

## ออกแบบระบบควบคุมรถติดตามแบบ AGV



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาวิศวกรรมเมคคาทรอนิกส์  
มหาวิทยาลัยเทคโนโลยีสุรนารี  
ปีการศึกษา 2566

# DESIGN CONTROL SYSTEM OF THE FOLLOWER AGV



A Thesis Submitted in Partial Fulfillment of the Requirements for the  
Degree of Master of Engineering in Mechatronics Engineering  
Suranaree University of Technology  
Academic Year 2023

## ออกแบบระบบควบคุมรถติดตามแบบ AGV

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นักศึกษานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษา  
ตามหลักสูตรปริญญาโทบริหารธุรกิจ

คณะกรรมการสอบวิทยานิพนธ์



(ผศ. ดร.ไตรฎา แข็งการ)  
ประธานกรรมการ



(รศ. ดร.จิระพล ศรีเสวีรัฐผล)  
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)



(ผศ. ดร.ชัยยุทธ์ สัมภาวะคุปต์)  
กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี



(รศ. ดร.ยุพาพร รักสกุลพิวัฒน์)  
รองอธิการบดีฝ่ายวิชาการและประกันคุณภาพ



(รศ. ดร.พรศิริ จงกล)  
คณบดีสำนักวิชาวิศวกรรมศาสตร์

เจษฎากร เหมือนงาม : ออกแบบระบบควบคุมรถติดตามแบบ AGV (DESIGN CONTROL SYSTEM OF THE FOLLOWER AGV)

อาจารย์ที่ปรึกษา : รองศาสตราจารย์ ดร.จิระพล ศรีเสรีภูมิผล, 92 หน้า.

คำสำคัญ: AGV/Image processing/การวัดระยะด้วยกล้อง

คลังสินค้าถือเป็นอีกหนึ่งพื้นที่ที่มีการพัฒนาอย่างรวดเร็ว เนื่องมาจากธุรกิจ E-Commerce มีความต้องการสูงในปัจจุบัน มีสาเหตุมาจากผู้บริโภคสั่งซื้อสินค้าผ่านช่องทางออนไลน์มากขึ้น จากการเก็บข้อมูล พบว่า มีการเติบโต 3 ปีซ้อนหลังตั้งแต่ปี 2563 - 2566 เฉลี่ย 26% ต่อปี ข้อมูลอ้างอิงจากศูนย์วิจัยกสิกรไทย ทำให้การบริหารจัดการคลังสินค้ามีความซับซ้อนมาก จึงจำเป็นต้องใช้เทคโนโลยีช่วยบริหารจัดการการทำงาน ซึ่งเทคโนโลยีที่ใช้กันอย่างแพร่หลายในอุตสาหกรรมโลจิสติกส์ ก็คือหุ่นยนต์เคลื่อนที่อัตโนมัติ AGV (Automated Guided Vehicle) เป็นยานพาหนะหรือที่เรียกง่าย ๆ ว่าเป็นรถที่ใช้ขนส่งภายในและภายนอกอาคารแบบไร้คนขับ โดยใช้หลักการการเคลื่อนที่ให้อยู่ในเส้นทาง โดยไม่ชนสิ่งกีดขวางด้วยการเคลื่อนที่ตามเส้น เลเซอร์ หรือใช้กล้องเป็นตัวบอกทิศทางการเคลื่อนที่ ช่วยขนย้ายสินค้า อุปกรณ์ที่มีน้ำหนักมากได้อย่างแม่นยำ แต่ก็มีข้อเสียที่ราคาค่อนข้างสูง เนื่องด้วยมีจำนวนเซ็นเซอร์หลายตัว ประกอบกับระบบนำทาง ผู้ทำวิจัยจึงสนใจที่จะออกแบบระบบควบคุมที่ใช้กับหุ่นยนต์ AGV ตัวตาม ซึ่งมีหน้าที่เคลื่อนที่ติดตาม AGV ตัวนำที่ถูกรับคำสั่งเส้นทางเคลื่อนที่มาจากผู้ใช้ โดยใช้กล้องและเซ็นเซอร์วัดระยะเป็นตัวตรวจรู้หลัก เพื่อลดจำนวนเซ็นเซอร์ ซึ่งเป็นเหตุทำให้ราคาต้นทุนของ AGV ถูกลง ซึ่งกล้องมีหน้าที่เป็นตัวตรวจรู้เพื่อตรวจจับสัญลักษณ์ QR code ที่มีข้อมูลเฉพาะเจาะจง ที่ติดอยู่กับ AGV ตัวนำ ซึ่งมีระยะการตรวจจับไม่เกิน 150 เซนติเมตร จากการทดสอบ เพื่อนำข้อมูลการติดตามไปสั่งการการบังคับเลี้ยวของชุดล้อหน้า และเซ็นเซอร์วัดระยะมีหน้าที่วัดระยะเพื่อรักษาระยะห่างระหว่าง AGV ตัวนำและตัวตาม ซึ่งสามารถรักษาระยะได้ตั้งแต่ 30 เซนติเมตร ถึง 150 เซนติเมตร โดยที่ AGV ตัวนำต้องมีความเร็วในการเคลื่อนที่ไม่เกิน 1 เมตรต่อวินาที และใช้เซ็นเซอร์อินฟราเรดในการส่งสัญญาณเข้าจอดในสถานีเมื่อเสร็จสิ้นภารกิจติดตาม ซึ่งสถานีจอดจะต้องมีค่าความเผื่อบวกลดด้านซ้ายและขวา 30 เซนติเมตรเมื่ออ้างอิงจากตัวรถ AGV ตัวตาม

สาขาวิชาวิศวกรรมเมคคาทรอนิกส์

ปีการศึกษา 2566

ลายมือชื่อนักศึกษา.....โศภฎากร.....

ลายมือชื่ออาจารย์ที่ปรึกษา.....[ลายเซ็น].....

CHETSADAKORN MUEANNGAM : DESIGN CONTROL SYSTEM OF THE FOLLOWER AGV. THESIS ADVISOR : ASSOC. PROF. JIRAPHON SRISERTPOL, Ph.D., 92 PP.

Keywords: AGV/Image Processing/Measure Distance By Webcam

A warehouse is considered another rapidly developing area due to the high demand from E-Commerce businesses today. This demand arises from consumers increasingly ordering products through online channels. Data collection shows that there has been an average annual growth of 26% over the past three years from 2020 to 2023, according to research from the Kasikorn Research Center. This makes warehouse management more complex, necessitating the use of technology to aid in operations. One widely used technology in the logistics industry is the Automated Guided Vehicle (AGV), AGVs navigate along predefined paths without colliding with obstacles, using methods such as following lines, lasers, or camera guidance. They accurately move heavy goods and equipment but have the drawback of being quite expensive due to the numerous sensors and navigation systems involved. Researchers are interested in designing a control system for a follower AGV, which moves by following a leader AGV that receives movement instructions from the user. The system uses a camera and distance sensors as primary detection methods to reduce the number of sensors and thus lower the cost of AGVs. The camera's role is to detect specific QR code symbols attached to the leader AGV, which can be detected from up to 150 centimeters away. The distance sensor maintains a distance of 30 to 150 centimeters between the leader and follower AGVs. The leader AGV must not exceed a speed of 1 meter per second and uses infrared sensors to signal when to dock at a station upon completing its tracking mission. The docking station should have a tolerance of plus or minus 30 centimeters on both the left and right sides, relative to the follower AGV.

School of Mechatronics Engineering  
Academic Year 2023

Student's Signature.....

Advisor's Signature.....

Chetsadakorn  
Srisertpol

## กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณ บุคคล และกลุ่มบุคคลต่าง ๆ ที่ได้กรุณาให้คำปรึกษา แนะนำและช่วยเหลือทั้งในด้านวิชาการและการดำเนินงานในงานวิจัย

ขอขอบคุณ รองศาสตราจารย์ ดร.จิระพล ศรีเสรีภูผล อาจารย์ที่ปรึกษางานวิจัย ที่กรุณาให้คำปรึกษาด้านวิชาความรู้ ทั้งทฤษฎีที่เกี่ยวข้องกับงานวิจัย และด้านปฏิบัติการ แนะนำ ชี้แนะให้เห็นถึงข้อบกพร่องและแนวทางการดำเนินการวิจัยที่ถูกต้อง

ขอขอบพระคุณ ประธานกรรมการสอบวิทยานิพนธ์รวมถึงคณะกรรมการสอบวิทยานิพนธ์ ได้แก่ ผู้ช่วยศาสตราจารย์ ดร.ไตรฎา แข็งการ และ ผู้ช่วยศาสตราจารย์ ดร.ชัยยุทธ์ สัมภาวะคุปต์ ที่ตรวจสอบความถูกต้องและครบถ้วนของวิทยานิพนธ์ และแนวทางการปรับปรุงวิทยานิพนธ์ซึ่งทำให้วิทยานิพนธ์นี้มีเนื้อหาที่สมบูรณ์และมีความหนักแน่นทางวิชาการ

สุดท้ายนี้ผู้วิจัยหวังว่า งานวิจัยนี้จะมีประโยชน์ไม่มากนักน้อย จึงขอยกส่วนดีงามทั้งหมดนี้ให้แก่ อาจารย์ที่ปรึกษาวิทยานิพนธ์ คณะกรรมการสอบวิทยานิพนธ์และเหล่าคณาจารย์ที่ได้มอบความรู้และประสบการณ์แก่ผู้วิจัยตลอดระยะเวลาที่ศึกษาในระดับบัณฑิตศึกษา และขอกราบขอบพระคุณ บิดา มารดา ที่ให้การเลี้ยงดูอบรมและให้การสนับสนุนแก่ผู้วิจัยตลอดมา

เจษฎากร เหมือนงาม

มหาวิทยาลัยเทคโนโลยีสุรนารี

# สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย) .....	ก
บทคัดย่อ (ภาษาอังกฤษ) .....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ญ
คำอธิบายสัญลักษณ์และคำย่อ .....	ฐ
<b>บทที่</b>	
<b>1 บทนำ.....</b>	<b>1</b>
1.1 ที่มาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย.....	3
1.3 ขอบเขตการวิจัย.....	3
1.4 ระเบียบวิธีวิจัย .....	3
1.5 สถานที่ดำเนินงานวิจัย.....	4
1.6 เครื่องมือที่ใช้ในการทำวิจัย.....	4
1.7 ประโยชน์ที่คาดว่าจะได้รับ .....	4
<b>2 ปรัชญาวรรณกรรมและงานวิจัยที่เกี่ยวข้อง.....</b>	<b>5</b>
2.1 ความหมายของ AGV (AUTOMATED GUIDED VEHICLE).....	5
2.1.1 ระบบการนำทางของ AGV.....	5
2.1.1.1 แลบแม่เหล็ก.....	5
2.1.1.2 สายไฟ .....	6
2.1.1.3 เลเซอร์ AGV.....	6
2.1.1.4 วิทยุทัศน์ AGV .....	7
2.1.1.5 RFID .....	7

## สารบัญ (ต่อ)

### หน้า

2.1.2	ประเภทของการขับเคลื่อน AGV.....	8
2.1.2.1	ระบบบังคับเลี้ยวแบบเฟืองต่าง (Differential Steering).....	8
2.1.2.2	ระบบบังคับเลี้ยวแบบใช้ล้อหมุน (Single or Dual Steering).....	8
2.1.2.3	ระบบบังคับเลี้ยวรอบทิศทาง (Omnidirectional Steering).....	9
2.1.2.4	ระบบบังคับเลี้ยวแบบหมุน (Swivel or Caster Steering).....	9
2.1.2.5	ระบบบังคับเลี้ยวแบบอัคร์แมน (Ackermann Steering).....	9
2.1.2.6	ระบบบังคับเลี้ยวแบบซิงโครนัส (Synchronous Steering).....	9
2.3	อัลกอริทึม PURE PURSUIT.....	9
2.3	หลักการทํางานของ QR CODE.....	15
2.4	หลักการทํางานของ ULTRASONIC SENSOR.....	18
2.5	การประมวลผลภาพ.....	21
2.5.1	ทฤษฎีสี.....	21
2.5.1.1	โมเดลสี (Color Models).....	21
2.5.1.2	การแก้ไขและปรับแต่งสี (Color Correction and Enhancement).....	23
2.5.1.3	การแบ่งส่วนตามสีและการตรวจจับวัตถุ (Color Segmentation and Object Detection).....	23
2.5.2	ประเภทของภาพตามความละเอียดของสี (Color Depth).....	24
2.5.2.1	ภาพขาวดำ (Binary Images) (1 บิต).....	24
2.5.2.2	ภาพขาวเทา grayscale (Grayscale Images) (8 บิต).....	24
2.5.2.3	ภาพสี (Color Images).....	24
2.5.3	การปรับแต่งคุณภาพของภาพ.....	24
2.5.3.1	ปรับแต่งค่าฮิสโตแกรม (Histogram Equalization).....	24
2.5.3.2	การกรองภาพ (Filtering).....	25
2.5.3.3	การตรวจจับขอบภาพ (Edge Detection).....	25
2.5.3.4	การปรับแต่งความคมชัด (Contrast Adjustment).....	25
2.6	ปริทัศน์วรรณกรรมที่เกี่ยวข้อง.....	25



## สารบัญ (ต่อ)

	หน้า
2.6.1 งานวิจัยที่เกี่ยวข้องกับการระบบควบคุม AGV.....	25
2.6.2 งานวิจัยที่เกี่ยวข้องกับการตรวจจับวัตถุ.....	27
2.7 สรุป.....	28
<b>3 วิธีดำเนินการวิจัย .....</b>	<b>30</b>
3.1 บทนำ.....	30
3.2 ชุดทดสอบการตรวจจับสัญลักษณ์ .....	31
3.2.1 กล้อง intel realsense depth camera d455 .....	31
3.2.2 สัญลักษณ์ .....	31
3.2.2.1 QR code .....	31
3.2.2.2 ภาพสีฟ้า.....	32
3.2.3 mini pc Dell Optiplex 7060 Micro.....	32
3.3 ชุดควบคุมสั่งงานการเคลื่อนที่ของ AGV ตัวตาม .....	33
3.3.1 บอร์ด Arduino Uno .....	33
3.3.2 บอร์ด Arduino Mega .....	33
3.3.3 เซนเซอร์.....	34
3.3.3.1 Ultrasonic sensor.....	34
3.3.3.2 Max sona.....	35
3.3.3.3 Infrared Photoelectric Sensors .....	35
3.4 ออกแบบโปรแกรมวัดระยะห่างและมุม .....	36
3.4.1 การวัดระยะห่างระหว่างวัตถุกับกล้อง .....	36
3.4.2 การวัดมุมระหว่างวัตถุกับกล้อง.....	40
3.5 ออกแบบการควบคุมสั่งงานการเคลื่อนที่และบังคับเลี้ยว .....	42
3.5.1 การควบคุมระยะห่างการเคลื่อนที่ระหว่างตัวนำและตัวตาม .....	42
3.5.2 การควบคุมการบังคับเลี้ยว.....	43
3.5 ออกแบบโปรแกรมการจอดและออกจากสถานี.....	45
3.6 การทดลอง.....	46
3.6.1 ทดสอบการตรวจจับสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงต่างกัน .....	46

## สารบัญ (ต่อ)

หน้า

3.6.2	ทดสอบการวัดระยะของสัญลักษณ์และเซนเซอร์.....	47
3.6.3	ทดสอบความแม่นยำของการจอดและออกจากสถานีเริ่มต้น .....	47
3.6.4	ทดสอบการเคลื่อนที่ติดตามทางตรง .....	48
3.7	สรุป.....	48
<b>4</b>	<b>ผลการวิจัยและอภิปรายผล.....</b>	<b>50</b>
4.1	บทนำ.....	50
4.2	เปรียบเทียบการตรวจจับสัญลักษณ์.....	50
4.3	เปรียบเทียบการวัดระยะห่างกล้องถึงสัญลักษณ์.....	51
4.3.1	วัดระยะห่างจากกล้องถึงสัญลักษณ์ (QRcode) ด้วยกล้อง.....	51
4.3.1.1	สภาพแวดล้อมความเข้มแสง 80 – 300 lux.....	52
4.3.1.2	สภาพแวดล้อมความเข้มแสง 300 – 650 lux.....	52
4.3.2	วัดระยะห่างจากกล้องถึงสัญลักษณ์ (Color) ด้วยกล้อง.....	52
4.3.2.1	สภาพแวดล้อมความเข้มแสง 80 – 300 lux.....	52
4.3.2.2	สภาพแวดล้อมความเข้มแสง 300 – 650 lux .....	53
4.4	ทดสอบการวัดขนาดมุมของสัญลักษณ์เทียบจุดกึ่งกลางกล้อง .....	53
4.5	เปรียบเทียบการวัดระยะห่างของเซนเซอร์.....	54
4.5.1	วัดระยะห่างวัตถุด้วยเซนเซอร์ ULTRASONIC .....	55
4.5.2	วัดระยะห่างวัตถุด้วยเซนเซอร์ MAX SONA .....	55
4.6	ทดสอบการเข้าจอดและออกจากสถานี.....	56
4.6.1	ทดสอบการเข้าจอด .....	56
4.6.2	ทดสอบการออกจากสถานี.....	56
4.7	ทดสอบการเคลื่อนที่ติดตาม AGV ตัวนำ .....	57
<b>5</b>	<b>สรุปและข้อเสนอแนะ .....</b>	<b>59</b>
5.1	สรุปผลการวิจัย.....	59
5.1.1	ผลการทดลองการตรวจจับสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงที่ แตกต่างกัน.....	59

## สารบัญ (ต่อ)

### หน้า

5.1.1.1	ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 80 ถึง 300 lux.....	59
5.1.1.2	ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 80 ถึง 300 lux.....	59
5.1.2	ผลการทดลองการวัดระยะห่างด้วยกล้อง.....	60
5.1.2.1	ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 80 ถึง 300 lux.....	60
5.1.2.2	ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 300 ถึง 650 lux.....	600
5.1.3	ผลการทดลองการวัดระยะห่างด้วยเซนเซอร์.....	60
5.1.4	ผลการทดสอบการเข้าจอดสถานีและเริ่มต้นออกจากสถานี.....	60
5.1.5	ผลการทดสอบการทำงานร่วมกันของการวัดระยะและการส่งสัญญาณควบคุมมอเตอร์.....	61
5.2	ข้อเสนอแนะ.....	61
รายการอ้างอิง	.....	61
ภาคผนวก		
ภาคผนวก ก.	โปรแกรมตรวจจับสัญลักษณ์และวัดระยะห่างด้วยกล้อง .....	66
ภาคผนวก ข.	โปรแกรมวัดมุมระหว่างสัญลักษณ์กับจุดกึ่งกลางเฟรม .....	71
ภาคผนวก ค.	รายละเอียดการต่อวงจรบอร์ดคอนโทรลและเซนเซอร์ของตัวรถ AGV โปรแกรมควบคุมมอเตอร์บังคับลิฟต์และรักษาระยะห่าง .....	74
ภาคผนวก ง.	บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในขณะศึกษา .....	83
ประวัติผู้เขียน	.....	92

## สารบัญตาราง

ตารางที่		หน้า
2.1	แสดงค่าของโหนดข้อมูลที่เก็บไว้ใน QR code .....	16
3.1	แสดงการทดลองตรวจจับสัญญาณ .....	47
3.2	แสดงการทดลองหาค่าความคาดเคลื่อนการจอดและออกจากสถานี .....	47



## สารบัญรูป

รูปที่		หน้า
1.1	เปอร์เซ็นต์การเพิ่มขึ้นของส่วนแบ่งทางตลาดการขายสินค้าออนไลน์.....	1
1.2	10 อันดับประเทศที่มีการเติบโต Ecommerce .....	2
2.1	AGV ที่ใช้แถบแม่เหล็กนำทาง.....	5
2.2	AGV ที่ใช้สายไฟนำทาง.....	6
2.3	AGV ที่ใช้เลเซอร์นำทาง.....	6
2.4	AGV ที่ใช้กล้องนำทาง .....	7
2.5	AGV ที่ใช้ RFID นำทาง.....	7
2.6	Differential Steering .....	8
2.7	แสดงการบังคับเลี้ยวแบบต่าง ๆ ของ AGV .....	9
2.8	แสดงการบังคับเลี้ยวแบบ Pure pursuit.....	10
2.9	แสดงโมเดลการเคลื่อนที่ไปยังจุดเป้าหมาย TP เมื่อปรับองศาของล้อหน้า $\delta$ ที่ไม่สอดคล้อง.....	11
2.10	แสดงการปรับองศาของล้อหน้าเท่ากับ 20 องศา .....	11
2.11	แสดงการปรับองศาของล้อหน้าเท่ากับ 15 องศา .....	12
2.12	แสดงการปรับองศาของล้อหน้าเท่ากับ 11.3 องศา.....	12
2.13	แสดงการคำนวณหาองศาบังคับเลี้ยวล้อหน้าโดยใช้หลักการสามเหลี่ยมหน้าจั่ว และกฎของไซน์.....	13
2.14	QR code เวอร์ชัน 1 (21×21).....	16
2.15	QR code เวอร์ชัน 4 (33×33).....	16
2.16	QR code เวอร์ชัน 10 (57×57).....	16
2.17	QR code เวอร์ชัน 25 (117×117).....	16
2.18	โครงสร้างของ QRcode .....	17
2.19	Ultrasonic sensor.....	19
2.20	แสดงหลักการทำงานของ Ultrasonic sensor .....	19
2.21	การสะท้อนของเสียงในวัตถุที่แตกต่างกัน.....	20

## สารบัญรูป (ต่อ)

รูปที่	หน้า	
2.22	แสดงโมดูล Timing Diagram.....	21
2.23	แสดงการเปรียบเทียบ RGB และ CMYK.....	22
2.24	แสดงการเปรียบเทียบ HSV และ HSL.....	22
3.1	แสดงการแผนดำเนินการวิจัย.....	30
3.2	กล้อง intel realsense depth camera d455.....	31
3.3	QR code ที่ใช้ในการทดลอง.....	31
3.4	Color ที่ใช้ในการทดลอง.....	32
3.5	Dell Optiplex 7060 Micro.....	32
3.6	Max Sona sensor .....	35
3.7	Infrared Photoelectric Sensors .....	36
3.8	แสดงการประยุกต์ใช้ทฤษฎีสามเหลี่ยมคล้าย.....	37
3.9	สัญลักษณ์ QR code ที่ใช้ในงานวิจัย.....	38
3.10	การตีกรอบสัญลักษณ์ QR code.....	38
3.11	การตัดแปลงภาพเป็นภาพขาวดำ.....	39
3.12	แสดงการคำนวณระยะห่างระหว่างพิกเซลสองจุด.....	39
3.13	แสดงการวัดระยะสัญลักษณ์ด้วยกล้อง.....	40
3.14	ความสัมพันธ์ระหว่างกล้องและสัญลักษณ์เมื่อมีค่าของมุม.....	41
3.15	แสดงการทำงานของฟังก์ชันวัดมุม.....	41
3.16	แสดงการทำงานของฟังก์ชันวัดมุม.....	42
3.17	ฟังก์ชันเงื่อนไขการสั่งงานควบคุมการเคลื่อนที่.....	42
3.18	Flow chart การควบคุมการเคลื่อนที่ติดตามของ AGV .....	43
3.19	แสดงเส้นรอบ threshold_left และ threshold_right.....	44
3.20	Flow chart เงื่อนไขการทำงานบังคับเลี้ยวล้อหน้า.....	45
3.21	Flow chart การควบคุมการเคลื่อนที่ติดตามของ AGV .....	46
4.1	แสดงการทดลองตรวจจับสัญลักษณ์ที่ความเข้มแสง 80 – 300 lux.....	50
4.2	แสดงการทดลองตรวจจับสัญลักษณ์ที่ความเข้มแสง 300 – 650 lux .....	51

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.3 กราฟเปรียบเทียบการวัดระยะของ QR code กับระยะมาตรฐาน ในช่วง 80 – 300 lux.....	52
4.4 กราฟเปรียบเทียบการวัดระยะของ QR code กับระยะมาตรฐาน ในช่วง 300 – 650 lux.....	52
4.5 กราฟเปรียบเทียบการวัดระยะของ Color กับระยะมาตรฐาน ในช่วง 80 – 300 lux.....	53
4.6 กราฟเปรียบเทียบการวัดระยะของ Color กับระยะมาตรฐาน ในช่วง 300 – 650 lux.....	53
4.7 กราฟแสดงการวัดมุม QR code.....	54
4.8 กราฟแสดงการวัดมุม Color.....	54
4.9 กราฟแสดงการวัดระยะห่างด้วย Ultrasonic.....	55
4.10 กราฟแสดงการวัดระยะห่างด้วย Max sona.....	55
4.11 กราฟแสดงค่าความคาดเคลื่อนของการเข้าจอดทางด้านซ้าย.....	56
4.12 กราฟแสดงค่าความคาดเคลื่อนของการเข้าจอดทางด้านขวา.....	56
4.13 กราฟแสดงค่าความคาดเคลื่อนของการออกจากสถานีทางด้านซ้าย.....	57
4.14 กราฟแสดงค่าความคาดเคลื่อนของการออกจากสถานีทางด้านขวา.....	57
4.15 กราฟแสดงความสัมพันธ์ระหว่างการวัดระยะกับการส่งสัญญาณรีเลย์.....	58

## คำอธิบายสัญลักษณ์และคำย่อ

$\gamma, \delta, \alpha$	=	มุม (องศา)
$l_d$	=	ระยะห่างของล้อหน้าและล้อหลังของ AGV (เซนติเมตร)
$R$	=	รัศมีของวงกลม (เซนติเมตร)
$\cong$	=	สัญลักษณ์แสดงความคล้ายกันของรูปเรขาคณิต



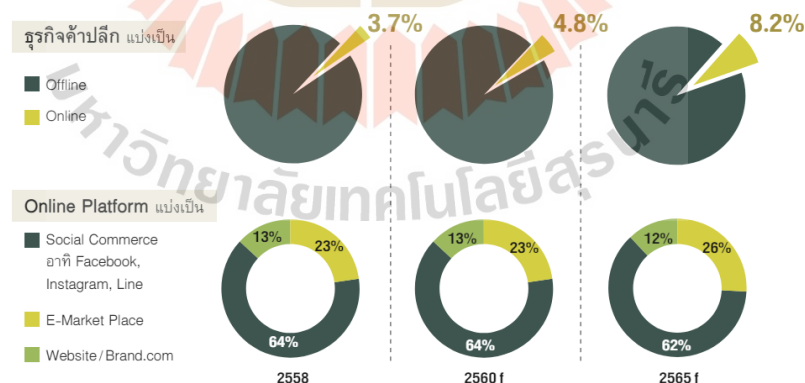


# บทที่ 1

## บทนำ

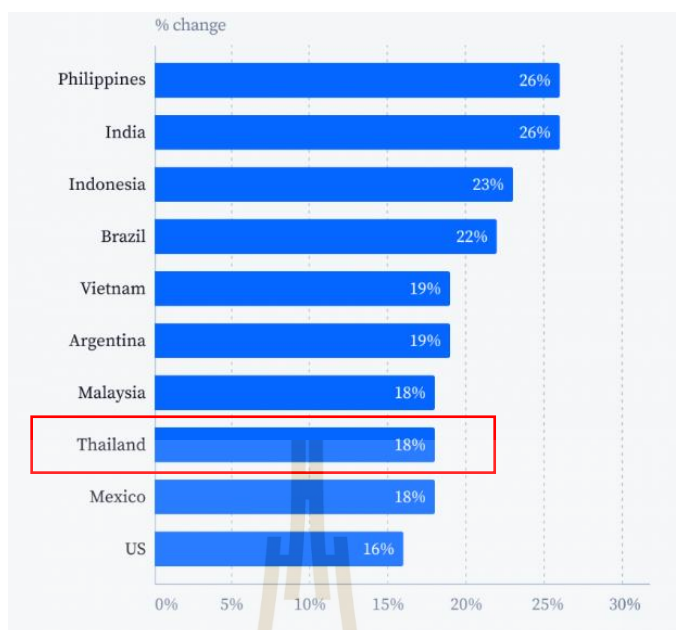
### 1.1 ที่มาและความสำคัญของปัญหา

คลังสินค้าถือเป็นอีกหนึ่งพื้นที่ที่มีการพัฒนาอย่างรวดเร็ว เนื่องมาจากธุรกิจ E-Commerce มีความต้องการของผู้บริโภคสูงในปัจจุบัน มีสาเหตุมาจากผู้บริโภคสั่งซื้อสินค้าผ่านช่องทางออนไลน์มากขึ้น และจากการสำรวจพบว่าการเติบโต 3 ปีย้อนหลังปี 2563 - 2566 เฉลี่ย 26% ต่อปี ข้อมูลอ้างอิงจากศูนย์วิจัยกสิกรไทย ดังในรูปที่ 1.1 ทำให้การบริหารจัดการคลังสินค้ามีความซับซ้อนมากขึ้น เมื่อมีการเติบโตที่สูงขึ้น การแข่งขันทางการตลาดก็สูงขึ้นเช่นกัน ทำให้ผู้ประกอบการจำเป็นต้องลดต้นทุนบางอย่าง เพื่อให้สามารถแข่งขันในตลาดได้ และแรงงานคนเป็นต้นทุนที่สามารถลดได้ อีกทั้งการเคลื่อนย้ายสินค้าบางอย่างในอุตสาหกรรมที่มีน้ำหนักมาก ไม่สามารถใช้แรงงานคนยกเพื่อเคลื่อนย้ายได้ จึงมีการเปลี่ยนไปใช้เทคโนโลยีในการทำงานทดแทน ซึ่งเทคโนโลยีที่ใช้กันอย่างแพร่หลายในอุตสาหกรรมโลจิสติกส์ ก็คือ หุ่นยนต์เคลื่อนที่อัตโนมัติ AGV (Automated Guided Vehicle) ซึ่งจากการสำรวจพบว่าประเทศไทยจัดอยู่ในอันดับที่แปดของประเทศที่มีการเติบโตของ Ecommerce ซึ่งมีเปอร์เซ็นต์การเติบโตที่ 18% เทียบกับการเปลี่ยนแปลง ดังรูปที่ 1.2



ที่มา : ETDA คำนวณและประมาณการโดยศูนย์วิจัยกสิกรไทย

รูปที่ 1.1 เปอร์เซ็นต์การเพิ่มขึ้นของส่วนแบ่งทางตลาดการขายสินค้าออนไลน์ (ETDA ศูนย์วิจัยกสิกรไทย, 2560)



รูปที่ 1.2 10 อันดับประเทศที่มีการเติบโต Ecommerce (2022,TIDIO)

AGV (Automated Guided Vehicle) เป็นยานพาหนะเคลื่อนที่อัตโนมัติ หรือที่เรียกง่าย ๆ ว่าเป็นรถที่ใช้ขนส่ง ที่ใช้ภายในและภายนอกอาคารแบบไร้คนขับ โดยใช้หลักการการเคลื่อนที่ให้อยู่ในเส้นทาง โดยไม่ชนสิ่งกีดขวางด้วยการเคลื่อนที่ตามเส้น เลเซอร์ หรือใช้กล้องเป็นตัวบอกทิศทาง การเคลื่อนที่ ช่วยขนย้ายสินค้าหรืออุปกรณ์ที่มีน้ำหนักมากได้อย่างแม่นยำ แต่ก็มีข้อเสียที่ราคาค่อนข้างสูง เนื่องจากความสามารถในการเคลื่อนที่อัตโนมัติจำเป็นต้องมีเซ็นเซอร์ตรวจจับหลายตัวเพื่อเป็นตัวตรวจรู้และระบบการจับเก็บข้อมูลเส้นทางที่ซับซ้อน ซึ่งจากการเก็บข้อมูลพบว่า AGV ที่มีความสามารถแบกรับโหลดไม่เกิน 100 กิโลกรัม มีราคา 200,000 – 300,000 บาท แต่หากเป็นรุ่นที่สามารถรับโหลดได้มากกว่า 100 กิโลกรัม จะมีราคาตั้งแต่ 300,000 – 3,000,000 บาท ยิ่งความสามารถแบกรับโหลดมาก AGV ก็จะมีราคาสูงแปรผันตามกัน ทำให้ระยะเวลาในการคืนทุนของมีระยะเวลาหลายปี และถ้าจำเป็นต้องมี AGV หลายตัวต้นทุนก็จะแปรผันสูงตามไปด้วย ผู้ทำวิจัยจึงสนใจที่จะออกแบบระบบควบคุมการเคลื่อนที่ที่ใช้ควบคุมหุ่นยนต์ AGV ตัวตาม ซึ่งมีหน้าที่เคลื่อนที่ติดตาม AGV ตัวนำที่มีราคาแพงเท่านั้น ทำให้สามารถลดจำนวนเซนเซอร์ลงได้ ส่งผลให้ราคาต้นทุนของ AGV ต่ำลง แต่สามารถแบกรับน้ำหนักโหลดได้สูง

งานวิจัยนี้นำเสนอการออกแบบระบบควบคุมการเคลื่อนที่ที่ใช้ควบคุม AGV ตัวตามให้สามารถเคลื่อนที่ติดตาม AGV ตัวนำ ด้วยการใช้อุปกรณ์เป็นตัวตรวจรู้ ตัวจับสัญญาณเฉพาะเจาะจงที่ติดอยู่กับ AGV ตัวนำ ในระยะที่ปรับค่าได้ เพื่อควบคุมสั่งการบังคับเลี้ยวซ้ายหรือเลี้ยวขวาของล้อหน้าหุ่นยนต์ AGV ตัวตาม และควบคุมทิศทางของล้อหลังเพื่อเคลื่อนที่เดินหน้าหรือถอยหลังทำให้

