แขนกลเคลื่อนย้ายวัตถุโดยใช้การประมวลผลภาพ

PICK AND PLACE ROBOTIC ARM VIA IMAGE PROOCESSING

สุพศิน โพธิ์โพน

สุรเกียรติ์ ตระการธนสกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2561

แขนกลเคลื่อนย้ายวัตถุโดยใช้การประมวลผลภาพ
PICK AND PLACE ROBOTIC ARM VIA IMAGE PROOCESSING

สุพศิน โพธิ์โพน
สุรเกียรติ์ ตระการธนสกุล

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2561

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PICK AND PLACE ROBOTIC ARM VIA IMAGE PROOCESSING

SUPHASIN POPONE
SURAKIAT TRAKARNTANASAKUL

A THESIS SUBMITTED IN PATIAL FULFILLMENT
OF THE REQUIRMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEER
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2018

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2561
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์ แขนกลเคลื่อนย้ายวัตถุโดยใช้การประมวลผลภาพ
PICK AND PLACE ROBOTIC ARM VIA IMAGE PROOCESSING

นักศึกษาผู้จัดทำ นายสุพศิน โพธิ์โพน **รหัสนักศึกษา** 58011357
นายสุรเกียรติ์ ตระการธนสกุล **รหัสนักศึกษา** 58011363

ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมการวัดคุม
ปีการศึกษา 2561

อาจารย์ผู้ควบคุมปริญญานิพนธ์	ลายมือชื่อ
ผศ. ดร.นรินทร์ ธรรมารักษ์วัฒนะ	
รศ. ทรงชัย วีระทวีมาศ	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	แขนกลเคลื่อนย้ายวัตถุโดยใช้การประมวลผลภาพ PICK AND PLACE ROBOTIC ARM VIA IMAGE PROOCESSING
นักศึกษาผู้จัดทำ	นายสุพศิน โพธิ์โพน **รหัสนักศึกษา** 58011357 นายสุรเกียรติ์ ตระการธนสกุล **รหัสนักศึกษา** 58011363
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.นรินทร์ ธรรมารักษ์วัฒนะ รองศาสตราจารย์ทรงชัย วีระทวีมาศ
ปีการศึกษา	2561

บทคัดย่อ

โครงการวิจัยชิ้นมีวัตถุประสงค์เพื่อศึกษาการทำงานของแขนกล และศึกษาทางด้านการประมวลผลภาพ เพื่อออกแบบระบบควบคุมแขนกลในการย้ายวัตถุด้วยการประมวลผลภาพ โดยจะประกอบด้วยส่วนหลักๆ 2 ส่วนคือ ส่วนฮาร์ดแวร์และส่วนซอฟต์แวร์ โดยฮาร์ดแวร์จะประกอบด้วยแขนกลโดยใช้เซอร์โวมอเตอร์ในการขับเคลื่อน และใช้ไมโครคอนโทรลเลอร์ในการควบคุม การทำงานเบื้องต้นจะเริ่มจากการจับภาพ และใช้การประมวลผลภาพในการจับจุดเพื่อหาตำแหน่งที่ต้องการแล้วนำตำแหน่งที่ได้มาประมวลผล เพื่อส่งคำสั่งไปควบคุมเซอร์โวมอเตอร์ของแขนกลในการเคลื่อนย้ายวัตถุตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis title	PICK AND PLACE ROBOTIC ARM VIA IMAGE PROOCESSING
Authors	Mr. Suphasin Popone
	Mr. Surakiat Trakarntanasakul
Thesis Advisor	Asst. Prof. Dr. Narin Tammarakvattana
	Assoc. Prof. Songchai Weerathaweemas
Year	2018

ABSTRACT

This project aims to study the operation of the robot arm, education of image processing and design robot arm control circuits using image processing. It consists of two main parts: hardware and software. The hardware consists of a robotic arm using a servo motor to drive. And a microcontroller to control. Preliminary work will start by capturing and using Image Processing in the capture point to the desired location. The position of the degrees were calculated for each section to send out each servo motor control to achieve the desired motion pattern.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี จึงขอขอบพระคุณเป็นอย่างสูงสำหรับอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.นรินทร์ ธรรมารักษ์วัฒนะ และ รองศาสตราจารย์ทรงชัย วีระทวีมาศ และอาจารย์ท่านอื่น ๆ ที่ให้คำแนะนำและคำปรึกษา ตลอดจนแนวคิดต่าง ๆ ในการพัฒนาโครงงาน ขอขอบคุณ บิดา มารดา ที่เป็นกำลังใจอย่างดีในการต่อสู้และฟันฝ่าอุปสรรคต่าง ๆ ในการทำงาน ขอขอบคุณพี่ๆ น้อง ๆ และเพื่อน ๆ ที่คอยช่วยเหลือในด้านต่าง ๆ และให้กำลังใจกันเสมอมา ขอขอบคุณภาควิชาวิศวกรรมการวัดคุมและควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่ให้ทั้งความรู้ และประสบการณ์ต่าง ๆ ที่ดีในการใช้ชีวิตในมหาวิทยาลัย

หวังเป็นอย่างยิ่งว่าปริญญานิพนธ์ฉบับนี้ จะก่อให้เกิดประโยชน์แก่ผู้ที่สนใจศึกษา และผู้ที่มีความสนใจและสามารถนำไปใช้ศึกษาและพัฒนาให้เกิดประโยชน์ต่อไป หากมีสิ่งใดผิดพลาด หรือขาดตกบกพร่อง ทางผู้จัดทำต้องขออภัยไว้ ณ ที่นี้ด้วย คุณค่าและประโยชน์อันพึงมีจากปริญญานิพนธ์ฉบับนี้ ขอมอบแด่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I
บทคัดย่อภาษาอังกฤษ.....II
กิตติกรรมประกาศ.....III
สารบัญ.....IV
สารบัญตาราง.....VI
สารบัญรูป.....VII

บทที่ 1 บทนำ.....1
1.1 ที่มาและความสำคัญ.....1
1.2 วัตถุประสงค์ของโครงการวิจัย.....1
1.3 หลักการและเหตุผลของโครงการ.....1
1.4 ขอบเขตของโครงการ.....2
1.5 ขั้นตอนการศึกษา.....2

บทที่ 2 แนวคิดและทฤษฎีที่เกี่ยวของ.....3
2.1 หลักการพื้นฐานของหุ่นยนต์.....3
2.1.1 ชนิดของหุ่นยนต์.....3
2.1.2 ลำดับขั้นความเป็นอิสระและการบังคับ.....4
2.2 การประมวลผลภาพ.....5
2.2.1 การคัดแยกสีและระบบสี.....6
2.2.2 การประมวลผลภาพสี.....8
2.3 เซอร์โวมอเตอร์.....9
2.4 บอร์ดอาดุยโน่คอนโทรลเลอร์(Arduino board).....11
2.5 กล้องเว็บแคม.....12

บทที่ 3 วิธีการดำเนินวิจัย.....13
3.1 หลักการทำงาน.....13
3.2 การประกอบตัวชิ้นงาน.....13
3.3 การพัฒนาทางด้านซอฟต์แวร์ในการควบคุมเซอร์โวมอเตอร์.....15
3.4 การคัดแยกสี.....16
3.5 การตรวจจับวัตถุ.....17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

3.5.1 ทำ Haar cascade....17
3.5.2 โหลดภาพเข้าตัวแปรและแปลงเป็นโหมดสีเทา....18
3.5.3 การตรวจจับ....18
3.6 การควบคุมแขนกล....19
3.6.1 การรับค่าพิกัดผ่านทาง usb port serial แบบ realtime....20
3.6.2 สั่งให้เซอร์โวมอเตอร์หมุนตามพิกัดที่ได้รับ....20
3.6.3 การเคลื่อนที่ของแขนกล....20
3.7 แผนผังการทำงานของแขนกล....21

บทที่ 4 การทดลองและผลการทดลอง....22
4.1 การจำแนกวัตถุ และจำแนกสี....22
4.1.1 การทดลองจำแนกวัตถุ....22
4.1.2 การทดลองจำแนกสี....25
4.2 การทดลองหยิบวัตถุที่ตำแหน่งใดๆ....30

บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ....34
5.1 สรุปผลการทดลอง....34
5.2 ข้อเสนอแนะ....34
บรรณานุกรม....35
ภาคผนวก....36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

2.1 แสดงรายชื่อสีและค่าที่แทนชื่อสีต่างๆ.....6
4.1 ตารางแสดงผลการจำแนกวัตถุจำนวน 50 ครั้ง แบ่งเป็น 2 รอบ.....23
4.1 ตารางแสดงผลการจำแนกวัตถุจำนวน 50 ครั้ง แบ่งเป็น 2 รอบ(ต่อ) 24
4.1 ตารางแสดงผลการจำแนกวัตถุจำนวน 50 ครั้ง แบ่งเป็น 2 รอบ(ต่อ) 25
4.2 แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 1.....26
4.2 แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 1 (ต่อ).....27
4.2 แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 1 (ต่อ).....28
4.3 แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 2.....28
4.3 แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 2 (ต่อ).....29
4.4 ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 1.....30
4.4 ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 1 (ต่อ).....31
4.4 ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 1 (ต่อ).....32
4.5 ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 2.....32
4.5 ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 2 (ต่อ).....33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 ภาพเปรียบเทียบร่างกายของมนุษย์กับแขนกลของหุ่นยนต์อุตสาหกรรม....3
2.2 ชนิดของการเคลื่อนที่....4
2.3 การเคลื่อนที่ที่มี 3 ลำดับขั้นความเป็นอิสระแบบเลื่อน....5
2.4 การเคลื่อนที่ที่มี 3 ลำดับขั้นความเป็นอิสระแบบหมุน....5
2.5 อุปกรณ์พื้นฐานสำหรับการประมวลผลภาพ....6
2.6 HSV Color Model....8
2.7 ลักษณะภายในของเซอร์โวมอเตอร์....9
2.8 เซอร์โวมอเตอร์ชนิด LDX-218....9
2.9 เซอร์โวมอเตอร์ชนิด QDS-15RO....10
2.10 ภาพรวมบอร์ด Arduino R3 และ พอร์ตเชื่อมต่อ....11
2.11 กล้องเว็บแคม OKER OP-002....12
3.1 ฐานของแขนกล....14
3.2 แขนกลส่วนล่าง....14
3.3 แขนกลส่วนบน....15
3.4 การเคลื่อนที่ของแขนกลจากตำแหน่ง 0 องศา ไปจนถึง 180 องศา....16
3.5 แสดงคำสั่งดึงไฟล์ที่สร้างไว้ และแปลงเป็นโหมดสีเทา....18
3.6 แสดงคำสั่งการตีกรอบล้อมรอบวัตถุ....19
3.7 แสดงคำสั่งให้จุดศูนย์กลางของกรอบเป็นตำแหน่ง....19
3.8 แสดงคำสั่งรับค่าพิกัดในแกน x และ y....20
3.9 แสดงคำสั่งให้เซอร์โวมอเตอร์หมุน....20
3.10 แสดง Flow chart การทำงานของแขนกล....21
4.1 วัตถุรูปกล่องสี่เหลี่ยม....22
4.2 แสดงการตรวจจับวัตถุ....23
4.3 แสดงการทดลองหยิบลูกบอลสีน้ำเงิน....25
4.4 แสดงการวางลูกบอลสีน้ำเงินในถาดสีน้ำเงิน....26
4.5 แสดงการหยิบวัตถุในตำแหน่งใดๆ....30
ผ.1 หน้าต่างโปรแกรมที่ใช้ภาษาไพธอนในการเขียน....36
ผ.2 หน้าต่างโปรแกรม Arduino IDE....45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญ

