, 14	-10, 12	-12, 5	-7, 2	-8, 10	0, 6	-20, 0	-8, 5	-17, 14
6	-3, 3	-13, 9	-16, 11	-9, 14	-10, 13	-4, 11	-16,7	-14, 1	-22, 1
7	-7, 4	-8, 6	-16, 17	-19, 8	-3, 7	-8, 9	-5, 0	-10, 2	-11, 10
8	3, 7	-4,6	-8, 10	-1, 14	-2, 12	-9, 10	-3, 4	-16, -4	-9, 13
9	-9, 7	-3, 2	-3, 5	1, 5	-5, 13	-12, 10	-11, 6	-14, 0	-12, 8
10	-8, 1	-3, 11	0, 8	-2, 8	-10, 18	-14, 7	-11, 9	-3, 7	-6, 1
11	-7, 15	-12, 15	-3, 13	-7, 15	-2, 2	-7, 2	-2,11	-21, 1	-11,2
12	-11,15	-7, 10	-7, 13	-9, 11	-12, 12	-4, 13	-7, 12	-15, 5	-10,5
13	-13, 3	-11, 8	-8, 15	-11, 18	1, 9	1, 7	-14, 9	-11, 5	-18, -2
14	-18, 16	-9, 11	-19, 20	-10, 6	-3, 14	-7, 14	-25, -4	-2, 3	-14, 3
15	-9, 15	-3, 13	-2, 19	-7, 5	-13, -2	-15, 17	-10, 1	-17, 4	-4, 10
16	-3, 12	-5, 15	-8, -1	-9, 4	-13, 4	-11, 12	-6, 8	-8, 9	-16, 5
17	-8, 23	-3, 7	-10, 1	-12, 4	-5, 6	-11, 5	-4, 7	-7, 10	-5, 9
18	-7, 14	-13, 5	-9, 8	-12, 7	-1, 1	-11, 2	-6, 7	-6, 9	-12, 1
19	-9, 12	-14, 17	4, 5	-9, 4	-11, 8	-13, 2	-12, -1	-11, 8	-5, 10
20	5,12	-11, 15	1, 13	-4, 13	-11, 12	-12, 5	-12, -5	-6, 8	-17, 8

จากผลการทดสอบในตารางที่ 4.9 จะนำค่าของตำแหน่ง x และ y ไปพลอตกราฟดังภาพที่ 4.34 เพื่อใช้ในการตรวจสอบว่าการหยอดเมล็ดนั้นทำสำเร็จหรือไม่ โดยจุดของการหยอดเมล็ดที่อยู่ภายในวงกลมเล็กนั้น จะเป็นตำแหน่งที่มีความคลาดเคลื่อนไม่เกิน  $\pm 20\%$  (จากจุดกึ่งกลางของกระถางที่ตำแหน่ง 70 มิลลิเมตร) ซึ่งใช้เป็นเกณฑ์ในการตัดสินว่าหุ่นยนต์สามารถหยอดเมล็ดได้สำเร็จ ส่วนการหยอดเมล็ดที่มีตำแหน่งนอกวงกลมดังกล่าว จะหมายถึงการที่หุ่นยนต์หยอดเมล็ดไม่สำเร็จ



ภาพที่ 4.34 กราฟแสดงตำแหน่งของการหยอดเมล็ดในกระถาง

พหุบัณฑิต ชีวะ

ตารางที่ 4.10 ระยะห่างของจุดหยอดเมล็ดกับจุดศูนย์กลางของกระถาง

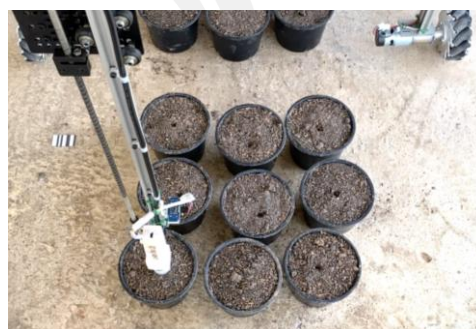
No.	Distance from Centroid (mm.)								
	Pot 1	Pot 2	Pot 3	Pot 4	Pot 5	Pot 6	Pot 7	Pot 8	Pot 9
1	9.22	9.06	6.40	2.24	6.40	5.00	17.12	6.40	3.61
2	2.24	11.40	9.90	6.32	9.22	8.94	6.71	6.08	6.00
3	8.06	6.08	14.56	10.00	9.22	8.06	13.60	3.00	7.07
4	10.82	8.94	18.87	13.45	10.82	2.83	12.21	5.83	12.81
5	14.14	15.62	13.00	7.28	12.81	6.00	20.00	9.43	22.02
6	4.24	15.81	19.42	16.64	16.40	11.70	17.46	14.04	22.02
7	8.06	10.00	23.35	20.62	7.62	12.04	5.00	10.20	14.87
8	7.62	7.21	12.81	14.04	12.17	13.45	5.00	16.49	15.81
9	11.40	3.61	5.83	5.10	13.93	15.62	12.53	14.00	14.42
10	8.06	11.40	8.00	8.25	20.59	15.65	14.21	7.62	6.08
11	16.55	19.21	13.34	16.55	2.83	7.28	11.18	21.02	11.18
12	18.60	12.21	14.76	14.21	16.97	13.60	13.89	15.81	11.18
13	13.34	13.60	17.00	21.10	9.06	7.07	16.64	12.08	18.11
14	24.08	14.21	27.59	11.66	14.32	15.65	25.32	3.61	14.32
15	17.49	13.34	19.10	8.60	13.15	22.67	10.05	17.46	10.77
16	12.37	15.81	8.06	9.85	13.60	16.28	10.00	12.04	16.76
17	24.35	7.62	10.05	12.65	7.81	12.08	8.06	12.21	10.30
18	15.65	13.93	12.04	13.89	1.41	11.18	9.22	10.82	12.04
19	15.00	22.02	6.40	9.85	13.60	13.15	12.04	13.60	11.18
20	13.00	18.60	13.04	13.60	16.28	13.00	13.00	10.00	18.79
Average	12.72	12.48	13.68	11.79	11.41	11.56	12.66	11.09	12.97
% Error	18.17	17.83	19.54	16.84	16.30	16.51	18.09	15.84	18.53

จากผลการทดสอบในตารางที่ 4.10 จะแสดงถึงค่าระยะห่างของตำแหน่งจุดศูนย์กลางของกระถางกับตำแหน่งจุดศูนย์กลางที่ Seed Injector แทนลงไปบนผิวดินในกระถางแต่ละใบ ซึ่งมีหน่วยเป็นมิลลิเมตร โดยค่าผลการทดสอบนี้จะใช้เพื่อประเมินว่า การทำงานของ Seed Injector ทำงานได้สำเร็จหรือไม่ในการหยอดเมล็ดในแต่ละครั้ง ตามเกณฑ์การประเมินที่หัวปลูกของหุ่นยนต์

จะต้องเคลื่อนที่ไปยังตำแหน่งของกระถาง และหยอดเมล็ดพันธุ์ลงบนกระถางได้ โดยมีความคลาดเคลื่อนในการหยอด  $\pm 20\%$  (หรือไม่เกินค่า  $\pm 14$  มิลลิเมตร) จากจุดกึ่งกลางของกระถาง ดังนั้นหากค่าระยะห่างของกระถางใบใดในตารางผลการทดสอบมีค่าเกิน  $\pm 14$  มิลลิเมตร จะถือว่า การหยอดเมล็ดในครั้งนั้นกระทำไม่สำเร็จ



(a) ด้านข้าง



(b) ด้านบน

ภาพที่ 4.35 การหยอดเมล็ดลงในกระถางของ Seed Injector



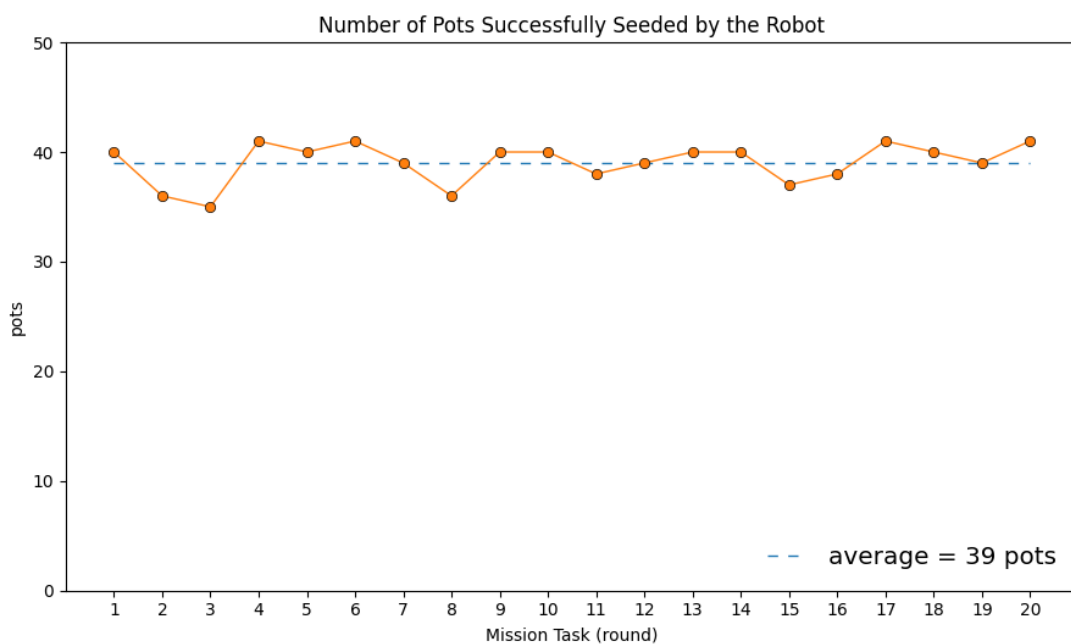
ภาพที่ 4.36 การวัดตำแหน่ง  $x$  และ  $y$  เพื่อเก็บผลการทดสอบ



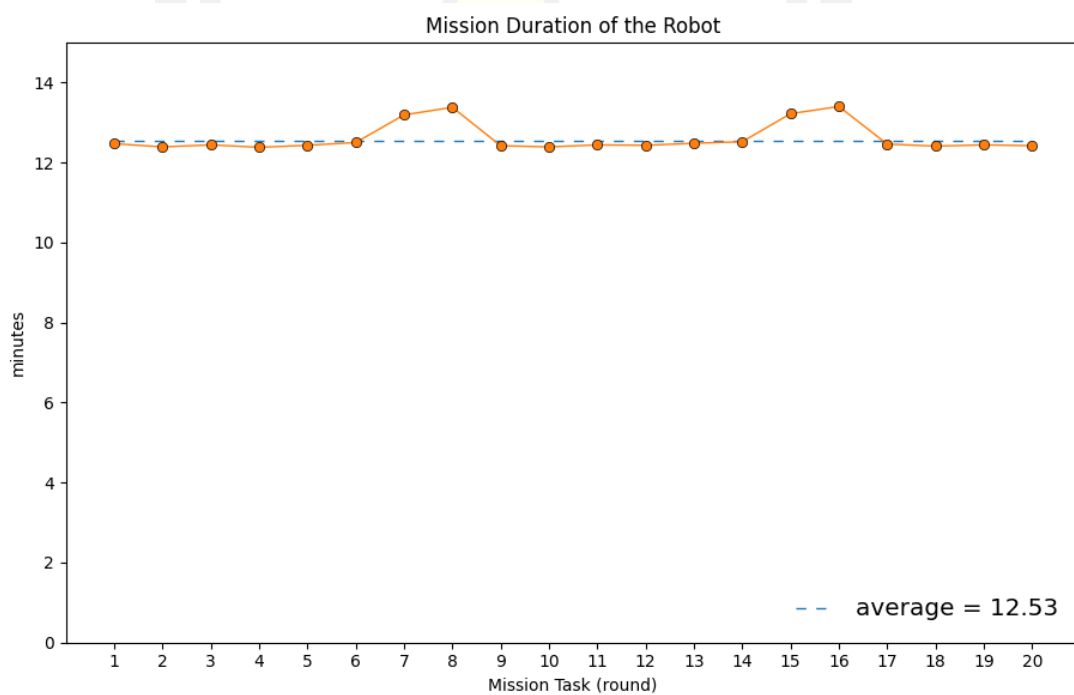
ตารางที่ 4.11 ผลการประเมินประสิทธิภาพของหุ่นยนต์