สามารถรักษาระยะห่างระหว่าง AGV ตัวนำและตัวตามให้เคลื่อนที่ติดตามอย่างเหมาะสม ด้วยการวัดระยะจากเซ็นเซอร์วัดระยะ และสามารถเคลื่อนที่ออกจากสถานีจอดเมื่อเริ่มต้นภารกิจติดตาม และเมื่อจบภารกิจติดตามสามารถเข้าจอดในสถานีจอดได้อย่างแม่นยำ

## 1.2 วัตถุประสงค์ของงานวิจัย

- 1.2.1 เพื่อศึกษาและพัฒนาระบบควบคุมการบังคับเลี้ยวเคลื่อนที่ของรถติดตามแบบ AGV
- 1.2.2 เพื่อศึกษาและพัฒนาการตรวจจับวัตถุที่ต้องเคลื่อนที่ติดตามด้วยวิชัน
- 1.2.3 เพื่อศึกษาและพัฒนาการรักษาระยะการเคลื่อนที่ติดตามด้วยเซ็นเซอร์วัดระยะ
- 1.2.4 เพื่อศึกษาและพัฒนาการเคลื่อนที่ออกจากสถานีและเข้าจอดในสถานีเมื่อเริ่มต้นและสิ้นสุดภารกิจติดตามแบบอัตโนมัติ

## 1.3 ขอบเขตการวิจัย

- 1.3.1 ออกแบบระบบควบคุมติดตามการเคลื่อนที่ของรถ AGV ตัวตาม โดยเคลื่อนที่ติดตาม AGV ตัวนำด้วยการบังคับเลี้ยวด้วยล้อหน้า
- 1.3.2 เคลื่อนที่ติดตามรถ AGV ตัวนำโดยการรักษาระยะตั้งแต่ 30 - 150 เซนติเมตรด้วยการใช้เซ็นเซอร์ Ultrasonic และกล้องเป็นตัววัดระยะ
- 1.3.3 ใช้กล้องเป็นวิชันในการตรวจจับวัตถุเฉพาะเจาะจงที่ติดอยู่กับ AGV ตัวนำ เพื่อสั่งการบังคับเลี้ยวด้วยล้อหน้า
- 1.3.4 ความเร็วของ AGV ตัวนำมีความเร็วไม่เกิน 1 เมตรต่อวินาที

## 1.4 ระเบียบวิธีวิจัย

- 1.4.1 รวบรวมข้อมูลงานวิจัยที่เกี่ยวข้องกับการบังคับเลี้ยว วัดระยะ และการตรวจจับวัตถุ
- 1.4.2 ศึกษาข้อมูลเกี่ยวกับการบังคับเลี้ยวของหุ่นยนต์ AGV
- 1.4.3 ศึกษาข้อมูลเกี่ยวกับการตรวจจับวัตถุด้วยกล้อง
- 1.4.4 ศึกษาข้อมูลเกี่ยวกับการวัดระยะด้วยเซ็นเซอร์ Ultrasonic
- 1.4.5 ออกแบบการทดลองของการตรวจจับวัตถุในความเข้มแสงที่แตกต่าง
- 1.4.6 ออกแบบการทดลองการเลือกใช้เซ็นเซอร์วัดระยะในประเภทต่าง ๆ เพื่อความเหมาะสม
- 1.4.7 พัฒนาโปรแกรมที่ใช้ตรวจจับและติดตาม AGV ตัวนำ
- 1.4.8 ทำการทดสอบการทำงานของโปรแกรมที่ได้พัฒนาขึ้นสามารถทำงานได้ตามเงื่อนไข
- 1.4.9 สรุปผลการศึกษาและจัดทำข้อเสนอแนะ

1.4.10 จัดทำเอกสาร และรายงานการวิจัย

## 1.5 สถานที่ดำเนินงานวิจัย

มหาวิทยาลัยเทคโนโลยีสุรนารี 111 ถนนมหาวิทยาลัย ตำบลสุรนารี อำเภอเมือง จังหวัดนครราชสีมา 30000

## 1.6 เครื่องมือที่ใช้ในการทำวิจัย

- 1.6.1 mini pc Dell Optix 1 เครื่อง
- 1.6.2 บอร์ดควบคุม Arduino UNO 1 บอร์ด
- 1.6.3 บอร์ดควบคุม Arduino Mega 2560 1 บอร์ด
- 1.6.4 กล้อง Intel deep D495 camera 1 ชิ้น
- 1.6.5 Ultrasonic sensor 1 ชิ้น
- 1.6.6 Infrared Photoelectric Sensors 1 ชิ้น
- 1.6.7 จอทัชสกรีน Nextion ขนาด 5 นิ้ว
- 1.6.8 แบตเตอรี่ลิเทียม 48 V
- 1.6.9 แผ่นสะท้อนแสงขนาด 20 x 20 เซนติเมตร

## 1.7 ประโยชน์ที่คาดว่าจะได้รับ

- 1.7.1 สามารถนำระบบควบคุมการเคลื่อนที่ของหุ่นยนต์ตัวตามให้เคลื่อนที่ติดตาม AGV ตัวนำ ได้อย่างถูกต้องและแม่นยำตามเงื่อนไขที่กำหนด
- 1.7.2 สามารถตรวจจับสัญลักษณ์เฉพาะเจาะจงที่ติดอยู่กับ AGV ตัวนำด้วยวิธีที่มีกล้องเป็นตัวตรวจรู้
- 1.7.3 ลดต้นทุนหุ่นยนต์เคลื่อนที่ AGV ตัวติดตามที่ใช้ในอุตสาหกรรม โดยการลดจำนวนเซ็นเซอร์ที่มีอยู่ใน AGV ปัจจุบัน
- 1.7.4 สามารถเคลื่อนที่ออกจากสถานีและเข้าจอดในสถานีเมื่อเริ่มต้นและสิ้นสุดภาระกิจติดตามแบบอัตโนมัติ

## บทที่ 2

### ปรัทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

#### 2.1 ความหมายของ AGV (Automated Guided Vehicle)

AGV (Automated Guided Vehicles) หรือ ระบบขนส่งและลำเลียงสินค้าอัตโนมัติ เป็นเทคโนโลยีหุ่นยนต์เคลื่อนที่ไร้คนขับ (AMR) ประเภทหนึ่ง ทำหน้าที่ในการขนส่งวัตถุดิบและสินค้าภายในโรงงาน คลังสินค้า และศูนย์กระจายสินค้า ช่วยเพิ่มผลผลิต ความปลอดภัย และประสิทธิภาพการทำงานด้วยการนำระบบจัดการวัสดุแบบอัตโนมัติมาใช้

##### 2.1.1 ระบบการนำทางของ AGV

ระบบนำทางที่ใช้ใน AGV โดยทั่วไป แบ่งออกได้เป็น 5 ประเภทหลัก

###### 2.1.1.1 แถบแม่เหล็ก

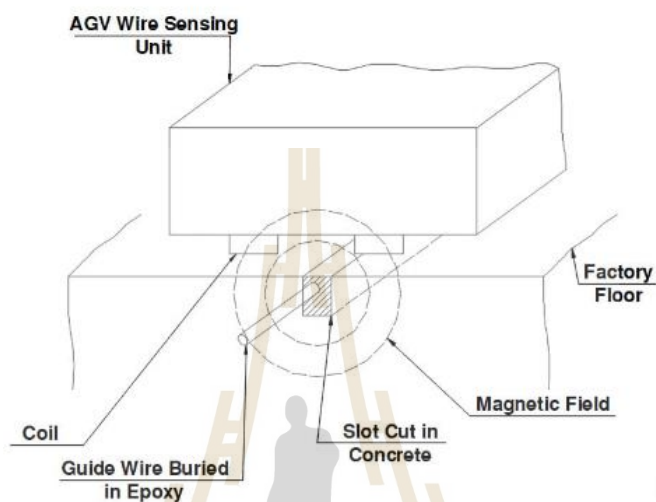
เป็นวิธีการนำทางที่เรียบง่ายและมีราคาไม่แพง แถบแม่เหล็กฝังอยู่ในพื้น และ AGV ตรวจจับแถบแม่เหล็กด้วยเซ็นเซอร์ใต้ตัวรถ ดังรูปที่ 2.1 วิธีนี้เหมาะสำหรับพื้นที่ที่มีเค้าโครงเรียบง่าย แต่แถบแม่เหล็กอาจเสียหายได้ง่ายและอาจเป็นอันตรายต่อคนเดินเท้า



รูปที่ 2.1 AGV ที่ใช้แถบแม่เหล็กนำทาง

### 2.1.1.2 สายไฟ

คล้ายกับแถบแม่เหล็ก แต่มีความยืดหยุ่นมากกว่าและไม่เป็นอันตรายต่อคนเดินเท้า AGV ใช้เซ็นเซอร์ใต้ตัวรถเพื่อตรวจจับสายไฟ ดังรูปที่ 2.2 วิธีนี้เหมาะสำหรับพื้นที่ที่มีเค้าโครงซับซ้อน แต่ต้องติดตั้งและบำรุงรักษาสายไฟเป็นประจำ



รูปที่ 2.2 AGV ที่ใช้สายไฟนำทาง

### 2.1.1.3 เลเซอร์ AGV

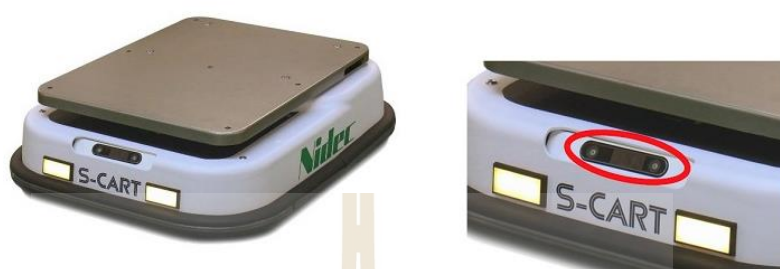
ใช้เลเซอร์เพื่อสร้างแผนที่ 3 มิติของสภาพแวดล้อม จากนั้นใช้แผนที่นี้เพื่อนำทางไปรอบ ๆ พื้นที่ทำงาน ดังรูปที่ 2.3 วิธีนี้แม่นยำสูง แต่มีราคาแพงและอาจมีปัญหาในสภาพแสงที่สว่างจ้าหรือมีฝุ่นละอองมาก



รูปที่ 2.3 AGV ที่ใช้เลเซอร์นำทาง

#### 2.1.1.4 วิสัยทัศน์ AGV

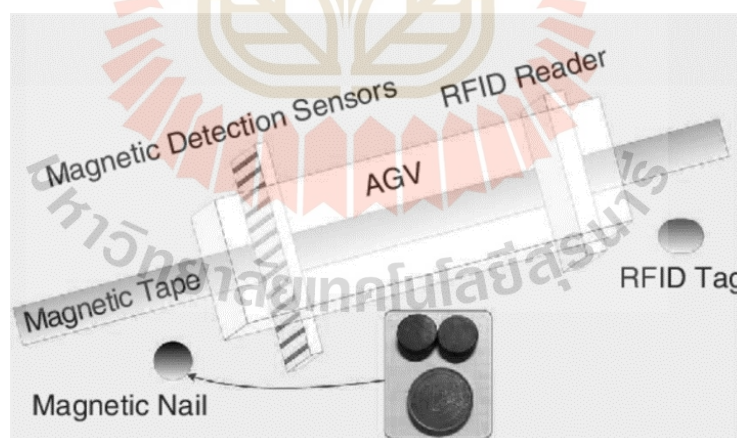
ใช้กล้องเพื่อดูสภาพแวดล้อม จากนั้นใช้ซอฟต์แวร์วิเคราะห์ภาพเพื่อระบุเส้นทางไปยังจุดหมายปลายทาง ดังรูปที่ 2.4 วิธีนี้มีความยืดหยุ่นสูง แต่ซับซ้อนและมีราคาแพง



รูปที่ 2.4 AGV ที่ใช้กล้องนำทาง

#### 2.1.1.5 RFID

ป้าย RFID ติดตั้งอยู่บนพื้นและ AGV ตรวจจับป้าย RFID ด้วยเซ็นเซอร์ใต้ตัวรถ ดังรูปที่ 2.5 วิธีนี้ทนทานกว่าแถบแม่เหล็กและไม่เป็นอันตรายต่อคนเดินเท้า แต่สัญญาณ RFID อาจถูกรบกวนจากวัตถุโลหะ



รูปที่ 2.5 AGV ที่ใช้ RFID นำทาง

ปัจจัยหลายประการที่มีผลต่อการเลือกระบบนำทางที่เหมาะสมสำหรับ AGV ดังนี้ เค้ําโครงพื้นที่ พื้นที่มีเค้ําโครงเรียบง่าย เหมาะกับการใช้แถบแม่เหล็กหรือสายไฟ พื้นที่มี เค้ําโครงซับซ้อนเหมาะกับการใช้เลเซอร์หรือวิสัยทัศน์ สภาพแวดล้อมที่มีแสงสว่างจ้าหรือมีฝุ่นละออง มาก ไม่เหมาะกับการใช้เลเซอร์ สภาพแวดล้อมที่มีวัตถุโลหะมาก ไม่เหมาะกับการใช้ RFID งบประมาณ แถบแม่เหล็กและสายไฟมีราคาไม่แพง เลเซอร์และวิสัยทัศน์มีราคาแพง ความต้องการ ด้านความแม่นยำ เลเซอร์และวิสัยทัศน์มีความแม่นยำสูง แถบแม่เหล็กและสายไฟมีความแม่นยำปาน กลาง ความต้องการด้านความยืดหยุ่น วิสัยทัศน์มีความยืดหยุ่นสูง แถบแม่เหล็กและสายไฟมีความ ยืดหยุ่นปานกลาง

## 2.1.2 ประเภทของการขับเคลื่อน AGV

### 2.1.2.1 ระบบบังคับเลี้ยวแบบเฟืองต่าง (Differential Steering)

บางครั้งเรียกว่า Skid Steering หรือ Tank Steering ใช้ล้อขับเคลื่อนอิสระ สองข้าง ดังรูปที่ 2.6 ควบคุมความเร็วและทิศทางของแต่ละล้อแยกกัน ซึ่งสามารถเลี้ยวโค้งแคบและ หมุนตัวอยู่กับที่ (zero radius turn)



รูปที่ 2.6 Differential Steering

### 2.1.2.2 ระบบบังคับเลี้ยวแบบใช้ล้อหมุน (Single or Dual Steering)

ใช้ล้อหนึ่งหรือหลายล้อที่สามารถหมุนเพื่อเปลี่ยนทิศทาง มักใช้ล้อหนึ่งหรือ สองล้อที่ด้านหน้าหรือด้านหลังของ AGV ที่สามารถบังคับเลี้ยวเพื่อนำทางรถ ให้การควบคุมที่นุ่มนวล และแม่นยำยิ่งขึ้น คล้ายกับรถยนต์



### 2.1.2.3 ระบบบังคับเลี้ยวรอบทิศทาง (Omnidirectional Steering)

ใช้ล้อที่สามารถเคลื่อนที่ไปในทุกทิศทาง เช่น ล้อ Mecanum หรือล้อรอบทิศทาง อนุญาตให้เคลื่อนที่ได้อย่างอิสระทุกทิศทาง รวมถึงการเคลื่อนที่ด้านข้างและทแยง เหมาะสำหรับสภาพแวดล้อมที่ต้องการความคล่องตัวและยืดหยุ่นสูง+

### 2.1.2.4 ระบบบังคับเลี้ยวแบบหมุน (Swivel or Caster Steering)

ใช้ล้อที่สามารถหมุนได้อย่างอิสระ มักใช้ร่วมกับล้อตายหรือล้อขับเคลื่อนอื่นๆ พบได้ทั่วไปใน AGV ที่ออกแบบมาสำหรับรับน้ำหนักเบาและเส้นทางเดินเรียบง่าย

### 2.1.2.5 ระบบบังคับเลี้ยวแบบอัคร์แมน (Ackermann Steering)

เลียนแบบกลไกการบังคับเลี้ยวของรถยนต์ ใช้การบังคับเลี้ยวล้อหน้า ขณะที่ล้อหลังวิ่งตาม เหมาะสำหรับ AGV ที่ต้องวิ่งตามเส้นทางโค้งเว้าอย่างราบรื่นและต้องการความเสถียรที่ความเร็วสูง

### 2.1.2.6 ระบบบังคับเลี้ยวแบบซิงโครนัส (Synchronous Steering)

ล้อทั้งหมดสามารถบังคับเลี้ยวและขับเคลื่อนได้อย่างสอดคล้องกัน ให้ความคล่องตัวสูงด้วยการเคลื่อนไหวของล้อที่ประสานงานกัน มีประโยชน์สำหรับการเดินเรือที่ซับซ้อนและการวางตำแหน่งที่แม่นยำ



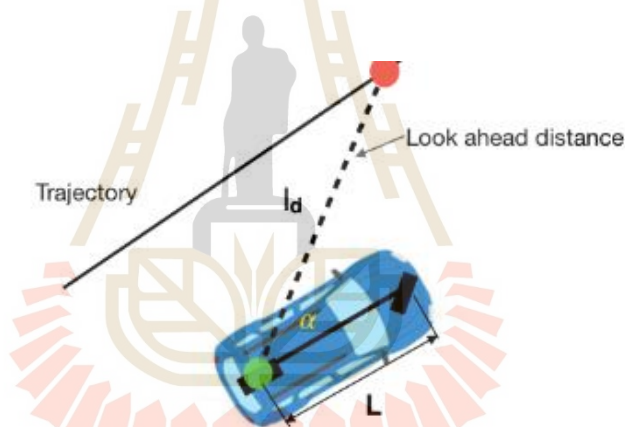
รูปที่ 2.7 แสดงการบังคับเลี้ยวแบบต่าง ๆ ของ AGV

## 2.3 อัลกอริทึม Pure Pursuit

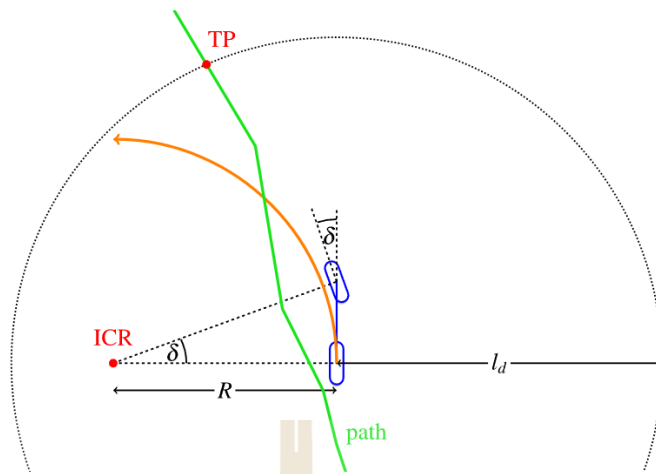
อัลกอริทึม Pure Pursuit เป็นอัลกอริทึมติดตามเส้นทางที่ใช้กันทั่วไปในหุ่นยนต์เคลื่อนที่ ซึ่งรวมถึง ยานพาหนะนำทางอัตโนมัติ (AGV) ด้วย วัตถุประสงค์หลักของอัลกอริทึมนี้ คือ การควบคุมทิศทางของยานพาหนะเพื่อให้วิ่งตามเส้นทางที่กำหนดไว้ล่วงหน้า โดยปรับมุมเลี้ยวเพื่อลดระยะห่างระหว่างตำแหน่งปัจจุบันของยานพาหนะกับจุดมองไปข้างหน้าบนเส้นทาง อัลกอริทึม Pure Pursuit ทำงานโดยปรับมุมเลี้ยวของยานพาหนะแบบไดนามิก เพื่อให้แน่ใจว่ายานพาหนะยังคงอยู่บนเส้นทาง

ที่ต้องการ อัลกอริทึมจะเลือกจุดบนเส้นทาง เรียกว่า จุดมองไปข้างหน้า (Look-ahead point) ซึ่งอยู่ห่างจากตำแหน่งปัจจุบันของยานพาหนะไปเป็นระยะทางหนึ่ง จากนั้น ระบบบังคับเลี้ยวของยานพาหนะจะถูกปรับเพื่อลดข้อผิดพลาดระหว่างทิศทางการมุ่งหน้าของยานพาหนะกับทิศทางไปยังจุดมองไปข้างหน้า

ขั้นตอนสำคัญในอัลกอริทึม Pure Pursuit กำหนดเส้นทางไว้ล่วงหน้าสำหรับยานพาหนะที่จะวิ่งตาม เส้นทางนี้มักแสดงเป็นชุดของจุดอ้างอิง (Waypoints) หรือเส้นโค้งต่อเนื่อง ดังรูปที่ 2.8 การคำนวณจุดมองไปข้างหน้า อัลกอริทึมจะระบุจุดมองไปข้างหน้าบนเส้นทาง จุดนี้ถูกเลือกตามระยะมองไปข้างหน้าที่กำหนดไว้ล่วงหน้าจากตำแหน่งปัจจุบันของยานพาหนะ การคำนวณมุมเลี้ยว มุมเลี้ยวจะถูกคำนวณตามรูปทรงของยานพาหนะและตำแหน่งของจุดมองไปข้างหน้าสัมพันธ์กับยานพาหนะ ดังรูปที่ 2.9 มุมเลี้ยวจะถูกปรับเพื่อบังคับยานพาหนะไปยังจุดมองไปข้างหน้า การควบคุมยานพาหนะ มุมเลี้ยวที่คำนวณได้จะถูกนำไปใช้กับกลไกบังคับเลี้ยวของยานพาหนะ เพื่อนำทางไปตามเส้นทาง

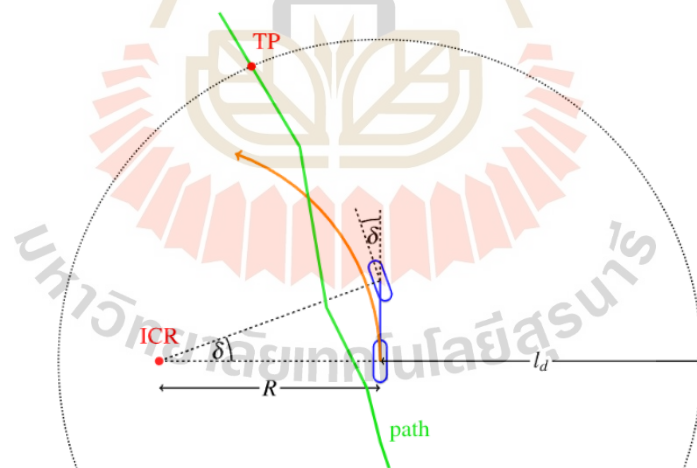


รูปที่ 2.8 แสดงการบังคับเลี้ยวแบบ Pure pursuit (Mario Theers and Mankaran Singh)

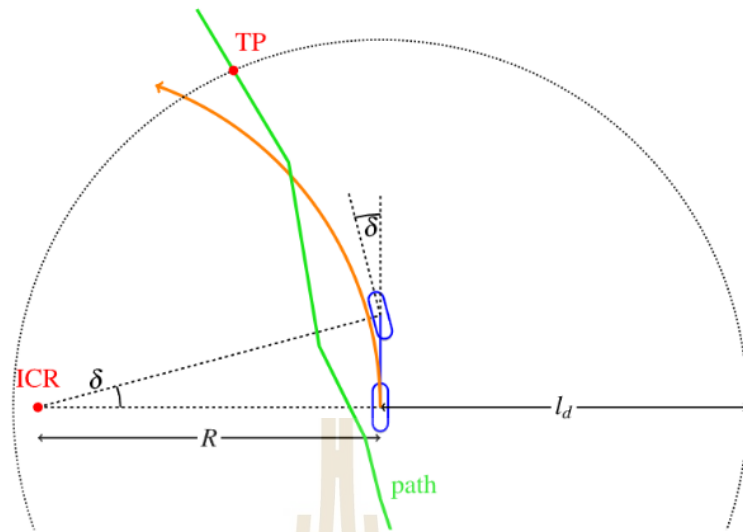


รูปที่ 2.9 แสดงโมเดลการเคลื่อนที่ไปยังจุดเป้าหมาย TP เมื่อปรับองศาของล้อหน้า  $\delta$  ที่ไม่สอดคล้อง (Mario Theers and Mankaran Singh)

ถ้าหากปรับเปลี่ยนค่าของมุมเลี้ยวล้อหน้า  $\delta$  ที่เหมาะสมจะทำให้การเคลื่อนที่ของ AGV ไปที่จุดเป้าหมาย ค่ามุมเลี้ยวยิ่งมีค่ามากจะส่งผลให้วงเลี้ยวกว้างตามมาด้วยดังรูปที่ 2.10

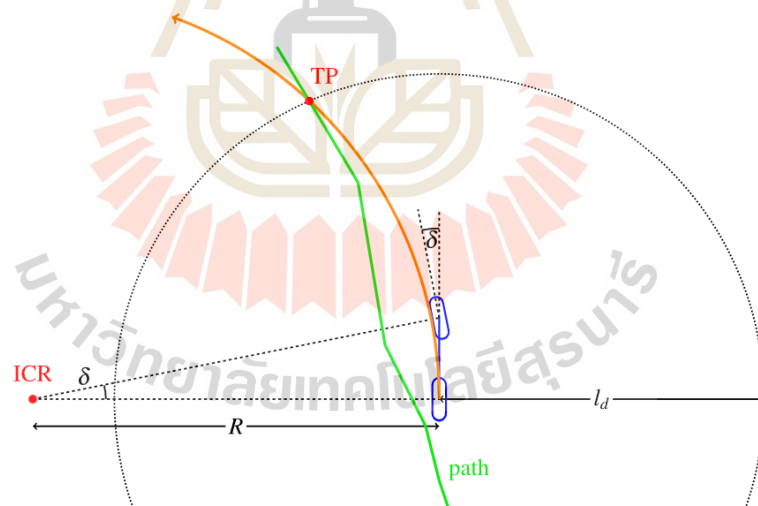


รูปที่ 2.10 แสดงการปรับองศาของล้อหน้าเท่ากับ 20 องศา (Mario Theers and Mankaran Singh)



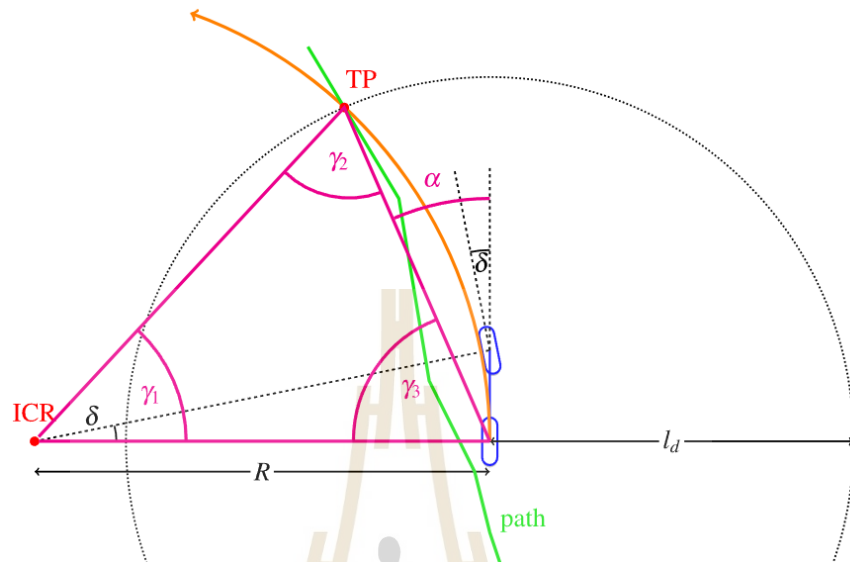
รูปที่ 2.11 แสดงการปรับองศาของล้อหน้าเท่ากับ 15 องศา (Mario Theers and Mankaran Singh)

ถ้าหากปรับองศาของล้อได้เหมาะสมจะทำให้การเคลื่อนที่ของ AGV สามารถไปถึงเป้าหมายได้อย่างถูกต้องดังรูปที่ 2.11 และรูปที่ 2.12



รูปที่ 2.12 แสดงการปรับองศาของล้อหน้าเท่ากับ 11.3 องศา  
(Mario Theers and Mankaran Singh)

ซึ่งการคำนวณองศาของบังเลี้ยวของล้อหน้าสามารถคำนวณได้จากสมการทางคณิตศาสตร์ โดยใช้หลักการของสามเหลี่ยมหน้าจั่วและกฎของไซน์



รูปที่ 2.13 แสดงการคำนวณองศาบังคับเลี้ยวล้อหน้า

โดยใช้หลักการสามเหลี่ยมหน้าจั่วและกฎของไซน์ (Mario Theers and Mankaran Singh) ระยะที่จะต้องกำหนดคือ ระยะห่างระหว่างจุด TP ถึงล้อหลัง จากนั้นลากเส้นตั้งฉากจากล้อหน้าและล้อหลังจะเกิดจุดตัด ICR ซึ่งจุดนี้จะเป็นศูนย์กลางของวงกลมสี่ล้อ ซึ่งเป็นวงเลี้ยวของการเคลื่อนที่ เมื่อได้จุด ICR จะเกิดสามเหลี่ยมหน้าจั่วที่มีด้าน ICR ถึง TP เท่ากับ ระยะ R จากสมบัติของรูปสามเหลี่ยมหน้าจั่วเมื่อมีด้านประกอบมุมสองมุมเท่ากัน จะส่งผลให้มุมของสามเหลี่ยมสองมุมมีค่าเท่ากันด้วยดังรูปที่ 2.13

$$\gamma_2 = \gamma_3 \quad (2.1)$$

จากรูปที่  $\gamma_3 + \alpha = 90$  เมื่อแทน  $\gamma_3$  ในสมการที่ 2.1 จะได้

$$\gamma_2 = \gamma_3 = 90 - \alpha \quad (2.2)$$

ซึ่งมุมภายในของรูปสามเหลี่ยมจะมีค่าเท่ากับ 180 องศา

$$180 = \gamma_1 + \gamma_2 + \gamma_3 = \gamma_1 + (90 - \alpha) + (90 - \alpha) \quad (2.3)$$

$$\gamma_1 = 2\alpha \quad (2.4)$$

จากกฎของไซน์

$$\frac{l_d}{\sin(\gamma_1)} = \frac{R}{\sin(\gamma_2)} \quad (2.5)$$

$l_d$  คือ ระยะจากล้อหลังไปถึงจุด TP แทน  $\gamma_2 = 90 - \alpha$  และ  $\gamma_1 = 2\alpha$  ในสมการที่

$$\frac{l_d}{\sin(2\alpha)} = \frac{R}{\sin(90 - \alpha)} \quad (2.6)$$

จากกฎมุมสองเท่าของตรีโกณมิติ  $\sin(2\alpha) = 2\sin(\alpha)\cos(\alpha)$

$$\frac{l_d}{2\sin(\alpha)\cos(\alpha)} = \frac{R}{\cos(\alpha)} \quad (2.7)$$

$$R = \frac{l_d}{2\sin(\alpha)} \quad (2.8)$$

$$\delta = \arctan \frac{L}{R} \quad (2.9)$$

$$\delta = \arctan \frac{2L\sin(\alpha)}{l_d} \quad (2.10)$$

เมื่อทราบระยะ TP และ มุมที่ล้อหลังกระทำกับจุด TP จะสามารถคำนวณหามุมบังคับเลี้ยวได้จากสมการที่ 2.10

ข้อดีของอัลกอริทึม Pure Pursuit ความเรียบง่าย อัลกอริทึม Pure Pursuit ใช้ได้ง่ายและประหยัดพลังงานในการคำนวณ การติดตามเส้นทางอย่างราบรื่น มันมีติดตามเส้นทาง ที่ราบรื่น ทำให้เหมาะสำหรับการใช้งานหลากหลาย การปรับแบบเรียลไทม์ อัลกอริทึมสามารถปรับได้แบบเรียลไทม์ เพื่อให้แน่ใจว่ายานพาหนะวิ่งตามเส้นทางได้อย่างแม่นยำ แม้ในสภาพแวดล้อมที่มีการเปลี่ยนแปลง

ข้อจำกัดและข้อควรพิจารณา ระยะมองไปข้างหน้า การเลือกระยะมองไปข้างหน้าที่เหมาะสมเป็นสิ่งสำคัญ หากระยะทางสั้นเกินไป ยานพาหนะอาจมีพฤติกรรมการสั่น หากยาวเกินไป ยานพาหนะอาจไม่วิ่งตามเส้นทางได้อย่างแม่นยำ ความโค้งของเส้นทาง อัลกอริทึมทำงานได้ดีบนเส้นทางที่มีทางโค้งที่ไม่หักมุมมากเกินไป

### 2.3 หลักการทำงานของ QR code

QR code (quick-response code) คือ เป็นประเภทหนึ่งของบาร์โค้ด เป็นบาร์โค้ดสองมิติ ที่ถูกสร้างขึ้นในปี 1994 โดยบริษัทญี่ปุ่น Denso Wave เพื่อใช้ในการติดป้ายบอกรหัสสินค้า ของอะไหล่รถยนต์ QR code ประกอบด้วยสี่เหลี่ยมสีดำที่เรียงต่อกันในกริดสี่เหลี่ยมบนพื้นหลังสีขาว ซึ่งสามารถถูกสแกนโดยอุปกรณ์ภาพ เช่น กล้อง และประมวลผลด้วย Reed-Solomon error จนกระทั่งภาพสามารถตีความหมายได้อย่างถูกต้อง ข้อมูลที่จำเป็นจะถูกดึงออกจากลายเส้นที่ปรากฏทั้งในส่วนของแนวนอนและแนวตั้ง QR code ใช้วิธีการเข้ารหัสที่มีมาตรฐานทั้งหมด 4 รูปแบบ (i) ตัวเลข, (ii) อัลฟานิวเมอริก, (iii) ไบต์หรือไบนารี และ (iv) คณิต ระบบ QR code ไม่ได้ถูกใช้แค่ในอุตสาหกรรมยานยนต์เท่านั้น เนื่องจากสามารถอ่านได้เร็วและเก็บข้อมูลได้มาก ซึ่งไม่เหมือนกับบาร์โค้ดแบบเดิมที่มีลักษณะเป็นมิติเดียวและถูกออกแบบให้สามารถสแกนด้วยลำแสง QR code ถูกตรวจจับโดยเซนเซอร์รูปภาพดิจิทัล 2 มิติ และถูกวิเคราะห์ประมวลผลที่ถูกโปรแกรมไว้ ประมวลผลจากตำแหน่งของสามกล่องที่แตกต่างกันที่มุมของภาพ QR code, โดยใช้กล่องเล็ก (หรือกล่องหลาย ๆ กล่อง) ที่มุมที่สี่เพื่อปรับขนาด, ทิศทาง, และมุมของมุมมอง จุดเล็กทั่วทั้ง QR code จากรูปถูกแปลงเป็นตัวเลขไบนารีและถูกตรวจสอบด้วย error-correcting อัลกอริทึม