ในกระบวนการผลิตในทางอุตสาหกรรม หุ่นยนต์แขนกลถือว่าเป็นส่วนสำคัญและได้เข้ามามีบทบาทเป็นอย่างมากในการผลิตหลายๆ ขั้นตอน โดยเฉพาะอย่างยิ่งในส่วนของการหยิบจับ คัดแยก หรือเคลื่อนย้ายวัตถุจากตำแหน่งหนึ่งไปยังอีกตำแหน่งหนึ่งซึ่งพบมากในกระบวนการผลิตแทบทุกประเภท ดังนั้นการควบคุมให้หุ่นยนต์แขนกลเคลื่อนที่และหยิบจับวัตถุได้อย่างถูกต้องและแม่นยำหรืออย่างน้อยที่สุดคือให้เกิดความผิดพลาดน้อยที่สุดเท่าที่จะเป็นไปได้ จึงเป็นสิ่งสำคัญที่ควรจะถูกพิจารณาเป็นลำดับต้นๆ ยกตัวอย่างเช่น การควบคุมแขนหุ่นยนต์ผ่าตัดระยะไกล จำเป็นจะต้องควบคุมแขนของหุ่นยนต์ให้มีความถูกต้องและแม่นยำมากที่สุด ซึ่งถ้าการควบคุมแขนของหุ่นยนต์ไม่มีความถูกต้องและแม่นยำที่เพียงพอ ก็อาจเกิดความเสียหายหรือเกิดความผิดพลาดก็เป็นได้ ดังนั้นเพื่อที่จะออกแบบตัวควบคุมให้หุ่นยนต์แขนกลทำงานได้อย่างมีประสิทธิภาพสูงสุด จำเป็นต้องอาศัยหลักการออกแบบตัวควบคุมให้มีความเหมาะสมกับการใช้งานของแขนกลนั้นด้วย รวมไปถึงการออกแบบตัวควบคุมเพื่อลดความผิดพลาดในการเคลื่อนที่ให้น้อยที่สุด

ทางผู้จัดทำจึงต้องการพัฒนาแขนกลให้สามารถเคลื่อนที่ย้ายวัตถุได้ตามต้องการได้ โดยอาศัยหลักการประมวลผลภาพในการหาตำแหน่งของวัตถุหรือตำแหน่งของเป้าหมายซึ่งสามารถหาได้จากการนำสัญญาณภาพของวัตถุ หรือเป้าหมายมาผ่านกระบวนการประมวลผลภาพ (Image Processing) โดยหลักการของการประมวลผลภาพนี้จะนำ Image acquisition มาช่วยแยกภาพที่ต้องการออกมาจากสิ่งแวดล้อมภายในภาพ โดยพิจารณาจากสีของวัตถุที่ต้องการ และเมื่อได้ข้อมูลภาพทั้งหมดก็ใช้วิธีการประมวลผลภาพสีและวิธีการประมวลผลภาพขาวดำ เพื่อช่วยในการหาตำแหน่งของวัตถุในภาพและสามารถให้ระบบส่งคำสั่งให้แขนกลหยิบจับหรือวางวัตถุได้ตามต้องการ

1.2 วัตถุประสงค์ของโครงการวิจัย

เพื่อพัฒนาหุ่นยนต์แขนกลให้สามารถเคลื่อนย้ายวัตถุหรือหยิบสิ่งของที่ต้องการได้อย่างถูกต้องและแม่นยำ

1.3 หลักการและเหตุผลของโครงการ

ในกระบวนการอุตสาหกรรมส่วนใหญ่ หุ่นยนต์แขนกลเริ่มเข้ามีบทบาทในส่วนของหลายๆ ขั้นตอนในกระบวนการต่างๆ โดยเฉพาะอย่างยิ่งในส่วนของการหยิบจับ คัดแยก หรือ เคลื่อนย้ายวัตถุจากอีกตำแหน่งหนึ่งไปยังอีกตำแหน่งหนึ่ง ทางผู้จัดทำจึงได้ทำการสร้างแขนกลเพื่อเคลื่อนย้ายหรือหยิบจับวัตถุโดยอาศัยระบบการประมวลผลภาพ (Image processing) การหาตำแหน่งของวัตถุหรือตำแหน่งของเป้าหมายสามารถหาได้จากการนำสัญญาณภาพของวัตถุ หรือเป้าหมายมาผ่านกระบวนการประมวลผลภาพ (Image processing) หลักการของการประมวลผลภาพนี้จะนำ Image acquisition มาช่วยแยก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ต้องการออกมาจากสิ่งแวดล้อมภายในภาพ โดยพิจารณาจากสีของวัตถุที่ต้องการ และเมื่อได้ข้อมูลภาพทั้งหมดก็ใช้วิธีการประมวลผลภาพสี และวิธีการประมวลผลภาพขาวดำเพื่อช่วยในการหาตำแหน่งของวัตถุในภาพ และสามารถหยิบจับหรือเคลื่อนย้ายวัตถุได้ตามต้องการ

1.4 ขอบเขตของโครงการ

1. สร้างหุ่นยนต์แขนกลต้นแบบที่มีข้อต่อ 6 แกนแบบอิสระที่สามารถหยิบจับวัตถุและเคลื่อนย้ายวัตถุได้
2. ใชระบบการประมวลผลภาพดิจิตอลผานทางโปรแกรมแมทแลปในการจำแนกสีและรูปทรงของวัตถุ และระบุตำแหนงของวัตถุ
3. พัฒนาโปรแกรมและทดสอบการเคลื่อนที่ของหุนยนต์แขนกลเพื่อใหเข้าถึงเป้าหมายได้อยางแมนยำที่สุด
4. วิเคราะหผลลัพธที่ได้ สรุปและจัดทำรายงานเผยแพรผลงานสูสาธารณะชน

1.5 ขั้นตอนการศึกษา

1. ศึกษาการเขียนโปรแกรมเพื่อควบคุมหุ่นยนต์แขนกลให้สามารถหยิบจับวัตถุและเคลื่อนย้ายวัตถุได้
2. ศึกษาวิธีการประมวลผลภาพเพื่อใชในการหาตำแหนงวัตถุหรือตำแหนงเปาหมายในการหยิบจับและเคลื่อนยายวัตถุของหุนยนต์แขนกล
3. ศึกษาการทำให้แขนกลสามารถหยิบจับวัตถุและเคลื่อนย้ายวัตถุโดยใช้การประมวลผลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

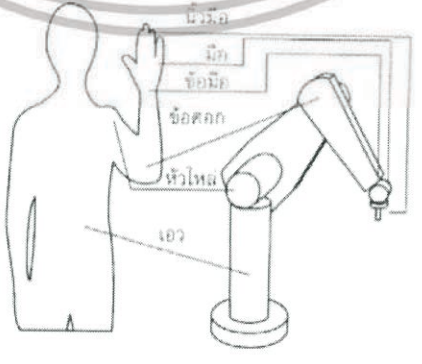
บทที่ 2
แนวคิดและทฤษฎีที่เกี่ยวข้อง

สำหรับแนวคิดและทฤษฎีที่เกี่ยวของกับงานวิจัยสามารถแบ่งออกเป็นสองส่วนที่สำคัญ คือ ส่วนแรกจะเป็นส่วนของหลักการพื้นฐานของหุ่นยนต์ ลำดับขั้นความเป็นอิสระและการบังคับของวัตถุเกร็ง ตลอดจนถึงการคำนวณหาแบบจำลองทางคณิตศาสตรของหุ่นยนต์แขนกล การหาตำแหนงของวัตถุและตำแหนงเป้าหมายในแกนสามมิติ ส่วนที่สองจะเป็นส่วนของทฤษฎีการประมวลผลภาพเพื่อนำไปใช้ในการหาตำแหน่งของวัตถุและตำแหน่งของเป้าหมายที่เหมาะสมที่สุด

2.1 หลักการพื้นฐานของหุ่นยนต์

หุ่นยนต์ (Robot) คือเครื่องจักรกลชนิดหนึ่งที่ทำงานด้วยการควบคุมแบบอัตโนมัติที่มีลักษณะโครงสร้างและการทำงานคล้ายหรือเสมือนกับมนุษย์ อีกทั้งสามารถทำงานที่ซ้ำและซับซ้อนดี รวมทั้ง งานที่มีความยากลำบากที่มนุษย์ไม่สามารถทำได้ เช่น งานสำรวจในพื้นที่บริเวณคับแคบ งานสำรวจใต้ท้องทะเลลึกหรืองานสำรวจดาวเคราะห์ที่ไม่มีสิ่งมีชีวิต หุ่นยนต์เป็นศาสตร์ทางวิศวกรรมที่รวมเอาวิศวกรรมเครื่องกล วิศวกรรมไฟฟ้า วิศวกรรมอิเล็คทรอนิกส์ วิศวกรรมคอมพิวเตอร์ และวิศวกรรมซอฟต์แวร์เข้าด้วยกันเพื่อสร้างหุ่นยนต์ขึ้น