No.	Total Pots	Number of Successful Pots	Mission Duration (Minute)	Accuracy
1	54	40	12.47	0.74
2	54	36	12.39	0.67
3	54	35	12.44	0.65
4	54	41	12.38	0.76
5	54	40	12.43	0.74
6	54	41	12.50	0.76
7	54	39	13.19	0.72
8	54	36	13.38	0.67
9	54	40	12.42	0.74
10	54	40	12.39	0.74
11	54	38	12.44	0.70
12	54	39	12.43	0.72
13	54	40	12.48	0.74
14	54	40	12.52	0.74
15	54	37	13.22	0.69
16	54	38	13.40	0.70
17	54	41	12.46	0.76
18	54	40	12.41	0.74
19	54	39	12.44	0.72
20	54	41	12.42	0.76
Average		39	12.53	0.72

จากตารางที่ 4.11 ผลการทดสอบเพื่อประเมินประสิทธิภาพพบว่า หุ่นยนต์สามารถทำภารกิจได้จนเสร็จสิ้นในทุกรอบของการทำงาน โดยสามารถหยอดเมล็ดลงกระถางได้บนพื้นที่ของทั้ง 6 Cell ใน Testbed นี้ ซึ่งความสำเร็จของการหยอดเมล็ดในการทำภารกิจแต่ละครั้ง แสดงไว้ดังกราฟในภาพที่ 4.37 โดยมีค่าเฉลี่ยของจำนวนกระถางที่หยอดเมล็ดได้สำเร็จ เท่ากับ 39 กระถาง และมีค่า Accuracy เท่ากับ 0.72



ภาพที่ 4.37 กราฟแสดงจำนวนกระถางที่หุ่นยนต์หยอดเมล็ดได้สำเร็จในการปฏิบัติการกิจ



ภาพที่ 4.38 กราฟแสดงเวลาที่หุ่นยนต์ใช้ในการปฏิบัติการกิจในแต่ละรอบ

จากกราฟแสดงเวลาที่หุ่นยนต์ใช้ในการปฏิบัติภารกิจ ในภาพที่ 4.38 นั้น มีค่าเฉลี่ยของ Mission Duration เท่ากับ 12.53 นาที ซึ่งการทำภารกิจในแต่ละรอบของหุ่นยนต์จะใช้เวลาในกระบวนการหยุดเมล็ด โดยเฉลี่ยคิดเป็น 13.37 วินาทีต่อกระถาง โดยคำนวณเป็นอัตราความเร็วในการปลูกพืชด้วยการหยุดเมล็ดได้ประมาณ 257 กระถางต่อชั่วโมง และวัดความเร็วในการเคลื่อนของหุ่นยนต์ (Velocity) จากจุดหนึ่งไปยังอีกจุดหนึ่ง คือ 6 เมตรต่อนาที หรือ 360 เมตรต่อชั่วโมง หากสังเกตจากตารางผลการทดลองในรอบที่ 6 – 8 และรอบที่ 14 – 16 จะพบว่า มีค่า Mission Duration ที่เพิ่มสูงขึ้นเรื่อย ๆ เนื่องจากสาเหตุของระดับแรงดันในแบตเตอรี่ลดลงจาก 11.9 V (วัดแรงดันเมื่อทำการชาร์จแบตเตอรี่จนเต็มและทำการเชื่อมต่อวงจรพร้อมทั้งเปิดการทำงานแบบ Full Load) เหลือเพียง 10.1 V ทำให้ประสิทธิภาพการจ่ายกระแสไฟฟ้าให้กับ Stepper Motor และ Gear Motor ลดลง ซึ่งจะต้องหยุดทำการทดสอบและชาร์จแบตเตอรี่ เพื่อป้องกันความคลาดเคลื่อนและแบตเตอรี่ชำรุดเสียหายจากการมีระดับแรงดันคงเหลือของเซลล์แบตเตอรี่ต่ำกว่า 9 V โดยทดสอบพบว่าหุ่นยนต์ต้นแบบที่ถูกพัฒนาขึ้นมี Running Time สูงสุดเฉลี่ยประมาณ 2 ชั่วโมง ก่อนที่ระดับแรงดันจะลดต่ำกว่า 9 V ที่การใช้งาน Full Load บนแบตเตอรี่ ประเภท Lithium-Ion ขนาด 30,000 mAh

สรุปภาพรวมความสามารถของต้นแบบหุ่นยนต์เพาะปลูกพืชแบบอัตโนมัติ “Lady Bug” ที่ถูกพัฒนาขึ้นในงานวิจัยนี้

1. มีต้นทุนในการพัฒนาต่ำ ขนาดของโครงสร้างเล็ก น้ำหนักเบาเพียง 20 กิโลกรัม
3. เคลื่อนที่ไปได้อิสระรอบทิศทางด้วย Mecanum Wheel จึงทำงานในพื้นที่จำกัดได้ดี
4. มีระบบประมวลผลภาพซึ่งช่วยให้การทำงานมีความยืดหยุ่น ตลอดจนทำให้สามารถเพิ่มหรือลดพื้นที่การทำงานได้
5. มีความเร็วในการเคลื่อนที่ 6 เมตรต่อนาที (360 เมตรต่อชั่วโมง) และความเร็วในการหยุดเมล็ด 13.37 วินาทีต่อกระถาง หรือในอัตราความเร็วการหยุดเมล็ด 257 กระถางต่อชั่วโมง
6. การหยุดเมล็ดมีค่า Accuracy เท่ากับ 0.72
7. สามารถตรวจสอบพารามิเตอร์ในสิ่งแวดล้อมที่หุ่นยนต์ทำงานอยู่ พร้อมทั้งควบคุม Actuator ที่ติดตั้งบนหุ่นยนต์ได้โดยผ่านทางระบบอินเทอร์เน็ตของสรรพสิ่งบนเครือข่าย GSM 3G
8. หุ่นยนต์สามารถเพิ่มฟังก์ชันการทำงานได้ในอนาคต เช่น การรดน้ำ การหยุดปุ๋ย และฉีดพ่นสารเคมี โดยการพัฒนา End-Effector ติดตั้งเพิ่มในแกน z
9. มี Running Time ต่อเนื่องเฉลี่ยเป็นระยะเวลา 2 ชั่วโมง โดยหุ่นยนต์สามารถทำภารกิจได้สูงสุด 9 รอบ และสามารถหยุดเมล็ดได้ประมาณ 486 กระถาง ข้อจำกัดใน Running Time เกิดจากการใช้งานพลังงานที่ค่อนข้างสูง เนื่องมาจากการทำงานของการประมวลผลของบอร์ด

Raspberry Pi และ Arduino รวมถึง Stepper Motor, Gear Motor และ Linear Actuator ที่มีจำนวนมากถึง 9 ชุด

ตารางที่ 4.12 ผลการประเมินประสิทธิภาพเปรียบเทียบกับงานวิจัยที่ผ่านมา

Solution	Experimental Results
Lady Bug	<ul style="list-style-type: none"> <li>- มีต้นทุนการพัฒนาระบบต่ำ</li> <li>- มีขนาดเล็ก น้ำหนัก 20 กิโลกรัม</li> <li>- สามารถปรับขนาดของพื้นที่การทำงานได้อัตโนมัติ โดยการตรวจจับ Tag</li> <li>- สามารถปรับเปลี่ยนการทำงานตามขนาดของกระถาง และตำแหน่งของการจัดวางกระถางในแปลงโดยอัตโนมัติ</li> <li>- มีความแม่นยำในการระบุตำแหน่งโดยการใช้กล้อง</li> <li>- มีการทำงานแบบ Automatic และ Dynamic</li> <li>- สามารถเคลื่อนที่อิสระรอบทิศทาง ด้วยความเร็ว 360 เมตรต่อชั่วโมง</li> <li>- Seed Injector มีกระบอกเก็บเมล็ด สามารถหยอดเมล็ดทันที ความเร็วในการทำงาน 257 กระถางต่อชั่วโมง (เฉลี่ย 13.37 วินาทีต่อกระถาง)</li> <li>- Monitor พารามิเตอร์ และควบคุมการทำงานได้ด้วยอินเทอร์เน็ทของสรรพสิ่งผ่าน MQTT Client และ Dashboard</li> <li>- ต้นแบบมีแค่ฟังก์ชันการปลูก แต่สามารถเพิ่มฟังก์ชันการทำงานได้ในอนาคต</li> <li>- ใช้แบตเตอรี่ Lithium-Ion ขนาด 30,000 mAh เป็นแหล่งจ่ายพลังงาน มี Running Time ต่อเนื่องเฉลี่ยเป็นระยะเวลา 2 ชั่วโมง</li> </ul>
CeRSAA [47]	<ul style="list-style-type: none"> <li>- มีต้นทุนการพัฒนาระบบที่สูง</li> <li>- มีขนาดใหญ่ แข็งแรง ใช้พื้นที่ในการติดตั้งมาก</li> <li>- มีพื้นที่การทำงานเป็นรูปสี่เหลี่ยมผืนผ้า ไม่สามารถปรับขนาดได้</li> <li>- มีความแม่นยำในการระบุตำแหน่งโดยการใช้กล้อง</li> <li>- มีการทำงานแบบ Semi-Automatic</li> <li>- หุ่นยนต์ไม่สามารถเคลื่อนที่ได้ มีเฉพาะแขนหุ่นยนต์เท่านั้นที่เคลื่อนที่ตามแนวตั้งฉากกับการเคลื่อนที่ทิศทางของพาเลทที่จัดวางกระถางต้นไม้ เมื่อทำงานจนครบแล้วพาเลทอันใหม่จะถูกเลื่อนเข้ามาแทนที่</li> <li>- ความเร็วในการทำงาน 420 กระถางต่อชั่วโมง (เฉลี่ย 8.57 วินาทีต่อกระถาง)</li> <li>- มีฟังก์ชันการทำงานที่หลากหลาย เช่น หยอดปุ๋ย และพ่นสารเคมี</li> <li>- ใช้กระแสในการทำงานที่สูง จึงต้องใช้แหล่งจ่ายไฟประเภท Switching Power Supply เพื่อให้เพียงพอกับการทำงาน</li> </ul>