ความจุของข้อมูล ปริมาณข้อมูลที่สามารถถูกแทนด้วยสัญลักษณ์ QR code ขึ้นอยู่กับประเภทข้อมูล (โหมดหรือชุดตัวอักษรที่ใส่), เวอร์ชัน (1, ..., 40, ที่ระบุขนาดรวมของสัญลักษณ์, เช่น  $4 \times$  เลขเวอร์ชัน + 17 ก็จะได้จำนวนจุดของแต่ละด้าน) ดังรูปที่ 2.14 – 2.17 และระดับการแก้ไขข้อผิดพลาด ความจุการเก็บข้อมูลสูงสุดคือที่เวอร์ชัน 40 และระดับการแก้ไขข้อผิดพลาด L (ต่ำ)

ตารางที่ 2.1 แสดงค่าของโหมดข้อมูลที่เก็บไว้ใน QR code

Input mode	Max. characters	Bits/char.	Possible characters, default encoding
Numeric only	7,089	$3\frac{1}{3}$	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Alphanumeric	4,296	$5\frac{1}{2}$	0–9, A–Z (upper-case only), space, \$, %, *, +, -, ., /, :
Binary/byte	2,953	8	ISO/IEC 8859-1
Kanji/kana	1,817	13	Shift JIS X 0208

ตัวอย่างเวอร์ชันของ QR code



รูปที่ 2.14 QR code เวอร์ชัน 1 (21×21)



รูปที่ 2.15 QR code เวอร์ชัน 4 (33×33)

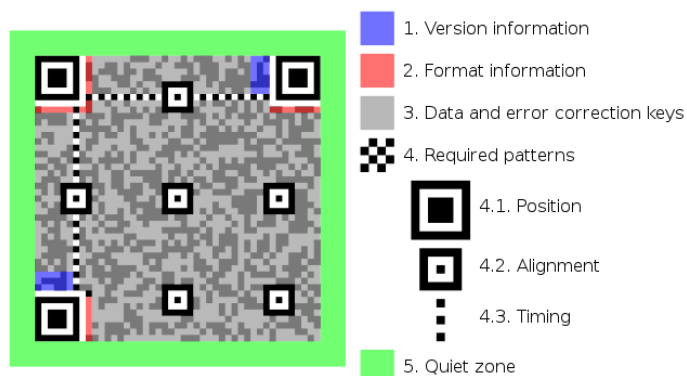


รูปที่ 2.16 QR code เวอร์ชัน 10 (57×57)



รูปที่ 2.17 QR code เวอร์ชัน 25 (117×117)





รูปที่ 2.18 โครงสร้างของ QRcode

Quiet zone เป็นพื้นที่รอบนอกของ QR code เป็นพื้นที่สีขาวเพื่อช่วยให้มองเป็นขอบเขตของ QR code และช่วยแยกแยะระหว่าง QR code กับสภาพแวดล้อม ดังรูปที่ 2.18

Detection marker เป็นสี่เหลี่ยมจัตุรัสอยู่ที่มุมซ้ายล่าง ซ้ายบน และ ขวาบน ช่วยให้การแสกนมีความแม่นยำและรวดเร็วในการอ่าน

Alignment pattern เป็นสี่เหลี่ยมจัตุรัสเล็ก จะเอาไว้ช่วยให้อ่านค่าได้เมื่อ QR code อยู่ในมุมที่แตกต่างกันหรือพื้นผิวที่โค้งงอ ซึ่งถ้า QR code มีขนาดใหญ่ก็จะมีจุด Alignment เยอะขึ้นตาม

Timing pattern เป็นเส้นตรงลักษณะตัวแอลเชื่อมระหว่างสี่เหลี่ยม Detect marker 3 จุด ช่วยให้ระบุตารางของข้อมูล และช่วยให้แสกนเนอร์รู้ขนาดของ QR code ด้วย

Version information ซึ่ง QR code มีทั้งหมด 40 เวอร์ชัน

Format information ในส่วนนี้จะมียุทธศาสตร์ความคลาดเคลื่อนของ error และยังบอกรายละเอียดรูปแบบมากที่สุดของข้อมูล ช่วยให้อ่านข้อมูลได้ง่ายขึ้น

Data and error correction keys ส่วนของข้อมูล ซึ่งถ้าข้อมูลหายไปไม่เกิน 30% จะยังสามารถอ่านข้อมูลได้อยู่

Error correction คือ กระบวนการและเทคนิคที่ใช้ในการตรวจจับและแก้ไขข้อผิดพลาดที่อาจเกิดขึ้นในการถ่ายโอนข้อมูลหรือการเก็บข้อมูล ซึ่งมักจะเป็นส่วนหนึ่งของระบบการสื่อสารหรือการจัดเก็บข้อมูลที่ต้องการความเสถียรและความถูกต้อง ในทางทฤษฎี การแก้ไขข้อผิดพลาดสามารถทำได้ในหลายวิธี แต่มีหลักการทั่วไปคือการเพิ่มข้อมูลเสริมเข้าไปในข้อมูลที่ถูกส่งหรือเก็บ เพื่อให้สามารถตรวจสอบและแก้ไขข้อผิดพลาดที่เกิดขึ้น หนึ่งในวิธีที่ใช้กันมาก คือ Reed-Solomon code ซึ่งเป็นระบบการเข้ารหัสที่สามารถใช้ในการแก้ไขข้อผิดพลาดได้โดยทั่วไป การใช้ Reed-Solomon code นั้น ข้อมูลเสริมถูกเพิ่มลงไปในการส่งหรือเก็บ และรหัสตรวจสอบข้อผิดพลาดถูกสร้างขึ้นเมื่อข้อมูลถูกรับหรือถูกอ่าน ระบบสามารถใช้รหัสตรวจสอบข้อผิดพลาดเพื่อตรวจสอบว่ามี

ข้อผิดพลาดหรือไม่และสามารถใช้ข้อมูลเสริมเพื่อแก้ไขข้อผิดพลาดที่พบ การแก้ไขข้อผิดพลาดเป็นสิ่งสำคัญในการสื่อสารและการจัดเก็บข้อมูล เพราะสภาพแวดล้อมและการส่งผ่านข้อมูลอาจทำให้เกิดข้อผิดพลาดได้ เช่น สัญญาณเสีย, การบิดผิดพลาด, หรือการทำลายข้อมูลในระหว่างการถ่ายโอน

Reed–Solomon code ใช้ 1 ใน 37 พหุนามที่แตกต่างกันที่มีดีกรีตั้งแต่ 7 ถึง 68 ขึ้นอยู่กับจำนวนข้อมูลการแก้ไขข้อผิดพลาดที่รหัสเพิ่มเข้าไป การเลือกใช้ระดับของพหุนามนี้ได้รับการบ่งชี้จากรูปแบบของ Reed–Solomon ที่ใช้ (เป็นรูปแบบที่มีระบบ BCH) โดยทั่วไปพหุนามเหล่านี้ เมื่อพุดถึงขั้นตอนของ Reed–Solomon นั้นมีความเสี่ยงในการสับสน จำนวนข้อมูลที่แตกต่างจากข้อมูลการแก้ไขข้อผิดพลาดภายในแต่ละ block ขึ้นอยู่กับ (i) เวอร์ชัน (ความยาวด้าน) ของสัญลักษณ์ QR และ (ii) ระดับการแก้ไขข้อผิดพลาดที่มีทั้งหมด 4 ระดับ ยิ่งมีระดับการแก้ไขข้อผิดพลาดสูง ความจุพื้นที่เก็บข้อมูลก็ยิ่งลดความสามารถในการแก้ไขข้อผิดพลาดโดยประมาณที่แต่ละระดับ

ระดับ L (ต่ำ)	7%	ของข้อมูลไบท์ที่สามารถฟื้นฟู
ระดับ M (กลาง)	15%	ของข้อมูลไบท์ที่สามารถฟื้นฟู
ระดับ Q (คลอไทด์)	25%	ของข้อมูลไบท์ที่สามารถฟื้นฟู
ระดับ H (สูง)	30%	ของข้อมูลไบท์ที่สามารถฟื้นฟู

ในสัญลักษณ์ QR ขนาดใหญ่ ข้อความถูกแบ่งเป็นบล็อกของรหัส Reed–Solomon หลายๆ บล็อก ขนาดของบล็อกถูกเลือกโดยไม่พยายามแก้ไขข้อผิดพลาดมากกว่า 15 ข้อผิดพลาดต่อบล็อก จะลดความซับซ้อนของอัลกอริทึมการถอดรหัส ทำให้เป็นไปได้น้อยกว่าความเสียหายที่ตำแหน่งเฉพาะใดเฉพาะหนึ่งในสัญลักษณ์ QR จะมีผลกระทบต่อความจุของบล็อกที่มีอยู่ด้วยการแก้ไขข้อผิดพลาด เป็นไปได้ที่จะสร้างรหัส QR ที่มีการประดับเพื่อทำให้มีความอ่านหรือดึงดูตตามากขึ้นหรือน่าสนใจ ต่อตามสายตาของมนุษย์และเพื่อผสมสี โลโก้ และลักษณะอื่น ๆ ในบล็อกรหัส QR การประดับเหล่านี้ ถือเป็นข้อผิดพลาด แต่รหัสยังสามารถสแกนได้ถูกต้อง ยังเป็นไปได้ที่ออกแบบรหัส QR ที่มีลักษณะศิลปะโดยไม่ลดความสามารถในการแก้ไขข้อผิดพลาดโดยการจัดการกับโครงสร้างทางคณิตศาสตร์ขั้นสูง อัลกอริทึมประมวลผลภาพก็ถูกใช้เพื่อลดข้อผิดพลาดในรหัส QR

## 2.4 หลักการทำงานของ Ultrasonic sensor

Ultrasonic sensor เป็นอุปกรณ์ที่ใช้วัดระยะทางโดยการส่งคลื่นเสียงและรวบรวมเสียงสะท้อนที่กลับมา ดังรูปที่ 2.19 สามารถวัดสิ่งของได้ไกลถึง 4.5 เมตร ทำให้เป็นเครื่องมือสำหรับวัดระยะทางใกล้และระยะทางไกลอย่างแม่นยำโดยไม่ต้องสัมผัสกับสิ่งของเป้าหมาย ซึ่งจำเป็นในการใช้งานหลายประเภท เช่น ระบบหลีกเลี่ยงสิ่งกีดขวางในหุ่นยนต์หรือยานพาหนะอัตโนมัติ เช่น เซอร์มี

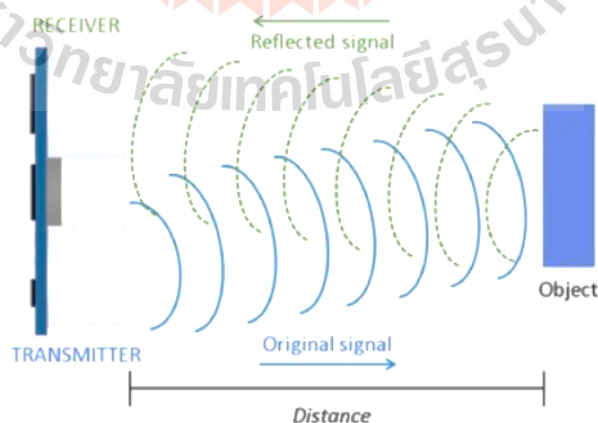
ความกว้างของ pulse ที่ปรับได้ช่วยให้การวัดความละเอียดสูงขึ้น เมื่อตั้งค่าไว้ที่ระดับต่ำกว่า แม้ว่าความแม่นยำของช่วงจะลดลงก็ตาม



รูปที่ 2.19 Ultrasonic sensor

เมื่อเปรียบเทียบกับเซ็นเซอร์อินฟราเรด (IR) ในการใช้งานการตรวจจับระยะใกล้ เซ็นเซอร์อัลตราโซนิกมีความไวต่อการรบกวนจากควัน ก๊าซ และอนุภาคในอากาศอื่น ๆ น้อยกว่า เซ็นเซอร์อัลตราโซนิกยังใช้เป็นเซ็นเซอร์ระดับเพื่อตรวจจับ ตรวจสอบ และควบคุมระดับของเหลวในภาชนะปิด เช่นถังสารเคมี สิ่งที่น่าสนใจที่สุด คือ เทคโนโลยีอัลตราซาวด์ช่วยให้อุตสาหกรรมทางการแพทย์สามารถถ่ายภาพอวัยวะภายใน ระบุเนื้องอก และรับประกันสุขภาพของทารกในครรภ์

หลักการทำงานของอัลตราโซนิก คือ การวัดเวลาที่ใช้ส่งสัญญาณโดยเครื่องส่งและแพร่กระจายกลับไปยังเครื่องรับ ดังรูปที่ 2.20 เซ็นเซอร์อัลตราโซนิกทำงานบนความถี่อัลตราโซนิก ความถี่ที่อยู่เหนือช่วงการได้ยินของมนุษย์เรียกว่า ความถี่เหนือเสียง ความถี่เหล่านี้สูงกว่า 20,000 Hz

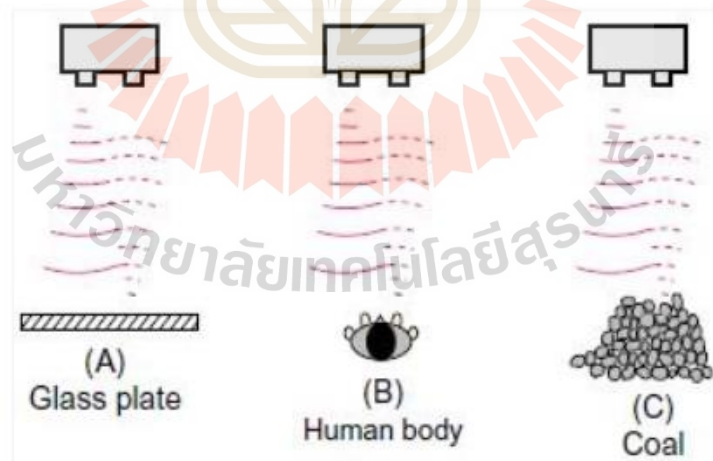


รูปที่ 2.20 แสดงหลักการทำงานของ Ultrasonic sensor

เซ็นเซอร์อัลตราโซนิกจะปล่อยคลื่นอัลตราโซนิกและแปลงเสียงที่สะท้อนให้เป็นสัญญาณไฟฟ้า อัลตราโซนิกเดินทางเร็วกว่าเสียงที่ได้ยิน และประกอบด้วยองค์ประกอบ 2 ส่วน ได้แก่ ตัวส่งสัญญาณ (ซึ่งใช้ piezoelectric crystal เพื่อปล่อยเสียง) และตัวรับ แม้ว่าเซ็นเซอร์บางตัวจะใช้ตัวส่งและตัวรับเสียงแยกกัน แต่ก็เป็นไปได้ที่จะเอาฟังก์ชันทั้งสองมารวมไว้ในอุปกรณ์ตัวเดียว โดยใช้องค์ประกอบอัลตราโซนิกเพื่อสลับระหว่างการส่งสัญญาณในวงจรที่ต่อเนื่องกัน เครื่องส่งสัญญาณของโมดูลจะส่งเสียงอัลตราโซนิก เสียงนี้จะสะท้อนออกมาหากมีวัตถุปรากฏอยู่ด้านหน้าเซ็นเซอร์อัลตราโซนิก เครื่องรับจะรับเสียงสะท้อนที่อยู่ในโมดูลเดียวกัน สัญญาณอัลตราโซนิกถูกแพร่กระจายโดยคลื่นที่มีมุม 30 องศา ดังรูปที่ 9 และมุมการวัดควรมีค่าอย่างน้อย 15 องศาเพื่อความแม่นยำสูงสุด ในกรณีนี้ วัตถุภายนอกที่ตกอยู่ภายใต้มุมการวัดนี้จะถูกรบกวนการกำหนดระยะห่างจากวัตถุ ซึ่งระยะทางถูกกำหนดโดยการวัดเวลาการเดินทางของเสียงและความเร็วเสียง

$$\text{ระยะทาง} = \frac{\text{เวลา} \times \text{ความเร็วเสียง}}{2} \quad (2.11)$$

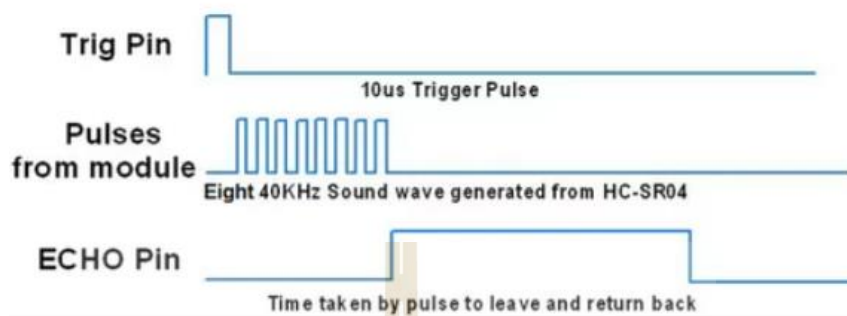
การสะท้อนของเสียงการตรวจจับจะแตกต่างกันไปตามการกำหนดค่า ดังรูปที่ 2.21 วัตถุแบน ได้แก่ ขางเหลว ก่องพลาสติก วัตถุเรียงเป็นแนว กระจ่าง ขวด ร่างกายมนุษย์ และ วัตถุที่เป็นเม็ด แน่ หิน ถ่านหิน



รูปที่ 2.21 การสะท้อนของเสียงในวัตถุที่แตกต่างกัน

ในการสร้างเสียงอัลตราโซนิก จำเป็นต้องกระตุ้น Trig Pin ให้สูงที่สุดอย่างน้อย 10 ไมโครวินาที จากนั้นโมดูลจะเริ่มส่งอัลตราซาวด์ระเบิดโซนิก 8 ครั้ง จากโมดูลที่ 40 KHz โดยจะรับทางเครื่องรับที่เรียกว่า Echo pin และจะคำนวณเวลาเอาท์พุทเพื่อวัดระยะทาง เพื่อป้องกันการรบกวนซึ่งกันและกัน

ระหว่างเซนเซอร์อัลตราโซนิกสามารถใช้งานแบบซิงโครนัสได้ โดยการปล่อยคลื่นอัลตราโซนิกจากเซนเซอร์แต่ละตัวพร้อมกัน



รูปที่ 2.22 แสดงโมดูล Timing Diagram

## 2.5 การประมวลผลภาพ

### 2.5.1 ทฤษฎีสี

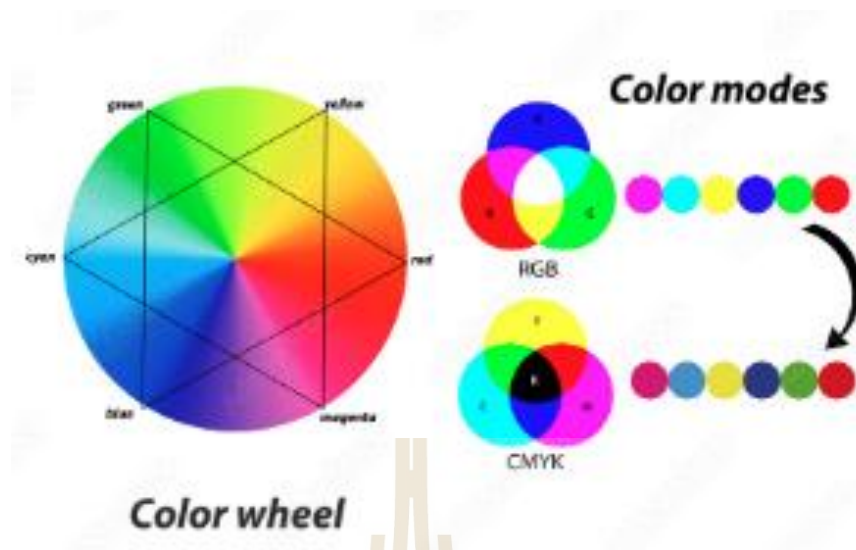
ทฤษฎีสีเป็นพื้นฐานสำคัญของการประมวลผลภาพ เนื่องจากมันให้หลักการและโมเดลสำหรับการทำความเข้าใจและปรับแต่งสีในภาพ ด้านล่างนี้เป็นวิธีการต่าง ๆ ที่ทฤษฎีสีเกี่ยวข้องกับการประมวลผลภาพ

#### 2.5.1.1 โมเดลสี (Color Models)

โมเดลสีที่แตกต่างกันถูกนำไปใช้ในการประมวลผลภาพเพื่อแสดงสีในรูปแบบที่เหมาะสมสำหรับการใช้งานต่าง ๆ

- 1) RGB (แดง, เขียว, น้ำเงิน)

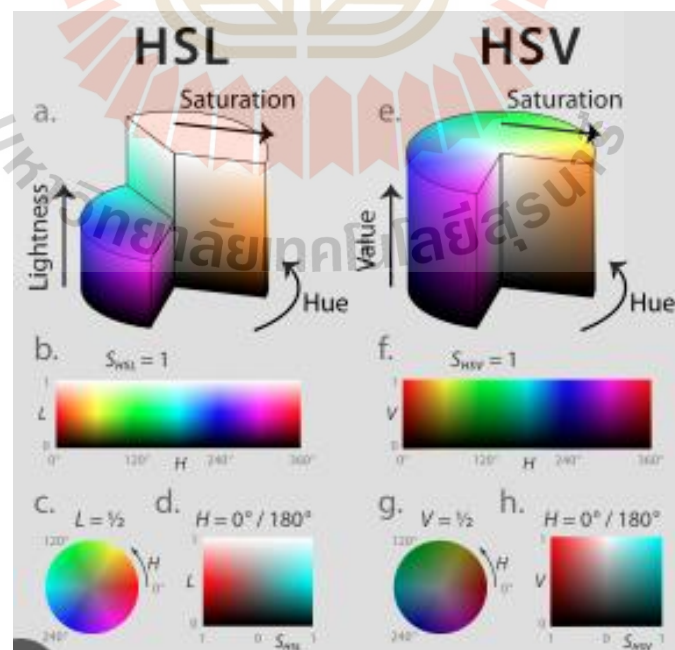
โมเดลสีที่นิยมใช้มากที่สุดในภาพดิจิทัลและหน้าจอแสดงผล สีแต่ละสีจะแสดงเป็นการผสมกันขององค์ประกอบสีแดง สีเขียว และสีน้ำเงิน งานประมวลผลภาพ เช่น การกรอง การปรับแต่ง และการวิเคราะห์ มักเริ่มต้นจากภาพ RGB



รูปที่ 2.23 แสดงการเปรียบเทียบ RGB และ CMYK

## 2) HSV (สี, ความอิ่มตัว, ค่า)

โมเดลสีทรงกระบอกที่แยกความส่องสว่างของภาพ (ความเข้ม) ออกจากความอิ่มตัวของสี (ข้อมูลสี) ซึ่งมีประโยชน์สำหรับงานต่าง ๆ เช่น การแบ่งส่วนตามสีและการตรวจจับวัตถุ โดยที่ข้อมูลสีจะต้องแยกออกจากสภาพแสง



รูปที่ 2.24 แสดงการเปรียบเทียบ HSV และ HSL

### 3) CMYK (ฟ้า, ม่วง, เหลือง, ดำ)

โมเดลสีแบบลบ (Subtractive Color Model) ที่ใช้ในการพิมพ์สี การทำความเข้าใจโมเดลนี้มีความสำคัญสำหรับงานที่เกี่ยวข้องกับการพิมพ์และการเตรียมภาพสำหรับสื่อสิ่งพิมพ์

### 4) Lab Color Space

ช่องว่างของสีตรงกันข้าม (Color-opponent space) ที่มีมิติ L สำหรับความสว่าง (Lightness) และ a และ b สำหรับมิติของสีตรงกันข้าม โมเดลนี้มีความสม่ำเสมอต่อการรับรู้ของมนุษย์มากขึ้น ซึ่งหมายความว่า การเปลี่ยนแปลงในช่องว่างของสีนั้นสัมพันธ์กับการมองเห็นของมนุษย์ได้ใกล้เคียงกัน มันมีประโยชน์สำหรับการแก้ไขสีและการแบ่งส่วนตามสี

#### 2.5.1.2 การแก้ไขและปรับแต่งสี (Color Correction and Enhancement)

ทฤษฎีสีช่วยในการปรับแต่งและปรับปรุงสีในภาพเพื่อให้ดูเป็นธรรมชาติมากขึ้นหรือเพื่อเน้นคุณสมบัติเฉพาะเจาะจง

##### 1) การปรับสมดุลแสงขาว (White Balance Adjustment)

แก้ไขโทนสีที่เกิดจากสภาพแสงที่แตกต่างกันเพื่อให้ภาพดูเป็นธรรมชาติมากขึ้น ซึ่งเกี่ยวข้องกับการปรับสมดุลสีตามอุณหภูมิของแหล่งกำเนิดแสง

##### 2) การแก้ไขแกมมา (Gamma Correction)

ปรับความส่องสว่างของสีเพื่อรองรับการตอบสนองที่ไม่เป็นเส้นตรงของอุปกรณ์แสดงผลและการมองเห็นของมนุษย์ ทำให้ภาพดูเป็นธรรมชาติมากขึ้น

##### 3) การปรับค่าฮิสโตแกรมให้เท่ากัน (Histogram Equalization)

เพิ่มความคมชัดของภาพโดยการกระจายค่าความเข้มที่พบมากที่สุด ซึ่งสามารถนำไปใช้กับช่องสัญญาณความส่องสว่างในช่องว่างของสีที่แยกความส่องสว่างและความอิ่มตัวของสี (เช่น HSV หรือ Lab)

#### 2.5.1.3 การแบ่งส่วนตามสีและการตรวจจับวัตถุ (Color Segmentation and Object Detection)

ข้อมูลสีมักถูกนำไปใช้ในการแบ่งภาพออกเป็นส่วน ๆ หรือเพื่อตรวจจับวัตถุเฉพาะเจาะจง

##### 1) การแบ่งธรรณภาคในช่องว่างของสี (Thresholding in Color Spaces)

การแยกวัตถุออกจากพื้นหลังตามสี ตัวอย่างเช่น การตรวจจับบริเวณสีเขียวในภาพพืชพรรณกรรมโดยใช้ช่องว่างของสี HSV

## 2) การจัดกลุ่ม (Clustering)

การจัดกลุ่มพิกเซลเป็นกลุ่มตามค่าสีของมัน เทคนิคเช่น K-means clustering สามารถนำมาใช้ในการแบ่งภาพออกเป็นส่วนๆ ที่มีสีใกล้เคียงกัน

ทฤษฎีสีเป็นพื้นฐานสำคัญสำหรับเทคนิคและวิธีการต่าง ๆ ในการประมวลผลภาพ ด้วยการเข้าใจว่าสีถูกแทนค่า รับรู้ และ ปรับแต่งอย่างไร เราจึงสามารถออกแบบ อัลกอริทึมการประมวลผลภาพเพื่อปรับปรุง วิเคราะห์ และแปลงภาพได้อย่างมีประสิทธิภาพ ทั้งในเชิงการมองเห็นและการคำนวณ

### 2.5.2 ประเภทของภาพตามความละเอียดของสี (Color Depth)

ความละเอียดของสี (Color Depth) คือจำนวนบิตที่ใช้ในการแสดงสีของพิกเซล เดียวภายในภาพ มันกำหนดจำนวนสีหรือเฉดสีที่สามารถแสดงได้ในภาพ ยิ่งความละเอียดของสีมาก ภาพก็จะมีสีที่สมจริงและซับซ้อนมากขึ้น มาดูประเภทหลักของภาพตามความละเอียดของสีกัน

#### 2.5.2.1 ภาพขาวดำ (Binary Images) (1 บิต)

ลักษณะ พิกเซลแต่ละจุดในภาพขาวดำแสดงด้วย 1 บิต ซึ่งหมายความว่า มันจะเป็นสีดำ (0) หรือสีขาว (1) ความละเอียดของสี 1 บิตต่อพิกเซล จำนวนสี 2 สี (ดำและขาว)

#### 2.5.2.2 ภาพขาวเทา grayscale (Grayscale Images) (8 บิต)

ลักษณะ พิกเซลแต่ละจุดในภาพเฉด grayscale แสดงด้วย 8 บิต อนุญาตให้มีเฉดสีเทาที่แตกต่างกันได้ 256 ระดับ ตั้งแต่ดำไปจนถึงขาว ความละเอียดของสี 8 บิตต่อพิกเซล จำนวนสี 256 เฉดสีเทา

#### 2.5.2.3 ภาพสี (Color Images)

ภาพสีแบบ indexed color (8 บิตขึ้นไป) ลักษณะ ภาพสีแบบ indexed color ใช้พาเลทสี (color map) โดยค่าของแต่ละพิกเซลจะเป็นดัชนี (index) ซึ่งไปยังสีในพาเลทนั้น ทั่วไปแล้วพาเลทจะมีสีได้ 256 สี (8 บิต) แต่ก็สามารถปรับเปลี่ยนได้ ความละเอียดของสี ปกติ 8 บิต ต่อพิกเซล แต่สามารถมากกว่านั้นได้ จำนวนสี สูงสุด 256 สี (สำหรับภาพ indexed color 8 บิต)

การประมวลผลภาพ (Image Processing) คือการใช้คอมพิวเตอร์วิเคราะห์ และปรับแต่งภาพดิจิทัลเพื่อปรับปรุงคุณภาพหรือดึงข้อมูลที่ต้องการออกมา เทคนิคการประมวลผล ภาพต่าง ๆ มีดังนี้

### 2.5.3 การปรับแต่งคุณภาพของภาพ

#### 2.5.3.1 ปรับแต่งค่าฮิสโตแกรม (Histogram Equalization)

เทคนิคนี้ใช้ปรับปรุงความคมชัด (contrast) ของภาพ โดยการกระจายค่า ความเข้มของแสง (intensity) ที่พบมากที่สุด คำนวณฮิสโตแกรม หาความถี่ของแต่ละระดับความเข้ม ของแสงในภาพ ฟังก์ชันการกระจายเชิงสะสม (Cumulative Distribution Function - CDF)



คำนวณค่า CDF จากฮิสโตแกรม การแม็พค่าความเข้มใช้ CDF เพื่อแปลงค่าความเข้มของแสงเดิมไปเป็นค่าใหม่ กระจายค่าพิกเซลเพื่อใช้ประโยชน์จากช่วงความเข้มทั้งหมด

### 2.5.3.2 การกรองภาพ (Filtering)

ชนิดของฟิลเตอร์ ฟิลเตอร์ลดสัญญาณรบกวน (Smoothing Filters) ลดสัญญาณรบกวนและทำให้ภาพเรียบเนียน ตัวอย่าง ฟิลเตอร์เกาส์ (Gaussian Filter) ใช้ฟังก์ชันเกาส์เพื่อลดสัญญาณรบกวนในภาพ ทำให้ภาพเรียบเนียน ฟิลเตอร์ค่ามัธยฐาน (Median Filter) แทนค่าของแต่ละพิกเซลด้วยค่ามัธยฐานของพิกเซลข้างเคียง มีประสิทธิภาพในการกำจัดสัญญาณรบกวน ฟิลเตอร์เพิ่มความคมชัด (Sharpening Filters) เพิ่มความคมชัดของขอบและรายละเอียดเล็ก ๆ ในภาพ ตัวอย่าง ฟิลเตอร์ลาปลาซ (Laplacian Filter): ใช้อนุพันธ์อันดับสองของภาพเพื่อเพิ่มความคมชัดของขอบ การเบลอแบบ Unmasking (Unsharp Masking): เพิ่มความคมชัดของขอบโดยการลบภาพที่เบลอจากภาพต้นฉบับ

### 2.5.3.3 การตรวจจับขอบภาพ (Edge Detection)

เทคนิคนี้ใช้ระบุการเปลี่ยนแปลงความเข้มของแสงที่สำคัญ เพื่อกำหนดขอบเขตของวัตถุภายในภาพ ตัวดำเนินการโซเบล (Sobel Operator): ใช้ขนาดของการไล่ระดับความเข้ม (gradient magnitude) เพื่อหาขอบภาพ ตัวตรวจจับขอบภาพแบบคานนี่ (Canny Edge Detector) อัลกอริทึมหลายขั้นตอนที่ให้การตรวจจับขอบภาพที่แม่นยำโดยการลดสัญญาณรบกวน การหาการไล่ระดับความเข้ม จากนั้นใช้การยับยั้งแบบไม่เกินค่าสูงสุด (non-maximum suppression) และการติดตามขอบภาพด้วยฮิสเทรีซิส (hysteresis)