2.1.1 ชนิดของหุ่นยนต์ หุ่นยนต์สามารถแยกได้หลากหลายรูปแบบตามลักษณะเฉพาะของการใช้งาน ได้แก่ การแบ่งแยกตามการใช้งาน การแบ่งแยกตามการเคลื่อนที่ การแบ่งแยกตามการควบคุมของการเคลื่อนที่ และการแบ่งแยกตามลักษณะภายนอกของแขนกล การแบ่งแยกตามการใช้งาน ได้แก่ หุ่นยนต์อุตสาหกรรม เป็นหุ่นยนต์ที่มีโครงสร้างคลายกับร่างกายของมนุษย์ คือ มีเอว ข้อศอก แขน และข้อมือ โดยคำว่า แขนกล หมายถึงแขนของหุ่นยนต์อุตสาหกรรม การออกแบบหุ่นยนต์อุตสาหกรรมเป็นการประยุกต์รวมเอาวิศวกรรมในหลายสาขาที่แตกต่างกัน ได้แก่ วิศวกรรมเครื่องกลและวิศวกรรมอุตสาหการจะทำหน้าที่ในการออกแบบสร้างหุ่นยนต์ใหม่โครงสร้างกลไกเชื่อมโยงกัน เลือกใชวัสดุที่มีความแข็งแรงและทนทาน ส่วนวิศวกรรมไฟฟาจะทำหน้าที่ในการเลือกใช้ชนิดของมอเตอร์และการจายพลังงานไฟฟ้าให้กับมอเตอร์ ดังแสดงในรูปที่ 2.1



รูปที่ 2.1 ภาพเปรียบเทียบร่างกายของมนุษย์กับแขนกลของหุ่นยนต์อุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์ส่วนใหญ่ที่ใช้ในงานอุตสาหกรรมจะเป็นหุ่นยนต์ที่ติดตั้งอยู่กับที่ ไม่สามารถเคลื่อนที่ได้ (Fixed Robot) เคลื่อนที่ไปมาได้เฉพาะแขนกล เช่น หุ่นยนต์ที่ใชในการหยิบจับและวางชิ้นงาน การเชื่อม และการพ่นสี หุ่นยนต์ประเภทนี้จะมีลักษณะโครงสรางที่ใหญ่และมีน้ำหนักมาก ใช้พลังงานจากแหล่งจ่ายภายนอก และจะมีการเขียนโปรแกรมกำหนดขอบเขตการเคลื่อนที่ของแขนกล ให้แขนกลของหุ่นยนต์สามารถเคลื่อนที่ไปมาได้ตามโปรแกรมที่กำหนดเอาไว้

2.1.2 ลำดับขั้นความเป็นอิสระและการบังคับ (Degrees of freedom (DOF) and constraints)

การเคลื่อนที่ของวัตถุเกร็ง ส่วนมากจะเป็นการเคลื่อนที่แบบเลื่อนและแบบหมุนรวมกัน การเคลื่อนที่แบบเลื่อนสามารถเคลื่อนที่เลื่อนไปได้ใน 1 แกน 2 แกน หรือ 3 แกน และการเคลื่อนที่แบบหมุนก็สามารถเคลื่อนที่หมุนได้ใน 1 แกน 2 แกน หรือ 3 แกน เช่นเดียวกัน ดังแสดงในรูปที่ 2.2

รูปที่ 2.2 ชนิดของการเคลื่อนที่

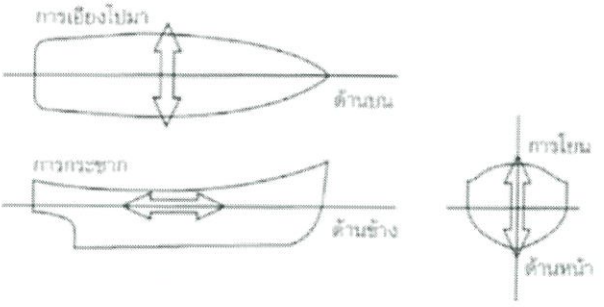
ลำดับขั้นความเป็นอิสระ คือจำนวนของส่วนประกอบการเคลื่อนที่ที่ต้องการควบคุมการเคลื่อนที่หรืออีกนิยามหนึ่งของลำดับขั้นความเป็นอิสระ คือจำนวนพิกัดทั่วไปที่เป็นอิสระต่อกัน ที่ใช้ในการอธิบายการเคลื่อนที่ของระบบ จากรูปที่ 2.2 วัตถุมีการเคลื่อนที่ทั้งแบบเลื่อน (ตามแกน X, Y และ Z) จึงมี 3 ลำดับขั้นความเปนอิสระ และแบบหมุน (รอบแกน X, Y, และ Z) จึงมี 3 ลำดับขั้นความเป็นอิสระรวมกันจะมี 6 ลำดับขั้นความเป็นอิสระ (6 DOF) จำนวนของลำดับขั้นความเป็นอิสระ คือจำนวนส่วนประกอบของการเคลื่อนที่ เพื่อต้องการควบคุมการเคลื่อนที่รอยต่อของชิ้นส่วนกลไกถูกบังคับให้เคลื่อนที่แบบเลื่อนเพียงอย่างเดียว จึงมี 1 ลำดับขั้นความเป็นอิสระและรอยต่อของชิ้นส่วนกลไกถูกบังคับให้มีการเคลื่อนที่แบบหมุนด้วย จึงมี 2 ลำดับขั้นความเป็นอิสระ

ในระนาบสามมิติและลำดับขั้นความเป็นอิสระ 6 ลำดับขั้นความเป็นอิสระของวัตถุแข็งเกร็ง (Rigid body) สามารถอธิบายโดยใช้ชื่อของการเดินเรือ ดังต่อไปนี้

1) การเคลื่อนที่แบบเลื่อน จะมี 3 ลำดับขั้นความเป็นอิสระ ได้แก่
   - การโยน (Heave) คือการเคลื่อนที่เชิงเส้นในแนวดิ่ง (ขึ้น-ลง)
   - การโอนเอียงไปมา (Sway) คือการเคลื่อนที่เชิงเส้นทางด้านข้าง
   - การกระชาก (Surge) คือการเคลื่อนที่เชิงเส้นในแนวนอน (ไปหน้า-ถอยหลัง)

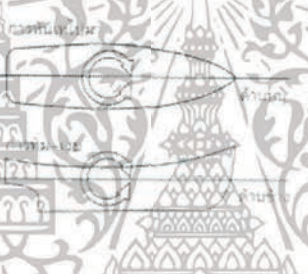
   ดังแสดงในรูปที่ 2.1.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 การเคลื่อนที่ที่มี 3 ลำดับขั้นความเปนอิสระแบบเลื่อน

2) การเคลื่อนที่แบบหมุน จะมี 3 ลำดับขั้นความเป็นอิสระ ได้แก่
- การโคลง (Roll) คือการเคลื่อนที่แบบหมนุรอบแกนนอน
- การทิ่ม-เงย (Pitch) คือการเคลื่อนที่แบบหมุนรอบแกนทแยง
- การหันเหไปมา (Yaw) คือการเคลื่อนที่แบบหมุนรอบแกนดิ่ง



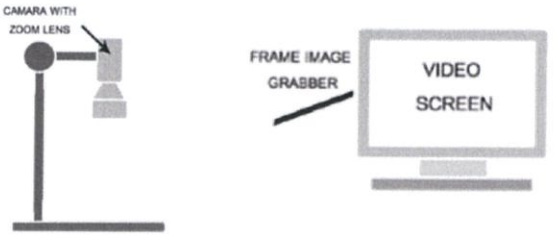
รูปที่ 2.4 การเคลื่อนที่ที่มี 3 ลำดับขั้นความเปนอิสระแบบหมุน

2.2 การประมวลผลภาพ (Image processing)

การประมวลผลภาพ คือการกระทำการอย่างใดอย่างหนึ่งกับภาพต้นฉบับ (Input image) เพื่อให้ได้ภาพผลลัพธ์ (Output image) ที่มีลักษณะของภาพเป็นไปตามที่ต้องการ ซึ่งการกระทำการกับภาพที่ใช้ในการประมวลผลภาพดิจิตอลมีอยู่มากมายหลายแบบ ซึ่งความเข้าใจเกี่ยวกับคุณลักษณะและการแยกแยะประเภทของการกระทำการกับภาพจะช่วยให้สามารถคาดคะเนภาพผลลัพธที่จะได้จากการกระทำการแตละแบบหรือประมาณความซับซ้อนของการกระทำการกับภาพที่จะนำไปใช้ได้ โดยขั้นตอนการประมวลผลภาพดิจิตอลประกอบไปด้วย 3 ส่วนคือ

1. ขั้นตอนการนำขอมูลภาพเขาสูเครื่องคอมพิวเตอร
2. ขั้นตอนการนำคอมพิวเตอรมาพัฒนาอัลกอริทึมเพื่อประมวลผลภาพ
3. ขั้นตอนการแสดงผลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 อุปกรณพื้นฐานสำหรับระบบประมวลผลภาพ

2.2.1 การคัดแยกสี

Image processing การประมวลผลภาพเพื่อที่จะได้ค่ามาวิเคราะห์นั้นประกอบด้วยหลายตัวแปรหรือหลายปัจจัย โดยส่วนที่มีความสำคัญมากที่สุดก็คงหนีไม่พ้นเรื่องของสี ซึ่งจะเป็นตัวบอกความแตกต่างระหว่างพื้นหลังและตัววัตถุหรือรูปทรงต่างๆ (Contour) และลำดับต่อมาคือหน่วยของภาพ หรือพิกเซลก็จะมีผลอีกเช่นกัน ส่วนในเรื่องของความละเอียด โดยความละเอียดยิ่งมากก็จะสามารถแยกส่วนของข้อมูลภาพได้ถูกต้องแม่นยำมากยิ่งขึ้น และแสงก็มีความสำคัญไม่แพ้กัน ต่อให้ภาพมีความละเอียดสูง แต่ถ้าแสงน้อยเกินไปอาจจะทำให้วิเคราะห์ไม่ตรงกับความจริง ดังนั้นปัจจัยพื้นฐานที่จะได้ข้อมูลจากภาพเพื่อนำไปวิเคราะห์ได้อย่างแม่นยำนั้นได้แก่ สี ปริมาณของพิกเซลและแสง ซึ่งมีรายละเอียด ดังนี้