ตารางที่ 4.12 (ต่อ) ผลการประเมินประสิทธิภาพเปรียบเทียบกับงานวิจัยที่ผ่านมา

Solution	Experimental Results
RIRRIG [48]	<ul style="list-style-type: none"> <li>- มีต้นทุนการพัฒนาระบบที่ต่ำ</li> <li>- มีขนาดใหญ่ แข็งแรง ใช้พื้นที่ในการติดตั้งมาก</li> <li>- พื้นที่การทำงานขึ้นอยู่กับขนาดโครงสร้างที่ครอบแปลงปลูก ปรับเปลี่ยนไม่ได้</li> <li>- ระบุตำแหน่งโดยการกำหนดในซอฟต์แวร์ควบคุม ไม่มีการใช้กล้อง</li> <li>- การทำงานไม่เป็นแบบ Automatic ตั้งการทำงานเอาไว้ล่วงหน้า</li> <li>- ถูกติดตั้งไว้ตายตัว ไม่สามารถเคลื่อนที่ได้</li> <li>- ต้นแบบมีแค่ฟังก์ชันการทำงานเดียว ไม่สามารถเพิ่มได้อีก</li> <li>- ใช้เซ็นเซอร์พื้นฐานทั่วไป ระบบไม่ซับซ้อน การบำรุงรักษาระบบกลไกง่าย</li> </ul>
FarmBot [49]	<ul style="list-style-type: none"> <li>- มีต้นทุนการพัฒนาระบบที่สูง</li> <li>- มีขนาดใหญ่ แข็งแรง ใช้พื้นที่ในการติดตั้งมาก</li> <li>- พื้นที่การทำงานขึ้นอยู่กับขนาดของรางที่ถูกติดตั้งบนแปลงปลูก ปรับเปลี่ยนไม่ได้</li> <li>- มีความแม่นยำในการระบุตำแหน่งโดยการใช้อัลตราซาวด์</li> <li>- มีการทำงานแบบ Automatic</li> <li>- ถูกติดตั้งอยู่กับกระบะปลูก ไม่สามารถเคลื่อนที่ได้</li> <li>- Seed Injector ต้องกลับไปดูเมล็ดที่ Seed Tray ซึ่งอยู่ที่จุดอ้างอิงทุกครั้ง เนื่องจากไม่มีกระบอกเก็บเมล็ด ทำให้สูญเสียเวลาโดยไม่จำเป็น</li> <li>- คุณสถานะและควบคุมการทำงานได้ด้วยระบบ Cloud Service</li> <li>- มีฟังก์ชันการทำงานที่หลากหลาย โดยใช้การเปลี่ยนชุด End-Effector และสามารถปลูกพืชได้หลายชนิด</li> <li>- ใช้พลังงานจาก Switching Power Supply</li> </ul>
SAASbot [52]	<ul style="list-style-type: none"> <li>- มีต้นทุนการพัฒนาระบบที่ต่ำ</li> <li>- มีขนาดเล็ก น้ำหนัก 24.8 กิโลกรัม</li> <li>- ปรับขนาดของพื้นที่การทำงานได้โดยใช้แผ่นยางวางเป็นรูปแบบรางครอบลงบน Cell ให้เกิดเป็นตาราง Grid เพื่อกำหนดขอบเขต</li> <li>- มีความแม่นยำในการระบุตำแหน่งโดยการใช้อัลตราซาวด์</li> <li>- มีการทำงานแบบ Automatic</li> <li>- สามารถเคลื่อนที่ได้</li> <li>- Seed Injector มีความเร็วในการทำงาน 90 ต้นต่อชั่วโมง</li> <li>- Monitor พารามิเตอร์ และควบคุมการทำงานได้ด้วย Mobile Application</li> <li>- มีฟังก์ชันการทำงานที่หลากหลาย</li> <li>- แบตเตอรี่ Lithium-Ion ขนาด 2,200 mAh มี Running Time เฉลี่ย 1 ชั่วโมง</li> </ul>

## บทที่ 5

### สรุปและอภิปรายผล

งานวิจัยนี้เป็นงานวิจัยเพื่อพัฒนาต้นแบบหุ่นยนต์ที่ใช้สำหรับทำการเพาะปลูกพืชบนกระถางที่ถูกจัดเรียงไว้ในโรงเรือน โดยเป็นการประยุกต์การทำงานของหุ่นยนต์เข้ากับเทคโนโลยีอินเทอร์เน็ตของสรรพสิ่ง การประมวลผลภาพ และปัญญาประดิษฐ์ ซึ่งสามารถสรุปผลการวิจัย อภิปรายผลสัมฤทธิ์ทางการวิจัย รวมทั้งข้อเสนอแนะและแนวทางในการทำวิจัยต่อ ดังนี้

#### 5.1 สรุปและอภิปราย

ปัจจุบันภาคเกษตรกรรมของโลกได้เข้าสู่ยุคของการใช้หุ่นยนต์เพื่อการเกษตรแล้ว ภายหลังจากวิกฤติของปัญหาการขาดแคลนแรงงานที่มีแนวโน้มสูงขึ้น สวนทางกลับการที่สินค้าเกษตรและอาหารมีความต้องการการบริโภคในตลาดโลกที่เพิ่มขึ้น ซึ่งการนำหุ่นยนต์เข้ามาใช้ในภาคการเกษตรนี้ นอกจากจะช่วยแก้ไขปัญหาการขาดแคลนแรงงานแล้ว ยังส่งผลให้สามารถผลิตสินค้าเกษตรได้ปริมาณสูงขึ้น มีต้นทุนที่ลดลง สามารถควบคุมคุณภาพการผลิตและการเก็บเกี่ยวได้อย่างแม่นยำ

อย่างไรก็ตาม การประยุกต์ใช้งานหุ่นยนต์ในภาคการเกษตรที่ผ่านมา พบว่าเป็นการพัฒนาต้นแบบให้ทำงานเฉพาะด้าน และมีหน้าที่แตกต่างกันไปตามประเภทงาน เช่น หุ่นยนต์ตรวจการณ์และเก็บข้อมูล หุ่นยนต์กำจัดแมลงและศัตรูพืช หุ่นยนต์เก็บเกี่ยวผลผลิต หุ่นยนต์ฉีดพ่นสารเคมี และหุ่นยนต์เพาะปลูกพืช โดยจะถูกนำไปใช้บนพื้นที่ขนาดใหญ่ในต่างประเทศ ซึ่งแตกต่างจากประเทศไทยที่มีการทำเกษตรกรรมในสเกลที่เล็กกว่า บนพื้นที่ของไร่ นา สวน หรือโรงเรือน

ดังนั้นในการวิจัยนี้จึงมุ่งเน้นการพัฒนาต้นแบบหุ่นยนต์ที่ใช้สำหรับทำการเพาะปลูกพืชแบบอัตโนมัติ เพื่อหยอดเมล็ดพืชบนกระถางพลาสติกกลมที่ถูกจัดวางไว้ในแปลงปลูกของโรงเรือน ซึ่งสามารถปรับขนาดพื้นที่ และปรับเปลี่ยนตำแหน่งการจัดวางของกระถางปลูกได้ จากผลการวิจัยสามารถสรุปและอภิปรายผลได้ดังนี้

1) การเคลื่อนที่ของหุ่นยนต์ที่ใช้ล้อ Mecanum Wheel นั้น จะสามารถเคลื่อนที่รอบทิศทางได้อย่างอิสระ โดยอาศัยชุด Roller ที่เอียงมุม 45 องศา บนชุดเฟรมล้อ ซึ่งทำการควบคุมทิศทางการหมุนแยกกันในแต่ละล้อ การเคลื่อนที่ไปในทิศทางต่าง ๆ จึงต้องใช้ความสัมพันธ์ในการหมุนของล้อทั้งสิ้น ด้วยหลักการดังกล่าว แต่ละล้อจะต้องสร้างแรงขับเคลื่อนกับเส้นทแยงมุมของเฟรมที่สอดคล้องกัน โดยมีการเปลี่ยนแปลงความเร็วในการหมุนและทิศทางของล้อ ซึ่งเป็นผลรวมของเวกเตอร์แรงจากแต่ละล้อ จึงส่งผลให้เกิดการเคลื่อนไหวเชิงเส้น รวมถึงการหมุนของโครงสร้างหุ่นยนต์ทำให้หุ่นยนต์สามารถเคลื่อนที่ได้อิสระ และสามารถทำงานในพื้นที่ซึ่งมีขนาดจำกัดได้

โดยในการวิจัยยังพบอีกว่า การเคลื่อนที่ของหุ่นยนต์นั้นยังมีข้อจำกัดในส่วนของการเคลื่อนที่บนพื้นผิวของพื้นที่ทำงาน (Workspace Surface) ซึ่งการเคลื่อนที่ที่จะมีความถูกต้องและแม่นยำนั้น หุ่นยนต์จะต้องทำงานอยู่บนพื้นผิวที่มีความราบ เรียบ แข็ง มีพื้นผิวที่สม่ำเสมอ และไม่มีสิ่งกีดขวาง หากหุ่นยนต์ทำงานอยู่บนพื้นผิวในลักษณะอื่น ๆ อาจส่งผลกระทบต่อให้เกิดการเคลื่อนที่ที่ผิดพลาดได้ ตัวอย่างเช่น การทำงานบนพื้นผิวดินทราย แม้จะมีความราบและเรียบ แต่เนื่องจากเม็ดทรายมีขนาดเล็กจึงสามารถเข้าไปขัดขวางการทำงานของชุด Roller ได้ โดยเข้าไปเกาะบนผิวยาง ทำให้ติดขัดจนส่งผลให้การหมุนของชุด Roller ทั้ง 4 ล้อ ไม่สัมพันธ์กันตามเงื่อนไขของการทำงานปกติ จึงส่งผลให้การเคลื่อนที่เกิดการไถล (Slip) ออกจากตำแหน่งที่ต้องการ

2) การนำเอาแพลตฟอร์มของหุ่นยนต์ต่างชนิดมาผนวกการทำงานเข้าด้วยกันอย่างเหมาะสม จะส่งผลให้แพลตฟอร์มใหม่ที่เกิดขึ้นนี้ เป็นการนำเอาประสิทธิภาพที่เป็นจุดเด่นของหุ่นยนต์ทั้ง 2 แพลตฟอร์มมารวมเข้าด้วยกัน เพื่อขยายขีดความสามารถ ตลอดจนแก้ไขจุดอ่อนของแพลตฟอร์มเดิม เช่นเดียวกับงานวิจัยนี้ที่นำหุ่นยนต์ในแพลตฟอร์ม Manipulator Robot ซึ่งมีหน้าที่ในการควบคุมอุปกรณ์ส่วนปลายที่ใช้ในการปลูก ให้ทำงานร่วมกับ Mobile Robot ซึ่งมีหน้าที่ควบคุมการเคลื่อนที่ของหุ่นยนต์ภายในแปลงปลูก กลายเป็นแพลตฟอร์มใหม่ที่มีประสิทธิภาพสูงขึ้น

3) การเคลื่อนที่ของระบบขับเคลื่อนทั้ง 3 แกน มีข้อสังเกตว่า ความแม่นยำของชุดแกน  $x$  และแกน  $y$  ถึงแม้จะเป็นระบบ Belt Driven System และใช้อุปกรณ์เช่นเดียวกัน แต่ก็มีปัจจัยที่ส่งผลให้เกิดความคลาดเคลื่อนที่แตกต่างได้ เนื่องจากแกน  $x$  มี Stepper Motor ขับเคลื่อนสายพานเพียงหนึ่งตัว ในขณะที่แกน  $y$  ต้องใช้ Stepper Motor ถึง 2 ตัวในการขับเคลื่อน 2 ด้าน รวมถึงความตึง-หย่อนของสายพานที่ต่างกัน ซึ่งทำให้เกิดความคลาดเคลื่อนขึ้นได้ ส่วนแกน  $z$  นั้น ใช้ระบบ Lead Screw Driven System พบว่ามีข้อจำกัดของการเกิด Backlash ในการหมุนของเพลาเกลียวที่ส่งผลให้การเคลื่อนที่ของแกน  $z$  จะไม่ได้ระยะที่เท่าเดิมในระหว่างการเคลื่อนที่ไปและเคลื่อนที่กลับ เกิดเป็นความคลาดเคลื่อนที่สะสมได้ ซึ่งในการวิจัยได้ปรับปรุงข้อผิดพลาดดังกล่าวด้วยการใช้ Anti-Backlash Nut Block เข้าป้องกันการ Slip ในระหว่างการทำงาน

4) แม้ว่าระบบการประมวลผลภาพจะสามารถตรวจจับกระถางได้ครบทุกใบในทุกการทดลอง แต่เมื่อนำตำแหน่งอ้างอิงในการทดลองมาเปรียบเทียบกับผลการตรวจจับที่ได้ จะพบว่ามีความคลาดเคลื่อนของพิกัด  $x$  และ  $y$  ในแต่ละครั้งที่ทำการทดสอบ ซึ่งเกิดขึ้นได้จากสถานะของแสงในสภาพสิ่งแวดล้อม เป็นปัจจัยที่ไม่สามารถควบคุมได้ เนื่องจาก Sensitivity ของ Hough Circle Transform ขึ้นอยู่กับสภาพแสงในขณะนั้น มีผลทำให้การตรวจจับขอบภาพมีการเพิ่ม หรือลดขนาดหรือขยับตำแหน่งไป ส่งผลให้การตรวจจับและคำนวณหาจุด Centroid ผิดพลาด

ข้อจำกัดในการตรวจจับกระถางของหุ่นยนต์ต้นแบบ พบว่าตำแหน่งของการจัดวางกระถางนั้น มีผลต่อการตรวจจับที่ถูกต้องของระบบการรับรู้ หากวางกระถางอยู่ในตำแหน่งที่ก้องไม่