### 2.5.3.4 การปรับแต่งความคมชัด (Contrast Adjustment)

การปรับแต่งความคมชัดเกี่ยวข้องกับการเปลี่ยนแปลงความแตกต่างระหว่างส่วนที่มืดที่สุดและสว่างที่สุดของภาพ การยืดเส้นตรงของคอนทราสต์ (Linear Contrast Stretching) ขยายช่วงของค่าความเข้มเพื่อครอบคลุมสเปกตรัมทั้งหมด ปรับแต่งค่าฮิสโตแกรมแบบปรับได้ตามพื้นที่ (Adaptive Histogram Equalization - AHE) ใช้การปรับแต่งค่าฮิสโตแกรมในบริเวณเล็ก ๆ (ไทล์) ของภาพ เพื่อปรับปรุง

## 2.6 ปรัชน์วรรณกรรมที่เกี่ยวข้อง

### 2.6.1 งานวิจัยที่เกี่ยวข้องกับการระบบควบคุม AGV

(Munadi M et al) การพัฒนาโมเดลรถยนต์ที่นำทางด้วยระบบอัตโนมัติ (AGV) ที่มีระบบนำทางเชื่อมต่ออย่างอัจฉริยะ โดย AGV นี้ใช้คอมพิวเตอร์วิชั่นและระบบควบคุม fuzzy สำหรับระบบช่วยในการเคลื่อนที่บนถนน อินพุตที่ใช้ในระบบควบคุม fuzzy โลจิก คือ ค่าการเปลี่ยนแปลงของเส้นทาง AGV สำหรับเส้นทางซ้ายและขวา ระบบนำทางใช้กล้องที่มีความไวสูงในการตรวจจับแสง

การมีความสว่างที่ต่ำหรือสูงเกินไปจะมีผลต่อประสิทธิภาพของระบบควบคุม ช่วงความสว่างที่แน่นอนจะมีผลต่อประสิทธิภาพของระบบช่วยในการถนนเป็นตัวควบคุม ถนนที่มีเส้นทางทางซ้ายและขวาถูกสร้างขึ้นเพื่อทดสอบประสิทธิภาพของระบบควบคุมการทำนายทางของคอมพิวเตอร์วิชั่น ผลการทดสอบแสดงให้เห็นว่าความสว่างที่เหมาะสมสำหรับระบบช่วยการถนนที่พัฒนาขึ้นคือ 110 ถึง 150 lux และ AGV สามารถทำการตามเส้นทางได้อย่างประสบความสำเร็จภายใต้ความสว่างเหล่านี้ ถึงแม้ว่าความผิดพลาดจะยังเกิดขึ้นบ้าง

(Sierra-Garcia et al) Automated Guided Vehicle (AGV) ได้รับการพัฒนาอย่างรวดเร็ว ระบบเหล่านี้มีปัญหาคอมพิวเตอร์ที่สอดคล้องกัน 2 ปัญหาหลัก คือการควบคุมความเร็วในทิศทางตรง ที่จำเป็นต้องรักษาการเคลื่อนที่ให้เป็นไปตามต้องการ เช่น ประสิทธิภาพการทำงานและเวลาหมุน และการควบคุมการติดตามเส้นทางที่จำเป็นเพื่อให้ได้ความแม่นยำที่เหมาะสมในการทำการโหลดและถอดของ ขณะที่เส้นทางมีความสั้นหรือมีการเปลี่ยนแปลงที่รุนแรง และการควบคุมการติดตามเส้นทางที่ท้าทายมากขึ้น ในกรณีนี้ กลยุทธ์ควบคุมที่ขั้นสูง การใช้เทคนิคที่ฉลาด เช่น machine learning (ML) สามารถเป็นประโยชน์ ดังนั้น การควบคุมฉลาดแบบผสมที่รวมกันระบบควบคุมที่ใช้ reinforcement learning (RL) กับตัวควบคุม PI ดั้งเดิมเพื่อแก้ไขทั้งสองปัญหาการควบคุมพร้อมกัน ในด้านหนึ่ง PI ถูกใช้ควบคุมความเร็วของล้อแต่ละล้อ ในด้านอื่น ๆ ค่าอ้างอิงของตัวควบคุมเหล่านี้ถูกคำนวณโดย RL เพื่อลดความผิดพลาดในการติดตามเส้นทางและรักษาความเร็วทิศทาง ซึ่งเปรียบเทียบกับตัวควบคุม PID ที่ถูกปรับแต่งโดยอัลกอริทึม ทำให้ AGV เรียนรู้วิธีการปรับปรุงการติดตามเส้นทางอย่างปรับตัวและในทางที่ดี นอกจากนี้ RL ยังช่วยให้รถยนต์เรียนรู้วิธีการปรับปรุงการติดตามเส้นทางในทางที่เป็นการปรับตัวและทำให้ AGV สามารถเผชิญกับการรบกวนหรือพารามิเตอร์ระบบที่ไม่ทราบที่อาจเปลี่ยนแปลงเนื่องจากการเสียดสีและความเสื่อมถอยของส่วนประกอบกลไกของ AGV การทดลองจำลองของกลยุทธ์ควบคุม ประสิทธิภาพของตัวควบคุมการติดตามของ RL เทียบกับ PID ที่ถูกปรับให้ได้ผลผิดพลาดน้อยกว่าร้อยละ 70 และค่าผิดพลาดสูงสุดเฉลี่ยก็ต่ำกว่าร้อยละ 48

(Liao, Xiaofei et al) การสร้างเส้นทางที่เหมาะสมจากตำแหน่งเริ่มต้นไปยังตำแหน่งเป้าหมายสำหรับ AGV ในสภาพแวดล้อมที่ซับซ้อน ทำให้เป็นปัญหาที่ยากต่อการสร้างเส้นทางอย่างที่ต้องการ เรานำเสนอเทคโนโลยี reinforcement learning เพื่อแก้ไขปัญหาที่ยากต่อการสร้างเส้นทาง อัลกอริทึม Sarsa ที่ใช้กลยุทธ์ simulated annealing สามารถนำ AGV สร้างเส้นทางที่เหมาะสม และเพิ่มอัตราความสำเร็จ สำหรับปัญหาที่อัลกอริทึม reinforcement learning ดั้งเดิมไม่สามารถประมวลผลข้อมูลเพียงพอในกรณีของ state space มีขนาดใหญ่ วิธี field potential ที่ผสมกับอัลกอริทึม deep q-network ถูกนำเสนอสำหรับการสร้างเส้นทางของ AGV อัลกอริทึมนี้สามารถนำ AGV ไปวาดเส้นทางที่เหมาะสมได้อย่างมีประสิทธิภาพ และแก้ปัญหาที่

อัลกอริทึม reinforcement learning ดั้งเดิมไม่สามารถจัดการกับพื้นที่ที่ซับซ้อนได้ อัลกอริทึมเหล่านี้ถูกนำไปใช้ในระบบวาดเส้นทางของ AGV เพื่อจำลองสถานะการเคลื่อนไหวของ AGV เดียวจากจุดเริ่มต้นไปยังจุดเป้าหมาย และยืนยันว่าอัลกอริทึมของเราสามารถทำให้การวาดเส้นทางของ AGV ได้อย่างมีประสิทธิภาพและเพิ่มประสิทธิภาพในการโลจิสติกส์คลังสินค้า

(Chen Yuanyuan et al) ทำการวิเคราะห์หลักการของตัวควบคุม Fuzzy PID และสร้างโมเดล เพื่อให้ได้ความสัมพันธ์ที่แท้จริงระหว่างรูปแบบการเคลื่อนไหวและความเร็วของล้อขับเคลื่อนแต่ละล้อ ทำการใช้อัลกอริทึมที่วัดและคืนกลับจากระบบควบคุม Fuzzy PID คือการใช้อัลกอริทึมขับเคลื่อนและค่าตำแหน่งขับเคลื่อนของ AGV เป็นอินพุตของคอนโทรลเลอร์ Fuzzy โดยใช้เทคโนโลยีการแก้ไขค่าผิดพลาดของ Fuzzy เพื่อปรับปรุงเส้นทางของ AGV และให้ความมั่นคงของการทำงานของ AGV ผ่านเทคโนโลยีการควบคุมการเคลื่อนไหวร่วมกัน ซอฟต์แวร์ MATLAB สร้างโมเดลจำลองร่วมของระบบควบคุม Fuzzy PID ของ AGV ผลลัพธ์แสดงให้เห็นว่า ผ่านการทดสอบการควบคุมด้วย PID และการแก้ไขด้วย Fuzzy PID ระบบควบคุม Fuzzy PID สามารถแก้ไขข้อบกพร่องได้มีประสิทธิภาพถึง 47.41% ในทดสอบด้านการเคลื่อนที่ทางไหนที่ประสิทธิภาพในการติดตามเส้นทางเพิ่มขึ้นถึง 30.76% ประสิทธิภาพในการแก้ไขระยะทางเพิ่มขึ้นถึง 69.56% ประสิทธิภาพในการแก้ไขมุมเพิ่มขึ้นถึง 69.56% และประสิทธิภาพเฉลี่ยเพิ่มขึ้นถึง 53.44% ผ่านการเปรียบเทียบทั้งสอง จึงสรุปได้ว่า ตัวควบคุม Fuzzy ที่ออกแบบในบทความนี้มีข้อดีในด้านความเร็วในการตอบสนอง และประสิทธิภาพในการแก้ไขการเบนของทาง

(Myungwook Park et al) กล่าวถึงระบบควบคุมการเคลื่อนไหวของรถยนต์อัตโนมัติในการติดตามเส้นทาง ระบบควบคุมการเคลื่อนไหวประกอบด้วยตัวติดตามเส้นทางและไดเรกทอรีพริมาทีฟ ตัวติดตามเส้นทางสร้างมุมการเคลื่อนไหวที่ต้องการโดยใช้ระยะเป้าหมายข้างหน้า ทิศทางของรถยนต์เพื่อปรับให้มุมการเคลื่อนไหวที่ต้องการ เมื่อเทียบกับวิธีการ pure pursuit ที่มีข้อดีในการลดการตัดมุมเคลื่อนไหวที่เป็นจุดอ่อนของวิธีการ pure pursuit ทั่วไป ตัวควบคุมการเคลื่อนไหวควบคุมการทำงานของรถเคลื่อนไหวเพื่อติดตามมุมการเคลื่อนไหวที่ต้องการ เครื่องยนต์เซอร์โวตั้งถูกติดตั้งเพื่อควบคุมพวงมาลัย และมันสามารถส่งกำลังทางการเคลื่อนไหวโดยใช้สายพานและลูกบิด เราออกแบบตัวควบคุมการเคลื่อนไหวที่ถูกนำมาใช้กับตัวควบคุมป้อน-อินที่กริดดิเวอร์เรนเซียล (PID) อย่างไรก็ตาม เนื่องจากมีแบนด์เดด ประสิทธิภาพในการติดตามเส้นทางและความมั่นคงของรถยนต์อัตโนมัติลดลง เพื่อที่จะเอาชนะแบนด์เดด พัฒนาตัวชดเชยแบนด์เดด ผลลัพธ์จากตัวชดเชยแบนด์เดดทำให้ประสิทธิภาพในการติดตามเส้นทางและความมั่นคงของรถยนต์อัตโนมัติเพิ่มขึ้น

## 2.6.2 งานวิจัยที่เกี่ยวข้องกับการตรวจจบบัตุ

(Shih-Hsuan Hung et al) นำเสนออัลกอริทึมใหม่เพื่อสร้างรหัส QR ด้วยเทคนิค micrography ซึ่งเป็นกราฟิกที่สามารถอ่านได้ทางเครื่องจักร โดยซ่อนรหัส QR ภายในภาพ

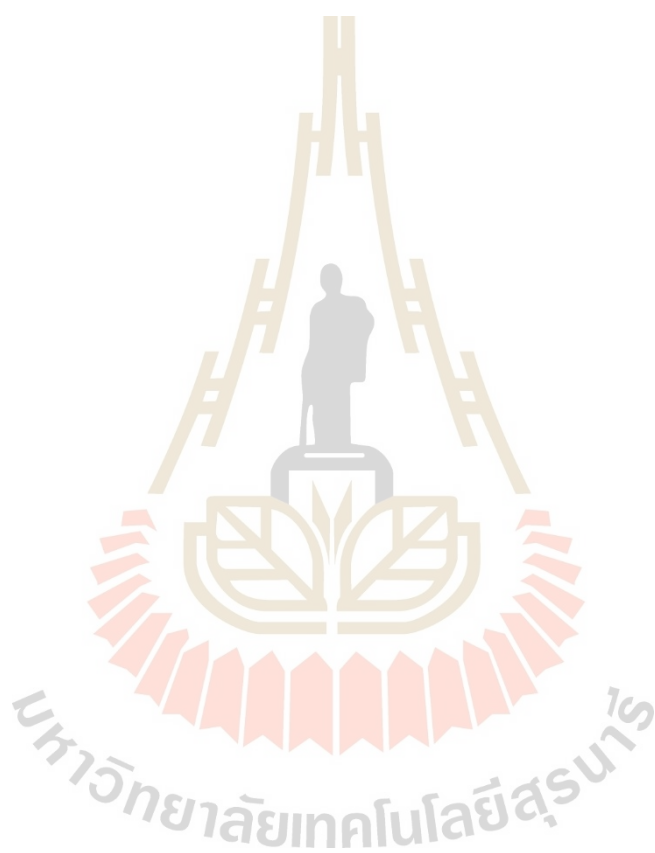
micrography ทำให้เกิดลักษณะที่ไม่เหมือนในวิธีการที่มีอยู่แล้วที่ใช้ร่วมกับภาพธรรมชาติหรือภาพระบายสี เราใช้ลักษณะที่ไม่มีความเทียบเท่าของ micrography เพื่อสร้างแบบจำลองที่สามารถโค้งงอได้เฉียบขาดของตัวหนังสือแต่ละตัวและปรับน้ำหนักแบบอักษรเพื่อให้เหมาะสมกับการซ่อนรหัส QR ภายใน micrography ทั้งกระบวนการนี้จะได้รับการควบคุมโดยชุดของ metric คุณภาพทางสายตาที่ถูกออกแบบมาเฉพาะสำหรับ micrography ร่วมกับมาตรการคุณภาพ QR code ที่ใหม่ที่มีมุ่งเน้นการสร้างความสัมพันธ์ระหว่างความเหมือนทางสายตาและความแข็งแกร่งในการถอดรหัสมาตรการคุณภาพ QR code ที่เสนอนี้ขึ้นอยู่กับแบบจำลองทางนิวเมริกที่เรียนรู้จากการถอดรหัสทดลองโดยใช้เครื่องถอดรหัสที่นิยมกับรหัส QR code เทียบเท่ากับรหัส QR code ที่ถูกสร้างขึ้นจากภาพที่ซ่อนรหัส ผลทดลองแสดงให้เห็นถึงประสิทธิภาพของวิธีการสร้าง micrography QR code ที่มีคุณภาพสูงจากหลายแหล่งทดสอบที่แตกต่างกัน

(Yiyang Wu et al) การแก้ปัญหาการติดตามเส้นทางของ AGV ในระบบติดตั้งและปรับแต่งโมดูลออปติคอลที่มีประสิทธิภาพสูง ได้ทำการออกแบบกลยุทธ์ควบคุม pure pursuit ที่มีการเปลี่ยนแปลงตามระยะเป้าหมายข้างหน้าตามความเร็วของ AGV และความโค้งของเส้นทางอ้างอิง โดยตั้งแบบจากรูปร่างของ AGV และ pure pursuit model ก่อนที่จะสร้างฟังก์ชันผสมด้วยการติดตามและความเสถียร ใช้อัลกอริทึม Particle Swarm Optimization (PSO) เพื่อปรับแต่งระยะเป้าหมายข้างหน้าของ pure pursuit model ของ AGV ในเงื่อนไขการทำงานที่หลากหลาย ระหว่างกระบวนการติดตาม AGV เลือกระยะที่ดูล่วงหน้าที่เหมาะสมตามความโค้งของเส้นทางอ้างอิงและความเร็วปัจจุบัน ผลการทดลองจำลองแสดงให้เห็นว่า กลยุทธ์ควบคุม pure pursuit ที่ปรับปรุงที่รวมข้อมูลความโค้งของเส้นทางอ้างอิงสามารถเพิ่มความสามารถในการปรับตัวของ AGV เมื่อติดตามเส้นทางที่ซับซ้อน ให้ความมั่นคงของการติดตามและความมั่นคงของการทำลาย

## 2.7 สรุป

จากการศึกษาทฤษฎีและปริทัศน์วรรณกรรมที่เกี่ยวข้องพบว่า การควบคุมการบังคับเลี้ยวของ AGV มีหลากหลายวิธี เมื่อเปรียบเทียบกันแล้ว พบว่าทฤษฎี pure pursuit model เป็นพื้นฐานของการหาองศาเพื่อได้มุมบังคับเลี้ยวที่แม่นยำไปยังจุดเป้าหมาย และการเคลื่อนที่ติดตามวัตถุ การตรวจจับวัตถุเป็นสิ่งที่สำคัญ เพราะระยะเวลาที่ใช้ในการตรวจจับนั้น จะส่งผลโดยตรงกับการประมวลผล และสั่งการบังคับเลี้ยวเพื่อติดตาม AGV ตัวนำ ซึ่งวัตถุที่จะใช้เป็นเป้าหมายจำเป็นต้องตรวจจับได้อย่างรวดเร็ว และแม่นยำ โดยที่มีสิ่งรบกวนทางด้านสิ่งแวดล้อมแสง ซึ่ง QR code เป็นวัตถุที่เหมาะสมที่จะใช้เป็นวัตถุที่ติดอยู่กับ AGV ตัวนำ เพราะมีความรวดเร็วในการตรวจจับและไม่ถูกรบกวนจากสิ่งแวดล้อมแสงได้ดี ดังนั้น งานวิจัยนี้จึงทำการออกแบบโปรแกรมติดตามวัตถุที่เป็น QR code ที่ถูกติดอยู่กับ AGV ตัวนำ โดยมีกล้องเป็นตัวตรวจรู้ และใช้ทฤษฎีของ pure pursuit

model ในการคำนวณมุมบังคับเลี้ยวของ AGV ตัวตาม และร่วมกับเซ็นเซอร์อัลตราโซนิกเป็นตัวตรวจ  
รู้เพื่อช่วยวัดระยะห่างระหว่างตัวนำและตัวตาม ให้เคลื่อนที่ติดตามด้วยระยะปรับค่าได้ ซึ่งจะ  
สามารถลดต้นทุนของ AGV ที่มีราคาสูง โดยตัวตามมีหน้าที่แค่เคลื่อนที่ตามตัวนำ โดยลดเซ็นเซอร์ที่  
ไม่จำเป็นออก แต่ยังสามารถทำงานแบกรับน้ำหนักได้เท่ากับหรือมากกว่า AGV ตัวนำ

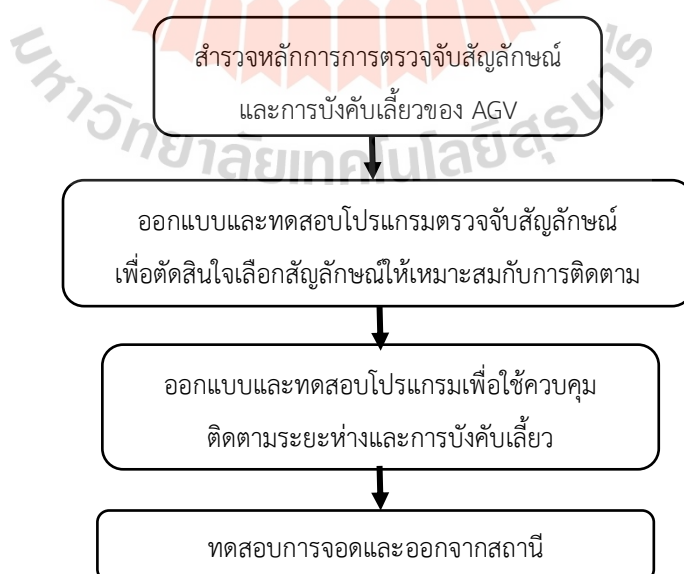


## บทที่ 3

### วิธีดำเนินการวิจัย

#### 3.1 บทนำ

ในงานวิจัยนี้ดำเนินการออกแบบโปรแกรมระบบควบคุมที่ใช้กับ AGV ตัวตาม เคลื่อนที่ติดตาม AGV ตัวนำ ใช้หลักการติดตามสัญลักษณ์ที่ติดอยู่กับ AGV ตัวนำ ใช้กล้องเป็นตัวตรวจรู้ เป็นอินพุตที่ใช้ในการการควบคุมการบังคับเลี้ยวล้อหน้า และใช้เซนเซอร์วัดระยะเป็นตัวช่วยในการวัดระยะ เพื่อรักษา ระยะห่างของ AGV ตัวนำในระยะที่กำหนด เพื่อลดจำนวนเซ็นเซอร์ที่มีราคาสูงและจำนวนมาก โดยผู้วิจัยนำเสนอการสร้างแบบจำลองทางคณิตศาสตร์เพื่อคำนวณองศาของมุมบังคับเลี้ยว ให้เคลื่อนที่ไป ยังเป้าหมายได้อย่างแม่นยำ และศึกษาเปรียบเทียบสัญลักษณ์ที่ติดอยู่กับ AGV ตัวนำเพื่อใช้ในการอ่าน และติดตาม โดยออกแบบการทดลองที่สภาพแวดล้อมของความเข้มแสงที่แตกต่างกัน ทดสอบระยะ ของมุมที่สามารถตรวจจับสัญลักษณ์ที่อยู่ทางซ้ายหรือทางขวาของ AGV ตัวตาม ทดสอบความแม่นยำ ของการวัดระยะด้วยเซนเซอร์วัดระยะเพื่อเปรียบเทียบกัน รวมถึงการทดสอบโปรแกรมการควบคุมการ บังคับเลี้ยวและการรักษาระยะกับรถ AGV ตัวนำ เพื่อทดสอบการเคลื่อนที่ติดตาม ในความเร็วของการ เคลื่อนที่ที่สอดคล้องกัน อีกทั้งยังทดสอบการเคลื่อนที่ออกจากสถานีจอด และการเข้าจอดเมื่อเสร็จสิ้น ภาระกิจติดตาม โดยภาพรวมของการดำเนินงานวิจัยแสดงในรูปที่ 3.1



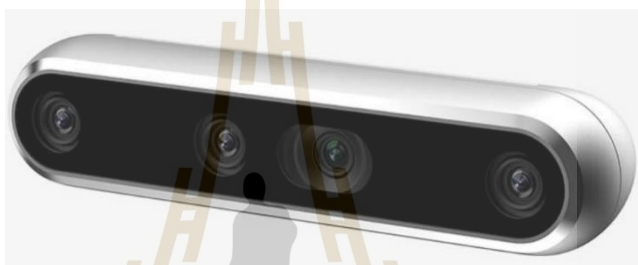
รูปที่ 3.1 แสดงการแผนดำเนินการวิจัย

### 3.2 ชุดทดสอบการตรวจจับสัญลักษณ์

งานวิจัยนี้ออกแบบโปรแกรมเพื่อควบคุมการเคลื่อนที่ของ AGV ตัวตาม โดยใช้สัญลักษณ์ติดตั้งที่ AGV ตัวนำ โดยใช้กล้องเป็นตัวตัวรู้ และใช้เซนเซอร์วัดระยะห่างระหว่างตัวนำและตัวตาม โดยมีอุปกรณ์การทดสอบเพื่อออกแบบโปรแกรมดังนี้

#### 3.2.1 กล้อง intel realsense depth camera d455

ระยะการมองเห็น 0.6 เมตร ถึง 6 เมตร เป็นกล้องสเตอริโอโคป ให้ความละเอียด 1280 x 720 พิกเซล ที่เฟรมเรทสูงสุด 90 เฟรมเรทต่อวินาที ขนาดของกล้อง 124 x 26 x 29 มิลลิเมตร ช่องเชื่อมต่อ USB type C ดังรูปที่ รูปที่ 3.2



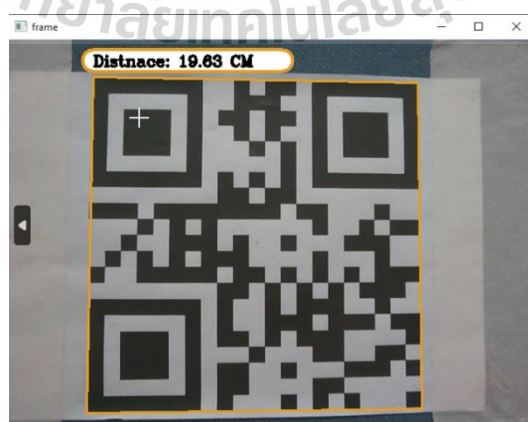
รูปที่ 3.2 กล้อง intel realsense depth camera d455

#### 3.2.2 สัญลักษณ์

เป็นสัญลักษณ์ที่ใช้ติดกับ AGV ตัวนำ เพื่อใช้วัดระยะและมุมเมื่อเทียบกับตัวตาม

##### 3.2.2.1 QR code

QR code ขนาดการเก็บข้อมูลเวอร์ชัน 1 21 x 21 ขนาด 26 x 21 เซนติเมตร



รูปที่ 3.3 QR code ที่ใช้ในการทดลอง

### 3.2.2.2 ภาพสีฟ้า

ภาพสีฟ้ามีขนาด 26 x 21 เซนติเมตร มีค่า RGB (5,100,250) ดังรูปที่ 3.4



รูปที่ 3.4 Color ที่ใช้ในการทดลอง

### 3.2.3 mini pc Dell Optiplex 7060 Micro

ใช้เป็นอุปกรณ์สื่อสารรับส่งข้อมูลของกล้อง ดังรูปที่ 3.5



รูปที่ 3.5 Dell Optiplex 7060 Micro



### 3.3 ชุดควบคุมสั่งงานการเคลื่อนที่ของ AGV ตัวตาม

#### 3.3.1 บอร์ด Arduino Uno

Arduino Uno เป็นหนึ่งในบอร์ดไมโครคอนโทรลเลอร์ที่ได้รับความนิยมมากที่สุดใครอบครัว Arduino ซึ่งถูกออกแบบมาเพื่อการเรียนรู้และการพัฒนางานทางด้านอิเล็กทรอนิกส์และโปรแกรมมิ่ง โดยเฉพาะสำหรับผู้เริ่มต้น นี่คืข้อมูลทีควรทราบเกี่ยวกับบอร์ด Arduino Uno

ข้อมูลทางเทคนิคของ Arduino Uno

ไมโครคอนโทรลเลอร์ (Microcontroller) : ATmega328P

แรงดันไฟฟ้าในการทำงาน (Operating Voltage) : 5V

แรงดันไฟฟ้าอินพุต (Input Voltage) : 7-12V (แนะนำ), 6-20V (ขีดจำกัด)

พินดิจิทัล I/O (Digital I/O Pins) : 14 พิน (ซึ่ง 6 พินสามารถใช้เป็นพิน PWM ได้)

พินอนาล็อกอินพุต (Analog Input Pins) : 6 พิน

กระแสสูงสุดต่อพิน I/O (DC Current per I/O Pin) : 20 mA

กระแสสูงสุดสำหรับพิน 3.3V (DC Current for 3.3V Pin) : 50 mA

แฟลชเมมโมรี่ (Flash Memory) : 32 KB (ATmega328P) โดย 0.5 KB ถูกใช้สำหรับบูต โหลดเดอร์ (Bootloader)

พอร์ตการเชื่อมต่อ USB : ใช้สำหรับการโปรแกรมและการสื่อสารกับคอมพิวเตอร์

พอร์ตการเชื่อมต่ออื่น ๆ : UART, SPI, I2C การเชื่อมต่อและการใช้งาน Arduino

พินและฟังก์ชันของ Arduino Uno :

พิน 0-13 (Digital I/O): ใช้สำหรับการรับหรือส่งสัญญาณดิจิทัล

พิน 0 (RX) และพิน 1 (TX): ใช้สำหรับการสื่อสาร Serial (UART)

พิน 3, 5, 6, 9, 10, 11 (PWM): ให้สัญญาณ PWM ที่ใช้ในการควบคุม

ความสว่างของ LED หรือความเร็วของมอเตอร์

พิน A0-A5 (Analog Input): ใช้สำหรับการอ่านค่าสัญญาณอนาล็อก

พิน 13: มี LED ในตัวที่เชื่อมต่อโดยตรง

#### 3.3.2 บอร์ด Arduino Mega

Arduino Mega 2560 เป็นบอร์ดไมโครคอนโทรลเลอร์ทีมีความสามารถและคุณสมบัติสูงกว่า Arduino Uno โดยเฉพาะในด้านจำนวนพิน I/O ทีมากกว่า เหมาะสำหรับโปรเจกต์ที่ต้องการการเชื่อมต่อกับอุปกรณ์หลายๆ ตัวหรืองานที่ต้องใช้ทรัพยากรมาก ข้อมูลทางเทคนิคของ Arduino Mega 2560

ไมโครคอนโทรลเลอร์ (Microcontroller): ATmega2560

แรงดันไฟฟ้าในการทำงาน (Operating Voltage): 5V

แรงดันไฟฟ้าอินพุต (Input Voltage): 7-12V (แนะนำ), 6-20V (ขีดจำกัด)

พินดิจิตอล I/O (Digital I/O Pins):

54 พิน (ซึ่ง 15 พินสามารถใช้เป็นพิน PWM ได้)

พินอนาล็อกอินพุต (Analog Input Pins): 16 พิน

กระแสสูงสุดต่อพิน I/O (DC Current per I/O Pin): 20 mA

กระแสสูงสุดสำหรับพิน 3.3V (DC Current for 3.3V Pin): 50 mA

แฟลชเมมโมรี่ (Flash Memory): 256 KB (ATmega2560) โดย 8 KB ถูกใช้

สำหรับบูตโหลดเดอร์ (Bootloader)

พอร์ตการเชื่อมต่อ USB: ใช้สำหรับการโปรแกรมและการสื่อสารกับคอมพิวเตอร์

พอร์ตการเชื่อมต่ออื่น ๆ: UART, SPI, I2C การเชื่อมต่อและการใช้งาน Arduino

พินและฟังก์ชันของ Arduino Mega 2560:

พิน 0-53 (Digital I/O): ใช้สำหรับการรับหรือส่งสัญญาณดิจิตอล

พิน 0 (RX) และพิน 1 (TX): ใช้สำหรับการสื่อสาร Serial (UART)

พิน 2-13, 44-46 (PWM): ให้สัญญาณ PWM ที่ใช้ในการควบคุมความสว่างของ

LED หรือความเร็วของมอเตอร์

พิน A0-A15 (Analog Input): ใช้สำหรับการอ่านค่าสัญญาณอนาล็อกจากเซนเซอร์

### 3.3.3 เซนเซอร์

เซนเซอร์ (Sensor) คือ อุปกรณ์ที่ใช้ในการตรวจจับหรือวัดค่าต่าง ๆ ในสภาพแวดล้อม เช่น อุณหภูมิ ความชื้น ความดัน แสง สี เสียง การเคลื่อนไหว การไหลของของเหลว และอื่นๆ แล้วแปลงข้อมูลเหล่านั้นเป็นสัญญาณไฟฟ้าหรือสัญญาณอื่นที่สามารถนำไปใช้ในการประมวลผลหรือต่อไปยังระบบควบคุมหรือแสดงผล

#### 3.3.3.1 Ultrasonic sensor

เซนเซอร์อัลตราโซนิก (Ultrasonic Sensor) เป็นอุปกรณ์ที่ใช้คลื่นเสียงที่มีความถี่สูงเกินกว่าช่วงที่มนุษย์สามารถได้ยิน (โดยทั่วไปจะอยู่ในช่วง 20 kHz ขึ้นไป) เพื่อวัดระยะทางหรือการตรวจจับวัตถุในสภาพแวดล้อม เซนเซอร์นี้ทำงานโดยส่งคลื่นเสียงออกไป แล้วตรวจจับคลื่นเสียงที่สะท้อนกลับมาจากวัตถุ รับแรงดันไฟฟ้ากระแสตรงเข้า 5 โวลต์ และกระแส 3 มิลลิแอมป์ ทำงานในสภาพแวดล้อมอุณหภูมิ 0 ถึง 70 องศาเซลเซียส ช่องสื่อสารเข้าพิน GPIO องศาของการตรวจวัดระยะไม่เกิน 15 องศา ระยะของการตรวจวัด 2 ถึง 4 เมตร ค่าความคลาดเคลื่อน 0.3 เซนติเมตร ขนาดของเซนเซอร์ 45 x 20 x 1.6 มิลลิเมตร หลักการทำงานของเซนเซอร์อัลตราโซนิก การส่งสัญญาณ (Transmission) เซนเซอร์จะส่งคลื่นเสียงความถี่สูงออกไปในรูปแบบของพัลส์ (pulse) ผ่านทางตัวส่ง (transmitter) ซึ่งโดยทั่วไปจะเป็นคริสตัลไพโซอิเล็กทริก (piezoelectric

crystal) การสะท้อนกลับ (Reflection) เมื่อคลื่นเสียงนี้เจอวัตถุในเส้นทาง คลื่นเสียงจะสะท้อนกลับไปยังเซนเซอร์ การรับสัญญาณ (Reception) ตัวรับ (receiver) ในเซนเซอร์จะตรวจจับคลื่นเสียงที่สะท้อนกลับมา การคำนวณระยะทาง (Distance Calculation) เซนเซอร์จะวัดเวลาที่ใช้ในการเดินทางของคลื่นเสียงจากตัวส่งไปยังวัตถุและกลับมายังตัวรับ

$$\text{ระยะทาง} = \frac{\text{ความเร็วเสียง} \times \text{เวลาเดินทาง}}{2} \quad (2.11)$$

ความเร็วของเสียงในอากาศปกติ คือ ประมาณ 343 เมตรต่อวินาที (m/s)