ระบบสี (Color system) เป็นรูปแบบที่เราใช้ในการนิยามหรือแทนค่าพื้นที่บนภาพ ซึ่งปกติแล้วจะนิยามในเชิงคณิตสาตร์ จะประกอบไปด้วยค่าของสี 3 หรือ 4 ค่า โดยระบบสีแบ่งออกได้ดังนี้

1) ระบบสี RGB ประกอบไปด้วยแม่สีทั้งหมด 3 สี ได้แก่ สีแดง สีเขียว และสีน้ำเงิน โดยการผสมสีในระบบนี้เป็นลักษณะของแสง ซึ่งแสงมีลักษณะเป็นคลื่นเมื่อใดที่แสงมาซ้อนทับกันจะทำให้เกิดการรวมตัวของความยาวคลื่น ดังนั้นจะทำให้เกิดแสงสีต่างๆ ใช้ในการแสดงผล ซึ่งอุปกรณ์ที่ใช้ในการแสดงผล (Output) ที่ได้ทางจะแสดงบนอุปกรณ์ที่เป็น โทรทัศน์ คอมพิวเตอร์ หรือ Smart phone ส่วน Input ของการรับค่าสีในระบบ RGB เช่น กล้องวิดีโอ (Webcam) และสแกนเนอร์ เป็นต้น ซึ่งระบบสี RGB จะใช้ในการวิเคราะห์ในระบบจำแนกป้ายทะเบียนรถยนต์อัตโนมัติ โดยจะเห็นได้จากตาราง เปรียบเทียบค่าสีในมาตรฐานต่างๆ ได้จากตารางที่ 2.1

ตารางที่ 2.1 ตารางรายชื่อสีและค่าที่แทนค่าสีต่างๆ

Color	Name	Hex Code #RRGGBB	Decimal Code (R,G,B)
	Black	#000000	(0,0,0)
	White	#FFFFFF	(255,255,255)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 (ต่อ) ตารางรายชื่อสีและค่าที่แทนค่าสีต่างๆ

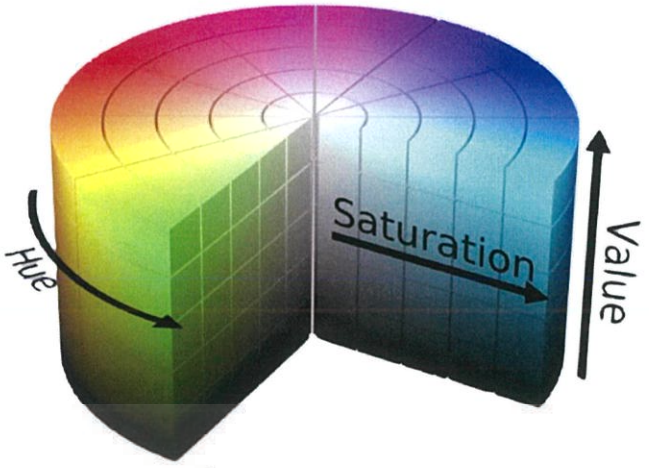
	Red	#FF0000	(255,0,0)
	Lime	#00FF00	(0,255,0)
	Blue	#0000FF	(0,0,255)
	Yellow	#FFFF00	(255,255,0)
	Cyan / Aqua	#00FFFF	(0,255,255)
	Magenta / Fuchsia	#FF00FF	(255,0,255)
	Silver	#C0C0C0	(192,192,192)
	Gray	#808080	(128,128,128)
	Maroon	#800000	(128,0,0)
	Olive	#808000	(128,128,0)

จะสังเกตุได้ว่าค่าสีขาวเป็นค่า 255 ในทุกแม่สี และสีดำจะมีค่า 0 ในทุกแม่สี ดังนั้นจะเป็นการง่ายในการวิเคราะห์องค์ประกอบที่เราสนใจในภาพได้ง่าย โดยองค์ประกอบที่เราสนใจนั้นก็คือ ป้ายทะเบียนและตัวอักษรภายในป้าย ซึ่งเราจะใช้คุณสมบัตินี้ในการแบ่งสีของภาพให้เป็นสีขาวและดำ

2) ระบบสี RYB ระบบสีนี้ ประกอบไปด้วยแม่สีทั้ง 3 ได้แก่ สีแดง สีเหลือง และสีน้ำเงิน เป็น มาตรฐานที่ใช้ในการผสมสีในงานศิลปะ เช่นการวาดรูป
3) ระบบสี CMYK ระบบสีนี้คือการผสมสี 4 สี ได้แก่ สีน้ำเงินอมเขียว สีแดงอมม่วง สีเหลือง และสีดำ ซึ่งใช้ในงานพิมพ์สี จะสังเกตุได้จากหมึกพิมพ์ของเครื่องปริ้น ink jet ทั่วไปจะมี 4 สี
4) ระบบสี HSV และ HSL เป็นระบบสีที่ปรับปรุงระบบสี RGB เพื่อให้ได้การแสดงผลสีของรูปทรงต่างๆได้อย่างมีคุณภาพ ใช้งานด้านคอมพิวเตอร์กราฟิกเพื่อความสมจริงของแสงและเงา

HSV color model เป็นการพิจารณาสีโดยใช้ Hue, Saturation, Value ซึ่งเป็นตัวแทนของระบบสี RGB เพื่อทำให้ง่ายขึ้นในการเราจะคัดแยกสี จริงแล้วทำแบบ RGB หรือ BGR ก็ได้แต่จะปรับแก้สีได้น้อยกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 HSV Color Model

Hue คือ ค่าสีของสีหลักหรือเรียกว่า เฉดสี
Saturation คือ ความบริสุทธิ์ของสี
Value คือ ความสว่างของสี ซึ่งสามารถวัดได้โดยค่าความเข้มของความสว่างของแต่ละสี

2.2.2 การประมวลผลภาพสี (Color image processing)

การประมวลผลภาพสีสามารถนำเทคนิคและวิธีการประมวลผลภาพระดับความเข้มเทาที่ได้กล่าวมาแล้วนำมาใช้ในการประมวลผลภาพสีได้ เช่น การประมวลผลภาพแบบจุด การปรับเพิ่มหรือลดความสว่าง การขยายฮีสโทแกรมภาพ เป็นต้น แต่เนื่องจากภาพสีในแบบจำลองสี RGB ประกอบไปด้วยข้อมูลภาพของแม่สีหลัก 3 สีที่ซ้อนกันอยู่ คือข้อมูลภาพของแม่สีแดง สีเขียว และสีน้ำเงิน ดังนั้นในการประมวลผลภาพสีนั้นจะทำการประมวลผลทีละแบนด์ของข้อมูลภาพของแต่ละแม่สีหลัก กล่าวคือ ขั้นตอนของการประมวลผลภาพในแต่ละแม่สีหลักนั้นจะมีขั้นตอนการประมวลผลภาพแยกออกจากกัน จากนั้นจึงนำผลลัพธ์ของภาพที่ประมวลผลได้ในแต่ละข้อมูลภาพของแม่สีหลักมารวมกันแสดงผลเป็นภาพ ผลลัพธ์ที่ได้จากการประมวลผลภาพสีทในบางกรณีภาพผลลัพธที่ได้จากการประมวลผลภาพสีในบางครั้งอาจมีผลทำให้ภาพผลลัพธ์มีความไม่สมบูรณหรือมีความผิดเพี้ยนของสีในภาพผลลัพธ์ได้ ดังนั้นเพื่อแก้ไขความผิดเพี้ยนของสีที่อาจจะเกิดขึ้นได้จากการทำการประมวลผลภาพทีละข้อมูลภาพหรือแบนด์ย่อยของแต่ละแม่สีหลักแยกจากกัน ก็สามารถที่จะทำการเปลี่ยนแปลงภาพสีในรูปแบบจำลอง RGB ให้แสดงผลอยู่ ในรูปของแบบจำลองสีแบบอื่นๆ จากนั้นจะทำการประมวลผลภาพเฉพาะส่วนของข้อมูลภาพที่แสดงถึงค่าความเข้มแสงหรือความสว่างของสีเทานั้น การประมวลผลภาพสสีามารถแบ่งออกได้เป็น 2 วิธีคือ

1) ทำการประมวลผลภาพในแต่ละแบนด์ย่อยของแบบจำลองสี RGB
2) ทำการเปลี่ยนแบบจำลองสี RGB เปนแบบจำลองสี YIQ แล้วทำการประมวลผลภาพเฉพาะในแบนดยอย Y เทานั้น (แบนด์ I และ Q ไมตองทำการประมวลผลภาพ) จากนั้นทำการเปลี่ยน แบบจำลองสี YIQ เป็นแบบจำลองสี RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 เซอร์โวมอเตอร์

เซอร์โวมอเตอร์ คือระบบควบคุมที่ประกอบด้วยไฟฟ้าคอนโทรลและเครื่องกล ซึ่งใช้สำหรับงานที่ต้องการควบคุมตำแหน่งความเร็ว แรงบิด ความแม่นยำและความรวดเร็ว เพื่อให้เครื่องกลและไฟฟ้าคอนโทรลทำงานสอดคล้องกันอย่างมีประสิทธิภาพ โดยใช้การควบคุมแบบป้อนกลับ (Feedback controller) ซึ่งคือระบบควบคุมที่มีการวัดค่าเอาต์พุตของระบบ แล้วนำมาเปรียบเทียบกับค่าอินพุตเพื่อควบคุมและปรับแต่งให้ค่าเอาต์พุตของระบบให้มีค่าเท่ากับหรือใกล้เคียงกับค่าอินพุต