สามารถจับภาพองค์ประกอบของกระถางได้ จะส่งผลให้ระบบการรับรู้ตรวจไม่พบการมีอยู่ของกระถางในภาพที่กำลังทำการตรวจสอบ ซึ่งองค์ประกอบของกระถางขั้นต่ำสุดที่ระบบจะสามารถตรวจสอบได้ จะต้องมีพื้นที่ภาพหน้าตัดของกระถาง  $\pm 50$  เปอร์เซ็นต์ โดยที่มีจุดกึ่งกลางของกระถางปรากฏอยู่ในภาพ และสามารถลากเส้นรัศมี R จากจุดกึ่งกลางไปยังขอบกระถางในภาพได้ด้วย ในส่วนของพื้นที่ที่กระถางถูกวาง ก็มีผลกระทบต่อการทำงานของระบบการรับรู้เช่นเดียวกัน ซึ่งพบว่า ถ้าพื้นที่ที่ใช้ในการวางกระถางนั้น หากมีความต่างของสี หรือ Contrast ของกระถางกับพื้นที่ที่จัดวางมาก จะส่งผลให้การตรวจจับกระถางทำได้อย่างมีประสิทธิภาพ คือ ตรวจจับจำนวนของกระถางได้ถูกต้อง มีขนาดและตำแหน่งของการจัดวางที่คลาดเคลื่อนจากค่าจริงเพียงเล็กน้อย ซึ่งความแตกต่างนี้จะสัมพันธ์กับความเข้มแสงในแต่ละ Pixel นั่นเอง

5) ระบบควบคุมการหยอดเมล็ดมีปริมาณของการหยอดเมล็ดต่อกระถางในแต่ละครั้งไม่คงที่ โดยมีปริมาณการหยอดแต่ละครั้ง คือ 1 – 4 เมล็ด โดยอัตราการหยอด 2 เมล็ดต่อกระถาง เป็นอัตราของการหยอดเมล็ดพันธุ์ที่เหมาะสม กรณีอัตราการงอกของเมล็ดอยู่ในระดับ 90 เปอร์เซ็นต์

ส่วนข้อจำกัดของระบบควบคุมการหยอดเมล็ดซึ่งพบในการวิจัยนั้น มีสาเหตุมาจากรูปร่าง ลักษณะ และขนาดของเมล็ดที่ไม่เท่ากัน ซึ่งเป็นปัจจัยทางธรรมชาติที่ไม่สามารถควบคุมได้ของเมล็ดพันธุ์พืช ทำให้การหยอดเมล็ดในแต่ละครั้งมีปริมาณของเมล็ดพันธุ์ที่ไม่สม่ำเสมอ ซึ่งชุด Seed Injector ในการวิจัยครั้งนี้ ถูกออกแบบมาเพื่อใช้กับเมล็ดพันธุ์ที่มีลักษณะเป็นทรงกลม มีขนาดเส้นผ่านศูนย์กลาง 2 มิลลิเมตร (ความคลาดเคลื่อน  $\pm 0.2$  มิลลิเมตร) ซึ่งมีอัตราการหยอดเมล็ดเท่ากับ 2 เมล็ดต่อกระถางในการหยอดแต่ละครั้ง

6) การทดสอบ Publish และ Subscribe ข้อมูลของ Topic จากหุ่นยนต์ในแบบ Real Time ด้วย MQTT Protocol พบว่า จำนวนครั้งในการรับ-ส่ง และข้อมูลจากชุดเซ็นเซอร์มีความถูกต้องของข้อมูล ถึงแม้หุ่นยนต์จะทำงานอยู่บนเครือข่าย GSM 3G ในพื้นที่ห่างไกลจากสถานีทวนสัญญาณที่ส่งผลให้การรับ-ส่งข้อมูลในระหว่างการทดสอบมีความเร็วที่ต่ำก็ตาม เนื่องจาก MQTT Protocol มีความรวดเร็วในการทำงาน ใช้ทรัพยากรในการทำงานและใช้พลังงานที่ต่ำ รวมถึงเป็นระบบ Event-Driven จึงสามารถรับข้อมูลได้ตลอดเวลาที่มีการ Publish ข้อมูล

7) ความแม่นยำของการหยอดเมล็ดพบว่า แม้การหยอดเมล็ดในแต่ละครั้งของหุ่นยนต์จะไม่อยู่จุดกึ่งกลางกระถาง แต่เมื่อวัดจากจุดศูนย์กลางของกระถางถึงจุดศูนย์กลางของหลุมที่ Seed Injector เจาะลงในผิวดิน มีความคลาดเคลื่อนเฉลี่ยอยู่ในช่วง  $\pm 20\%$  ซึ่งยังอยู่ในพื้นที่ที่เหมาะสมในการปลูกพืชให้เติบโตได้ตามปกติ เมื่อพืชเจริญเติบโตขึ้นจะไม่ล้มกระถางไปเบียดต้นที่อยู่ข้างเคียงจนเกินไป จึงถือว่าเป็นตำแหน่งที่ยอมรับได้สำหรับการปลูกพืช ซึ่งพบว่าความคลาดเคลื่อนของตำแหน่งในการหยอดเมล็ดนี้ สามารถเกิดได้ตั้งแต่ขั้นตอนของการตรวจจับกระถางด้วยการประมวลผลภาพ ไปจนถึงขั้นตอนการเคลื่อนที่ในระบบขับเคลื่อนสายพานของชุด CNC



8) การทดสอบเพื่อประเมินประสิทธิภาพของหุ่นยนต์พบว่า หุ่นยนต์สามารถเคลื่อนที่ไปทำภารกิจได้สำเร็จทุกรอบการทำงาน โดยสามารถหยุดเมื่อดึงลงกระถางได้ทุกใบ ซึ่งมีค่า Accuracy ในการหยุดเมล็ด เท่ากับ 0.72 ซึ่งพบว่าหุ่นยนต์มี Mission Duration เฉลี่ยแต่ละรอบ 12.53 นาที ซึ่งใช้เวลาในการหยุดเมล็ดเฉลี่ย 13.37 วินาทีต่อกระถาง คำนวณเป็นความเร็วในการหยุดเมล็ดได้ที่อัตรา 257 กระถางต่อชั่วโมง โดยหุ่นยนต์มีความเร็วในการเคลื่อนที่ 6 เมตรต่อนาที หรือ 360 เมตรต่อชั่วโมง

9) ระดับแรงดันไฟฟ้าในแบตเตอรี่ส่งผลกระทบต่อประสิทธิภาพการจ่ายกระแสไฟฟ้าให้กับ Stepper Motor และ Gear Motor หากสังเกตจากผลการทดลองจะพบว่า ในการทดลองรอบที่ 6 – 8 และรอบที่ 14 – 16 มีค่าของ Mission Duration ที่เพิ่มสูงขึ้นเป็นลำดับ เนื่องจากจากระดับแรงดันของแบตเตอรี่ลดลงจาก 11.9 V เหลือเพียง 10.1 V จึงมีความจำเป็นจะต้องหยุดการทดลองและทำการชาร์จแบตเตอรี่ใหม่ให้เต็มก่อนแล้วจึงกลับมาทำการทดลองอีกครั้ง เพื่อป้องกันแบตเตอรี่เสียหายและการได้ผลการทดลองที่คลาดเคลื่อน โดยหุ่นยนต์ต้นแบบที่ถูกพัฒนาขึ้นมี Running Time ได้สูงสุดเฉลี่ย 2 ชั่วโมง ที่ Full Load บนแบตเตอรี่ ประเภท Lithium-Ion ขนาด 30,000 mAh

## 5.2 ผลสัมฤทธิ์ทางการวิจัย

ผลสัมฤทธิ์ทางการวิจัย แบ่งออกเป็น 2 ส่วน คือ

### 1. องค์ความรู้ที่ได้จากงานวิจัยนี้

1) การพัฒนาหุ่นยนต์เพื่อนำไปใช้งานในภาคการเกษตรนั้น จะต้องมีการกำหนด Task ของการทำงานให้มีความชัดเจนก่อนการออกแบบทุกครั้ง เช่น ต้องการให้หุ่นยนต์ทำงานอะไร ทำงานที่ไหน อยู่ในสภาพแวดล้อมลักษณะใด ตลอดจนมีข้อจำกัดใดบ้างในการทำงาน เพื่อที่จะถูกนำมาใช้เป็นข้อมูลสำคัญในการกำหนดเงื่อนไข และออกแบบโครงสร้างทางกลไกให้สอดคล้องกับภาระงานที่หุ่นยนต์ได้รับมอบหมาย เพราะในภาคการเกษตรนั้น มีกิจกรรมที่หลากหลาย รวมถึงกิจกรรมเหล่านั้นก็มีรูปแบบที่แตกต่างกัน เช่น การตรวจสอบพื้นที่แปลง การหยุดเมล็ด การฉีดพ่นสารเคมี การกำจัดวัชพืช และการเก็บเกี่ยว ซึ่งรูปแบบดังกล่าวนี้จะส่งผลต่อการพัฒนากลไกและเงื่อนไขในการทำงานของหุ่นยนต์

2) รูปแบบ Platform ของหุ่นยนต์ที่แตกต่างกันนั้น สามารถนำมา Integrate เข้าด้วยกันเพื่อสร้างเป็น Platform ใหม่ ซึ่งมีการทำงานที่ยืดหยุ่นและมีสมรรถนะที่เพิ่มสูงขึ้นจากความสามารถเฉพาะตัวของหุ่นยนต์แต่ละ Platform ที่ถูกเลือกมาได้ แต่ทั้งนี้ต้องเกิดจากการเลือก Platform ของหุ่นยนต์ที่มีกลไกซึ่งสามารถทำงานร่วมกันได้เท่านั้น

3) โครงสร้างทางกลไกของหุ่นยนต์ที่ถูกพัฒนาขึ้น หากมีความยืดหยุ่นในการทำงาน รวมถึงสามารถทำงานที่มีลักษณะเป็น Multi-Tasking ได้นั้น ผู้วิจัยจะสามารถเพิ่มรูปแบบในการทำงานของหุ่นยนต์ได้ง่าย โดยเพียงแค่ทำการออกแบบและพัฒนาอุปกรณ์ส่วนปลาย (End-Effector) ใหม่เพิ่มเติมเท่านั้น

4) สภาพแวดล้อมในการทำงานมีผลกระทบต่อประสิทธิภาพ และความสำเร็จในการทำภารกิจของหุ่นยนต์ ไม่ว่าจะเป็นระดับของแสงสว่างในพื้นที่ ลักษณะของพื้นผิว ตลอดจนรูปแบบและขนาดของพื้นที่ในการทำงาน เป็นต้น

5) การนำ การประมวลผลภาพ ปัญญาประดิษฐ์ และอินเทอร์เน็ทของสรรพสิ่ง เข้ามาช่วยควบคุมการทำงานของหุ่นยนต์นั้น จะทำให้การทำงานของหุ่นยนต์มีความยืดหยุ่น สามารถตัดสินใจในการทำเงื่อนไขต่าง ๆ ได้ด้วยตัวเอง มีการทำงานที่เป็นระบบอัตโนมัติมากขึ้น แตกต่างจากรูปแบบเดิม ซึ่งมีการทำงานภายใต้เงื่อนไขที่จำกัด มีลักษณะการทำงานแบบเป็นเส้นตรง ขาดความยืดหยุ่นเมื่อต้องตัดสินใจเพื่อทำงานตามเงื่อนไขที่หุ่นยนต์พบด้วยตนเอง

## 2. การนำไปใช้ประโยชน์

ต้นแบบของหุ่นยนต์จะถูกนำมาใช้ในงานด้านการเพาะปลูกโดยการหยอดเมล็ดลงบนกระถางในแปลงปลูกภายในโรงเรือน ซึ่งเป็นงานที่สามารถใช้แรงงานหุ่นยนต์เข้ามาทดแทนได้ เพื่อลดความยุ่งยากของงานที่ทำเป็นประจำ ประหยัดเวลา ลดการใช้แรงงานมนุษย์ ตลอดจนลดความเสี่ยงในการสัมผัสกับสารเคมีหรือยาฆ่าแมลงที่เป็นอันตรายของมนุษย์ลง นอกจากนี้หุ่นยนต์ต้นแบบในงานวิจัยนี้ ยังสามารถพัฒนาต่อยอดเพื่อเพิ่มขีดความสามารถในการทำงาน เช่น การรดน้ำ การหยอดปุ๋ย หรือการฉีดพ่นสารเคมีและยาฆ่าแมลงได้อีกในอนาคต โดยไม่จำเป็นต้องพัฒนาโครงสร้างหลักของหุ่นยนต์ขึ้นใหม่ เพียงแค่ทำการออกแบบและพัฒนาอุปกรณ์ในชุดการทำงานส่วนปลาย หรือ End-Effector จากนั้นจึงนำส่วนของ Mechanic ที่พัฒนาขึ้นใหม่นี้มาเชื่อมต่อเข้ากับ Hardware และระบบ MCU ของโครงสร้างหลัก แล้วจึงทำการเขียนโค้ดในการควบคุมการทำงานต่อไป