### 3.3.3.2 Max sona

MaxSonar เป็นผลิตภัณฑ์เซนเซอร์อัลตราโซนิกที่ผลิตโดย MaxBotix Inc. เซนเซอร์ MaxSonar ใช้สำหรับการวัดระยะทางและการตรวจจับวัตถุในหลากหลายการใช้งาน ดังรูปที่ 3.6 ทั้งในด้านอุตสาหกรรม หุ่นยนต์ ยานยนต์ และการวิจัยทางวิทยาศาสตร์ คุณสมบัติของเซนเซอร์ MaxSonar ความแม่นยำสูง (High Accuracy) เซนเซอร์ MaxSonar มีความสามารถในการวัดระยะทางได้แม่นยำและเชื่อถือได้ ช่วงการวัดที่กว้าง (Wide Measurement Range) MaxSonar มีช่วงการวัดที่กว้างตั้งแต่ไม่กี่เซนติเมตรถึงหลายเมตร ทำให้เหมาะสมกับการใช้งานหลากหลาย ความเร็วในการตอบสนอง (Fast Response Time) เซนเซอร์ MaxSonar มีการตอบสนองรวดเร็ว สามารถใช้ในระบบที่ต้องการการวัดระยะทางแบบเรียลไทม์ การทนต่อสภาพแวดล้อม (Environmental Robustness) เซนเซอร์ MaxSonar ออกแบบให้ทนทานต่อสภาพแวดล้อมที่หลากหลาย เช่น ฝุ่น น้ำ ความร้อน และความเย็น



รูปที่ 3.6 Max Sona sensor

### 3.3.3.3 Infrared Photoelectric Sensors

Infrared Photoelectric Sensors (เซนเซอร์ตรวจจับวัตถุด้วยแสงอินฟราเรด) เป็นอุปกรณ์ที่ใช้ในการตรวจจับวัตถุและวัดระยะทางโดยใช้แสงอินฟราเรด ดังรูปที่ 3.7

ซึ่งมีประโยชน์อย่างมากในหลากหลายแอปพลิเคชันในอุตสาหกรรม หุ่นยนต์ ระบบรักษาความปลอดภัย และอื่น ๆ หลักการทำงานของเซนเซอร์อินฟราเรดโฟโตอิเล็กทริก เซนเซอร์อินฟราเรดโฟโตอิเล็กทริกทำงานโดยการส่งสัญญาณแสงอินฟราเรดจากตัวส่ง (transmitter) ไปยังตัวรับ (receiver) และตรวจจับการสะท้อนหรือการถูกบล็อกของสัญญาณนั้นเพื่อวัดระยะทางหรือการตรวจจับวัตถุ โดยทั่วไปมีสองประเภทหลัก แบบสะท้อนกลับ (Reflective Type) ตัวส่งและตัวรับอยู่ในตำแหน่งเดียวกัน และวัดสัญญาณแสงที่สะท้อนกลับมาจากวัตถุ มักใช้สำหรับการตรวจจับวัตถุที่อยู่ใกล้ แบบทะลุผ่าน (Through-beam Type) ตัวส่งและตัวรับอยู่ตรงข้ามกัน และตรวจจับการถูกขัดขวางของสัญญาณแสงเมื่อวัตถุผ่านระหว่างตัวส่งและตัวรับ มักใช้ในระบบรักษาความปลอดภัยหรือในอุตสาหกรรมที่ต้องการการตรวจจับที่มีความแม่นยำสูง



รูปที่ 3.7 Infrared Photoelectric Sensors

### 3.4 ออกแบบโปรแกรมวัดระยะห่างและมุม

#### 3.4.1 การวัดระยะห่างระหว่างวัตถุกับกล้อง

เพื่อกำหนดระยะทางจากกล้องไปยังวัตถุหรือเครื่องหมาย จะใช้ความทฤษฎีความคล้ายของรูปสามเหลี่ยม ดังรูปที่ 3.8 ความคล้ายคลึงกันของสามเหลี่ยม เป็นหลักการพื้นฐานในวิชาเรขาคณิตที่อธิบายว่าเมื่อใดที่สามเหลี่ยมสองรูปจะคล้ายคลึงกัน สามเหลี่ยมที่คล้ายคลึงกันจะมีรูปร่างเหมือนกัน แต่ขนาดอาจแตกต่างกันโดยเฉพาะแล้ว สามเหลี่ยมสองรูปจะคล้ายคลึงกันเมื่อ

## 3.4.1.1 มุม-มุม

ระบุว่าหากมุมที่สอดคล้องกันสองมุมในสามเหลี่ยมสองรูปมีความสอดคล้องกัน (มีขนาดวัดเท่ากัน) สามเหลี่ยมสองรูปนั้นจะคล้ายคลึงกัน

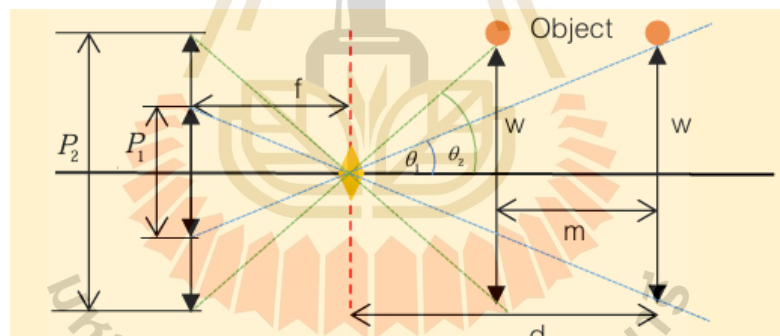
## 3.4.1.2 ด้าน-ด้าน-ด้าน

หากทุกด้านที่สอดคล้องกันของสามเหลี่ยมสองรูปมีความสัดส่วนเท่ากัน (มีอัตราส่วนเท่ากัน) สามเหลี่ยมสองรูปนั้นจะคล้ายคลึงกัน

## 3.4.1.3 ด้าน-มุม-ด้าน

ใช้เมื่อด้านที่สอดคล้องกันสองด้านของสามเหลี่ยมสองรูปมีความสัดส่วนเท่ากัน และมุมที่รวมโดยด้านเหล่านั้นมีความสอดคล้องกัน สามเหลี่ยมสองรูปนั้นจะคล้ายคลึงกัน เมื่อสามเหลี่ยมสองรูปคล้ายกัน จะทำให้ขนาดของมุมที่มันมี มีค่าเท่ากันด้วย และส่งผลให้ขนาดของด้านจะมีขนาดเป็นอัตราส่วนกัน ถ้าสามเหลี่ยม  $\triangle ABC \cong \triangle EDF$  จะได้

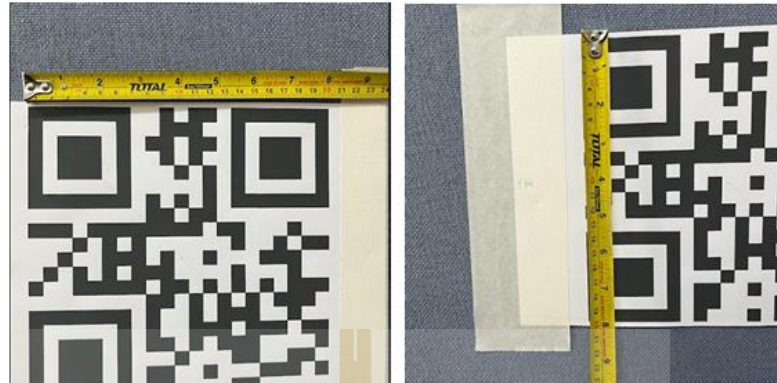
$$\frac{AB}{DE} = \frac{BC}{EF} = \frac{CA}{FD} \quad (2.12)$$



รูปที่ 3.8 แสดงการประยุกต์ใช้ทฤษฎีสามเหลี่ยมคล้าย

- $f$  คือ ทางยาวโฟกัสจากเลนส์ถึงเซนเซอร์กล้อง
- $P_1, P_2$  คือ ความกว้างของภาพสัญญาณมีหน่วยเป็นพิกเซล
- $W$  คือ ขนาดความกว้างจริงของสัญญาณ

*d* คือ ระยะห่างระหว่างกล้องถึงสัญลักษณ์มีหน่วยเป็นเซนติเมตร



รูปที่ 3.9 สัญลักษณ์ QR code ที่ใช้ในงานวิจัย

จากความคล้ายของรูปสามเหลี่ยม

$$\tan(\theta) = \frac{\left(\frac{P_1}{2}\right)}{f} = \frac{\left(\frac{w}{2}\right)}{d} \quad (2.13)$$

$$\frac{P_1}{f} = \frac{w}{d} \quad (2.14)$$

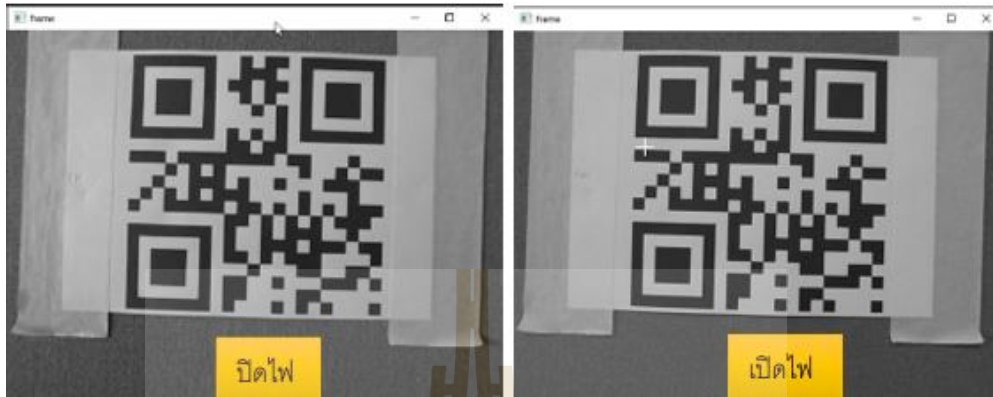
$$f = \frac{P_1 \times d}{w} \quad (2.15)$$

เริ่มต้นด้วยการใช้ Edge detection ตรวจจับ QR code จากนั้นทำการตีกรอบ เพื่อหาพิกัดจุด  $(x_1, y_1)$  และ  $(x_2, y_2)$  เพื่อนำไปคำนวณหาค่าระยะห่างของสัญลักษณ์ในหน่วยพิกเซล ดังรูปที่ 3.14



รูปที่ 3.10 การตีกรอบสัญลักษณ์ QR code

ก่อนที่จะทำการตีกรอบเพื่อหาค่าพิกัด ดังรูปที่ 3.10 จะทำการดัดแปลงภาพให้เป็นภาพขาวดำ ดังรูปที่ 3.11 เพื่อตัดการรบกวนจะสภาพแวดล้อมแสง



รูปที่ 3.11 การดัดแปลงภาพเป็นภาพขาวดำ

เมื่อทราบพิกัดกึ่งกลางขอบของสัญลักษณ์จากการตีกรอบสัญลักษณ์ที่ตรวจจับได้ นำค่าพิกัดไปคำนวณหา  $P_1$  จากสมการที่ 2.16 จะได้ระยะห่างระหว่างจุด 2 จุด

$$P_1 = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2.16)$$



รูปที่ 3.12 แสดงการคำนวณระยะห่างระหว่างพิกเซลสองจุด

นำค่า  $P_1$  ที่ได้จากการคำนวณที่ระยะความห่างจากสัญลักษณ์ถึงกล้องที่ระยะเริ่มต้นในงานวิจัยนี้ใช้ระยะเริ่มต้นที่ 100 เซนติเมตร และค่าของความกว้างของสัญลักษณ์ ดังนั้น  $w = 21$  เซนติเมตร  $d = 100$  เซนติเมตร นำไปแทนในสมการที่ 2.17

$$f = \frac{P_1 \times d}{w} \quad (2.17)$$

เมื่อได้ค่า  $f$  ทางยาวโฟกัสของระยะเริ่มต้น นำไปแทนในสมการเพื่อหาค่าของระยะห่างระหว่างสัญลักษณ์และกล้องเมื่อมีค่าเปลี่ยนแปลงไป ซึ่งสัมพันธ์กับค่า  $P_1$

$$d = \frac{w \times f}{P_1} \quad (2.18)$$

ค่า  $f$  ยังคงใช้ค่าเดิมที่คำนวณได้ในขั้นตอนเริ่มต้น และค่า  $w$  เป็นค่าที่ไม่เปลี่ยนแปลงได้จากการกำหนดขนาดของสัญลักษณ์ในขั้นเริ่มต้น



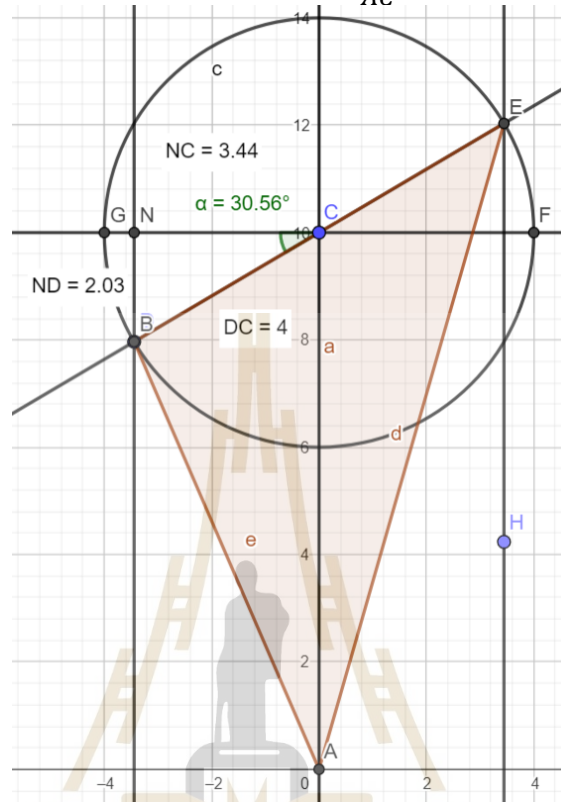
รูปที่ 3.13 แสดงการวัดระยะสัญลักษณ์ด้วยกล้อง

#### 3.4.2 การวัดมุมระหว่างวัตถุกับกล้อง

เมื่อเราได้ระยะห่างระหว่างสัญลักษณ์กับกล้อง  $d$  เราจะนำไปแทนในสมการความสัมพันธ์ของสามเหลี่ยมมุมฉาก  $\Delta ACN$  เพื่อหามุมที่สัญลักษณ์ทำกับกึ่งกลางเฟรมของกล้อง จุด  $A$  เป็นตำแหน่งของกล้อง จุด  $C$  เป็นระยะห่างระหว่างกล้องถึงสัญลักษณ์ในขณะที่อยู่กลางเฟรม จุด  $N$  เป็นตำแหน่งของสัญลักษณ์ที่อยู่ด้านซ้ายหรือขวาของเฟรมภาพ แทนค่าระยะห่างที่ได้จากการคำนวณในสมการที่ 2.19



$$\tan(\alpha) = \frac{NC}{AC} \quad (2.19)$$



รูปที่ 3.14 ความสัมพันธ์ระหว่างกล้องและสัญลักษณ์เมื่อมีค่าของมุม



รูปที่ 3.15 แสดงการทำงานของฟังก์ชันวัดมุม

### 3.5 ออกแบบการควบคุมสั่งงานการเคลื่อนที่และบังคับเลี้ยว

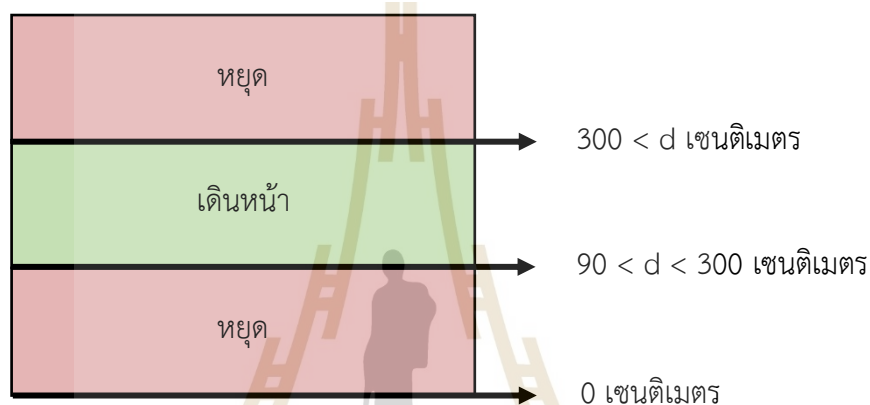
#### 3.5.1 การควบคุมระยะห่างการเคลื่อนที่ระหว่างตัวนำและตัวตาม

Logic ของการเคลื่อนที่ติดตามทางตรง แบ่งออกเป็นสามช่วง ดังรูปที่ 3.20

ช่วงที่หนึ่ง : ถ้าวัตถุอยู่ในระยะน้อยกว่า 90 เซนติเมตร สั่งงานให้หยุดเคลื่อนที่เพื่อรักษาระยะห่าง

ช่วงที่สอง : ถ้าวัตถุอยู่ในระยะมากกว่า 90 เซนติเมตร ถึง 300 เซนติเมตร จำสั่งงานให้เคลื่อนที่  
เดินหน้า

ช่วงที่สาม : ถ้าวัตถุอยู่ในระยะมากกว่า 300 เซนติเมตร สั่งงานให้หยุดเคลื่อนที่เพื่อรักษาระยะห่าง



รูปที่ 3.16 แสดงข้อกำหนดเงื่อนไขการสั่งงานควบคุมการเคลื่อนที่

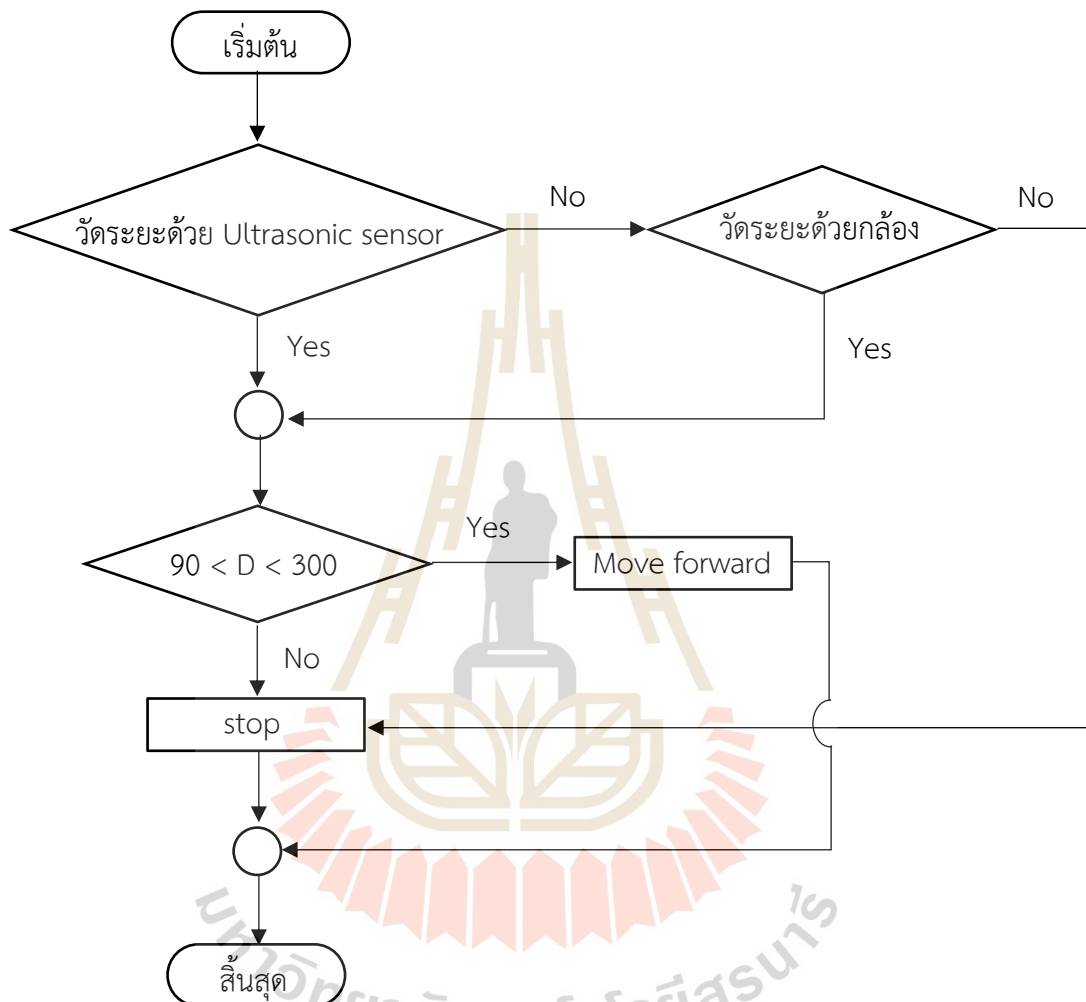
```
def thinking(Distance, center_x):
    if Distance <= disBack:
        direction = "B"
    elif Distance >= disGo:
        direction = "D"
    elif disBack < Distance <= disGo or Distance < limit:
        direction = "N"

    ser.write(direction.encode())
    print(direction)
    time.sleep(0.1)

    return direction
```

รูปที่ 3.17 ฟังก์ชันเงื่อนไขการสั่งงานควบคุมการเคลื่อนที่

การวัดระยะจะใช้เซ็นเซอร์อัลตราโซนิกในการวัดระยะเป็นหลักหากเซ็นเซอร์อัลตราโซนิกไม่สามารถวัดระยะได้จะทำการเปลี่ยนตัววัดระยะเป็นกล้องในการวัดระยะ logic การทำงานเป็นไปตาม Flow chart ที่แสดงดังรูปที่ 3.22



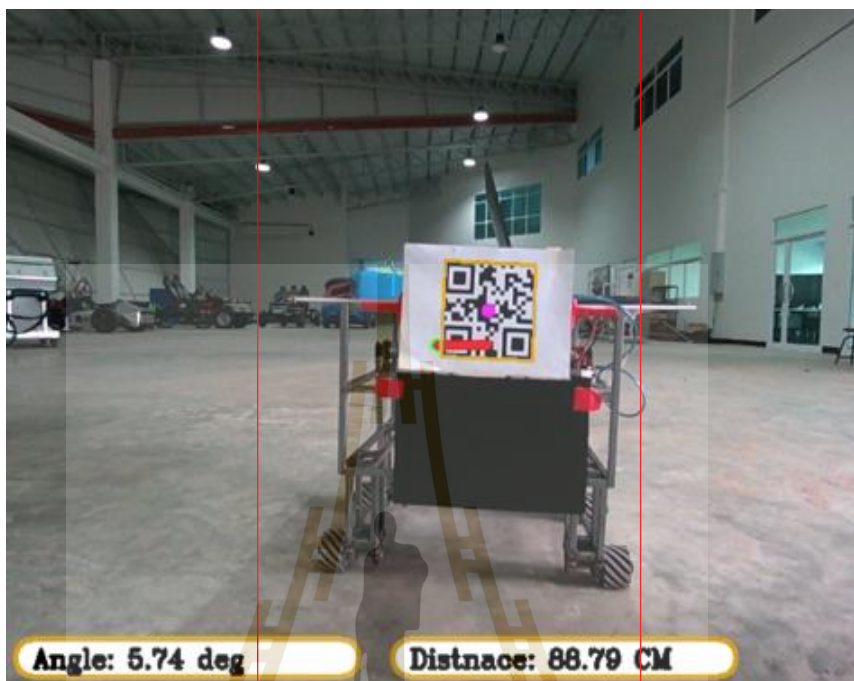
รูปที่ 3.18 Flow chart การควบคุมการเคลื่อนที่ติดตามของ AGV

### 3.5.2 การควบคุมการบังคับเลี้ยว

เมื่อทราบองศาระหว่างสัญลักษณ์กับจุดกึ่งกลางเฟรม (ตำแหน่งของกล้อง) logic ของการบังคับเลี้ยวล้อหน้า จะทำการกำหนดเส้นทางซ้ายและทางขวาเป็นขอบเขตของการบังคับเลี้ยวเป็นสามช่วง

ช่วงที่หนึ่ง : ถ้าตำแหน่งกึ่งกลางของสัญลักษณ์มีค่าพิคัดแกน  $x$  น้อยกว่าค่า  $threshold\_left$  จะสั่งงานให้เลี้ยวซ้าย

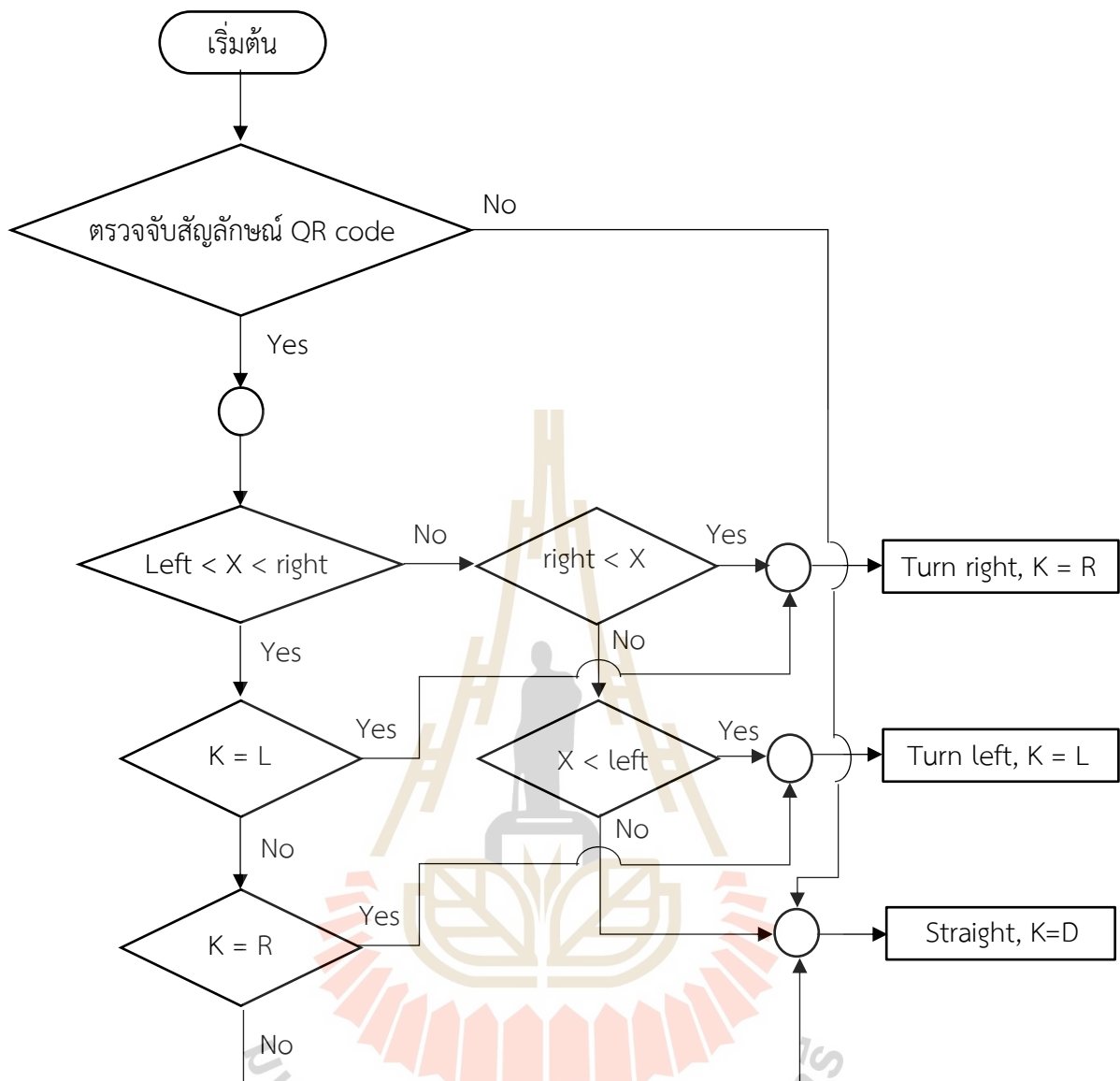
ช่วงที่สอง : ถ้าตำแหน่งกึ่งกลางของสัญลักษณ์มีค่าพิกัดแกน x มากกว่าค่า  $\text{threshold\_right}$  จะสั่งงานให้เลี้ยวซ้าย



รูปที่ 3.19 แสดงเส้นกรอบ  $\text{threshold\_left}$  และ  $\text{threshold\_right}$

ช่วงที่สาม : ถ้าตำแหน่งกึ่งกลางของสัญลักษณ์มีค่าพิกัดแกน x มากกว่าค่า  $\text{threshold\_left}$  แต่ไม่เกิน  $\text{threshold\_right}$  จะพิจารณาว่าเงื่อนไขก่อนถูกส่งคำสั่งอะไรมาก่อน เงื่อนไขที่หนึ่ง ถ้าคำสั่งครั้งก่อนหน้า สั่งล้อเลี้ยวซ้าย จะทำการสั่งงานให้ล้อเลี้ยวขวา เพื่อเป็นการคืนล้อให้อยู่ในตำแหน่งกึ่งกลาง หรือ ล้อตรง

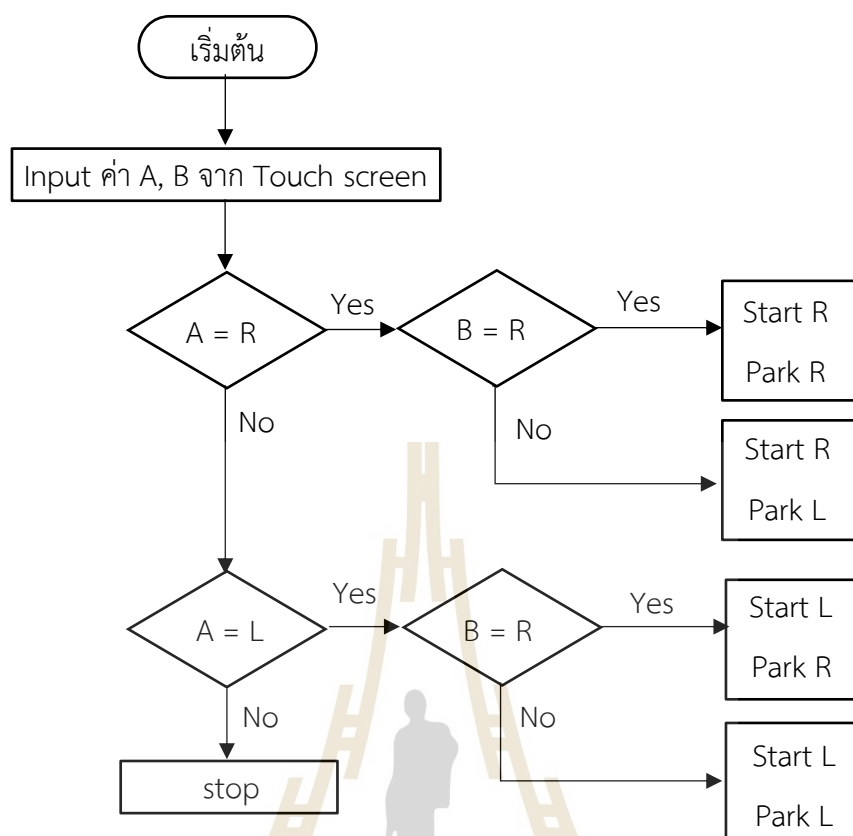
เงื่อนไขที่สอง ถ้าคำสั่งครั้งก่อนหน้า สั่งล้อเลี้ยวขวา จะทำการสั่งงานให้ล้อเลี้ยวซ้าย เพื่อเป็นการคืนล้อให้อยู่ในตำแหน่งกึ่งกลาง หรือ ล้อตรง แสดง logic การทำงานดังรูปที่ 3.24



รูปที่ 3.20 Flow chart เงื่อนไขการทำงานบังคับเลี้ยวล้อหน้า

### 3.5 ออกแบบโปรแกรมการจอดและออกจากสถานี

การสั่งงานเข้าจอดเมื่อเสร็จภาระกิจเคลื่อนที่ติดตามและออกจากสถานีจอดเพื่อเริ่มเคลื่อนที่ติดตามจะเป็นคำสั่งทำตามขั้นตอนที่ผู้วิจัยได้เรียงลำดับไว้ เพียงแต่การออกจากสถานีจอดและเข้าจอดจะเริ่มต้นจากทางด้านซ้ายหรือด้านขวาจะสามารถป้อนอินพุตตามที่ใช้ต้องการได้ ผ่านการทัชสกรีน ลำดับขั้นตอนการทำงานของการเข้าจอดและออกจากสถานีเป็นไปตามรูปที่ 3.25



รูปที่ 3.21 Flow chart การจอดและออกจากสถานี

### 3.6 การทดลอง

#### 3.6.1 ทดสอบการตรวจจับสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงต่างกัน

เนื่องจากการเคลื่อนที่ติดตาม AGV ตัวนำ AGV ตัวตามจะต้องมีตัวตรวจรู้ ซึ่งในงานวิจัยนี้ใช้เป็นกล้อง เพื่อให้ทราบตำแหน่งของ AGV ตัวนำว่าสถานะขณะนั้นอยู่ในตำแหน่งทางด้านซ้ายหรือทางด้านขวาของ AGV ตัวตาม เพื่อที่จะนำสถานะนี้ไปควบคุมคำสั่งการบังคับเลี้ยวล้อหน้า เพื่อให้ AGV ตัวตามยังคงเคลื่อนที่ติดตามกันไปได้ในกรณีที่ AGV ตัวนำมีการเคลื่อนที่เลี้ยวหรือเคลื่อนที่ไม่เป็นเส้นตรง ดังนั้นการทดลองนี้จึงออกแบบเพื่อทดสอบความสามารถในการตรวจจับสัญลักษณ์ 2 สัญลักษณ์ระหว่างคิวอาร์โค้ดและภาพสี ทำการทดลองหาความสามารถของการตรวจจับสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงสองเงื่อนไข เพื่อเลือกสัญลักษณ์ที่มีความสามารถในการตรวจจับในสภาพแวดล้อมความเข้มแสงที่หลากหลาย ซึ่งเงื่อนไขแรก คือความเข้มแสงที่ 80 ถึง 300 lux คือสภาพความเข้มแสงเวลากลางวันที่ไม่มีการเปิดหลอดไฟในสถานที่ทำการทดลอง และเงื่อนไขที่สอง คือความเข้มแสงที่ 300 ถึง 650 lux คือสภาพความเข้มแสงเวลากลางวันและกลางคืนที่มีการเปิดหลอดไฟในสถานที่ทำการทดลอง ดังแสดงในตารางที่ 3.1