รูปที่ 2.7 ลักษณภายในของเซอร์โวมอเตอร์

เซอร์โวมอเตอร์ในปัจจุบันมีจำหน่ายอยู่หลากหลายรูปแบบ ทั้งแบบกระแสตรง (DC motor) และแบบกระแสสลับ (AC motor) ด้วยคุณสมบัติเด่นของเซอร์โวมอเตอร์ คือ สามารถให้แรงบิดที่สูง (มีหน่วยเป็นนิวตันเมตร) มีทำงานได้รวดเร็ว สามารถเคลื่อนที่ได้อย่างแม่นยำและทำงานได้เงียบ จึงมีการนำเอาเซอร์โวมอเตอร์ไปประยุกต์ใช้งานอย่างหลากหลาย เซอร์โวมอเตอร์ที่ใช้งานมีดังนี้

1) LOBOT LDX-218 17KG ชุดเกียร์แบบมีแรงบิดขนาดใหญ่ Dual shaft digital servo

รูปที่ 2.8 เซอร์โวมอเตอร์ชนิด LDX-218

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียด

- ยี่ห้อ: LOBOT
- รายการ: LOBOT LDX-218
- ความเร็วในการทำงาน (7.4V): 0.16 วินาที / 60 °
- แรงบิดของบูธ (6.6V): 15 กก. ซม
- แรงบิดของบูธ (7.4V): 17 กก. ซม
- ขนาด: 40X20X40.5 มม
- น้ำหนัก: 60 กรัม
- เชื่อมต่อ ลวด ความยาว: 300 มม
- การใช้งาน: สำหรับรุ่น RC

2) Magic-Q QDS-15RO 220 ° 15KG แรงบิดขนาดใหญ่เซอร์โวดิจิตอล Biaxial สำหรับหุ่นยนต์

รูปที่ 2.9 เซอร์โวมอเตอร์ชนิด QDS-15RO

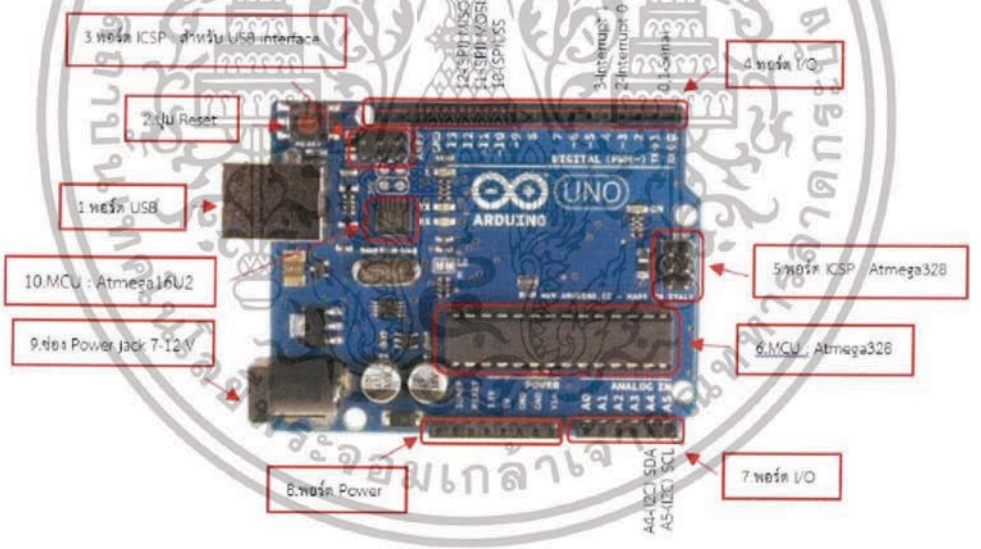
รายละเอียด

- ยี่ห้อ: Magic-Q
- รายการ: QDS-15PO
- PW: 600μs-2400μs
- แรงดันไฟฟ้า: 6-8.4V
- ความเร็วในการทำงาน (8.4V): 0.18sec / 60 °
- แรงบิดของบูธ (6.0V): 13kg / cm
- แรงบิดของบูธ (8.4V): 17kg / cm
- ขนาด: 40 * 20 * 40.5 มม
- น้ำหนัก: 60g
- ความยาวของสาย: 45 ซม.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 บอร์ดอาดุยโน่คอนโทรลเลอร์ (Arduino board)

Arduino คือ โครงการที่นำชิปไอซีไมโครคอนโทรลเลอร์ตระกูลต่างๆ มาใช้ร่วมกันในภาษาซี ซึ่งภาษาซีนี้มีลักษณะเฉพาะ คือมีการเขียนไลบารี่ของ Arduino ขึ้นมาเพื่อให้การสั่งงานไมโครคอนโทรลเลอร์ที่แตกต่างกัน สามารถใช้งานโค้ดตัวเดียวกันได้ โดยตัวโครงการได้ออกบอร์ดทดลองมาหลายๆ รูปแบบเพื่อใช้งานกับ IDE ของตนเอง สาเหตุหลักที่ทำให้ Arduino เป็นนิยมมาก เนื่องจากซอฟแวร์ที่ใช้งานร่วมกันสามารถโหลดได้ฟรีและตัวบอร์ดทดลองยังถูกแจกแปลน ทำให้ผู้ผลิตจีนนำไปผลิตและขายออกตลาดมาในราคาที่ถูก จากที่ได้กล่าวไปแล้ว Auduino นั้นได้ใช้ชิป AVR เป็นหลักใน Auduino แทบรุ่น สาเหตุมาจากไมโครคอนโทรเลอร์ของตะกูล AVR นั้นมีความทันสมัย ในชิปในบางตัวสามารถเชื่อมต่อผ่าน USB ได้โดยตรง สามารถใช้กับคอมพิวเตอร์สมัยใหม่ได้เป็นอย่างดี และในไมโครคอนโทรเลอร์ตะกูล AVR ยังมีส่วนของโปรแกรมพิเศษที่เรียกว่า Bootloader อยู่ในระดับล่างกว่าส่วนโปรแกรมปกติ ซึ่งจะเป็นส่วนโปรแกรมที่จะถูกเรียกขึ้นมาก่อนการเรียกโปรแกรมปกติ ทำให้สามารถเขียนคำสั่งให้ทำงานใดๆ ก็ได้ ก่อนการเรียกโปรแกรมปกติ ทำให้ Arduino นั้นอาศัยส่วนโปรแกรมแกรมพิเศษนี้ในการทำให้ชิปสามารถโปรแกรมผ่านพอร์ตอนุกรมชนิด UART ได้ จึงทำให้การเขียนโปรแกรมลงไปในชิปใช้เพียง USB to UART ก็เพียงพอแล้ว แต่การโปรแกรมด้วยการใช้โปรโตคอล UART มีข้อเสียคือต้องใช้เวลาในการบูตเข้าโปรแกรมปกติประมาณ 1–2 วินาที



รูปที่ 2.10 ภาพรวมบอร์ด Arduino R3 และ พอร์ทเชื่อมต่อ

จากภาพเบื้องต้นสามารถอธิบายได้ดังนี้

1) USB Port: ใช้สำหรับต่อกับ Computer เพื่ออับโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
2) Reset Button: เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ICSP Port ของ Atmega16U2: เป็นพอร์ตที่ใช้โปรแกรม Visual com port บน Atmega16U2
4) I/O Port: Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx, Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM
5) ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6) MCU: Atmega328 เป็น MCU ที่ใช้บนบอร์ด Arduino
7) I/O Port: นอกจากจะเป็น Digital I/O แล้ว ยังเปลี่ยนเป็นช่องรับสัญญาณอนาล็อก ตั้งแต่ขา A0-A5
8) Power Port: ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขาไฟเลี้ยง +3.3 V, +5V, GND, V_{in}
9) Power Jack: รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V
10) MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่าน Atmega16U2

2.5 กล้องเว็บแคม

ในโครงงานนี้ได้ใช้กล้องเว็บแคม OKER OP-002 ซึ่งสามารถแสดงผลได้ดี และมีระบบในการปรับสมดุลของภาพเพื่อให้ได้สีของภาพที่เป็นธรรมชาติสมจริง และสามารถปรับความคมชัดผ่านเลนส์ได้

รูปที่ 2.11 กล้องเว็บแคม OKER OP-002

รายละเอียด

- Resolution: 300k pixels และสามารถ ได้สูงถึง 12 0M pixels โดยซอฟแวร์
- Frame rate: 30 fps
- USB 1.1/2.0 interface
- Info noise rate: 48db
- Focus range: 30 mm-infinitives
- Compatibility: Windows 2000/XP/Vista/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
วิธีดำเนินการวิจัย

ในบทนี้จะกล่าวถึงวิธีดำเนินการวิจัยตลอดจนขั้นตอนการทำงานต่างๆ ของหุ่นยนต์แขนกลที่ใช้การประมวลผลภาพในการเคลื่อนย้ายวัตถุหรือหยิบจับสิ่งของจากตำแหน่งหนึ่งไปยังอีกตำแหน่งหนึ่งโดยมีการใช้ข้อมูลภาพที่ได้จากกล้องดิจิตอลโดยใช้วิธีการประมวลผลภาพ โดยงานวิจัยชิ้นนี้เป็นงานวิจัยที่มุ่งเน้นในเรื่องราวการใช้อัลกอริทึมของการพัฒนาทางด้านซอฟท์แวร์ผสมผสานกับการใช้การประมวลผลภาพเพื่อควบคุมหุ่นยนต์แขนกล