## 5.3 ปัญหาและอุปสรรค

1. มีการใช้งบประมาณในการวิจัยค่อนข้างสูง เพราะมีการแยกพัฒนาหุ่นยนต์ออกเป็น 4 ส่วน คือ 1. Motion 2. Perception 3. Action และ 4. IoT ก่อนนำมาประกอบเพื่อทำงานร่วมกันในขั้นตอนสุดท้าย นอกจากนี้ ในการตั้งงบประมาณเพื่อใช้ในการจัดซื้อวัสดุอุปกรณ์พัฒนา Prototype ต้นแบบหุ่นยนต์ ผู้วิจัยจะต้องกั้นงบประมาณเพิ่มเติมอีกอย่างน้อยร้อยละ 30 เพื่อรับมือกับปัญหาที่เกิดขึ้นในระหว่างการดำเนินงานวิจัย หากอุปกรณ์เกิดการชำรุดเสียหาย หรือมีการปรับปรุงความสามารถเพิ่มเติมเพื่อให้หุ่นยนต์สามารถทำงานได้ตามภารกิจที่กำหนดไว้

2. ผลกระทบจากภาวะการระบาดของโรคโควิด 19 ส่งผลกระทบต่อการดำเนินงานวิจัย เกิดปัญหาในการจัดซื้อวัสดุอุปกรณ์-การขนส่งจากต่างประเทศ การประสานงาน การส่งเอกสาร รวมถึงการเข้า-ออกเพื่อขอใช้เครื่องจักรในการพัฒนาโครงสร้างของหุ่นยนต์ที่ต้นสังกัดของผู้วิจัย

3. การออกแบบในทางทฤษฎีเมื่อถูกนำมาพัฒนาเป็นหุ่นยนต์ต้นแบบจริง จะพบว่ามีรายละเอียดปลีกย่อยที่สามารถนำมาสู่ปัญหาในการปฏิบัติการกิจของหุ่นยนต์ในโลกความจริงได้ตลอดเวลา เพื่อให้การทำงานของหุ่นยนต์ถูกต้อง จึงมีการประยุกต์และปรับเปลี่ยนเทคนิคหรืออุปกรณ์อยู่เสมอ ซึ่งการกระทำดังกล่าวจะส่งผลกระทบต่องบประมาณและระยะเวลาในการวิจัยที่เพิ่มขึ้น

4. อุปกรณ์ทางกลไกหรืออิเล็กทรอนิกส์บางชิ้นมีรูปแบบการทำงานเฉพาะตัว ไม่สามารถดัดแปลงหรือประยุกต์จากชิ้นส่วนที่มีขายทั่วไปในท้องตลาดได้ จึงจะต้องออกแบบชิ้นงานด้วยซอฟต์แวร์ 3D แล้วนำมาสร้างชิ้นด้วยเครื่อง 3D Printer ซึ่งในการพิมพ์ชิ้นส่วนแต่ละครั้งใช้เวลาค่อนข้างนานกว่าที่จะได้ Prototype ออกมา และ Prototype ที่ได้จะต้องถูกนำมาทดสอบเสียก่อน หากใช้งานไม่ได้หรือมีความบกพร่องของชิ้นส่วน ก็จะต้องเริ่มขบวนการปรับแก้ไขแบบ แล้วนำมาพิมพ์ใหม่อีกครั้ง จนกว่าจะได้ Prototype ที่ทำงานได้ถูกต้อง

5. เมื่อใช้งานหุ่นยนต์ตามปกติไปในช่วงระยะเวลาหนึ่ง อุปกรณ์อาจเกิดความเสียหายในระหว่างการใช้งานได้ ซึ่งแบ่งออกเป็น 2 ส่วน คือ

อุปกรณ์ทางกลไก ความเสียหายสามารถเกิดขึ้นได้จากการทำงานตามปกติ เนื่องจากการทำงานของหุ่นยนต์ในทุกครั้งมีการเคลื่อนที่ทางกล มีผลให้เกิดการสั่นไหว ซึ่งส่งผลให้น็อตสามารถสละสล้างกันเสียดกันเหล่านั้น เกิดการคลายตัวจนหลวมหลุดได้ ความตึง-หย่อนของชุดสายพานขับเคลื่อนก็เช่นเดียวกัน เนื่องจากใช้น็อตเป็นตัวปรับความตึง ซึ่งการป้องกันปัญหาในเบื้องต้น คือการใช้น้ำยาล็อคเกลียว (Thread Locker) หยอดเพื่อถ่วงน็อตคลาย

อุปกรณ์ทางอิเล็กทรอนิกส์ ความเสียหายมักเกิดกับชุดจ่ายพลังงานไฟฟ้า ซึ่งเป็นแบตเตอรี่ประเภทลิเทียมไอออน (Lithium-Ion Battery) มีสาเหตุมาจากการใช้งานหุ่นยนต์จนมีระดับของแรงดันภายในเซลล์ของแบตเตอรี่ลดต่ำกว่า 9 V หรือใช้งานจนแบตเตอรี่หมด ซึ่งจะส่งผลให้แบตเตอรี่เสื่อมสภาพและไม่สามารถนำมาชาร์จเพื่อประจุไฟฟ้าใหม่ได้อีก ในการบำรุงดูแลรักษานั้น ควรเก็บแบตเตอรี่ให้อยู่ภายในอุณหภูมิไม่เกิน 40 องศาเซลเซียสและความชื้นที่เหมาะสม หลีกเลี่ยงการชาร์จภายใต้อุณหภูมิที่สูง ไม่ใช้งานจนมีแรงดันต่ำกว่า 9 V หรือใช้งานจนแบตเตอรี่หมด หากไม่ได้ใช้งานหุ่นยนต์เป็นระยะเวลานาน ควรทำการชาร์จแบตเตอรี่ให้มีไฟอยู่ในระดับ 50% ของความจุ เพื่อยืดอายุของแบตเตอรี่ และนำกลับมาทำการชาร์จให้แบตเตอรี่เต็มอีกครั้งก่อนการใช้งานหุ่นยนต์

ดังนั้นเพื่อให้หุ่นยนต์มีความพร้อมในการทำงานและประสิทธิภาพในการทำงานสูงสุด จึงควรวางแผนงานซ่อมบำรุงเชิงป้องกัน (Preventive Maintenance: PM) เมื่อมีการทำงาน

ตามระยะเวลา (Time Base) ครบ 1 เดือน หรือตามปริมาณการใช้งาน (Usage Base) การหยุดเมล็ด จำนวน 4,000 กระถาง หากเกณฑ์ใดถึงก่อนให้ดำเนินการบำรุงรักษาทันที ดังรายการต่อไปนี้

- 1) ตรวจสอบน็อตตามจุดเชื่อมต่อต่าง ๆ บนโครงสร้างหุ่นยนต์
- 2) ตรวจสอบชุดล้อ Mecanum และทำความสะอาดชุด Roller ของล้อ
- 3) ตรวจสอบความตึง-หย่อนของชุดขับเคลื่อนสายพาน
- 4) ทำความสะอาดและใส่ซิลิโคนหล่อลื่นหรือจารบีขาวในชิ้นส่วนที่มีการเคลื่อนไหว
- 5) ตรวจสอบชุดรางกระดูกงู (Cable Drag Chain) และสายเคเบิลที่อยู่ภายในราง
- 6) ตรวจสอบและทำความสะอาดชุดกล้อง Raspberry Pi Camera และ MCU-Cam
- 7) ตรวจสอบสายไฟ สายเคเบิล และจุดเชื่อมต่อทางวงจรไฟฟ้า
- 8) ตรวจสอบระดับแรงดันของแบตเตอรี่ Lithium-Ion ในแต่ละเซลล์
- 9) ตรวจสอบการเชื่อมต่อของระบบ GSM 3G และ MQTT Broker
- 10) ตรวจสอบการทำงานของชุดอุปกรณ์เซ็นเซอร์

#### 5.4 แนวทางในการพัฒนาต่อ

1. การพัฒนาระบบการทำงานของหุ่นยนต์ให้สามารถทำงานได้ในลักษณะของ Multi-taking จากเดิมที่ Prototype ที่เป็นต้นแบบมีความสามารถเพียงแค่การหยุดเมล็ดพันธุ์ลงบนกระถางซึ่งถูกวางอยู่ในแปลงปลูก ให้มีความสามารถที่หลากหลายเพิ่มขึ้นและสามารถทำงานได้พร้อมกันในคราวเดียว เช่น การรดน้ำ การหยุดปุ๋ย การตรวจวิเคราะห์พืช การกำจัดวัชพืช และการฉีดพ่นสารเคมีหรือยาฆ่าแมลง ซึ่งจากการที่หุ่นยนต์ต้นแบบทำงานเฉพาะบนกระถางแต่ละใบแยกกันนั้น หากเพิ่มขีดความสามารถดังกล่าวเข้าไป จะส่งผลให้เกิดความแม่นยำในการทำงาน ลดความเสี่ยงและป้องกันอันตรายจากการใช้สารเคมีที่เป็นอันตรายกับมนุษย์

2. การพัฒนาชุดหัวปลูก (Seed Injector) เพื่อให้สามารถใช้กับเมล็ดพันธุ์รูปทรงอื่น ๆ นอกเหนือจากเมล็ดพันธุ์ที่มีลักษณะเป็นทรงกลม มีแนวทางในการพัฒนา 2 รูปแบบ คือ

1) การพัฒนาชุดส่วนปลายของหัวปลูกให้มีความยืดหยุ่นด้วยวัสดุประเภทยางเทอร์โมพลาสติก โพลียูรีเทน (Thermoplastic Polyurethanes: TPU) ที่มีความยืดหยุ่นกว่าพลาสติกชนิดอื่น โดยมีลักษณะคล้ายคลึงกับซิลิโคน มีความนิ่ม ยืดหยุ่นได้ และทนแรงกระแทกได้ดี ซึ่งปกติปลายหัวปลูกที่ทำจาก TPU จะหดรัดตัวป้องกันไม่ให้เมล็ดพันธุ์สามารถไหลออกมา เมื่อถูกก้านปลูกดันออกมา ส่วนปลายของหัวปลูกก็จะขยายตัวออกพร้อมกับปล่อยให้เมล็ดสามารถไหลออกมาได้ ในขณะเดียวกันเมื่อก้านปลูกเคลื่อนที่กลับตำแหน่งเดิม ส่วนปลายหัวปลูกก็จะหดรัดตัวลงอีกครั้งเพื่อป้องกันไม่ให้เมล็ดไหลออกมา