### 3.6.2 ทดสอบการวัดระยะของสัญลักษณ์และเซนเซอร์

การติดตามการเคลื่อนที่ของ AGV ตัวนำนั้น จะต้องมีการเคลื่อนที่ติดตามแบบรักษาระยะห่าง นอกจากการทราบตำแหน่งสถานะของ AGV ตัวนำ ว่าอยู่ในตำแหน่งทางซ้ายหรือขวาแล้ว การทดสอบการวัดระยะของสัญลักษณ์และเซนเซอร์จึงจำเป็นต่อการตัดสินใจเลือก เพื่อใช้เป็นสัญญาณตรวจรู้ในการคิดวิเคราะห์คำสั่งของการเคลื่อนที่ ดังแสดงในตารางที่ 3.1

ตารางที่ 3.1 แสดงการทดลองตรวจจับสัญลักษณ์

ระยะห่าง (เซนติเมตร)	สภาพแวดล้อมความเข้มแสง (lux)	สัญลักษณ์	
20 – 200	80 – 300	QR code	Color
20 - 200	300 - 650		

### 3.6.3 ทดสอบความแม่นยำของการจอดและออกจากสถานีเริ่มต้น

เมื่อเสร็จสิ้นภารกิจการเคลื่อนที่ติดตาม AGV ตัวตามจะต้องสามารถเข้าสถานีจอดที่มีพื้นที่จำกัด และเป็นขอบเขตที่ชัดเจน ดังนั้นการจอดและออกจากสถานีจำเป็นจะต้องมีการทำซ้ำเพื่อให้สามารถทำงานในพื้นที่ขอบเขตจำกัดได้ ด้วยการตรวจวัดการสะท้อนกลับของเซนเซอร์ Infrared Photoelectric เพื่อบ่งบอกว่าถึงสถานีจอด จากนั้นคำสั่งการทำงานจอดจะเริ่มต้นเช่นเดียวกับกับการเริ่มต้นภารกิจติดตาม การเคลื่อนที่ออกจากสถานีจอดมากยิ่งขึ้นเส้นทางติดตามนั้น AGV ตัวตามนั้นจะต้องมีตำแหน่งการเริ่มต้นที่ตัวตรวจรู้ซึ่งเป็นกล้องจะต้องสามารถตรวจจับสัญลักษณ์ที่ติดอยู่กับ AGV ตัวตามได้ เพื่อให้ยังคงเคลื่อนที่ติดตามและรักษาระยะห่าง การทดสอบนี้จึงทดสอบหาค่าความคาดเคลื่อนของการจอดและออกจากสถานีจอด และหาตำแหน่งของป้ายสะท้อนแสงของเซนเซอร์ Infrared Photoelectric ที่มีผลต่อตำแหน่งการจอด

ตารางที่ 3.2 แสดงการทดลองหาค่าความคาดเคลื่อนการจอดและออกจากสถานี

	ตำแหน่งการจอด	ตำแหน่งการออกจากสถานี
ค่าความคาดเคลื่อน	ตำแหน่งจอดซ้าย	เริ่มต้นทางซ้าย
	ตำแหน่งจอดขวา	เริ่มต้นทางขวา

### 3.6.4 ทดสอบการเคลื่อนที่ติดตามทางตรง

การเคลื่อนที่รักษาระยะห่างนั้นนอกจากการวัดระยะด้วยเซนเซอร์และกล้องแล้ว ยังจำเป็นต้องสั่งงานรีเลย์ที่ทำหน้าที่สั่งงานให้ AGV ติดตามเคลื่อนที่เดินหน้า หรือหยุดการเคลื่อนที่เพื่อรักษาระยะห่าง ดังนั้นการทดสอบนี้จึงทดสอบการทำงานร่วมกันของตัวตรวจรู้ที่ใช้กล้องและเซนเซอร์วัดระยะกับการสั่งงานรีเลย์ที่ทำหน้าที่ควบคุมมอเตอร์ขับเคลื่อนล้อหลัง ทำการทดลองในระยะ 20 ถึง 200 เซนติเมตร เพื่อตรวจสอบการทำงานร่วมกันของโปรแกรมการบังคับเลี้ยวและติดตามกับการส่งสัญญาณรีเลย์เพื่อควบคุมการทำงานของมอเตอร์ล้อหลัง

## 3.7 สรุป

การดำเนินงานวิจัยการออกแบบโปรแกรมวัดระยะห่างระหว่าง AGV ตัวนำกับ AGV ตัวตามที่ใช้ตัวตรวจรู้เป็นกล้องและเซนเซอร์วัดระยะ ที่เป็นค่าตัวแปรสำคัญในการควบคุมสั่งงานการเคลื่อนที่ติดตามและการบังคับเลี้ยวของล้อหน้า เพื่อให้ทราบถึงพฤติกรรมและความสามารถของสัญลักษณ์ที่ใช้ รวมถึงการวัดระยะในแต่ละสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงที่หลากหลายเพื่อหาช่วงของการทำงานที่ยังมีประสิทธิภาพ อีกทั้งทำการทดลองเพื่อหาค่าความคาดเคลื่อนของการจอดเข้าสถานีเมื่อจบภารกิจติดตาม จะต้องทราบถึงค่าความคาดเคลื่อนเพราะพื้นที่การจอดมีขอบเขตที่ชัดเจนและจำกัด ซึ่งแบ่งขั้นตอนการดำเนินการทดลองเป็นดังนี้

ขั้นตอนที่ 1 ทดสอบเพื่อหาระยะการตรวจจับของสัญลักษณ์ QR code และ Color ในสภาพแวดล้อมความเข้มแสงสองช่วง ช่วงที่หนึ่ง 80 – 300 lux และช่วงที่สอง 300 – 650 lux

ขั้นตอนที่ 2 ทดสอบความแม่นยำในการวัดระยะด้วยกล้องของสัญลักษณ์ QR code และ Color ในสภาพแวดล้อมความเข้มแสงสองช่วง ช่วงที่หนึ่ง 80 – 300 lux และช่วงที่สอง 300 – 650 lux เพื่อเปรียบเทียบหาสัญลักษณ์ที่เหมาะสมกับการใช้ติดอยู่ที่ AGV ตัวตาม

ขั้นตอนที่ 3 ทดสอบการวัดมุมของสัญลักษณ์ QR code และ Color ในช่วงมุมมอง 0 – 30 องศา เพื่อใช้ในการควบคุมสั่งงานบังคับเลี้ยวล้อหน้า

ขั้นตอนที่ 4 ทดสอบการวัดระยะห่างด้วยเซนเซอร์ระหว่าง Ultrasonic sensor และ Max sona sensor เพื่อหาระยะมุมมองข้างหน้าของการตรวจวัดระยะ และหาความแม่นยำของการวัดระยะระหว่าง AGV ตัวนำ และ AGV ตัวตาม

ขั้นตอนที่ 5 ออกแบบซอฟต์แวร์การควบคุมการสั่งงานมอเตอร์ล้อหลังเพื่อเคลื่อนที่ติดตามและรักษาระยะห่างระหว่าง AGV ตัวนำ กับ AGV ตัวตาม รวมถึงการออกแบบซอฟต์แวร์การบังคับเลี้ยวล้อหน้าเพื่อเคลื่อนที่เลี้ยวตาม AGV ตัวนำ

ขั้นตอนที่ 6 ทดสอบซอฟต์แวร์การเคลื่อนที่ติดตาม AGV ตัวนำว่าสามารถรักษาระยะห่างในช่วงที่กำหนด และสามารถบังคับเลี้ยวติดตามได้ในกรณีที่ AGV ตัวนำเคลื่อนที่ไม่เป็นเส้นตรง



ขั้นตอนที่ 7 ทดสอบการเข้าจอดในสถานีจอดเมื่อเสร็จสิ้นภารกิจการติดตามและการเคลื่อนที่ออกจากสถานี เพื่อหาระยะความคาดเคลื่อนของการจอดและออกจากสถานี และทดสอบซอฟต์แวร์การเคลื่อนที่ติดตามพร้อมกับการออกจากสถานีจนถึงการเข้าจอดเมื่อจบภารกิจ

ขั้นตอนที่ 8 วิเคราะห์และสรุปผลการทดลอง



## บทที่ 4

### ผลการวิจัยและอภิปรายผล

#### 4.1 บทนำ

ในบทนี้นำเสนอผลการทดลองเปรียบเทียบการตรวจจับสัญลักษณ์ที่ใช้ติดตั้งกับ AGV ตัวนำ เพื่อหาความสามารถในการตรวจจับในสภาพแวดล้อมความเข้มแสงในแต่ละช่วง ประกอบการตัดสินใจเลือกใช้เป็นสัญลักษณ์เพื่อติดตาม เปรียบเทียบการวัดมุมมองของสัญลักษณ์ เพื่อหาค่าความคลาดเคลื่อนของการวัดมุม ที่มีผลกับการสั่งการบังคับเลี้ยว และเปรียบเทียบเซนเซอร์วัดระยะ เพื่อหาระยะและองศาการตรวจวัดที่แม่นยำกับช่วงการทำงาน ทดสอบหาค่าความคลาดเคลื่อนของการเคลื่อนที่ออกจากสถานีและการเข้าจอดในสถานี เพื่อนำไปกำหนดขอบเขตของพื้นที่สถานีจอด อีกทั้งยังนำเสนอการตรวจสอบความสามารถของโปรแกรมที่ใช้ในการควบคุมบังคับเลี้ยว เพื่อเคลื่อนที่ติดตาม AGV ตัวนำว่าสามารถเคลื่อนที่ติดตามได้จริงตามวัตถุประสงค์ของงานวิจัย

#### 4.2 เปรียบเทียบการตรวจจับสัญลักษณ์

##### 4.2.1 ตรวจจับสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงช่วง 80 – 300 lux

การตรวจจับสัญลักษณ์ระหว่าง QR code และ Color ในสภาพแวดล้อมความเข้มแสงช่วง 80 – 300 lux คือสภาพแวดล้อมที่ปิดไฟโดยใช้แสงธรรมชาติ ช่วงเวลากลางวัน

80 - 300 lux		
Distance(cm)	Qrcode	Color
20	Y	Y
30	Y	Y
40	Y	Y
50	Y	N
60	Y	N
70	Y	N
80	Y	N
90	Y	N
100	Y	N
110	Y	N
120	Y	N
130	Y	N
140	Y	N
150	Y	N
160	N	N
170	N	N
180	N	N
190	N	N
200	N	N

รูปที่ 4.1 แสดงการทดลองตรวจจับสัญลักษณ์ที่ความเข้มแสง 80 – 300 lux

เมื่อ Y แทน สามารถตรวจจับได้ N แทน ไม่สามารถตรวจจับได้

#### 4.2.2 ตรวจจับสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงช่วง 300 – 650 lux

การตรวจจับสัญลักษณ์ระหว่าง QR code และ Color ในสภาพแวดล้อมความเข้มแสงช่วง 300 – 650 lux คือสภาพแวดล้อมที่เปิดไฟ

300 - 650 lux		
Distance(cm)	Qrcode	Color
20	Y	Y
30	Y	Y
40	Y	Y
50	Y	Y
60	Y	Y
70	Y	Y
80	Y	Y
90	Y	Y
100	Y	Y
110	Y	Y
120	Y	Y
130	Y	Y
140	Y	Y
150	Y	Y
160	N	Y
170	N	Y
180	N	N
190	N	N
200	N	N

รูปที่ 4.2 แสดงการทดลองตรวจจับสัญลักษณ์ที่ความเข้มแสง 300 – 650 lux

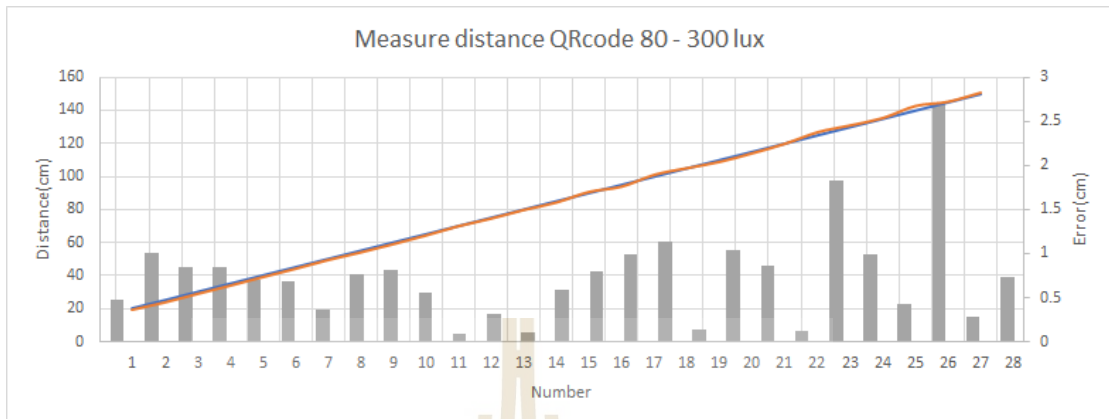
เมื่อ Y แทน สามารถตรวจจับได้ N แทน ไม่สามารถตรวจจับได้

#### 4.3 เปรียบเทียบการวัดระยะห่างกล้องถึงสัญลักษณ์

##### 4.3.1 วัดระยะห่างจากกล้องถึงสัญลักษณ์ (QRcode) ด้วยกล้อง

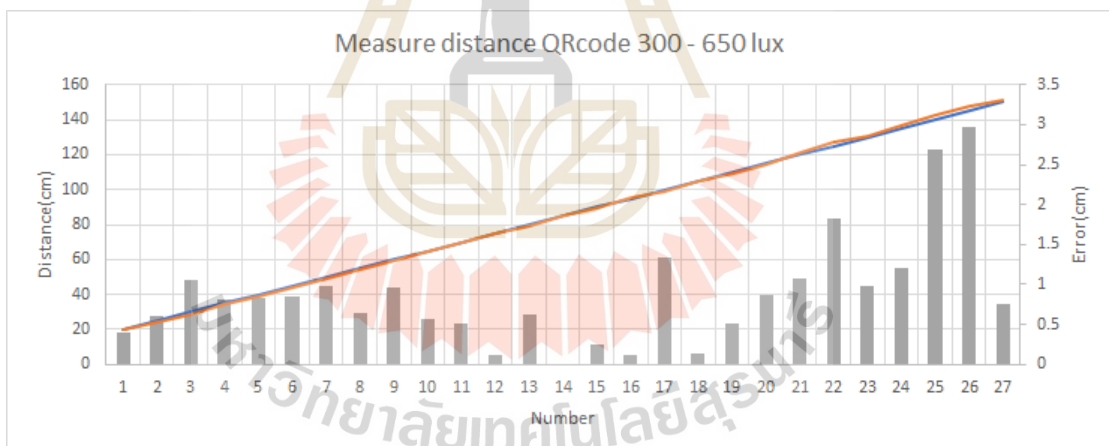
การทดลองวัดระยะจะทำการทดลองในสภาพแวดล้อมที่แตกต่างกันสองช่วง ช่วงแรก 80 – 300 lux คือช่วงที่ความเข้มแสงต่ำ และจะทำการทดลองกับสัญลักษณ์ทั้งสองแบบ คือ QR code กับ Color

#### 4.3.1.1 สภาพแวดล้อมความเข้มแสง 80 – 300 lux



รูปที่ 4.3 กราฟเปรียบเทียบการวัดระยะของ QR code กับระยะมาตรฐานในช่วง 80 – 300 lux

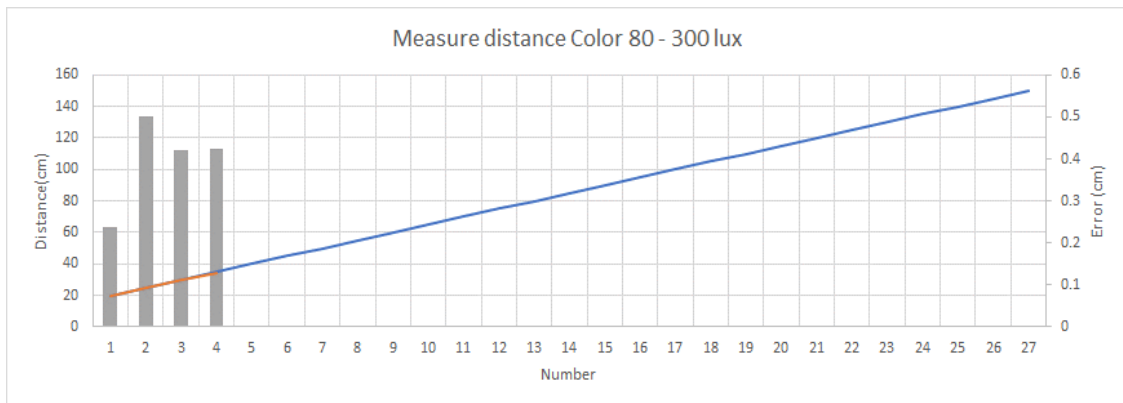
#### 4.3.1.2 สภาพแวดล้อมความเข้มแสง 300 – 650 lux



รูปที่ 4.4 กราฟเปรียบเทียบการวัดระยะของ QR code กับระยะมาตรฐานในช่วง 300 – 650 lux

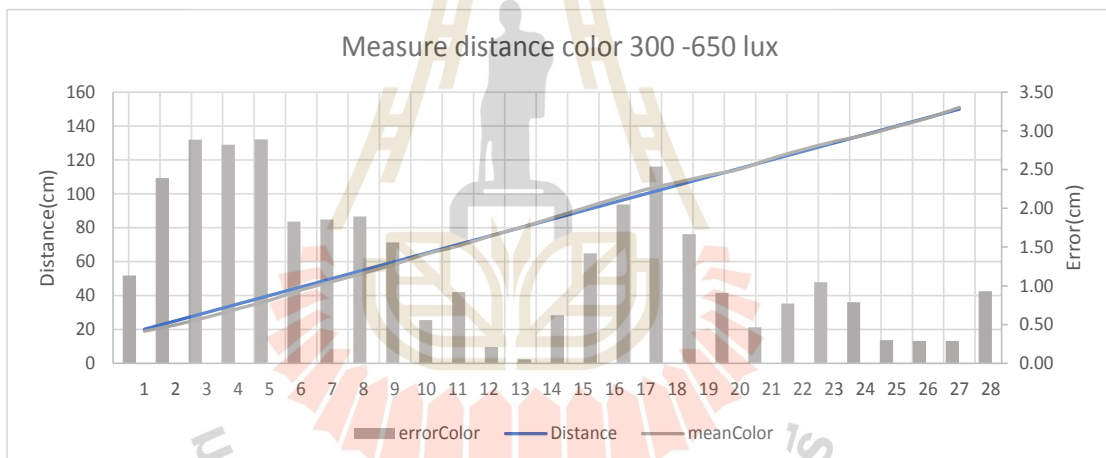
### 4.3.2 วัดระยะห่างกล้องถึงสัญลักษณ์ (Color) ด้วยกล้อง

#### 4.3.2.1 สภาพแวดล้อมความเข้มแสง 80 – 300 lux



รูปที่ 4.5 กราฟเปรียบเทียบการวัดระยะของ Color กับระยะมาตรฐานในช่วง 80 – 300 lux

#### 4.3.2.2 สภาพแวดล้อมความเข้มแสง 300 – 650 lux

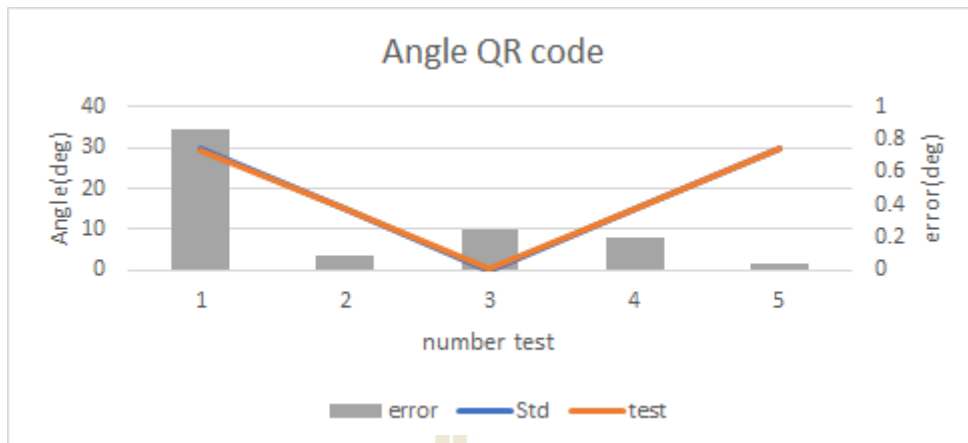


รูปที่ 4.6 กราฟเปรียบเทียบการวัดระยะของ Color กับระยะมาตรฐานในช่วง 300 – 650 lux

#### 4.4 ทดสอบการวัดขนาดมุมของสัญลักษณ์เทียบจุดกึ่งกลางกล้อง

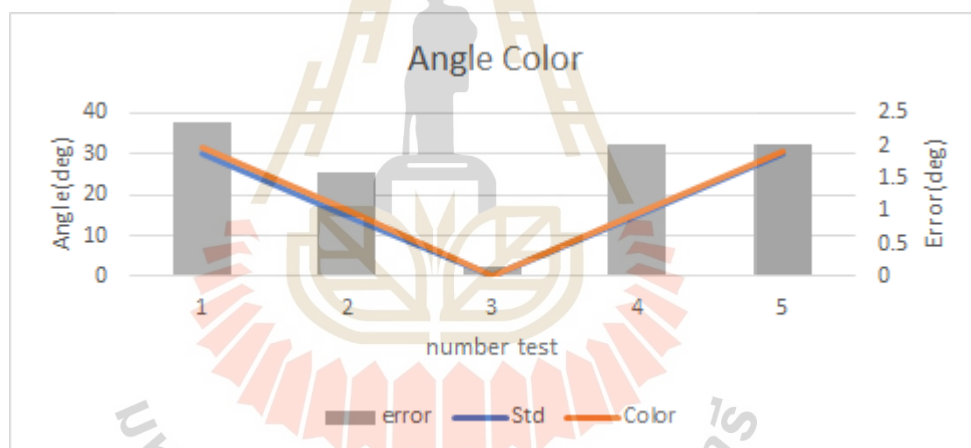
การทดสอบวัดมุมมีความสำคัญกับการบังคับเลี้ยว หากได้ข้อสรุปของการวัดระยะในสัญลักษณ์แล้ว ยังคงต้องทดลองเรื่องของการวัดมุมอีกด้วย เพราะการวัดระยะทำได้แค่คำนวณเพื่อหาค่าสั่งการติดตามแต่ยังขาดอินพุตของการบังคับเลี้ยว ดังนั้นจึงต้องทำการทดลองการวัดมุมเพื่อประกอบการตัดสินใจเลือกใช้สัญลักษณ์

##### 4.4.1 วัดขนาดมุมสัญลักษณ์ (QR code) เทียบจุดกึ่งกลางกล้อง ช่วง 300 – 650 lux



รูปที่ 4.7 กราฟแสดงการวัดมุม QR code

#### 4.4.2 วัดขนาดมุมสัญลักษณ์ (Color) เทียบจุดกึ่งกลางกล้อง ช่วง 300 – 650 lux

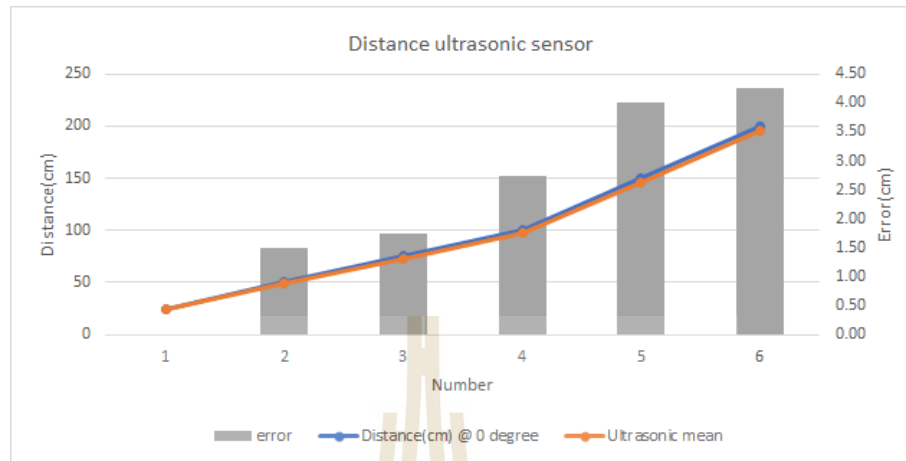


รูปที่ 4.8 กราฟแสดงการวัดมุม Color

#### 4.5 เปรียบเทียบการวัดระยะห่างของเซนเซอร์

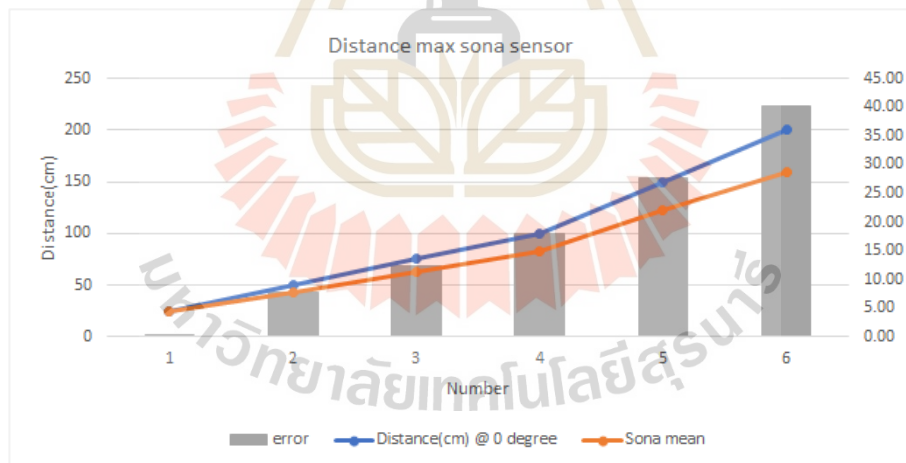
การวัดระยะด้วยเซนเซอร์สำคัญไม่น้อยไปกว่าการวัดระยะด้วยสัญลักษณ์เพราะการวัดระยะห่างเป็นตัวแปรสำคัญในการออกคำสั่งเดินทางหรือหยุดเคลื่อนที่ ซึ่งมีผลต่อการติดตามการเคลื่อนที่ ดังนั้นจึงจำเป็นต้องมีตัวตรวจรู้ในด้านการวัดระยะสำรองซึ่งในงานวิจัยนี้จะใช้เซนเซอร์วัดระยะเป็นตัวตรวจรู้หลัก และการวัดระยะด้วยกล้องเป็นตัววัดระยะสำรอง โดยการทดลองจะเปรียบเทียบระหว่าง Ultrasonic กับ Max Sona เพื่อตรวจสอบความแม่นยำและองศาด้านหน้าของการวัดระยะ

#### 4.5.1 วัดระยะห่างวัตถุด้วยเซนเซอร์ Ultrasonic



รูปที่ 4.9 กราฟแสดงการวัดระยะห่างด้วย Ultrasonic

#### 4.5.2 วัดระยะห่างวัตถุด้วยเซนเซอร์ Max Sona

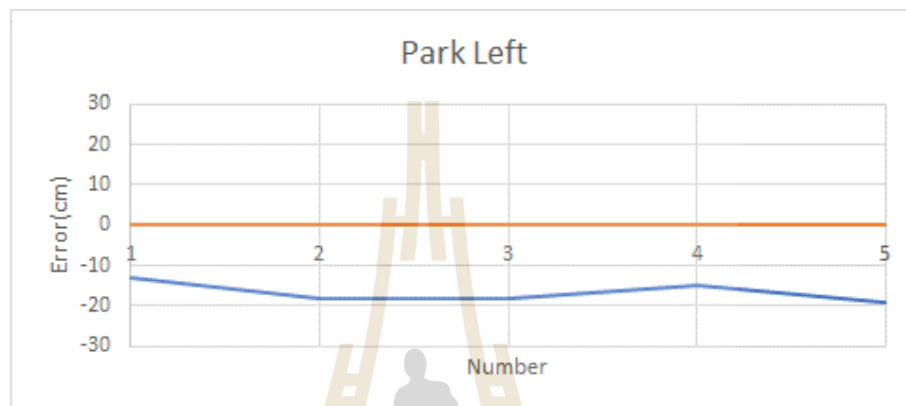


รูปที่ 4.10 กราฟแสดงการวัดระยะห่างด้วย Max Sona

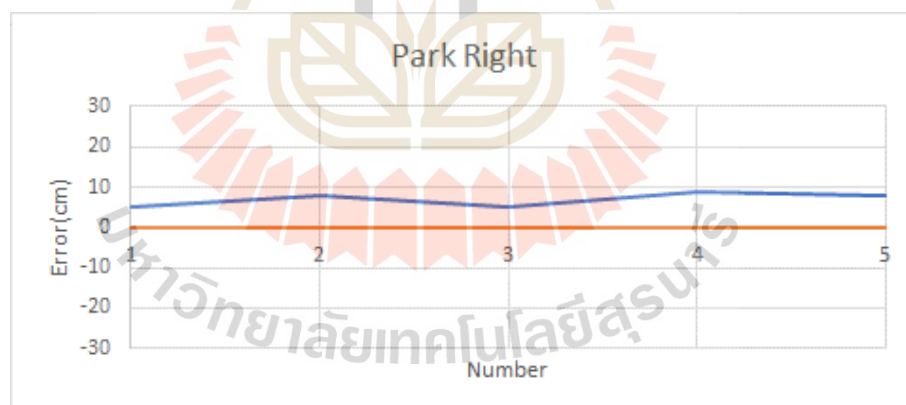
## 4.6 ทดสอบการเข้าจอดและออกจากสถานี

### 4.6.1 ทดสอบการเข้าจอด

ทดลองเข้าจอดในสถานีเมื่อเสร็จสิ้นภารกิจเคลื่อนที่ติดตาม การจอดจำเป็นต้องทราบค่าความคลาดเคลื่อนของการจอด ซึ่งมีผลต่อการระบุขอบเขตของสถานีจอด ดังนั้นการทดลองนี้จะทำการเข้าจอดซ้ำทั้งหมด 5 ครั้ง เพื่อหาค่าความคลาดเคลื่อนเมื่อเทียบกับจุดอ้างอิง



รูปที่ 4.11 กราฟแสดงค่าความคลาดเคลื่อนของการเข้าจอดเมื่อสถานีจอดอยู่ทางซ้าย



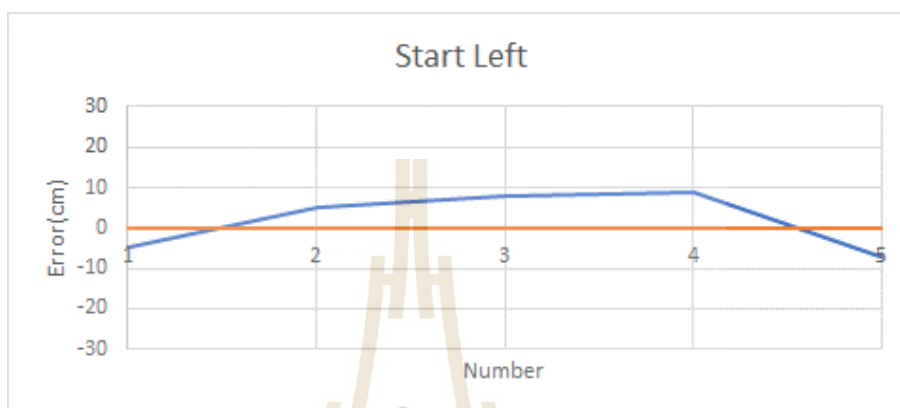
รูปที่ 4.12 กราฟแสดงค่าความคลาดเคลื่อนของการเข้าจอดเมื่อสถานีจอดอยู่ทางขวา

### 4.6.2 ทดสอบการออกจากสถานี

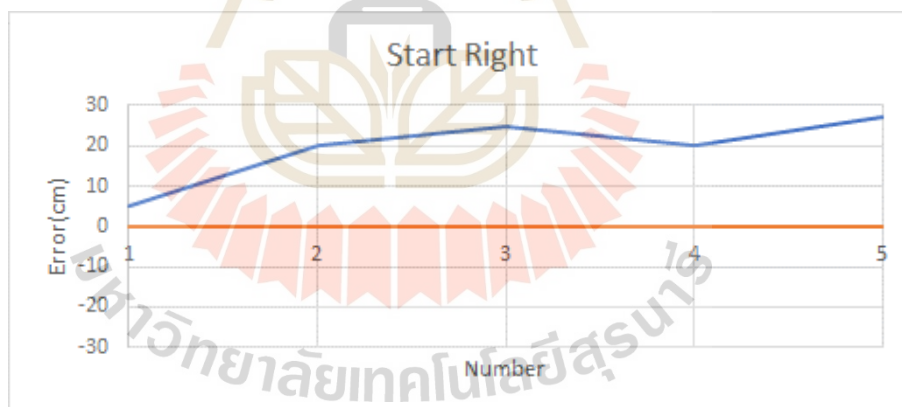
ทดลองออกจากสถานี เมื่อเริ่มภารกิจเคลื่อนที่ติดตาม เมื่อรถ AGV ตัวตามเคลื่อนที่ออกจากสถานีจำเป็นต้องเห็นสัญลักษณ์ที่ถูกติดตั้งไว้บน AGV ตัวนำ เพื่อรับค่าอินพุตไปสั่งการบังคับเลี้ยว หากสถานะเริ่มต้นเมื่อเคลื่อนที่ออกจากสถานี AGV ตัวตามไม่สามารถตรวจจับ