3.1 หลักการทำงาน

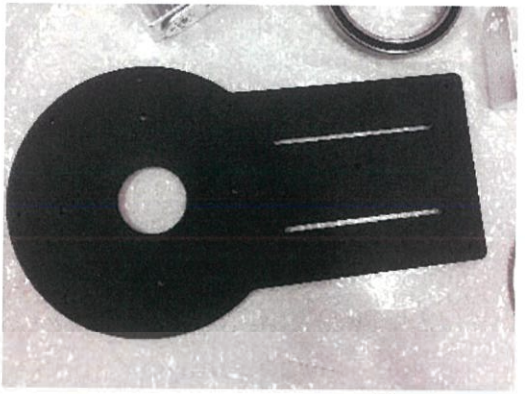
ระบบการทำงานของแขนกลในการเคลื่อนย้ายวัตถุหรือสิ่งของนั้น มี 2 รูปแบบการทำงาน โดยฟังก์ชันแรกเริ่มจากตัวของแขนกลเคลื่อนที่ไปหยิบลูกบอล 3 สี คือ สีน้ำเงิน สีเหลืองและสีแดง แล้วจะนำไปตรวจสอบที่กล้อง Webcam ว่าตรวจจับเป็นสีอะไรถ้าเป็นสีแดงตัวแขนกล จะนำไปวางใส่ถ้วยสีแดงและสีอื่นๆ ตามตำแหน่งที่ได้กำหนดไว้ และฟังก์ชันที่สอง ตัวแขนกลจะเคลื่อนที่ตามตำแหน่งของวัตถุที่ถูกจับได้ด้วยกล้อง Webcam ตามพิกัดของ X และ Y โดยการทำงานของทั้งสองฟังก์ชันเริ่มต้นโดยกล้องรับภาพที่ติดอยู่เหนือของตัวแขนกลจะทำการรับภาพวัตถุที่ต้องการจะเคลื่อนย้ายหรือหยิบแล้วส่งภาพของวัตถุนั้นเข้าสู่กระบวนการประมวลผลภาพเพื่อใช้หาค่าของสีหรือพิกัดตำแหน่งของวัตถุที่ต้องการ จากนั้นจะส่งค่าพิกัดตำแหน่งของวัตถุไปยังไมโครคอนโทรลเลอร์ เพื่อใช้ในการควบคุมการหมุนของเซอร์โวเมอเตอร์ในการหยิบวัตถุและนำไปวางในตำแหน่งที่ต้องการ

3.2 การประกอบตัวชิ้นงาน

ในการประกอบตัวแขนกลซึ่งเป็นองค์ประกอบหลักที่มีความสำคัญ คือต้องมีความแข็งแรงและทนทาน ทางผู้จัดทำจึงได้ทำการสั่งทำโครงตัวแขนกลที่มีความทนทานและแข็งแรงเพื่อให้ตัวแขนกลมีประสิทธิภาพในการทำงานมากที่สุด โดยชิ้นงานจะประกอบด้วย 3 ส่วนหลัก ต่างๆ ดังนี้

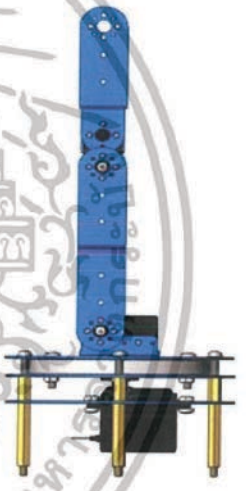
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ฐานของตัวแขนกล



รูปที่ 3.1 ฐานของแขนกล

2 .แขนกลส่วนล่าง



รูปที่ 3.2 แขนกลส่วนล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แขนกลส่วนบน

รูปที่ 3.3 แขนกลส่วนบน

3.3 การพัฒนาทางด้านซอฟต์แวร์ในการควบคุมเซอร์โวมอเตอร์

ในการควบคุมเซอร์โวมอเตอร์แต่ละแกนนั้นจะมีคำสั่งพื้นฐานดังนี้

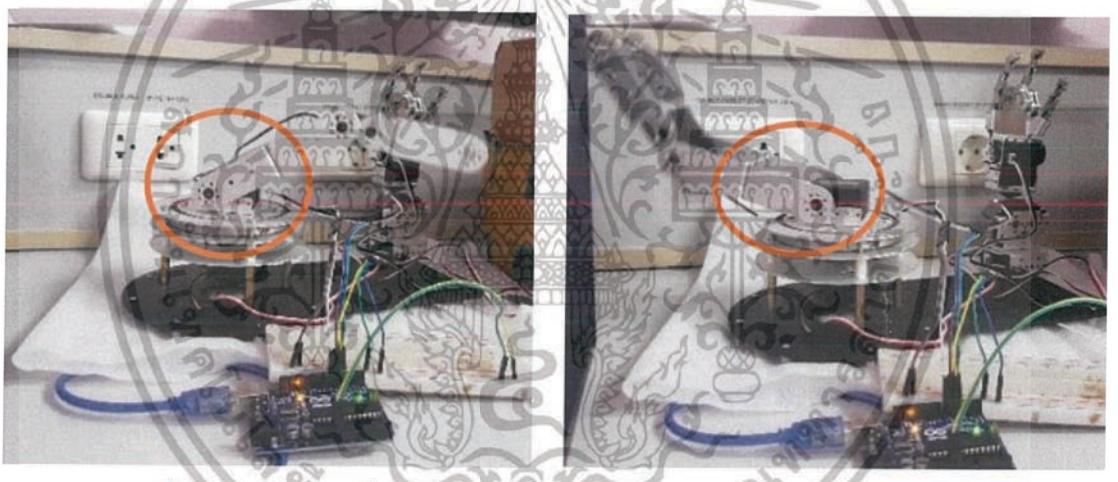
```
#include <Servo.h>

// STEP 1.                          ประกาศ Obj เพื่อการควบคุม Servo
Servo myservo1;  //                 servo ตัวที่ 1
Servo myservo2;  //                 servo ตัวที่ 2
int pos = 0;     //                 ค่าตำแหน่งเริ่มต้นของ Servo
void setup()
{
  // STEP 2.                        กำหนดให้ขาที่ต้องการเป็นขาควบคุม Servo ตัวนั้นๆ
  myservo1.attach(5);  //           กำหนดให้ขา D5 เป็นขาควบคุม Servo1
  myservo2.attach(6);  //           กำหนดให้ขา D6 เป็นขาควบคุม Servo2
void loop()
{
  // STEP 3.                        สั่งให้ Servo หมุนไปในตำแหน่งที่ต้องการ
  // ------------------------------------------------------
  // สั่งให้หมุนไป..
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  for(pos = 0; pos < 180; pos += 2)  // รอบการสั่งให้หมุนไปจาก 0 ไปจนถึง 180 องศา ทีละ 2
step
  {
    myservo1.write(pos);
    myservo2.write(pos);
    delay(15);                  // หน่วงเวลารอ servo ทำงานให้เสร็จ (เพราะ mcu ทำงานเร็วกว่าservo)
  }
  // สั่งให้หมุนกลับ
  for(pos = 180; pos>=1; pos -= 2)     // รอบการสั่งให้หมุนกลับจาก 180 มาที่ 0
องศา
    myservo1.write(pos);
    myservo2.write(pos);
    delay(15);                        // หน่วงเวลา
```

จากตัวอย่างเป็นการอธิบายการสั่งการควบคุมเซอร์โวเพียง 2 ตัว



รูปที่ 3.4 การเคลื่อนที่ของแขนกลจากตำแหน่งที่ 0 องศา ไปจนถึง 180 องศา

3.4 การคัดแยกสี

ในการตรวจจับวัตถุจากการติดตามสีด้วย OpenCV กับ Python มีคำสั่งใช้งานดังนี้

[cv2.inRange(hsv,ค่าสี HSV ช่วงเริ่มต้น , ค่าสี HSV ช่วงสิ้นสุด)]

เป็นคำสั่งใช้ตรวจจับสีที่อยู่ในช่วงสี HSV ที่กำหนด ในการกำหนดค่าสี HSV ช่วงเริ่มต้น - สิ้นสุด ส่วนใหญ่จะใช้โมดูล Numpy เข้ามาช่วยเรื่องการเขียนช่วงข้อมูลสี HSV

[cv2.bitwise_and(cv2.bitwise_and(src1, src2[, dst[, mask]]))]

1. src1 - ชุดข้อมูลแรก
2. src2 - ชุดข้อมูลที่สอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. dst - ชุดข้อมูลผลลัพธ์ที่มีขนาดและชนิดเดียวกันเป็นชุดข้อมูลสำหรับการป้อนค่า
4. mask – เป็นเหมือนหน้ากากที่รับค่ามารวมด้วย

ตัวอย่างโค้ด ตรวจจับช่วงสีเหลืองด้วย OpenCV กับ Python

```
import cv2 ( เรียกเปิดกล้อง webcam )
import numpy as np
cap = cv2.VideoCapture(0) # รับวิดีโอจากกล้อง
while(1):
    # รับข้อมูลจากเว็บแคม
    _, frame = cap.read()
    # แปลงสี BGR ไปยัง HSV
    hsv = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    # ช่วงของสีเหลืองในระบบ HSV
    lower_yellow = np.array([10,100,100], dtype=np.uint8)
    upper_yellow = np.array([30,255,255], dtype=np.uint8)
    # จำกัดภาพ HSV รับเฉพาะสีเหลือง
    yellow = cv2.inRange(hsv, lower_yellow, upper_yellow)
    # Bitwise-AND mask และภาพต้นฉบับ
    mask = cv2.bitwise_and(frame,frame, mask= yellow)
    cv2.imshow('frame',frame)
    cv2.imshow('yellow',yellow)
    cv2.imshow('mask',mask)
    k = cv2.waitKey(5) & 0xFF
    if k == 27:
        break
```

3.5 การตรวจจับวัตถุ

การตรวจจับวัตถุโดยใช้ Haar cascade เป็นส่วนที่สำคัญที่สุดในส่วนของ Image processing เพราะ การที่จะเลือกหยิบวัตถุได้ จำเป็นที่จะต้องจำแนกวัตถุได้เพื่อตรวจจับวัตถุนั้น แล้วนำไปหาพิกัดของวัตถุ เพื่อนำพิกัดไปควบคุมการหมุนของแขนกล

3.5.1 ทำ Haar cascade

1. ทำการเทรน วัตถุที่ต้องการ โดยใช้รูป Negative และ Positive
2. เมื่อได้ไฟล์วัตถุที่ต้องการเทรน บันทึกเป็นไฟล์ .XML

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 โหลดภาพเข้าตัวแปรและแปลงเป็นโหมดสีเทา