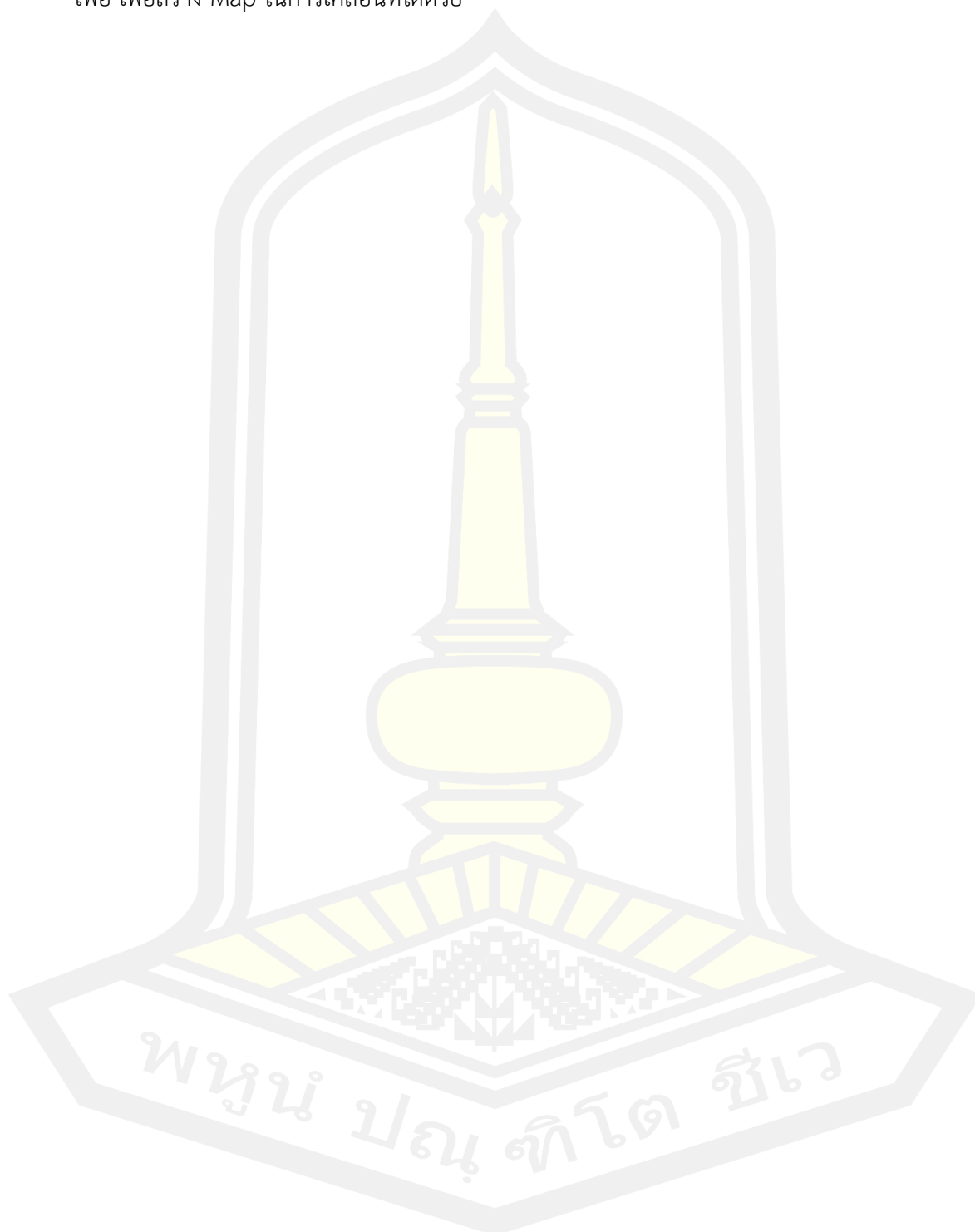
2) การพัฒนาชุดส่วนปลายของหัวปลุกโดยใช้เข็มโลหะ (Needle) ขนาด 18G (1.2 มิลลิเมตร x 25 มิลลิเมตร) ซึ่งมีรูปร่างคล้ายเข็มฉีดยาแต่มีปลายเรียบ ร่วมกับการทำงานของปั๊มสุญญากาศ (Vacuum Pump) เพื่อทำการดูดเมล็ดขึ้นมาจากถาดเก็บเมล็ด (Seeds Tray) ซึ่งจะถูกติดตั้งไว้ใกล้กับตำแหน่งจุด Home ของ CNC เมื่อหัวปลุกเคลื่อนที่ถึงตำแหน่งที่กำหนดแล้ว หัวปลุกก็จะเคลื่อนตัวลงเพื่อแทงลงไปบนดินพร้อมกับตัดการทำงานของปั๊มสุญญากาศเพื่อปล่อยให้เมล็ดตกลงในกระถาง แต่วิธีการในรูปแบบนี้จะเกิดความสูญเสียทางด้านเวลาขึ้น เนื่องจากทุกรอบของการหยอดเมล็ด หัวปลุกจะต้องเคลื่อนที่กลับไปยังถาดที่ทำการเก็บเมล็ดเสมอ เพื่อทำการดูดเมล็ดมาใช้ในการหยอดแต่ละครั้ง

3. การพัฒนาศักยภาพของระบบการเรียนรู้สภาพแวดล้อมให้กับหุ่นยนต์ โดยการทำงานของ AI ร่วมกับ Light Detection and Ranging (LiDAR) ในการตรวจจับและคาดคะเนระยะทางของวัตถุในโลกจริง (Real World) เพื่อสร้างแผนที่ในโลกเสมือน 3 มิติขึ้นในระบบเคลื่อนที่นำทาง ทำให้หุ่นยนต์มีการเคลื่อนที่และหลบหลีกสิ่งกีดขวางได้อย่างมีประสิทธิภาพมากกว่าระบบเดิมที่ใช้เพียงแค่ MCU Cam เพื่อตรวจสอบ Sign Tag แล้วตัดสินใจทำตามชุดโค้ดคำสั่งควบคุมการเคลื่อนที่ซึ่งได้กำหนดไว้แล้วเท่านั้น

4. การพัฒนาแนวทางในการประสานหุ่นยนต์หลายตัวให้เป็นระบบ (Swarm Robotics) ด้วย AI และ IoT ซึ่งเป็นวิธีการที่เน้นโครงสร้างการทำงานของหุ่นยนต์แบบเรียบง่ายหลายตัวแทนการใช้หุ่นยนต์ที่มีความซับซ้อนเพียงตัวเดียว ผูกหุ่นยนต์จะมีลักษณะคล้ายกันกับฝูงมด ไม่มีหุ่นยนต์ตัวใดในกลุ่มที่ซับซ้อนมาก แต่เมื่อรวมกันแล้วหุ่นยนต์จะสามารถทำงานที่ยากและซับซ้อนได้ โดยหุ่นยนต์ต้นแบบเดิมในการวิจัยนั้นต้องการส่วนประกอบทางฮาร์ดแวร์ที่ซับซ้อน มีประสิทธิภาพในการประมวลผลของระบบที่ค่อนข้างสูงเพื่อให้งานที่ได้รับมอบหมายสำเร็จ ในขณะที่หุ่นยนต์ที่อยู่ภายใต้ Swarm Robotics โครงสร้างค่อนข้างเรียบง่าย ใช้งบประมาณของการพัฒนาที่ต่ำ แต่มีประสิทธิภาพเมื่อร่วมมือกันทำงาน

5. การพัฒนาระบบ Station Dock โดยให้หุ่นยนต์สามารถตรวจจับสถานะของตัวเองได้ว่ามีปริมาณพลังงานของแบตเตอรี่ เมล็ดพันธุ์พืช น้ำ ปุ๋ย ยาฆ่าแมลง หรือสารเคมีที่หุ่นยนต์บรรทุกไปใช้ในการปฏิบัติการกิจนั้น มีปริมาณคงเหลือในปัจจุบันเท่าไรแบบ Real Time หากมีปริมาณที่อยู่ในระหว่างค่า Max และค่า Min ที่กำหนดเอาไว้ ระบบจะอนุญาตให้หุ่นยนต์สามารถปฏิบัติการกิจต่อได้ แต่หากพบว่าปริมาณที่เท่ากับหรือต่ำกว่าค่า Min ระบบจะสั่งให้หุ่นยนต์หยุดภารกิจ แม้จะเป็นแค่ปริมาณระดับพลังงาน เมล็ดพันธุ์ น้ำ ปุ๋ย ยาฆ่าแมลง หรือสารเคมี เพียงอย่างเดียวอย่างหนึ่งก็ตาม หุ่นยนต์จะเคลื่อนที่กลับไปยัง Dock Station เพื่อทำการ Charge แบตเตอรี่ และเติมวัสดุปลูกให้มีปริมาณที่จะพร้อมกับการปฏิบัติการกิจต่อไปเสียก่อน จึงจะอนุญาตให้หุ่นยนต์กลับไปทำงานต่อได้ ซึ่งในการทำงานรูปแบบ Dock Station นี้ หุ่นยนต์จะต้องมีความสามารถในการระบุและจดจำตำแหน่ง

ก่อนหน้าและปัจจุบันของตัวเองได้ รวมถึงหุ่นยนต์จะต้องสามารถเรียนรู้สภาพแวดล้อมการทำงาน เพื่อ เพื่อสร้าง Map ในการเคลื่อนที่ได้ด้วย



## บรรณานุกรม

- [1] เออาร์ไอพี. เกษตรกรอัจฉริยะ ภายใต้โครงการ Smart Farmer/Smart Officer. eLEADER 2556; 23[2]: 48-51.
- [2] Kaloxylos A, Eigenmann R, Teye F, ZoiPolitopoulou, Wolfert S, Shrank C, et al. Farm Management Systems and the Future Internet Era. Computers and Electronics in Agriculture 2012; 89: 130-144.
- [3] ชีรพงษ์ มังคะวัฒน์. การใช้เทคโนโลยีสารสนเทศในระบบการจัดการฟาร์ม. ส่งเสริมและนิเทศศาสตร์เกษตร 2554; 7[2]: 102-109.
- [4] Bechar A, Vigneault C. Agricultural robots for field operations: Concepts and components. Biosystems Engineering 2016; 149:94-111.
- [5] Oberti R, Marchi M, Tirelli P, Calcante A, IritiaEmanuel M, MarkoHočevar T, et al. Selective Spraying of Grapevines for Disease Control Using a Modular Agricultural Robot. Biosystems Engineering 2016; 146: 203-215.
- [6] Bloch V, Degani A, Bechar A. A methodology of orchard architecture design for an optimal harvesting robot. Biosystems Engineering 2018; 166:126-137.
- [7] Gonzalez-de-Soto M, Emmi L, Perez-Ruiz M, Aguera J, Gonzalez-de-Santos P. Autonomous systems for precise spraying – Evaluation of a robotised patch sprayer. Biosystems Engineering 2016; 146:165-182.
- [8] Ishigure Y, Hirai K, Kawasaki H. A pruning robot with a power-saving chainsaw drive. International Conference on Mechatronics and Automation; 4-7 August 2013; Takamatsu, Japan. IEEE; 1223-1228.
- [9] Drach U, Halachmi I, Pnini T, Izhaki I, Degani A. Automatic herding reduces labour and increases milking frequency in robotic milking. Biosystems Engineering 2017; 155:134-141.
- [10] Zhang C, Gao H, Zhou J, Cousins A, Pumphrey MO, Sankaran S. 3D Robotic System Development for High-throughput Crop Phenotyping. IFAC-PapersOnLine 2016; 49[16]: 242-247.
- [11] Comba L, Gay P, Aimonino DR. Robot Ensembles for Grafting Herbaceous Crops. Biosystems Engineering 2016; 146:227-239.

- [12] Ruangurai P, Ekpanyapong M, Pruetong C, Watewai T. Automated Three-wheel Rice Seeding Robot Operating in Dry Paddy Fields. Maejo International Journal of Science and Technology 2015; 9[3]: 403-412.
- [13] Smith K. How robots will feed the world. [online]. 8 August 2017 [Cited 1 September 2018]; <http://www.freshplaza.com/article/179319/How-robots-will-feed-the-world>.
- [14] Asimov I. Runaround. United States: Street & Smith; 1942.
- [15] Berry A. Robot Design Handbook. New York: R.R.Donnelley & Son Company; 1988.
- [16] พงศ์แสน พิทักษ์วัชร. พื้นฐานของหุ่นยนต์: กลศาสตร์ของหุ่นยนต์แบบอนุกรม. กรุงเทพฯ: สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย; 2557.
- [17] Joshi S. Common Robot Configurations. [online]. 26 September 2013 [cited 5 November 2018]; <https://nptel.ac.in/courses/112103174/module7/lec5/3.html>.
- [18] วโรตม ตูจันดา. การวิเคราะห์และควบคุมหุ่นยนต์อุตสาหกรรม. กรุงเทพฯ: สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย; 2559.
- [19] Kawasaki Heavy Industries. How Are Industrial Robots Built? A Guide on the Components and the Movement of Robot Arms. [online]. 22 May 2018 [cited 5 November 2018]; <https://robotics.kawasaki.com/ja1/xyz/en/1804-03/>.
- [20] Lynch K, Peshkin M. Introduction to Robotics Mechanics, Planning and Control. Illinois: Northwestern University; 2016.
- [21] Thornton J. Kinematic Chains & Robots. [online]. 2015 [cited 3 November 2018]; <https://slideplayer.com/slide/8393990/>.
- [22] Keramas J. Robot Technology Fundamentals. Delmar Publishers; 1999.
- [23] Niku S. Introduction to Robotics Analysis, Systems, Application. Prentice Hall; 2001.
- [24] Marín JAG. New Concepts in Automation and Robotic Technology for Surface Engineering. Stuttgart: University of Stuttgart; 2010.
- [25] Saha SK. Introduction to Robotics. 2nd. New Delhi: McGraw Hill; 2014.
- [26] บุญธรรม ภัทราจารกุล. หุ่นยนต์อุตสาหกรรม. กรุงเทพฯ: ซีเอ็ดยูเคชั่น; 2555.
- [27] Shamshiri R, Weltzien C, Hameed I, Yule I, Grift T, Balasundram S, et al. Research and Development in Agricultural Robotics: A Perspective of Digital Farming.