สัญลักษณ์ได้จะส่งผลให้ AGV ตัวตามไม่ได้รับอิทธิพลของการบังคับเลี้ยว เพื่อให้เงื่อนไขการเริ่มต้นเหมาะสมกับการเคลื่อนที่ติดตาม และนำค่าความคลาดเคลื่อนไปตั้งค่า AGV ตัวนำให้หยุดรอในขอบเขตที่เผื่อค่าความคลาดเคลื่อนแล้ว จำเป็นต้องทดลองเพื่อหาค่าความคลาดเคลื่อนของการออกจากสถานี ซึ่งทดลองออกจากสถานีทำซ้ำทั้งหมด 5 ครั้ง



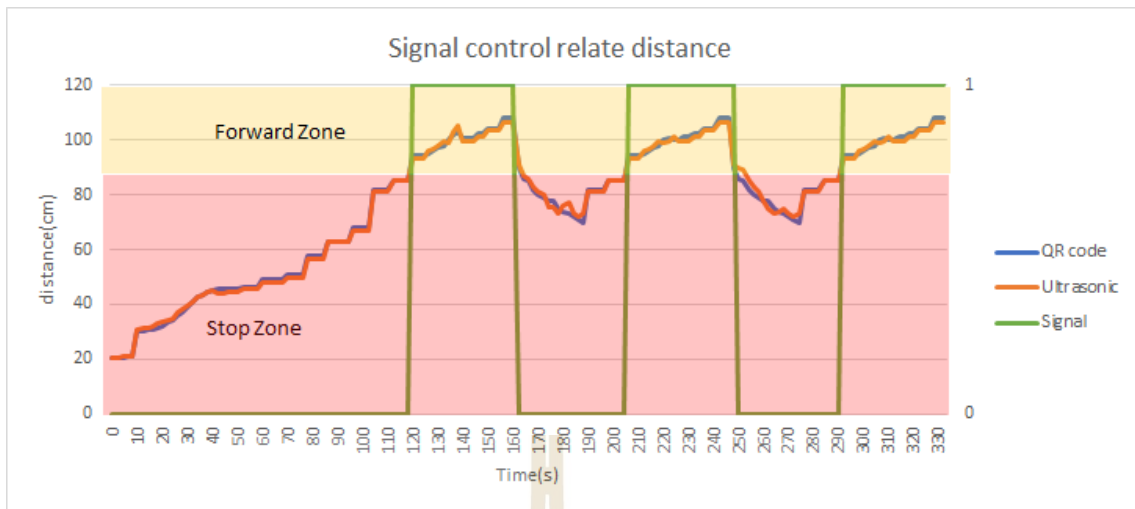
รูปที่ 4.13 กราฟแสดงค่าความคลาดเคลื่อนของการออกจากสถานีทางด้านซ้าย



รูปที่ 4.14 กราฟแสดงค่าความคลาดเคลื่อนของการออกจากสถานีทางด้านขวา

#### 4.7 ทดสอบการเคลื่อนที่ติดตาม AGV ตัวนำ

เมื่อทดสอบการวัดระยะด้วยกล้องและเซนเซอร์จนได้ข้อสรุป จึงนำมาทดสอบร่วมกับโปรแกรมการควบคุมสั่งการชุดขับเคลื่อนล้อหลังให้เคลื่อนที่สัมพันธ์กับระยะห่าง โดยการส่งสัญญาณให้รีเลย์ที่ควบคุมมอเตอร์ชุดล้อหลังทำงาน สัมพันธ์กับระยะห่างในช่วงตั้งแต่ 20 – 120 เซนติเมตร



รูปที่ 4.15 กราฟแสดงความสัมพันธ์ระหว่างการวัดระยะกับการส่งสัญญาณรีเลย์



## บทที่ 5

### สรุปและข้อเสนอแนะ

#### 5.1 สรุปผลการวิจัย

งานวิจัยนี้ได้ศึกษาการเคลื่อนที่ติดตาม AGV และการวัดระยะห่างจาก AGV ตัวนำและตัวตาม ด้วยกล้องเป็นตัวตรวจรู้ โดยใช้หลักการตรวจจับ QR code ร่วมกับเซนเซอร์ Ultrasonic ในการวัดระยะ เพื่อออกแบบโปรแกรมที่ใช้กับ AGV ตัวตาม ให้สามารถเคลื่อนที่ติดตาม AGV ตัวนำในระยะที่กำหนดและสามารถบังคับเลี้ยวเพื่อติดตามในเส้นทางได้อย่างแม่นยำ อีกทั้งยังสามารถเข้าจอดในสถานีจอดเมื่อเสร็จสิ้นภารกิจการเคลื่อนที่ติดตาม และออกจากสถานีเริ่มต้นเพื่อเคลื่อนที่ติดตาม AGV ตัวนำได้ ซึ่งผลการทดลองของวิจัยสามารถสรุปได้ดังนี้

##### 5.1.1 ผลการทดลองการตรวจจับสัญลักษณ์ในสภาพแวดล้อมความเข้มแสงที่แตกต่างกัน

###### 5.1.1.1 ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 80 ถึง 300 lux

การทดสอบตรวจจับสัญลักษณ์ทดลองในช่วงระยะ 20 ถึง 200 เซนติเมตร พบว่า สัญลักษณ์ QR code สามารถตรวจจับเพื่อวัดระยะได้ในระยะการตรวจจับตั้งแต่ 20 เซนติเมตร ถึง 150 เซนติเมตร เมื่อมากกว่า 150 เซนติเมตรจะไม่สามารถตรวจจับได้ ในขณะที่ Color สามารถตรวจจับเพื่อวัดระยะได้ในระยะการมองตั้งแต่ 20 เซนติเมตร ถึง 40 เซนติเมตร ถ้าหากระยะมากกว่า 40 เซนติเมตรจะไม่สามารถตรวจจับได้ เนื่องจากความเข้มแสงมีผลต่อขอบเขต RGB ของสีที่ใช้ตรวจจับ จึงสรุปได้ว่าความสามารถในการตรวจจับของสัญลักษณ์ QR code มีความไกลกว่าถึง 110 เซนติเมตร ซึ่งมีความมากกว่า 275 เปอร์เซ็นต์

###### 5.1.1.2 ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 300 ถึง 650 lux

เมื่อทดลองในสภาพแวดล้อมแสงที่มีความเข้มแสงมากขึ้นในช่วง 300 ถึง 650 lux พบว่าสัญลักษณ์ QR code สามารถตรวจจับเพื่อวัดระยะได้ในระยะการมองตั้งแต่ 20 เซนติเมตร ถึง 150 เซนติเมตร ในขณะที่ Color สามารถตรวจจับเพื่อวัดระยะได้ในระยะการมองตั้งแต่ 20 เซนติเมตร ถึง 170 เซนติเมตร ซึ่งมีความสามารถในการตรวจจับที่มากกว่า QR code ถึง 20 เซนติเมตร หรือคิดเป็น 11.3 เปอร์เซ็นต์

ดังนั้น การตรวจจับเพื่อวัดระยะต่อหรือเพื่อส่งค่าพิกัดไปใช้ในการคำนวณเพื่อบังคับเลี้ยวล้อหน้า สัญลักษณ์ QR code จึงมีความเหมาะสมมากกว่าที่จะใช้เป็นสัญลักษณ์ติดไว้กับ AGV ตัวนำ เพราะสามารถทำงานได้ครอบคลุมในช่วงความเข้มแสงที่กว้างมากกว่าคือช่วง 80 ถึง 650 lux

### 5.1.2 ผลการทดลองการวัดระยะห่างด้วยกล้อง

#### 5.1.2.1 ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 80 ถึง 300 lux

การวัดระยะห่างด้วยตัวตรวจรู้ที่เป็นกล้อง ทำการทดลองเปรียบเทียบสัญลักษณ์สองสัญลักษณ์ ระหว่าง QR code และ Color ในระยะตั้งแต่ 20 ถึง 150 เซนติเมตร ในช่วงสภาพแวดล้อมความเข้มแสง 80 ถึง 300 lux พบว่า QR code สามารถตรวจจับและวัดระยะได้ครอบคลุมช่วงความเข้มแสงและมีค่าความผิดพลาดมากที่สุดเท่ากับ 2.5 เซนติเมตรที่ระยะการตรวจวัด 140 เซนติเมตร เมื่อเทียบกับ Color ซึ่งมีความสามารถตรวจจับในช่วงความเข้มแสงดังกล่าวได้แค่ระยะ 40 เซนติเมตร

#### 5.1.2.2 ในสภาพแวดล้อมที่มีความเข้มแสงในช่วง 300 ถึง 650 lux

แต่เมื่อทำการทดลอง ในช่วงสภาพแวดล้อมความเข้มแสง 300 ถึง 650 lux ซึ่งมีความสว่างมากกว่าช่วงความเข้มแสงการทดลองก่อน พบว่าทั้ง QR code และ Color สามารถตรวจจับและวัดระยะได้ครอบคลุมช่วงความเข้มแสง 300 ถึง 650 lux และมีค่าความผิดพลาดมากที่สุดสำหรับ QR code เท่ากับ 3 เซนติเมตรที่ระยะการตรวจวัด 150 เซนติเมตร เมื่อเทียบกับ Color มีค่าความผิดพลาดมากที่สุดเท่ากับ 3 เซนติเมตรที่ระยะการตรวจวัด 130 เซนติเมตร

ดังนั้น การวัดระยะเพื่อส่งค่าระยะห่างไปใช้ในการคำนวณเพื่อเพื่อควบคุมการทำงานของล้อหลัง สัญลักษณ์ QR code จึงมีความเหมาะสมมากกว่าที่จะใช้เป็นสัญลักษณ์ติดไว้กับ AGV ตัวนำ เพราะสามารถทำงานได้ครอบคลุมในช่วงความเข้มแสงที่กว้างมากกว่าคือช่วง 80 ถึง 650 lux แต่อย่างไรก็ตาม ถ้าหากเปรียบเทียบกันในเรื่องของการวัดระยะทั้งสองสัญลักษณ์เหมาะสมทั้งคู่เพราะค่าความคลาดเคลื่อนมีค่าไม่เกิน 5 เซนติเมตร ในระยะการวัดที่ใกล้เคียงกัน

### 5.1.3 ผลการทดลองการวัดระยะห่างด้วยเซนเซอร์

การวัดระยะห่างด้วยเซนเซอร์เป็นค่าตัวแปรที่สำคัญที่สุดในการควบคุมการติดตาม AGV ตัวนำ เพราะระยะที่วัดได้จะถูกส่งไปคำนวณเพื่อสั่งการควบคุมรีเลย์มอเตอร์ล้อหลัง และการวัดระยะของงานวิจัยนี้จะใช้เซนเซอร์เป็นตัวตรวจรู้หลัก และใช้กล้องวัดระยะเป็นตัวตรวจรู้สำรองจากการทดลองจึงสรุปได้ว่า Ultrasonic มีความแม่นยำมากกว่า Max Sona อยู่ 15.75 เปอร์เซ็นต์ ที่มุมด้านหน้าการวัดระยะเท่ากับ 15 องศาเท่ากัน ดังนั้นงานวิจัยนี้จึงเลือกใช้ Ultrasonic เนื่องจากมีความแม่นยำมากกว่าและมีราคาที่ต่ำกว่า Max Sona ถึง 10 เท่า

### 5.1.4 ผลการทดสอบการเข้าจอดสถานีและเริ่มต้นออกจากสถานี

การทดสอบทำซ้ำของการจอดและการเริ่มต้นออกจากสถานี ได้ข้อสรุปดังนี้ การเริ่มต้นออกจากสถานีทางด้านซ้ายมีค่าความคลาดเคลื่อนมากที่สุดไม่เกิน 10 เซนติเมตร และ การเริ่มต้นออกจากสถานีทางด้านขวามีค่าความคลาดเคลื่อนมากที่สุดไม่เกิน 30 เซนติเมตร จึงสรุปได้ว่าการเคลื่อนที่ออกจากสถานีทางด้านซ้ายจะมีค่าความคลาดเคลื่อนน้อยกว่าทางด้านขวาเพราะฉะนั้น

การเริ่มต้นออกจากสถานีควรมีค่าความคลาดเคลื่อนทั้งด้านซ้ายและด้านขวา 30 เซนติเมตรเป็นอย่างน้อย

สำหรับการทดสอบเข้าจอดในสถานีทางด้านซ้ายมีค่าความคลาดเคลื่อนมากที่สุดไม่เกิน 20 เซนติเมตร ในขณะที่การเข้าจอดในสถานีทางด้านขวามีค่าความคลาดเคลื่อนมากที่สุดไม่เกิน 10 เซนติเมตร จึงสรุปได้ว่าการเคลื่อนที่เข้าจอดในสถานีทางด้านขวาจะมีค่าความคลาดเคลื่อนน้อยกว่าทางด้านซ้ายเพราะฉะนั้นการเข้าจอดในสถานีควรมีค่าความคลาดเคลื่อนทั้งด้านซ้ายและด้านขวา 20 เซนติเมตรเป็นอย่างน้อย เพื่อให้เข้าจอดได้ในพื้นที่จอดที่จำกัด และป้ายสะท้อนแสงควรวางในตำแหน่งถัดจากสถานีจอด 1 เมตร จึงจะทำให้เข้าจอดในสถานีจอดได้อย่างแม่นยำ

### 5.1.5 ผลการทดสอบการทำงานร่วมกันของการวัดระยะและการส่งสัญญาณควบคุมมอเตอร์

การทดสอบการทำงานร่วมกันระหว่างการวัดระยะและการส่งงานควบคุมรีเลย์ที่ส่งสัญญาณให้มอเตอร์หลังทำงานขับ AGV ตัวตามเพื่อรักษาระยะห่าง พบว่าเมื่อระยะการวัดมีค่ามากกว่า 90 เซนติเมตร รีเลย์จะส่งสัญญาณให้มอเตอร์ขับหลังทำงานเดินหน้า จนกว่าระยะการวัดจะมีค่าน้อยกว่า 90 เซนติเมตร ถ้าหากน้อยกว่า 90 เซนติเมตร รีเลย์จะหยุดสั่งการทำงานทำให้มอเตอร์ขับหลังหยุดทำงาน แต่จากผลการทดลองดังรูปที่ 4.15 จะพบว่า ระยะการวัดจะมีค่าลดลงไปถึง 70 เซนติเมตรแล้วค่อย ๆ เพิ่มขึ้นเป็น 90 เซนติเมตร เหตุใดจึงเป็นเช่นนั้นเพราะการหยุดสั่งงานของรีเลย์ไม่ได้ทำให้รถ AGV หยุดทันที รถ AGV มีน้ำหนักจึงเกิดการไถลของตัวรถ ทำให้ระยะตกลงไปที่ 70 เซนติเมตร

## 5.2 ข้อเสนอแนะ

5.2.1 การบังคับเลี้ยวที่ใช้กล้องเป็นตัวตรวจรู้เพียงตัวเดียว จะส่งผลให้เกิดปัญหาเคลื่อนที่ไม่ติดตามได้ในกรณีที่ไม่สามารถตรวจจับสัญลักษณ์ที่ติดอยู่กับ AGV ตัวนำ ดังนั้นผู้วิจัยจึงมีข้อเสนอแนะว่าควรมีตัวตรวจรู้สำรอง โดยการเพิ่มเซ็นเซอร์รูปแบบอื่น เพื่อให้ทราบตำแหน่งของ AGV ตัวนำว่าตำแหน่ง ณ ขณะนั้นอยู่ทางด้านซ้ายหรือขวาของ AGV ตัวตาม

5.2.2 ฮาร์ดแวร์การบังคับเลี้ยวควรเป็นการควบคุมแบบปิด เพื่อที่จะได้ทราบถึงสถานะของล้อหน้าในขณะที่มีการสั่งงานบังคับเลี้ยวในองศาที่ต้องการ งานวิจัยนี้ใช้ระบบการบังคับเลี้ยวแบบเปิด เมื่อสั่งการบังคับล้อหน้าตามองศาที่ต้องการ อาจเกิดข้อผิดพลาดได้เนื่องจากฮาร์ดแวร์บริเวณข้อต่อเกิดระยะฟรีและได้รับผลโดยตรงกับน้ำหนักของ AGV ยิ่งน้ำหนักมากการสั่งงานมอเตอร์จำเป็นต้องจ่ายกระแสมากเพื่อเพิ่มทอร์คให้มอเตอร์ยังคงบังคับเลี้ยวได้ ถ้าหากมอเตอร์บังคับเลี้ยวล้อหน้าเป็นการควบคุมแบบเปิด ระบบจะไม่สามารถทราบได้ว่า ณ ขณะนั้นตำแหน่งของล้อหน้าอยู่ที่ตำแหน่งใด จึงส่งผลให้การเคลื่อนที่ติดตามเกิดข้อผิดพลาดได้

5.2.3 การลดปัญหาการเกิดระยะฟรีในอุปกรณ์บังคับเลี้ยวสามารถแก้ไขได้ด้วยวิธีการเปลี่ยนรูปแบบการควบคุมเป็น Differential drive ซึ่งสามารถควบคุมการเคลื่อนที่แบบระบบปิดได้ด้วยการติดตั้งเซนเซอร์เอนโค้ดเดอร์ตรวจวัดตำแหน่งการหมุนของมอเตอร์หน้า แต่อาจจะส่งผลให้ราคาต้นทุน AGV สูงขึ้นตามไปด้วย



## รายการอ้างอิง

- Sierra-Garcia, J.Enrique. Combining reinforcement learning and conventional control to improve automatic guided vehicles tracking of complex trajectories. *Expert Systems*. Volume 41, Issue 2, Article number e13076. Accepted: 31 May 2022, doi: 10.1111/exsy.13076.
- Wikipedia contributors. Automated guided vehicle. Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 1 Mar. 2024, [online], <https://en.wikipedia.org/>.
- R. P. Saputra, E. Rijanto. Automatic Guided Vehicles System and Its Coordination Control for Containers Terminal Logistics Application. *ArXiv*. March 7, 2021, doi: 10.48550/arXiv.2104.08331.
- Munadi M, Radityo B, Ariyanto M, Taniai Y. Automated guided vehicle (AGV) lane-keeping assist based on computer vision and fuzzy logic control under varying light intensity. *Journal of Robotics and Intelligent Systems Engineering*, doi: 10.1016/j.rineng.2023.101678.
- Park M. Development of Steering Control System for Autonomous Vehicle Using Geometry-Based Path Tracking Algorithm. *ETRI Journal*. 2015 May; 37(3), doi: 10.4218/etrij.15.2314.0123.
- Diab M. Self-Driving Car Lane-keeping Assist using PID and Pure Pursuit Control. 2020 International Conference on Innovation and Intelligence for Informatics. Computing and Technologies (3ICT). 2020 December; doi: 10.1109/3ICT51146.2020.9311987.
- Toropov E, Tumasov A, Vashurin A, Butin D, Stepanov E. Hardware-in-the-Loop Testing of Vehicle's Electronic Stability Control System. *Applied Engineering Letters*, 8(2), 2023:70–79, doi: 10.18485/aeletters.2023.8.2.4.

- Park M. Development of lateral control module for zone (u-turn) maneuver of vehicle/driver cooperative autonomous driving system. In 2016 IEEE Transportation Electrification Conference and Expo. Asia-Pacific (ITEC Asia-Pacific). 2016 June 4; Busan: p.1-5, doi: 10.1109/ITEC-AP.2016.7513080.
- Wei W, Ke W, Yuanyuan C. Application of exception recognition and fuzzy decision in surveillance. In 2010 International Conference on Computer, Mechatronics, p.14, doi:10.1109/CMCE.2010.5610198.
- Hung S. Micrography QR Codes. IEEE Transactions on Visualization and Computer Graphics; 2020 September 1; 26(9):2834-2847, doi: 10.1109/TVCG.2019.2896895.
- Chen R, Yu Y, Xu X, Wang L, Zhao H, Tan H-Z. Adaptive Binarization of QR Code Images for Fast Automatic Sorting in Warehouse Systems. Sensors. 2019; 19(24):5466, doi: 10.3390/s19245466.
- Karrach L, Pivarčiová E, Bozek P. Recognition of Perspective Distorted QR Codes with a Partially Damaged Finder Pattern in Real Scene Images. Applied Sciences. 2020; 10(21):7814, doi: 10.3390/app10217814.
- Magnier B, Hayat K. Revisiting Mehrotra and Nichani's Corner Detection Method for Improvement with Truncated Anisotropic Gaussian Filtering. Sensors (Basel). 2023 Oct 23; 23(20):8653, doi: 10.3390/s23208653.
- R. P. Saputra and E. Rijanto, "Automatic Guided Vehicles System and Its Coordination Control for Containers Terminal Logistics Application," ArXiv, March 7, 2021, DOI: 10.48550/arXiv.2104.08331.
- Munadi, M., Radityo, B., Ariyanto, M., & Taniai, Y. "Automated guided vehicle (AGV) lane-keeping assist based on computer vision and fuzzy logic control under varying light intensity" Journal of Robotics and Intelligent Systems Engineering, DOI: 10.1016/j.rineng.2023.101678
- Sierra-Garcia, J.Enrique. "Combining reinforcement learning and conventional control to improve automatic guided vehicles tracking of complex trajectories. Expert Systems" Volume 41, Issue 2, Article number e13076. Accepted: 31 May 2022. DOI: 10.1111/exsy.13076
- Hung, S.-H."Micrography QR Codes". IEEE Transactions on Visualization and Computer Graphics, 26(9), 2834-2847. DOI: 10.1109/TVCG.2019.2896895.



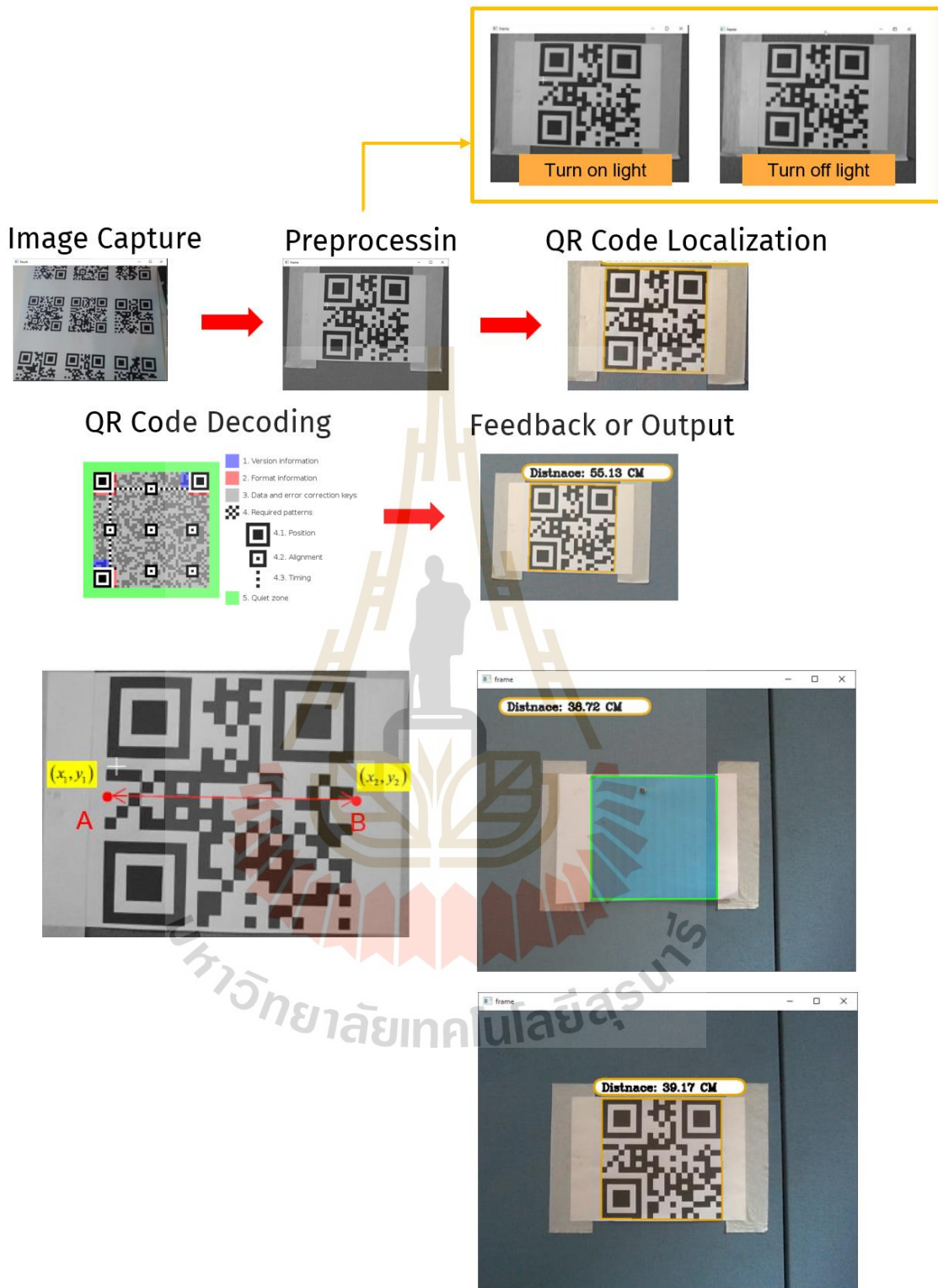
- Wei, W., Ke, W., & Yuanyuan, C. "Application of exception recognition and fuzzy decision in surveillance" In 2010 International Conference on Computer, Mechatronics, (pp.14).DOI:10.1109/CMCE.2010.5610198.
- Park, M. W. (2016). "Development of lateral control module for zone (u-turn) maneuver of vehicle/driver cooperative autonomous driving system". In 2016 IEEE Transportation Electrification Conference and Expo, Asia-Pacific (ITEC Asia-Pacific) (pp. 1-5). DOI: 10.1109/ITEC-AP.2016.7513080.
- Wu, Y. (2021). Steering Wheel AGV Path Tracking Control Based on Improved Pure Pursuit Mode. Journal of Physics Conference Series, 2093(1), 012005. DOI: 10.1088/1742-6596/2093/1/012005.
- Park, M. (2015). "Development of Steering Control System for Autonomous Vehicle Using Geometry-Based Path Tracking Algorithm". ETRI Journal, 37(3). DOI: 10.4218/etrij.15.2314.0123.
- Xu, L. "Path Tracking of a 4WIS-4WID Agricultural Machinery Based on Variable Look-Ahead Distance" Applied Sciences, 12(17), 8651. DOI: 10.3390/app12178651.
- Sukhil, V., & Behl, M. "Adaptive Lookahead Pure-Pursuit for Autonomous Racing" arXiv. Computer Science Robotics ,17 Nov 2021. DOI: 10.48550/arXiv.2111.08873.
- Diab, M. "Self-Driving Car Lane-keeping Assist using PID and Pure Pursuit Control" 2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT). DOI: 10.1109/3ICT51146.2020.9311987.
- Chen, R., Yu, (2019, December 11). "Adaptive Binarization of QR Code Images for Fast Automatic Sorting in Warehouse Systems. Sensors." 19(24), 5466. DOI: 10.3390/s19245466
- Chakravorty, P. (2018, September). "What Is a Signal? Lecture Notes". IEEE Signal Processing Magazine, 35(5), 175-177. DOI: 10.1109/MSP.2018.2832195.
- Zhang, L. (2023). Research on Vehicle Detection Algorithms Based on YOLOv5s. IAENG International Journal of Computer Science, 50(4), Article number IJCS\_50\_4\_34.
- Carotenuto, R. (2019, July). An Indoor Ultrasonic System for Autonomous 3-D Positioning. IEEE Transactions on Instrumentation and Measurement, 68(7), 2507-2518. DOI: 10.1109/TIM.2018.2866358.



ภาคผนวก ก

โปรแกรมตรวจจับสัญญาณและวัดระยะห่างด้วยกล้อง

มหาวิทยาลัยเทคโนโลยีสุรนารี



```

import cv2 as cv
from datetime import datetime
import time
import numpy as np
import math
import pyzbar.pyzbar as pyzbar
import betterLook
import time
import csv
import serial

serial_port = 'COM3'
ser = serial.Serial(serial_port, 9600, timeout=1)
time.sleep(2)

# define the fonts
fonts = cv.FONT_HERSHEY_COMPLEX
Pos = (50,50)

# colors (BGR)
WHITE = (255, 255, 255)
BLACK = (0, 0, 0)
MAGENTA = (255, 0, 255)
GREEN = (0, 255, 0)
CYAN = (255, 255, 0)
GOLD = (0, 255, 215)
YELLOW = (0, 255, 255)
ORANGE = (0, 165, 230)
RED = (0,0,255)

# Variable
camID = 0

# Define threshold variables
threshold_left = 220
threshold_right = 420

disBack = 50
disGo = 90

# Real world measured Distance and width of QR code
KNOWN_DISTANCE = 110 # Cm
KNOWN_WIDTH = 16 # Cm

disBack = 0.5
disGo = 0.9
limit = 0.3
upper = 3

```

```

# functions

# finding angle
def findangle(knowDistance,focalLength,widthInImage):

    Width = (widthInImage * knowDistance)/focalLength
    angle_rad = math.atan(Width/knowDistance)
    angle_deg = math.degrees(angle_rad)

    return angle_deg

# finding Distance between two points
def eucaldainDistance(x, y, x1, y1):
    eucaldainDist = math.sqrt((x1 - x) ** 2 + (y1 - y) ** 2)

    return eucaldainDist

# focal length finder function
def focalLengthFinder(knowDistance, knownWidth, widthInImage):
    focalLength = ((widthInImage * knowDistance) / knownWidth)

    return focalLength

def distanceFinder(focalLength, knownWidth, widthInImage):
    distance = ((knownWidth * focalLength) / widthInImage)

    return distance

```

```

=====
cam = cv.VideoCapture(camID)

height, width = 480,640
# Set the desired width and height for the webcam capture
cam.set(cv.CAP_PROP_FRAME_WIDTH, height)
cam.set(cv.CAP_PROP_FRAME_HEIGHT, width)

refernceImage = cv.imread("100cm.png")
# getting the width of QR code in the reference image
Rwidth = DetectQRcode(refernceImage,0,0,threshold_left, threshold_right)

# finding the focal length
focalLength = focalLengthFinder(KNOWN_DISTANCE, KNOWN_WIDTH, Rwidth[0])

wheel = 0
none_count = 0
#time.sleep(60)
=====
while True:
    ret, frame = cam.read()

    #capture_frame()
    cen_x = width // 2 # Calculate the x-coordinate of the center
    cen_y = height // 2 # Calculate the y-coordinate of the center

    # Draw a circle at the center of the frame
    cv.circle(frame, (cen_x, cen_y), 5, (0, 255, 0), -1)

    cv.line(frame, (threshold_left, 0), (threshold_left, 480), (0, 0, 255), 1)
    cv.line(frame, (threshold_right, 0), (threshold_right, 480), (0, 0, 255), 1)

    # finding width of QR code width in the frame
    codewidth = DetectQRcode(frame, cen_x, cen_y, threshold_left, threshold_right)

```

```

def DetectQRcode(image, center_x, center_y, threshold_left, threshold_right):
    direction = 0
    codewidth = 0
    x, y = 0, 0
    euclaDistance = 0
    global Pos3,Pos1
    # convert the color image to gray scale image
    Gray = cv.cvtColor(image, cv.COLOR_BGR2GRAY)

    # create QR code object
    objectQRcode = pyzbar.decode(Gray)
    for obDecoded in objectQRcode:
        myData = obDecoded.data.decode('utf-8')
        rect_points = obDecoded.rect

        left, top, width, height = rect_points[0], rect_points[1], rect_points[2], rect_points[3]
        center_xQR = int(left + width / 2)
        center_yQR = int(top + height / 2)

        cv.circle(image, (center_xQR, center_yQR), 5, (255, 0, 255), -1)
        #cv.line(image, (center_xQR, center_y), (center_x, center_y), (0, 0, 255), 5)

        points = obDecoded.polygon
        if len(points) > 4:
            hull = cv.convexHull(
                np.array([points for point in points], dtype=np.float32))
            hull = list(map(tuple, np.squeeze(hull)))
        else:
            hull = points

        n = len(hull)
        # draw the lines on the QR code
        for j in range(0, n):

            cv.line(image, hull[j], hull[(j + 1) % n], ORANGE, 2)

        # finding width of QR code in the image
        x, x1, x2, x3 = hull[0][0], hull[1][0], hull[2][0], hull[3][0]
        y, y1, y2, y3 = hull[0][1], hull[1][1], hull[2][1], hull[3][1]

        Pos0 = hull[0]
        Pos1 = hull[1]
        Pos2 = hull[2]
        Pos3 = hull[3]

        # using Eucaldain distance finder function to find the width
        euclaDistance = eucaldainDistance(x, y, x1, y1)

        # using Eucaldain distance finder function to find the width
        PointAngleDistance = eucaldainDistance(center_xQR, center_y, center_x, center_y)

        # retruing the Eucaldain distance/ QR code width other words
        return euclaDistance,PointAngleDistance,center_xQR

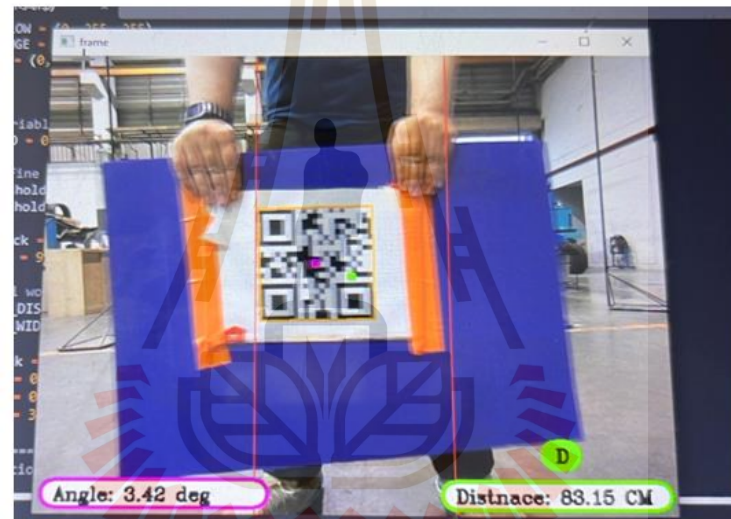
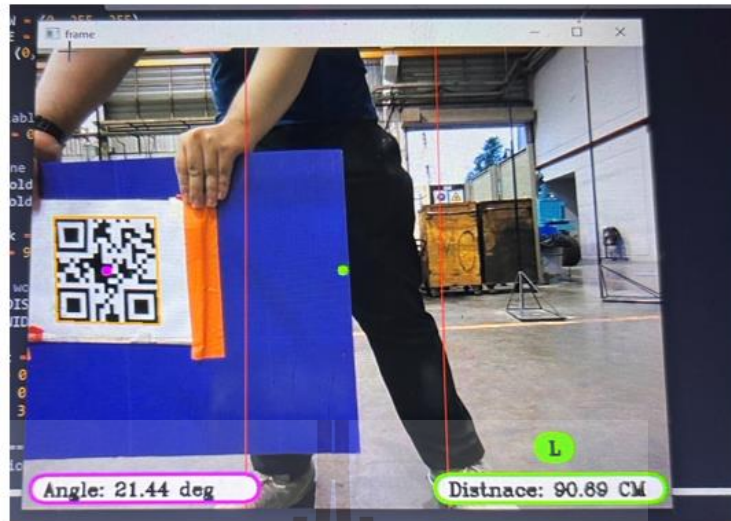
```