1. สร้างไฟล์ Python และนำไฟล์ .XML ที่สร้างไว้
2. ทำการแปลงเป็นโหมดสีเทา

```
face_cascade = cv2.CascadeClassifier('haarcascade_eye.xml')

cap = cv2.VideoCapture(0)

while 1:
    ret, img = cap.read()
    cv2.resizeWindow('img', 500,500)
    cv2.line(img,(500,250),(0,250),(0,255,0),1)
    cv2.line(img,(250,0),(250,500),(0,255,0),1)
    cv2.circle(img, (250, 250), 5, (255, 255, 255), -1)
    gray  = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, 1.3)
```

รูปที่ 3.5 แสดงคำสั่งดึงไฟล์ที่สร้างไว้ และแปลงเป็นโหมดสีเทา

3.5.3 การตรวจจับ

เมื่อเจอรูปที่เข้าโหมดสีเทามาแล้วนั้น มาตรวจหาวัตถุตามไฟล์ .XML โดยมีขั้นตอนคือ

1) โหลดภาพ
2) นำภาพเข้าสู่โหมดสีเทา
3) ตรวจหาจากไฟล์ .XML
4) เมื่อเจอวัตถุ จะทำการตีกรอบล้อมรอบวัตถุที่ต้องการ
5) เมื่อได้กรอบวัตถุ จุดศูนย์กลางของกรอบ จะเป็นพิกัด นำพิกัดที่ได้ ส่งเป็นเอาต์พุตไปที่ ardruino

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for (x,y,w,h) in faces:
        cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,0),5)
        roi_gray  = gray[y:y+h, x:x+w]
        roi_color = img[y:y+h, x:x+w]

        arr = {y:y+h, x:x+w}
        print (arr)

        print ('X :' +str(x))
        print ('Y :'+str(y))
        print ('x+w :' +str(x+w))
        print ('y+h :' +str(y+h))
```

รูปที่ 3.6 แสดงคำสั่งการตีกรอบล้อมรอบวัตถุ

```
xx = int(x+(x+h))/2
        yy = int(y+(y+w))/2

        print (xx)
        print (yy)

        center = (xx,yy)

        print("Center of Rectangle is :", center)
        data = "X{0:d}Y{1:d}Z".format(xx, yy)
        print ("output = '" +data+ "'")
        arduino.write(data)
```

รูปที่ 3.7 แสดงคำสั่งให้จุดศูนย์กลางของกรอบเป็นตำแหน่ง

3.6 การควบคุมแขนกล

แขนกลจะเคลื่อนที่ไปหยิบวัตถุ ในตำแหน่งใดๆ โดยรับค่าพิกัดจากการทำ image processing โดยจะเชื่อมต่อกับ Arduino ผ่านทางพอร์ต Usb serial port แบบ Real time จากนั้น Arduino จะไปควบคุมการหมุนของเซอร์โวมอเตอร์ให้ไปหยิบวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.1 การรับค่าพิกัดผ่านทาง usb port serial แบบ realtime

```
}
 if(Serial.available() > 0)
 {
   if(Serial.read() == 'X')
   {
     x = Serial.parseInt();
     if(Serial.read() == 'Y')
     {
       y = Serial.parseInt();
      Pos();
     }
   }
   while(Serial.available() > 0)
   {
     Serial.read();
   }
```

รูปที่ 3.8 แสดงคำสั่งรับค่าพิกัดในแกน x และ y

3.6.2 สั่งให้เซอร์โวมอเตอร์หมุนตามพิกัดที่ได้รับ

```
void Pos()
{
  if(prevX != x || prevY != y)
  {
    int servoX = map(x, 600, 0, 70, 179);
    int servoY = map(y, 450, 0, 179, 95);
    servoX = min(servoX, 179);
    servoX = max(servoX, 70);
    servoY = min(servoY, 179);
    servoY = max(servoY, 95);