- International Journal of Agricultural and Biological Engineering 2018; 11[4]: 1-14.
- [28] Milkomax. Milking Robot Presentation and Operation. [online]. 2010 [cited 28 September 2018]; <https://milkomax.com/en/milking-robot/>.
- [29] McAndrew A. An Introduction to Digital Image Processing with Matlab. Melbourne, Australia: Victoria University of Technology; 2004.
- [30] Gonzalez RC, Woods RE. Digital Image Processing. 3rd. Canada: Pearson Prentice Hall.
- [31] Chityala R, Pudipeddi S. Image Processing and Acquisition using Python. New York: CRC Press; 2014.
- [32] ชนารักษ์ อธิษฐาน และคณะ. เทคโนโลยีสมองกลฝังตัว. กรุงเทพฯ: ส.ส.ท.; 2549.
- [33] Arduino.cc. Arduino Industrial 101. [online]. 2017 [cited 1 September 2018]; <https://store.arduino.cc/usa/arduino-industrial-101>.
- [34] อติสร เตือนตรานนท์. Trends in Sensor Technology. Horizon 2558; 5[4]: 4-10.
- [35] AliExpress. 45 IN 1 Sensors Modules. [online]. 2018 [cited 3 September 2018]; <https://www.aliexpress.com/item/45-in-1-Sensors-Modules-Starter-Kit-For-arduino-better-than-37in1-sensor-kit-37-in/32818304151.html>.
- [36] Rouse M. Internet of Things (IoT) definition. [online]. 2013 [Cited 8 September 2018]; <http://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT>.
- [37] NIIT Technologies. Internet of Things. [online]. 2017 [cited 3 September 2018]; <https://www.niit-tech.com/offering/digital-services/emerging-technology/offering/internet-things>.
- [38] OASIS. MQTT Version 5.0. [online]. 2019 [cited 3 May 2020]; <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html>.
- [39] MQTT.org. MQTT: The Standard for IoT Messaging. [online]. 2020 [cited 3 May 2020]; <https://mqtt.org/getting-started/>.
- [40] Jacoff A, Messina E, Evans J. Performance evaluation of autonomous mobile robots. Industrial Robot: An International Journal 2002; 29[3]:
- [41] Lampe A, Chatila R. Performance Measure For The Evaluation of Mobile Robot Autonomy. 2006 IEEE International Conference on Robotics and Automation; May 2006; Orlando, Florida. 4057-4062.

- [42] Kanjilal D, Singh D, Reddy R, Mathew PJ. Smart Farm: Extending Automation to The Farm Level. *International Journal of Scientific & Technology Research* 2014; 3[7]:
- [43] Roham VS, Pawar GA, Patil AS, Rupnar PR. Smart Farm using Wireless Sensor Network. *International Journal of Computer Applications* 2015;
- [44] Boonchieng E, Chieochan O, Saokaew A. Smart Farm: Applying the Use of NodeMCU, IOT, NETPIE and LINE API for a Lingzhi Mushroom Farm in Thailand. *IEICE Transactions on Communications* 2018; E101B[1]: 16-23.
- [45] Ko MH, Ryuh BS, Kim KC, Suprem A, Mahalik NP. Autonomous Greenhouse Mobile Robot Driving Strategies From System Integration Perspective: Review and Application. *Ieee-Asme Transactions on Mechatronics* 2015; 20[4]: 1705-1716.
- [46] Naik NS, Shete VV, Danve SR. Precision Agriculture Robot for Seeding Function. *International Conference on Inventive Computation Technologies (ICICT)*; 26-27 August 2016; Coimbatore, India.
- [47] Belforte G, Deboli R, Gay P, Piccarolo P, Aimonino DR. Robot design and testing for greenhouse applications. *Biosystems Engineering* 2006; 95[3]: 309-321.
- [48] Batista AVdA, Albiero D, Viana TVdA, Monteiro LdA, Chioderoli CA, Sousa IRSd, et al. Multifunctional Robot at low cost for small farms. *Ciencia Rural* 2017; 47[7]:
- [49] Aronson RL. The FarmBot Whitepaper. [Online]. 2019 [cited 13 May 2019]; <https://farm.bot/pages/whitepaper>.
- [50] FarmBot. FarmBot Genesis Prototype. [Online]. 2019 [cited 13 May 2019]; <https://farm.bot/>.
- [51] FarmBot. FarmBot Software. [Online]. 2019 [cited 13 May 2019]; <https://software.farm.bot/docs>.
- [52] Athukorala A, Ranasinghe N, Herath K, Jayasekara P. Scalable Autonomous AgronomicalSmartbot. *14th IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications (MESA)*; 2-4 July 2018; Oulu, Finland. 1-6.
- [53] Hemming J, Bac W, Tuijl Bv, Barth R, Bontsema J, Pekkeriet E, et al. A robot for harvesting sweet-pepper in greenhouses. *International Conference of*

Agricultural Engineering; 6-10 July 2014; Zurich.

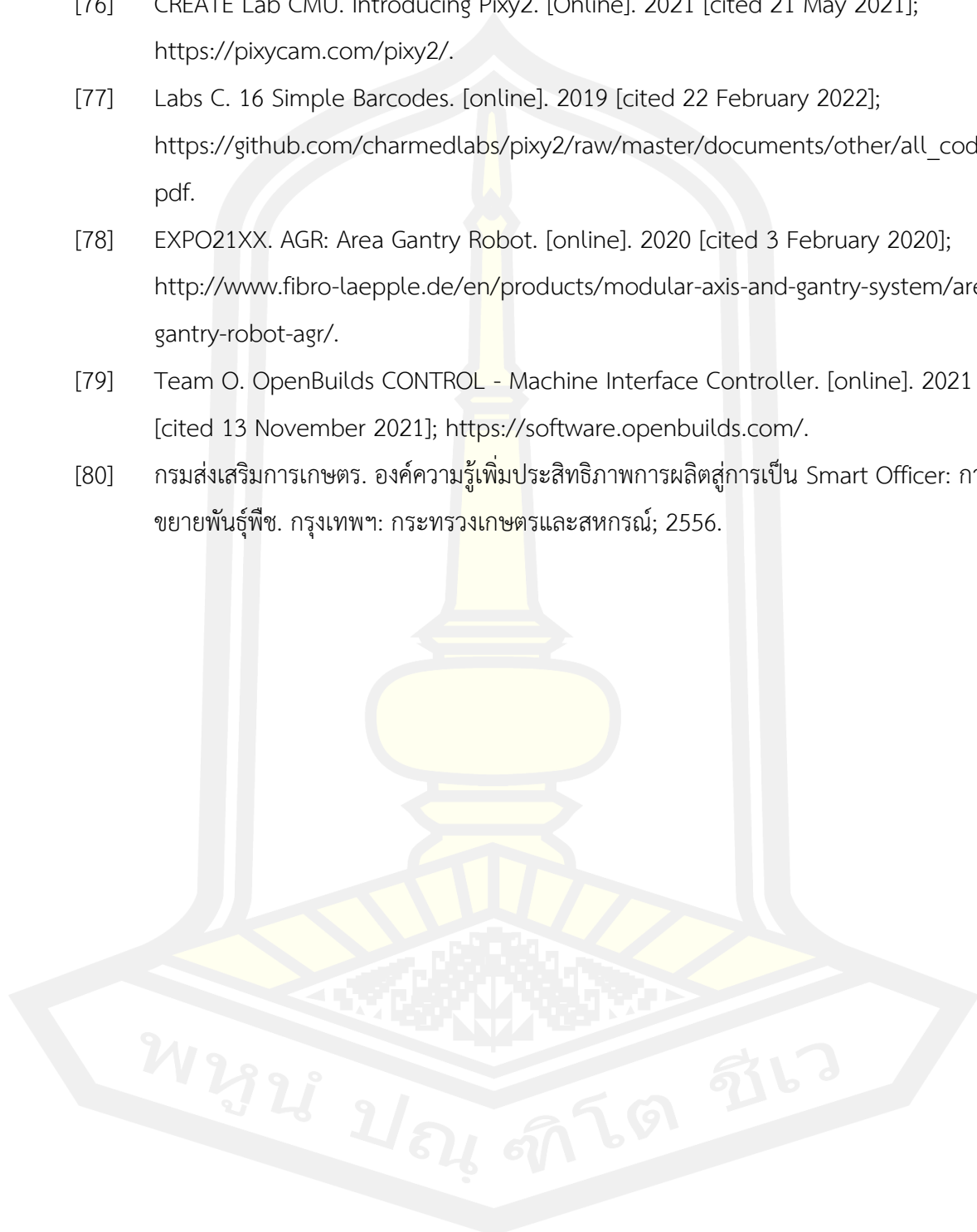
- [54] Ishak W, Ismail W. Research and Development of Oil Palm Harvester Robot at Universiti Putra Malaysia. *International Journal of Engineering and Technology* 2010; 7[2]: 87-94.
- [55] Stein M, Bargoti S, Underwood J. Image Based Mango Fruit Detection, Localisation and Yield Estimation Using Multiple View Geometry. *Sensors* 2016; 16[11]:
- [56] Van Henten E J, Hemming J, Van Tuijl B A J, Kornet J G, Meuleman J, Bontsema J, et al. An Autonomous Robot for Harvesting Cucumbers in Greenhouses. *Autonomous Robots* 2002; 13[3]: 241–258.
- [57] Underwood JP, Hung C, Whelan B, Sukkarieh S. Mapping almond orchard canopy volume, flowers, fruit and yield using lidar and vision sensors. *Computers and Electronics in Agriculture* 2016; 130:83-96.
- [58] Bargoti S, Underwood JP. Image Segmentation for Fruit Detection and Yield Estimation in Apple Orchards. *Journal of Field Robotics* 2017; 34[6]: 1039–1060.
- [59] Qingchun F, Xiu W, Wengang Z, Quan Q, Kai J. A new strawberry harvesting robot for elevated-trough culture. *International Journal of Agricultural and Biological Engineering* 2012; 5[2]: 1-8.
- [60] Tanigaki K, Fujiura T, Akase A, Imagawa J. Cherry-harvesting robot. *Computers and Electronics in Agriculture* 2008; 63[1]: 65-72.
- [61] Mehta S S, MacKunis W, Burks T F. Robust visual servo control in the presence of fruit motion for robotic citrus harvesting. *Computers and Electronics in Agriculture* 2016; 123:362-375.
- [62] Nuske S, Wilshusen K, Achar S, Yoder L, Narasimhan S, Singh S. Automated Visual Yield Estimation in Vineyards. *Journal of Field Robotics* 2014; 31[5]: 837–860.
- [63] Lili W, Bo Z, Jinwei F, Xiaohan H, Shu W, Yashuo L, et al. Development of a tomato harvesting robot used in greenhouse. *International Journal of Agricultural and Biological Engineering* 2017; 10[4]: 140-149.
- [64] Hung C, Underwood J, Nieto J, Sukkarieh S. A Feature Learning Based Approach for Automated Fruit Yield Estimation. *Field and Service Robotics* 2015; 105:485-

498.