ภาคผนวก ข

โปรแกรมวัดมุมระหว่างสัญลักษณ์กับจุดกึ่งกลางเฟรม

มหาวิทยาลัยเทคโนโลยีสุรนารี





```

=====
while True:
    ret, frame = cam.read()

    #capture_frame()
    cen_x = width // 2 # Calculate the x-coordinate of the center
    cen_y = height // 2 # Calculate the y-coordinate of the center

    # Draw a circle at the center of the frame
    cv.circle(frame, (cen_x, cen_y), 5, (0, 255, 0), -1)

    cv.line(frame, (threshold_left, 0), (threshold_left, 480), (0, 0, 255), 1)
    cv.line(frame, (threshold_right, 0), (threshold_right, 480), (0, 0, 255), 1)

    # finding width of QR code width in the frame
    codewidth = DetectQRcode(frame, cen_x, cen_y, threshold_left, threshold_right)

    if codewidth is not None:
        none_count = 0

        # print("not none")
        Distance = distanceFinder(focalLength, KNOWN_WIDTH, codewidth[0])
        # cv.putText(frame, f"Distance: {Distance}", (50,50), fonts, 0.6, (GOLD), 2)
        Position = thinking(Distance,codewidth[2])

        S_angle = findangle(Distance,focalLength,codewidth[1])
        #print(S_angle)

        betterLook.showText(frame, f"Distnace: {round(Distance,2)} CM", (420,480), GREEN)
        betterLook.showText(frame, f"Angle: {round(S_angle,2)} deg", (20,480), MAGENTA)
        betterLook.showDirec(frame, Position ,(520,440), GREEN)

    elif codewidth is None:
        betterLook.showText(frame, "Not detect", (420,480), RED)
        betterLook.showText(frame, "Not detect", (20,480), RED)

    cv.imshow("frame", frame)
    key = cv.waitKey(1)

    if cv.waitKey(1) & 0xFF == ord('q'):
        break

cam.release()
cv.destroyAllWindows()

```

```

# finding angle
def findangle(knowDistance,focalLength,widthInImage):

    Width = (widthInImage * knowDistance)/focalLength
    angle_rad = math.atan(Width/knowDistance)
    angle_deg = math.degrees(angle_rad)

    return angle_deg

```



ภาคผนวก ค

รายละเอียดการต่อวงจรบอร์ดคอนโทรลและเซนเซอร์ของตัวรถ AGV  
โปรแกรมควบคุมมอเตอร์บังคับเลี้ยวและรักษาระยะห่าง

มหาวิทยาลัยเทคโนโลยีสุรนารี



```
char input;
char stat;
char prev_stat = ' ';
char prev_input = ' ';

//=====varcontrol=====
const int LED4 = 5; //relay_4
const int LED5 = 6; //relay_5

//=====state=====
unsigned long interval_2 = 900; //set time for front wheel turn little
unsigned long interval_3 = 900; //set time for front wheel turn little
//=====function=====

void leftwheel_little() {
    digitalWrite(LED4, HIGH);
    delay(interval_2);
    digitalWrite(LED4, LOW);
}

void rightwheel_little() {
    digitalWrite(LED4, HIGH);
    digitalWrite(LED5, HIGH);
    delay(interval_3);
    digitalWrite(LED4, LOW);
    digitalWrite(LED5, LOW);
}

//=====setup=====
void setup() {
    Serial.begin(9600);
    pinMode(LED4, OUTPUT);
    pinMode(LED5, OUTPUT);
}
```

```
void loop() {  
  
  if (Serial.available() > 0) {  
    input = Serial.read();  
    Serial.print(input);  
  }  
  
  //=====logic=====  
  if (input == 'R'){  
    if (prev_input == 'L' || prev_input == 'D' || prev_input == ' '){  
      rightwheel_little();  
    }  
    prev_input = input;  
  }  
  
  else if (input == 'L'){  
    if (prev_input == 'R' || prev_input == 'D' || prev_input == ' '){  
      leftwheel_little();  
    }  
    prev_input = input;  
  }  
  
  else if (input == 'D'){  
    if (prev_input == 'R' ){  
      leftwheel_little();  
    }  
    if (prev_input == 'L' ){  
      rightwheel_little();  
    }  
    prev_input = input;  
  }  
  // Serial.print("prev_stat: ");  
  // Serial.println(prev_stat);  
  // Serial.print("stat: ");  
  // Serial.print(stat);  
}
```

```
#define MAX_INPUTS 4

const int receiverPin = 7; // Receiver for forward/backward control
const int receiverPin1 = 8;

bool firstForward = true;
bool firstBackward = true;

const int Relay_R = 12; // ตัวขาเห็นหน้าไปทางเดียวกับรถ
const int Relay_L = 9;

char inputs[MAX_INPUTS];
int inputIndex = 0;
int count_start = 0;
int count_park = 0;
bool countbreak = true;

char input;
char stat;
char prev_stat = ' ';
char prev_input = ' ';

unsigned long lastUltraCheckTime = 0;
const unsigned long ultraCheckInterval = 500;

//=====varcontrol=====
const int LED1 = 2; //relay_1
const int LED2 = 3; //relay_2
const int LED3 = 4; //relay_3
const int LED4 = 5; //relay_4
const int LED5 = 6; //relay_5

//=====varbutton=====
const int buttonPin_1 = 13; //sensor for park
```

```

//=====ultrasonic=====

int trigPin = 10;    // TRIG pin
int echoPin = 11;   // ECHO pin

const int disGo = 80;
const int upper = 300;

float duration_us, disUltra_cm;

//=====state=====
int button1State = 0;
int button2State = 0;

int lastButton1State = 0;
int lastButton2State = 0;

unsigned long interval = 1800; //time turn front wheel
unsigned long interval_1 = 9200; // time start
unsigned long interval_2 = 900; //set time for front wheel turn little
unsigned long interval_3 = 800; //set time for front wheel turn little

```

```

//=====setup=====
void setup() {
  Serial.begin(9600);
  pinMode(receiverPin, INPUT);
  pinMode(receiverPin1, INPUT);

  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(LED4, OUTPUT);
  pinMode(LED5, OUTPUT);
  pinMode(buttonPin_1, INPUT_PULLUP);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(Relay_R, OUTPUT);
  pinMode(Relay_L, OUTPUT);
}

```

```
//=====function=====
void stop() {
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
    firstForward = true;
    firstBackward = true;
}

void forward() {
    digitalWrite(LED1, HIGH);
    digitalWrite(LED3, HIGH);
}

void backward() {
    digitalWrite(LED1, HIGH);
}

void shortcircuit() {
    delay(400);
    digitalWrite(LED2, HIGH);
    delay(200);
    digitalWrite(LED2, LOW);
}

void leftwheel() {
    digitalWrite(LED4, HIGH);
    delay(interval);
    digitalWrite(LED4, LOW);
}

void leftwheel_1() {
    digitalWrite(LED4, HIGH);
    delay(interval_2);
    digitalWrite(LED4, LOW);
}

void rightwheel() {
    digitalWrite(LED4, HIGH);
    digitalWrite(LED5, HIGH);
    delay(interval);
    digitalWrite(LED4, LOW);
    digitalWrite(LED5, LOW);
}
```



```

void loop() {

  // Read the pulse widths from the receiver channels
  unsigned long pulseWidth = pulseIn(receiverPin, HIGH);
  unsigned long pulseWidth1 = pulseIn(receiverPin1, HIGH);

  if (pulseWidth1 < 1300) {
    digitalWrite(LED4, HIGH);
    //Serial.println("Receiver 2 Action: LED1 ON");
  } else if (pulseWidth1 > 1500) {
    digitalWrite(LED4, HIGH);
    digitalWrite(LED5, HIGH);
    //Serial.println("Receiver 2 Action: LED4 and LED5 ON");
  } else if (pulseWidth1 >= 1300 && pulseWidth1 <= 1500) {
    digitalWrite(LED4, LOW);
    digitalWrite(LED5, LOW);
    //Serial.println("Receiver 2 Action: LED4 and LED5 OFF");
  }

  // Process serial input
  if (Serial.available() > 0) {
    input = Serial.read();
    //Serial.println(input);
    if (inputIndex < MAX_INPUTS) {
      inputs[inputIndex] = input;
      inputIndex++;
      Serial.print(inputs[0]);
      Serial.print(inputs[1]);
      Serial.println(inputs[2]);
    }

    if (inputs[2] == 'S' && input != 'K') {
      count_start ++;
      if (inputs[0] == 'A' && inputs[1] != 'B' && count_start == 1){
        startleft();
      } else if (inputs[0] == 'B' && inputs[0] != 'A' && count_start == 1){
        startright();
      }
    }

    if (input == 'K') {
      inputIndex = 0;
      count_start = 0;
      count_park =0;
      memset(inputs, 0, sizeof(inputs));
    }
  }
}

```

```

//=====logic=====
if ( disUltra_cm < disGo+10 ){
  stop();
  //brake();
  prev_stat = stat;
  stat = '1';
}
if ( disUltra_cm > upper-10 ){
  if (pulseWidth > 1500) { // Forward
    forward();
    if (firstForward) {
      shortcircuit();
      firstForward = false; // Set the flag to false after first call
    }
  }
  if (pulseWidth < 1300) { // Forward
    digitalWrite(LED1, HIGH);
    shortcircuit();
    delay(2000);
    digitalWrite(LED1, LOW );
  }
}

if (disGo-10 <= disUltra_cm && disUltra_cm <= upper-10 && timmer > 120 ){
  forward();
  prev_stat = stat;
  if (prev_stat == '1' ) {
    shortcircuit();
  }
  stat = '2';
}

//Serial.print("prev_stat: ");
//Serial.println(prev_stat);
//Serial.print("stat: ");
// Serial.println(stat);
}

```

```

//=====sensor=====
button1State = digitalRead(buttonPin_1);
if (button1State != lastButton1State) {
  if (button1State == LOW) {
    count_park ++;
    if (inputs[1] == 'C' && inputs[1] != 'D' && count_park == 1) {
      parkleft();
    }
    else if (inputs[1] == 'D' && inputs[1] != 'C' && count_park == 1) {
      parkright();
    }
  }
}
lastButton1State = button1State;
delay(50);
}

//=====ultrasonic=====
unsigned long currentMillis = millis();
unsigned long timmer = currentMillis/1000;

if (currentMillis - lastUltraCheckTime >= ultraCheckInterval) {
  lastUltraCheckTime = currentMillis;
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration_us = pulseIn(echoPin, HIGH);
  disUltra_cm = ((0.017 * duration_us) + 5);
}

```

ภาคผนวก ง

บทความทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างการศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี

## รายชื่อบทความที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

เจษฎากร เหมือนงาม และจิระพล ศรีเสริฐผล. (2567) “การจำลองพฤติกรรมของระบบระบายความร้อนด้วยไฟฟ้าของรถบัสโดยสาร ด้วยโปรแกรม MATLAB Simulink” งานประชุมวิชาการ The 2nd International Conference on Mechanics, Electronics, Automation and Automatic Control. วันที่ 16 มิถุนายน พ.ศ.2567 จำนวน 7 หน้า.

Chetsadakorn M. , Nopparut Kh. , Jiraphon S. and Natthawut P. , (2024) **Exploring Techniques for Estimating Distance in Leader-Follower AGV Systems.** The 2 nd International Conference on Mechanics, Electronics, Automation and Automatic Control. 16 – 18 June. Ordos. China.



# Exploring techniques for estimating distance in leader-follower AGV systems

Chetsadakorn Muenngam, Nopparut Khaewnak, Jiraphon Srisertpol and Natthawut Prasoetkam<sup>1</sup>

*Mechatronics Engineering Program, School of Mechanical Engineering, Suranaree University of Technology, 111 University avenue, Muang district, Nakhon Ratchasima, 30000, Thailand*

ORCID ID: Jiraphon Srisertpol <https://orcid.org/0000-0002-4341-3708>

**Abstract.** The application of Automated Guided Vehicles (AGVs) for workpiece transportation in production processes is commonplace, particularly in the context of Industry 4.0. However, the high cost associated with AGVs presents a significant challenge. This paper proposes an approach to AGV tracking using a combination of camera-based QR code detection and ultrasonic sensors. The objective is to regulate the distance between leading and following AGVs, thereby addressing the operational costs associated with expensive equipment. Experimental investigations were conducted to assess the performance of QR symbols and color bars under varying lighting conditions, ranging from 200 lux to 650 lux. Accurate distance measurements were achieved within a range of 150 cm. In lower lighting conditions (100 lux to 250 lux), the RGB color bar detection failed at distances below 40 cm, whereas QR Code detection remained robust. Angle measurements demonstrated accurate readings up to 30 degrees from the camera's center, within distances of 50 to 125 cm, with an anticipated deviation of no more than 0.5 degrees. Subsequently, QR code symbols were chosen as the experimental control symbol, complemented by ultrasonic sensors for inter-vehicle distance maintenance. During practical testing, if the measured distance falls below 60 cm. while moving in a straight line, the motor ceases operation. Conversely, if the distance exceeds 80 cm, the motor activates, allowing the follower AGV to uphold the specified distance behind the leader AGV.

**Keywords.** QR code, ultrasonic, follower AGV, distance control

## 1. Introduction

Warehousing is another rapidly evolving area, driven by the high demand in the current E-commerce business landscape [1]. The surge in online orders from consumers has led to significant growth in warehousing activities, averaging a 26% annual increase from 2020 to 2023, according to data from the Kasikorn Research Center. With this heightened growth comes increased market competition, necessitating cost reductions to remain competitive. One way to achieve this is by leveraging technology to replace labor-intensive tasks, such as utilizing Automated Guided Vehicles (AGVs) in logistics operations [2]. AGVs are autonomous vehicles commonly used in logistics [3], offering the potential to streamline operations and reduce labor costs.

AGV (Automated Guided Vehicle) is a type of vehicle commonly referred to as an unmanned transport vehicle. It is utilized for transporting goods both within and outside

<sup>1</sup> Corresponding Author: Natthawut Prasoetkarn, Email: natthawut.p@g.sut.ac.th.

buildings. Operating on the principle of adhering to a predetermined path without encountering obstacles, AGVs navigate by following a line, utilizing laser guidance, or employing cameras to detect movement direction. They incorporate PID control and fuzzy logic for lane monitoring [4, 5, 6, 7], alongside side control for zone regulation (including U-turns) within the vehicle/driver cooperative autonomous driving system [8]. Adaptation to dynamically changing environmental colors within image sequences [9] enhances AGVs' precision in maneuvering cargo and heavy equipment. However, a notable drawback is their high cost, resulting in prolonged return on investment. Moreover, the expense escalates with the need for multiple AGVs. Consequently, researchers are keen on developing more cost-effective AGVs that uphold quality standards and fulfill their intended functions, thus reducing overall business expenditure.

This research presents the concept of using a single guiding vehicle (AGV) along with multiple following vehicles, where the AGV leader does not necessarily need to carry a heavy load, while the AGV follower can handle increased weight. Therefore, a motion control system for AGV following is designed using cameras to detect specific symbols attached to the adjustable AGV within range. The system maintains distances by measuring with distance sensors and commands the AGV to move accordingly, including obstacle avoidance.

## 2. Materials

From the research studied, it was found that there are equipment used for AGVs such as lidar cameras, ultrasonic sensors, and so on. Researchers attempted to reduce equipment costs, so they used the equipment as shown in Table 1.

**Table 1.** Experimental equipment.

Equipment	Specification
Computer	Intel(R) Core (TM) i7-8700 CPU @ 3.20GHz 16 GB ram, 64-bit
Camera	The Intel® RealSense™ D455, Up to 90 fps, Depth Accuracy <2% at 4 m
Controller	Arduino Mega 2560, Voltage 0-5 volts, receive/send values, 54 digital input/output pins, connected by USB
QR code	Dimension 21x25 cm
Color paper	Dimension 21x25 cm
Lux Light Meter	UNI-T UT383S
Light meter	Digital light meter UT383S
Distance Sensor	Ultrasonic sensor Arduino module.



(a)



(b)

**Figure 1.** Hardware communication (a) and Warehouse area ranging from 100 lux to 300 lux (b).

From Table 1. Install a camera and ultrasonic sensor at the semi-central front of the car with a height of 30 cm. from the ground. The camera will be connected to the computer via USB type C for image processing, while the ultrasonic sensor will be connected to an Arduino. The Arduino will send distance measurement values to the

computer for processing and control the operation of a relay to control the motors show in Figure 1(a).

In the experiment, experimenting with QR code detection and color image recognition in a warehouse area ranging from 100 lux to 300 lux show in Figure 1(b), without any other interfering light. At the lower light intensity of 100 lux, object detection, whether QR codes or color images, may be affected. Therefore, the experimental control for the movement of the vehicle will be conducted at 300 lux as the standard. The results of the experiment comparing QR code detection with color image recognition will be demonstrated accordingly.

### 3. Methods

The research is divided into two main parts. The first part focuses on determining the distance from the following vehicles to the symbols. This can be achieved by using both cameras to measure the distance from the color stripes and QR codes within the range of 30 - 150 cm and using ultrasonic sensors to measure distances within the same range. The experiment will be conducted with light intensity values ranging from 100 to 650 lux to be used in the operation of the following vehicles.

#### 3.1. Finding distance

3.1.1 Finding the distance from a QR code (Quick Response code) [10], which is a type of barcode, involves utilizing a two-dimensional barcode that was created in 1994 by the Japanese company Denso Wave for labeling automotive parts. A QR code consists of black squares arranged in a square grid on a white background. QR codes are utilized to identify, track, or inspect items in production, storage, and distribution processes. Additionally, they are employed to assist visually impaired individuals and the blind in obtaining relevant information [11, 12].

When receiving images from the camera, they will be compared with the original image. Then, the angle detection process [13] will be utilized to find the angles. Subsequently, the coordinates  $(x_1, y_1)$  and  $(x_2, y_2)$  will be determined to find the line width using equation (1).

$$D = \frac{f \times W}{d} \quad (1)$$

$f$  is focal length (focal length from the lens to the camera sensor)

$W$  is the width of the actual object.

$d$  is the width of the camera image.

$D$  is the distance between the camera and the actual object.

From the experiment, Reacher tested the values of  $f$  and  $W$  by utilizing the prototype image and measuring the distance between the camera and the QR code. The QR code has a size of 16x16 cm. Using Equation 1, Reacher obtained the value of  $f$ . Additionally, when measuring the width of the image, the camera allows us to determine the distance between the vehicle and the QR code based on Equation 1. When the distance is less than

R, which is the specified distance, R is transmitted to the vehicle control board. Conversely, when the distance is greater than F, which is the specified distance, F is sent to the vehicle control board. Lastly, when the value falls between R and F, S is transmitted to the vehicle control board.

3.1.2 Measuring the distance from the square color with a size of 16x16 cm, where blue color was used in the test. Subsequently, the angle detection process [12] will be employed to find  $(x_1, y_1)$  and  $(x_2, y_2)$  in order to determine the width equation (2).

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (2)$$

3.1.3 Determining the distance from an ultrasonic sensor entails measuring the time between sending a signal and receiving it. To calculate the distance from the measured time, the speed of the wave is multiplied by two.

### 3.2. Controlling the operation of the vehicle in moving forward and stopping

- When the control board receives the letter 'F' through the serial port and the ultrasonic sensor measures a distance greater than 80, it will cause the vehicle to move forward.
- When the control board receives the letter 'S' through the serial port, the vehicle to stop operating.

## 4. Experiment results

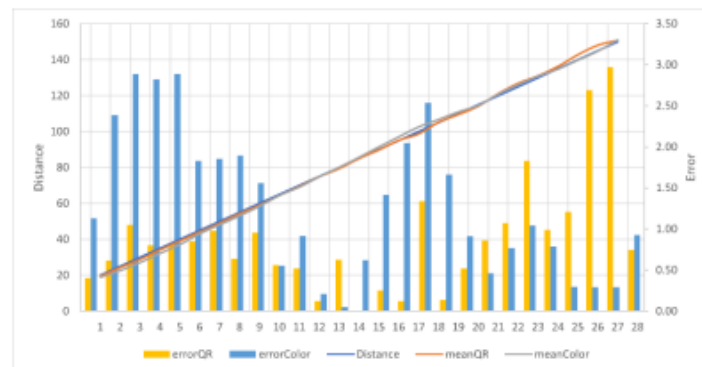
### 4.1. Comparison distance measurement from camera to object by QR code and color scenario natural light

Conduct experiments to detect objects and measure the distance from the camera to the objects at a distance not exceeding 150 cm. The objects include QR codes and color images, each tested three times to calculate the average. The experiments will be conducted under lighting conditions ranging from 200 lux to 650 lux or in illuminated environments where both QR codes and color images can be detected and measured accurately, with a margin of error not exceeding 3 cm as a Figure 2.

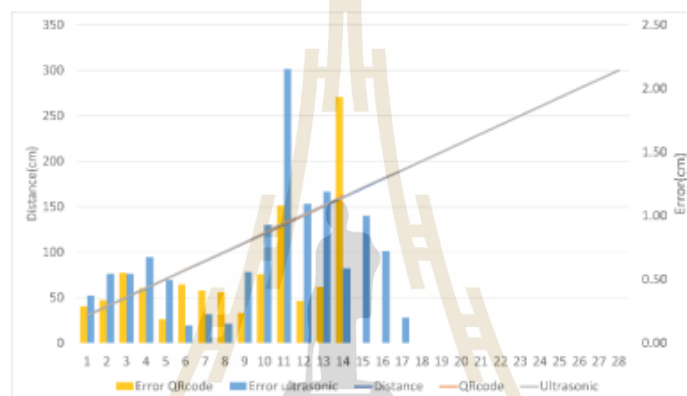
### 4.2. Comparison distance measurement from camera to object by QR code and Color scenario turn off light

This experiment is similar to Experiment 4.1 but differs in light intensity. In this experiment, the light intensity ranges from 100 lux to 250 lux. It was found that at lower light intensities, object detection for distance measurement was not possible because the low light intensity caused changes in the RGB values of the color images, leading to undetectable objects as a Figure 3.

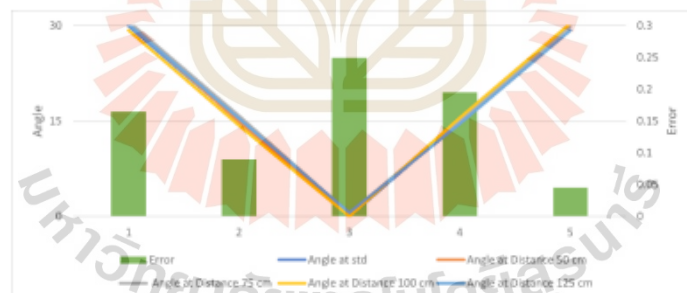




**Figure 2.** Graph comparison distance measurement from camera to object by QR code and Color natural light.



**Figure 3.** Graph comparison distance measurement from camera to object by QR code and Color turn off light.



**Figure 4.** Angle measurement from camera to object by QR code.

#### 4.3. Angle measurement from camera to object by QR code

This experiment involves measuring the angle between the camera and the QR code. It was found that the maximum angle measurable from the camera's centerline does not exceed 30 degrees. The experiment was conducted at four distances: 50, 75, 100, and 125 cm. The results show that the margin of error in angle measurement using the camera does not exceed 0.5 degrees as a Figure 4.

From experiments 4.1, 4.2, and 4.3, it can be concluded that objects detected for distance measurement should ideally be QR codes because they are less affected by low

light conditions (below 250 lux) compared to color images. The maximum measurable distance should not exceed 150 cm, and the angle of detection should not exceed 30 degrees. Therefore, this information will be utilized to control the relay and motors for the movement of the AGV (Automated Guided Vehicle).

#### 4.4. The results of controlling the relay for the motors driving the AGV to move forward and stop

The results of the experiment controlling the relay to operate the motors based on distance measurements are as follows: If the distance measured from the camera or ultrasonic sensor is less than 80 cm, indicating that the distance is too short, the motors will not function. However, if the distance measured is greater than 80 cm, the relay will be activated, causing the AGV to move forward. When the AGV moves closer and the distance becomes less than 80 cm. again, the vehicle will be immediately commanded to stop as a Figure 5.

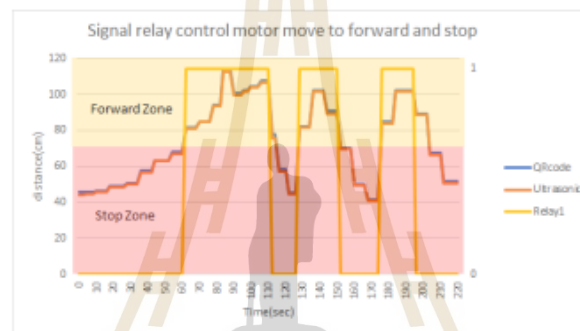


Figure 5. Results of controlling the relay for the motors driving the AGV to move forward and stop.

## 5. Conclusions

In the industrial transportation sector, there are two main types of automated vehicles: Automated Guided Vehicles (AGVs) and Autonomous Mobile Robots (AMRs). These systems come with high costs. Therefore, there is a concept to use cameras to track autonomous vehicles using symbols. In this research, two types of symbols were used: QR codes and color images. When tested under light conditions ranging from 80 lux to 650 lux, it was found that QR codes were better at detecting distances within the range of 30 to 150 cm compared to color images. Therefore, QR codes were chosen as the symbol for testing with AGV.

In testing the operation of the drive wheels, if the distance measured is greater than 80 cm, the vehicle moves forward; if it is between 60 and 80 cm, the vehicle stops; and if it is less than 60 cm, the vehicle reverses. For the front wheels, if the angle is greater than 15 degrees to the right, the vehicle turns right; if it is between 0 and 15 degrees, the wheels stay straight; and if the angle is greater than 15 degrees to the left, the vehicle turns left. Testing the actual movement of the vehicle in symbol tracking along a straight path showed that the vehicle's operations could be controlled according to the specified conditions.

Based on the testing, when the light is below 100 lux, the QR code cannot be detected. Additionally, when the camera reaches its maximum range, the vehicle stops

according to the conditions. Furthermore, if the speed of the leading vehicle is lower than the following vehicle, it causes the following vehicle to run and then stop repeatedly.

Recommendations:

1. Ensure that the speeds of the leading and following vehicles are similar to each other. This will prevent the following vehicle from having to repeatedly accelerate and decelerate.

2. Increase the detection range of the camera or consider implementing alternative methods to enhance detection capabilities. This will make the vehicle's movement smoother and more consistent.

## References

- [1] Sierra-Garcia, J.Enrique. Combining reinforcement learning and conventional control to improve automatic guided vehicles tracking of complex trajectories. *Expert Systems*. Volume 41, Issue 2, Article number e13076. Accepted: 31 May 2022, doi: 10.1111/exsy.13076.
- [2] Wikipedia contributors. Automated guided vehicle. Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 1 Mar. 2024, [online], <https://en.wikipedia.org/>.
- [3] R. P. Saputra, E. Rijanto. Automatic Guided Vehicles System and Its Coordination Control for Containers Terminal Logistics Application. *ArXiv*. March 7, 2021, doi: 10.48550/arXiv.2104.08331.
- [4] Munadi M, Radityo B, Ariyanto M, Taniai Y. Automated guided vehicle (AGV) lane-keeping assist based on computer vision and fuzzy logic control under varying light intensity. *Journal of Robotics and Intelligent Systems Engineering*, doi: 10.1016/j.rineng.2023.101678.
- [5] Park M. Development of Steering Control System for Autonomous Vehicle Using Geometry-Based Path Tracking Algorithm. *ETRI Journal*. 2015 May; 37(3), doi: 10.4218/etrij.15.2314.0123.
- [6] Diab M. Self-Driving Car Lane-keeping Assist using PID and Pure Pursuit Control. 2020 International Conference on Innovation and Intelligence for Informatics. *Computing and Technologies (3ICT)*. 2020 December; doi: 10.1109/3ICT51146.2020.9311987.
- [7] Toropov E, Tumasov A, Vashurin A, Butin D, Stepanov E. Hardware-in-the-Loop Testing of Vehicle's Electronic Stability Control System. *Applied Engineering Letters*, 8(2), 2023:70–79, doi: 10.18485/aeletters.2023.8.2.4.
- [8] Park M. Development of lateral control module for zone (u-turn) maneuver of vehicle/driver cooperative autonomous driving system. In 2016 IEEE Transportation Electrification Conference and Expo. Asia-Pacific (ITEC Asia-Pacific). 2016 June 4; Busan: p.1-5, doi: 10.1109/ITEC-AP.2016.7513080.
- [9] Wei W, Ke W, Yuanyuan C. Application of exception recognition and fuzzy decision in surveillance. In 2010 International Conference on Computer, Mechatronics, p.14, doi:10.1109/CMCE.2010.5610198.
- [10] Hung S. Micrography QR Codes. *IEEE Transactions on Visualization and Computer Graphics*; 2020 September 1; 26(9):2834–2847, doi: 10.1109/TVCG.2019.2896895.
- [11] Chen R, Yu Y, Xu X, Wang L, Zhao H, Tan H-Z. Adaptive Binarization of QR Code Images for Fast Automatic Sorting in Warehouse Systems. *Sensors*. 2019; 19(24):5466, doi: 10.3390/s19245466.
- [12] Karrach L, Pivarčiová E, Bozek P. Recognition of Perspective Distorted QR Codes with a Partially Damaged Finder Pattern in Real Scene Images. *Applied Sciences*. 2020; 10(21):7814, doi: 10.3390/app10217814.
- [13] Magnier B, Hayat K. Revisiting Mehrotra and Nichani's Corner Detection Method for Improvement with Truncated Anisotropic Gaussian Filtering. *Sensors (Basel)*. 2023 Oct 23; 23(20):8653, doi: 10.3390/s23208653.

## ประวัติผู้เขียน

นาย เจษฎากร เหมือนงาม เกิดเมื่อวันที่ 19 กันยายน 2539 ที่โป่งน้ำร้อน จังหวัดจันทบุรี เริ่มการศึกษาในระดับประถมศึกษาที่โรงเรียนอรรณาลัย ระดับมัธยมศึกษาปีที่ 1-6 ที่โรงเรียน มวกเหล็กวิทยา ต่อมาเข้าศึกษาต่อระดับอุดมศึกษาที่มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัด นครราชสีมา สำเร็จการศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมเมคคาทรอนิกส์ เมื่อ พ.ศ. 2561 ปฏิบัติงานในบริษัท เวสเทิร์น ดิจิตอล (ประเทศไทย) จำกัด เป็นระยะเวลา 1 ปี และบริษัท ซีเกท เทคโนโลยี (ประเทศไทย) จำกัด เป็นระยะเวลา 1 ปี และได้รับทุน OROG จากมหาวิทยาลัย เทคโนโลยีสุรนารี เข้าศึกษาต่อในระดับมหาบัณฑิตศึกษา สาขาวิชาวิศวกรรมเมคคาทรอนิกส์ ณ มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา

ในขณะที่ศึกษาต่อระดับมหาบัณฑิตได้มีประสบการณ์เป็นผู้สอนวิชาปฏิบัติการและเป็นผู้ช่วย สอนของสาขาวิชาวิศวกรรมเครื่องกลและวิศวกรรมเมคคาทรอนิกส์ จำนวน 3 รายวิชา ดังนี้ (1) กรรมวิธีการผลิตเบื้องต้น (2) เขียนแบบวิศวกรรม 1 และ (3) วิชาการจำลองและระบบควบคุม

นำเสนอผลงานทางวิชาการจำนวน 1 บทความ ภายในงานประชุมวิชาการระดับชาติ The 2nd International Conference on Mechanics, Electronics, Automation and Automatic Control ณ ประเทศจีน รายละเอียดปรากฏในภาคผนวก ง.

มหาวิทยาลัยเทคโนโลยีสุรนารี