- [65] Nan L, Chunlong Z, Ziwen C, Zenghong M, Zhe S, Ting Y, et al. Crop positioning for robotic intra-row weeding based on machine vision. *International Journal of Agricultural and Biological Engineering* 2015; 8[6]: 20-29.
- [66] Hu J, Yan X, Ma J, Qi C, Francis K, Mao H. Dimensional synthesis and kinematics simulation of a high-speed plug seedling transplanting robot. *Computers and Electronics in Agriculture* 2014; 107:64-72.
- [67] Huang Y-J, Lee F-F. An automatic machine vision-guided grasping system for Phalaenopsis tissue culture plantlets. *Computers and Electronics in Agriculture* 2010; 70[1]: 42-51.
- [68] Shamshiri R, Ismail WIW. Design and Simulation of Control Systems for a Field Survey Mobile Robot Platform. *Engineering and Technology* 2013; 6[13]: 2307-2315.
- [69] Kelly A. *Mobile Robotics: Mathematics, Models, and Methods*. 1st. New York: Cambridge University Press; 2013.
- [70] Cook G. *Mobile Robots: Navigation, Control and Remote Sensing*. 1st. New Jersey: Wiley; 2011.
- [71] Rubio F, Valero F, Llopis-Albert C. A Review of Mobile Robots: Concepts, Methods, Theoretical Framework and Applications. *International Journal of Advanced Robotic Systems* 2019; 1-22.
- [72] Salih JEM, Rizon MJM, Yaacob S, Adom A. Designing omni-directional mobile robot with mecanum wheel. *American Journal of Applied Sciences* 2006; 3[5]: 1831-1835.
- [73] Circuits M. H-Bridges – the Basics. [online]. July 2019 [cited 1 November 2019]; <https://www.modularcircuits.com/blog/articles/h-bridge-secrets/h-bridges-the-basics/>.
- [74] Seeed Technology. 4WD Mecanum Wheel Robot Kit Series. [Online]. 2017 [cited 10 September 2019]; [http://wiki.seeedstudio.com/4WD\\_Mecanum\\_Wheel\\_Robot\\_Kit\\_Series](http://wiki.seeedstudio.com/4WD_Mecanum_Wheel_Robot_Kit_Series).
- [75] Adascalitei F, Doroftei I. Practical Applications for Mobile Robots Based on Mecanum Wheels - a Systematic Survey. *Romanian Review Precision Mechanics*,

Optics and Mechatronics 2011; 21-29.

- [76] CREATE Lab CMU. Introducing Pixy2. [Online]. 2021 [cited 21 May 2021]; <https://pixycam.com/pixy2/>.
- [77] Labs C. 16 Simple Barcodes. [online]. 2019 [cited 22 February 2022]; [https://github.com/charmedlabs/pixy2/raw/master/documents/other/all\\_codes.pdf](https://github.com/charmedlabs/pixy2/raw/master/documents/other/all_codes.pdf).
- [78] EXPO21XX. AGR: Area Gantry Robot. [online]. 2020 [cited 3 February 2020]; <http://www.fibro-laepple.de/en/products/modular-axis-and-gantry-system/area-gantry-robot-agr/>.
- [79] Team O. OpenBuilds CONTROL - Machine Interface Controller. [online]. 2021 [cited 13 November 2021]; <https://software.openbuilds.com/>.
- [80] กรมส่งเสริมการเกษตร. องค์ความรู้เพิ่มประสิทธิภาพการผลิตสู่การเป็น Smart Officer: การขยายพันธุ์พืช. กรุงเทพฯ: กระทรวงเกษตรและสหกรณ์; 2556.





## ประวัติผู้เขียน

ชื่อ	นายอภิรักษ์ ทูลธรรม
วันเกิด	6 สิงหาคม 2521
สถานที่เกิด	อ.เมือง จ.ยโสธร
สถานที่อยู่ปัจจุบัน	126 หมู่ 1 สนามม้าซอย 1 ต.ขวัญเมือง อ.เสลภูมิ จ.ร้อยเอ็ด 45120
ตำแหน่งหน้าที่การงาน	ผู้ช่วยผู้อำนวยการฝ่ายพัฒนาหลักสูตร
สถานที่ทำงานปัจจุบัน	วิทยาลัยเทคโนโลยีอุตสาหกรรมศรีสงคราม มหาวิทยาลัยนครพนม
ประวัติการศึกษา	พ.ศ. 2546 วศ.บ. วิศวกรรมไฟฟ้า มหาวิทยาลัยขอนแก่น พ.ศ. 2556 วท.ม. วิทยาการคอมพิวเตอร์ มหาวิทยาลัยมหาสารคาม พ.ศ. 2559 วท.บ. เทคโนโลยีสารสนเทศ มหาวิทยาลัยนครพนม พ.ศ. 2565 ปริญญาโท วิทยาการคอมพิวเตอร์ มหาวิทยาลัยมหาสารคาม
ทุนวิจัย	Problem Evaluation & Solution to Protect Internet Banking Solution to Protect Internet Banking สำนักงานคณะกรรมการวิจัยแห่งชาติ ปีงบประมาณ 2557 การสร้างมูลค่าเพิ่มและการส่งเสริมช่องทางการตลาดของของผลิตภัณฑ์ปลาสามบ้านท่าบ่อ สำนักงานคณะกรรมการการอุดมศึกษา ปีงบประมาณ 2560 การพัฒนาระบบจัดการฟาร์มมะเดื่อฝรั่งอัจฉริยะในเชิงพาณิชย์ สำนักงานคณะกรรมการวิจัยแห่งชาติ ปีงบประมาณ 2561 อินเทอร์เน็ตของสรรพสิ่งสำหรับหุ่นยนต์ปัญญาประดิษฐ์เพื่อสนับสนุนการเพาะปลูกอัตโนมัติ สำนักงานคณะกรรมการวิจัยแห่งชาติ ปีงบประมาณ 2564
ผลงานวิจัย	อภิรักษ์ ทูลธรรม และ สมนึก พ่วงพรพิทักษ์, “การทดลองเพื่อประเมินปัญหาการโจมตีด้วยการเปลี่ยเอสเอสแอล”, การประชุมวิชาการทางคอมพิวเตอร์และเทคโนโลยีสารสนเทศ (CIT 2013); 24 มกราคม 2556; สกลนคร, ไทย. อภิรักษ์ ทูลธรรม และ สมนึก พ่วงพรพิทักษ์, “Click2Enforce: โปรแกรมส่วนขยายของเบราว์เซอร์เพื่อป้องกันการโจมตีจากการเปลี่ยเอสเอส

เอสแอล”, การประชุมทางวิชาการระดับชาติด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ครั้งที่ 9 (NCCIT 2013); 9–10 พฤษภาคม 2556; กรุงเทพฯ, ไทย.

อภิรักษ์ ทูลธรรม และ สมนึก พ่วงพรพิทักษ์, “Click2Enforce: โปรแกรมส่วนขยายของเบราว์เซอร์เพื่อป้องกันการโจมตีจากการเปลี่ยเอสเอสแอล”, วารสารเทคโนโลยีสารสนเทศ, กรกฎาคม–ธันวาคม 2556, Vol. 9 No. 2, ISBN: 1685-8573.

สมนึก พ่วงพรพิทักษ์ และ อภิรักษ์ ทูลธรรม, “การประเมินวิธีแก้ไขปัญหาการโจมตีด้วยการเปลี่ยเอสเอสแอล”, วารสารเทคโนโลยีสารสนเทศ, มกราคม–มิถุนายน 2557, Vol. 10 No. 1, ISBN: 1685-8573.

อภิรักษ์ ทูลธรรม และ ภูมินทร์ ชงมา, “Blows2Safe: เครื่องช่วยเซ่นเซอร์ไร้สายสำหรับการตรวจหาแอลกอฮอล์ในลมหายใจเพื่อป้องกันผลกระทบจากแอลกอฮอล์ต่อการขับชี่ยานพาหนะ”, การประชุมทางวิชาการระดับชาติด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ครั้งที่ 11 (NCCIT 2015); 2–3 กรกฎาคม 2558; กรุงเทพฯ, ไทย.

อภิรักษ์ ทูลธรรม และ อุมาพร จันโสภา, ความพึงพอใจระบบบริการการจัดการเรียนการสอนแบบออนไลน์มูเดิ้ลและกูเกิ้ลคลาสรูมในบทบาทของผู้สอน, วารสารมนุษยศาสตร์และสังคมศาสตร์ มหาวิทยาลัยมหาสารคาม; ฉบับพิเศษ 2559. หน้า. 84-91. ISSN 0859-5992.

อุมาพร จันโสภา และ อภิรักษ์ ทูลธรรม, การประยุกต์ใช้เครื่องช่วยสังคมเพสบูคเพื่อเพิ่มประสิทธิภาพของระบบการจัดการเรียนออนไลน์มูเดิ้ล, วารสารวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยมหาสารคาม; ฉบับพิเศษ 2559. หน้า 518-524. ISSN 686-9664.

อภิรักษ์ ทูลธรรม และ อุมาพร จันโสภา, การประยุกต์ใช้การเรียนรู้แบบผสมผสานเพื่อเพิ่มผลสัมฤทธิ์ทางการเรียนในรายวิชาการสื่อสารข้อมูลและเครือข่ายคอมพิวเตอร์, การประชุมทางวิชาการระดับชาติด้านคอมพิวเตอร์และเทคโนโลยีสารสนเทศ ครั้งที่ 12 (NCCIT 2016); 7–8 กรกฎาคม 2559; หน้า 139-144 .ขอนแก่น, ไทย.

อภิรักษ์ ทูลธรรม, ชารินี พรหมภักดี และ อนุชาวดี ไชยทองศรี, “RMUTI-Smart Login: แนวคิดระบบพิสูจน์ตัวตนเพื่อป้องกันการโจมตีจากการเปลี่ยเอสเอสแอล”, การประชุมวิชาการและนำเสนอผลงานวิจัย



ระดับชาติ ครั้งที่ ๒ และสัมมนาวิชาการ ราชวมงคลสุรินทรวิชาการ ครั้งที่ ๖; สุรินทร์, ไทย.

ชารินทร์ พรหมภักดี, อภิรักษ์ ทูลธรรม และ อนุชาวดี ไชยทองศรี, “การเพิ่มประสิทธิภาพการเรียนการสอนในรายวิชาทฤษฎีทางด้านคอมพิวเตอร์ โดยให้ผู้เรียนมีส่วนร่วมในการเรียนการสอน กรณีศึกษาวิชาโครงสร้างข้อมูลและอัลกอริทึม”, การประชุมวิชาการและนำเสนอผลงานวิจัยระดับชาติ ครั้งที่ ๒ และสัมมนาวิชาการ ราชวมงคลสุรินทรวิชาการ ครั้งที่ ๖; สุรินทร์, ไทย.

อนุชาวดี ไชยทองศรี, อภิรักษ์ ทูลธรรม และ ชารินทร์ พรหมภักดี, “กระบวนการสร้างความสัมพันธ์การใช้งานฐานข้อมูลออนไลน์ด้วยวิธีการออกแบบการพัฒนาเชิงวัตถุ กรณีศึกษา มหาวิทยาลัยเทคโนโลยีราชมงคลวิทยาเขตสกลนคร”, การประชุมวิชาการและนำเสนอผลงานวิจัยระดับชาติ ครั้งที่ ๒ และสัมมนาวิชาการ ราชวมงคลสุรินทรวิชาการ ครั้งที่ ๖; สุรินทร์, ไทย.

Apirak Tooltham, “The Hidden Threat Behind Internet of Things”, In Proceedings of The 6th Rajamangala University of Technology International Conference; 1–3 September 2015; Nakhonratchasima, Thailand.

Poramin Bheganon, Patcharee Sriputta, Boonchoo Srikudkao, Natthakit Srikan-janapert, Jantima Polpinij and Apirak Tooltham, “Automatic Legal Cases Assembling by the Spherical k-mean Clustering”, In Proceedings of The 24th International Symposium on Artificial Life and Robotics: AROB 2019; 23 – 25 January 2019; Beppu, Japan.