    servoHor.write(servoX);
    servoVer.write(servoY);
  }
}
```

รูปที่ 3.9 แสดงคำสั่งให้เซอร์โวมอเตอร์หมุน

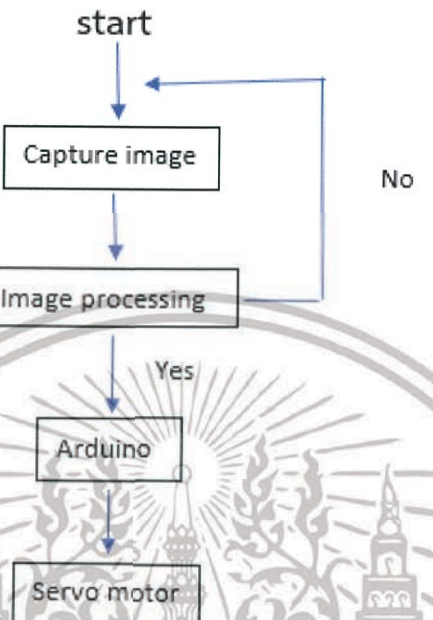
3.6.3 การเคลื่อนที่ของแขนกล

เซอร์โวมอเตอร์ตัวอื่นๆ จะเคลื่อนที่ก้มลงไปเพื่อหยิบวัตถุ โดยมอเตอร์จะทำงานสัมพันธ์กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 แผนผังการทำงานของแขนกล

การทำงานของหุ่นยนต์แขนกลเคลื่อนย้ายวัตถุที่ใช้การประมวลผลภาพสามารถเขียนแสดงได้ดังแผนภาพในรูปที่ 3.10



รูปที่ 3.10 แสดง Flow chart การทำงานของแขนกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 การจำแนกวัตถุ และจำแนกสี

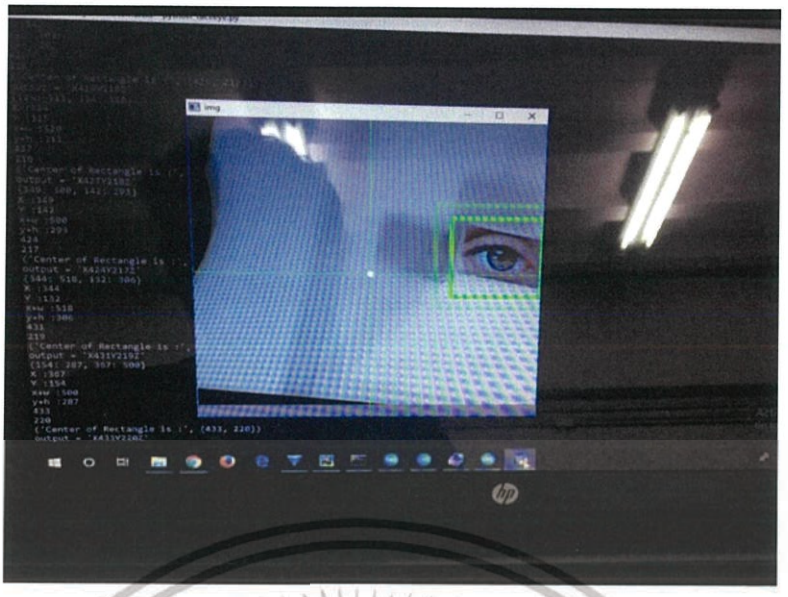
การทดลองในหัวข้อนี้คือการจำแนกวัตถุและการระบุตำแหน่งของวัตถุในรูปแบบระบบพิกัด x, y โดยใช้วิธีการประมวลผลภาพ ซึ่งวัตถุที่นำมาทดลองจะเป็นวัตถุทรงสี่เหลี่ยมผืนผ้า ทำจากแผ่นฟิวเจอร์บอร์ด และได้ทำการติดรูปตาไว้ที่วัตถุ โดยวัตถุจะวางอยู่บนพื้นซึ่งอยู่ระนาบเดียวกับฐานของแขนกล ส่วนกล้องที่ใช้รับภาพจะติดอยู่ที่ขาตั้งกล้องและวางไว้ตำแหน่งที่สูงกว่าความสูงของแขนกล

4.1.1 การทดลองจำแนกวัตถุ

ในการทดลองนี้ได้ทำการทดลองจำแนกวัตถุที่ต้องการ โดยที่วัตถุที่ต้องการจะติดรูปตาไว้ จากนั้นกล้องเว็บแคมจะตรวจจับภาพของวัตถุที่มีรูปตาติดอยู่ แล้วทำการส่งพิกัดไปให้โปรแกรมที่ได้ทำการเขียนโปรแกรมไว้เพื่อส่งค่าไปควบคุมเซอร์โวมอเตอร์ให้ไปหยิบวัตถุตามที่ต้องการ

รูปที่ 4.1 วัตถุรูปกล่องสี่เหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงการตรวจจับวัตถุ

ในการทดสองนี้จะแสดงผลลัพธ์จากการทดลองจำแนกวัตถุที่ต้องการ โดยวัตถุที่ต้องการจะติดรูปตาไว้ที่วัตถุ เมื่อกล้องจับภาพและเจอรูปตา กล้องจะทำการตรวจจับเฉพาะรูปตา ซึ่งทำการทดลอง 20 ครั้ง ทั้งหมด 2 รอบรอบละ 10 ครั้ง ผลการทดลองโดยใช้การประมวลผลภาพสามารถแสดงผลลัพธ์ได้ดังตารางที่ 4.1

ตารางที่ 4.1 ตารางแสดงผลการจำแนกวัตถุจำนวน 50 ครั้ง แบ่งเป็น 2 รอบ

ครั้งที่	รอบที่ 1	รอบที่ 2
	สำเร็จ / ไม่สำเร็จ	สำเร็จ / ไม่สำเร็จ
1	✓	✓
2	✓	✗
3	✗	✓
4	✓	✓
5	✓	✓
6	✓	✓
7	✗	✓
8	✓	✓
9	✓	✗
10	✓	✓
11	✓	✓
12	✓	✗
13	✓	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 (ต่อ) ตารางแสดงผลการจำแนกวัตถุจำนวน 50 ครั้ง แบ่งเป็น 2 รอบ

ครั้งที่	รอบที่ 1	รอบที่ 2
	สำเร็จ / ไม่สำเร็จ	สำเร็จ / ไม่สำเร็จ
14	✓	✓
15	✓	✓
16	✓	✗
17	✓	✓
18	✗	✓
19	✓	✓
20	✓	✗
21	✓	✗
22	✓	✓
23	✓	✓
24	✓	✓
25	✗	✓
26	✓	✓
27	✓	✗
28	✗	✓
29	✓	✓
30	✓	✓
31	✓	✗
32	✗	✓
33	✓	✓
34	✓	✗
35	✓	✓
36	✗	✓
37	✓	✓
38	✓	✗
39	✗	✓
40	✓	✓
41	✓	✓
42	✓	✗
43	✓	✓
44	✓	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

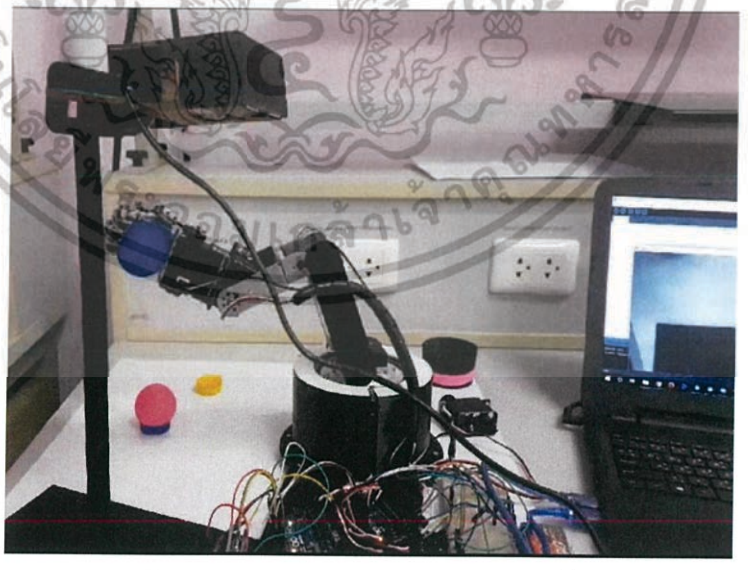
ตารางที่ 4.1 (ต่อ) ตารางแสดงผลการจำแนกวัตถุจำนวน 50 ครั้ง แบ่งเป็น 2 รอบ

ครั้งที่	สีแดง	สีน้ำเงิน
	สำเร็จ / ไม่สำเร็จ	สำเร็จ / ไม่สำเร็จ
45	✗	✓
46	✓	✓
47	✓	✗
48	✓	✓
49	✓	✓
50	✓	✓

จากตารางแสดงให้เห็นถึงผลลัพธ์ของการจำแนกและตรวจจับวัตถุที่ต้องการ จำนวนทั้งหมด 100 ครั้ง โดยแบ่งเป็น 2 รอบ รอบละ 50 ครั้ง พบว่า รอบที่ 1 จากการจำแนกทั้งหมด 50 ครั้งพบว่ามีการจำแนกสำเร็จทั้งหมด 41 ครั้ง และไม่สำเร็จทั้งหมด 9 ครั้ง และจากการทดลองรอบที่ 2 จากการจำแนกทั้งหมด 50 ครั้งพบว่ามีการจำแนกสำเร็จทั้งหมด 39 ครั้ง และไม่สำเร็จทั้งหมด 11 ครั้ง จากการทดลองทั้งหมดเมื่อนำความสำเร็จมาคำนวณแล้ว พบว่ามีเปอร์เซ็นต์ความสำเร็จอยู่ที่ 80 เปอร์เซ็นต์

4.1.2 การทดลองจำแนกสี

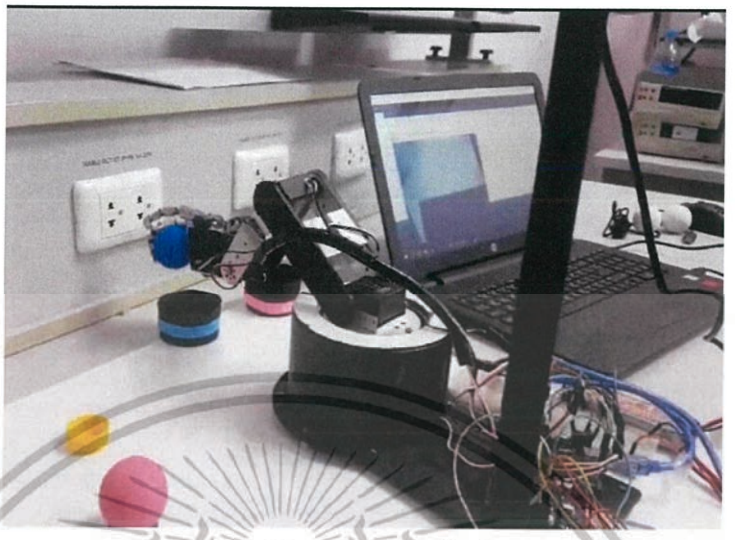
ในการทดลองนี้ได้ทำการทดลองจำแนกวัตถุที่ต้องการโดยการทดลองนี้จะใช้ลูกบอลสีจำนวน 2 สี โดยให้แขนกลหยิบลูกบอลแต่ละสีจากนั้นกล้องเว็บแคมจะตรวจจับสีของลูกบอลแล้วกล้องจะทำการคัดแยกว่าเป็นสีอะไรแล้วจากนั้นจะส่งตัวแปรของแต่ละสี ไปให้โปรแกรมที่ได้ทำการเขียนโปรแกรมไว้เพื่อส่งค่าไปควบคุมเซอร์โวมอเตอร์ให้ไปวางในตำแหน่งที่ต้องการ



รูปที่ 4.3 แสดงการทดลองหยิบลูกบอลสีน้ำเงิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 4.3 แขนกลได้ทำการหยิบลูกบอลแล้วนำลูกบอลไปตรวจสอบที่กล้อง เพื่อดูว่าตรวจจับได้เป็นสีอะไรจากการทดลองได้ผลลัพธ์เป็นสีน้ำเงิน



รูปที่ 4.4 แสดงการวางลูกบอลสีน้ำเงินในถาดสีน้ำเงิน

ในการทดลองนี้ แสดงผลลัพธ์จากการทดลองการจำแนกสีของลูกบอลสีน้ำเงินและสีแดง ทั้งหมด 20 ครั้ง เป็นจำนวน 2 รอบ รอบละ 10 ครั้ง ผลการทดลองโดยใช้การประมวลผลภาพสามารถแสดงผลลัพธ์ได้ดังตารางที่ 4.2

ตารางที่ 4.2 แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 1

ครั้งที่	สีแดง	สีน้ำเงิน
	สำเร็จ / ไม่สำเร็จ	สำเร็จ / ไม่สำเร็จ
1	✓	✓
2	✓	✓
3	✓	✓
4	✓	✓
5	✓	✓
6	✗	✓
7	✓	✓
8	✓	✓
9	✓	✓
10	✓	✓
11	✓	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 (ต่อ) แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 1

12	✓	✓
13	✓	✓
14	✓	✓
15	✓	✓
16	✓	✓
17	✓	✓
18	✓	✓
19	✓	✓
20	✓	✗
21	✓	✓
22	✓	✓
23	✓	✓
24	✓	✓
25	✓	✓
26	✓	✓
27	✓	✓
28	✓	✓
29	✓	✓
30	✓	✓
31	✓	✓
32	✓	✓
33	✓	✓
34	✓	✓
35	✓	✓
36	✓	✓
37	✓	✓
38	✗	✓
39	✓	✓
40	✓	✓
41	✓	✓
42	✓	✓
43	✓	✓
44	✓	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 (ต่อ) แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 1

45	✓	✓
46	✓	✓
47	✓	✓
48	✓	✓
49	✓	✓
50	✓	✓

ตารางที่ 4.3 แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 2

ครั้งที่	สีแดง	สีน้ำเงิน
	สำเร็จ / ไม่สำเร็จ	สำเร็จ / ไม่สำเร็จ
1	✓	✓
2	✓	✓
3	✓	✓
4	✓	✓
5	✓	✓
6	✓	✓
7	✓	✓
8	✓	✓
9	✓	✓
10	✓	✓
11	✓	✓
12	✓	✓
13	✓	✓
14	✓	✓
15	✓	✓
16	✓	✓
17	✓	✓
18	✓	✓
19	✓	✓
20	✓	✓
21	✓	✓
22	✗	✓
23	✓	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 (ต่อ) แสดงผลการทดลองหยิบลูกบอลสีแดงและสีน้ำเงิน 50 ครั้ง รอบที่ 2

24	✓	✓
25	✓	✓
26	✓	✓
27	✓	✓
28	✓	✓
29	✓	✓
30	✓	✓
31	✓	✓
32	✓	✓
33	✓	✓
34	✓	✓
35	✓	✓
36	✓	✓
37	✓	✓
38	✓	✓
39	✓	✓
40	✓	✓
41	✓	✓
42	✓	✓
43	✓	✓
44	✓	✓
45	✓	✓
46	✓	✓
47	✓	✓
48	✓	✓
49	✓	✓
50	✓	✓

จากตารางแสดงให้เห็นถึงผลลัพธ์ของแขนกลที่ทำการหยิบลูกบอลสีน้ำเงินและสีแดง ที่จำนวนทั้งหมด 100 ครั้ง แบ่งเป็น 2 รอบ รอบละ 50 ครั้ง โดยหยิบสีน้ำเงินสำเร็จทั้งหมดจำนวน 100 ครั้ง มีความผิดพลาดเกิดขึ้น 1 ครั้ง เปอร์เซ็นต์ความสำเร็จคิดเป็น 99 เปอร์เซ็นต์ และทำการหยิบลูกบอลสีแดงอีก 100 ครั้ง พบว่าหยิบสำเร็จทั้งหมด 97 ครั้ง มีความผิดพลาดเกิดขึ้น 3 ครั้ง เปอร์เซ็นต์ความสำเร็จคิดเป็น 97 เปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองหยิบวัตถุที่ตำแหน่งใด ๆ

รูปที่ 4.5 แสดงการหยิบวัตถุในตำแหน่งใดๆ

การทดลองหัวข้อนี้จะเป็นการทดลองหยิบวัตถุที่ต้องการ ซึ่งวัตถุจะวางในตำแหน่งใดๆ บนระนาบ โดยจะทำการทดลองทั้งหมด รอบละ 10 ครั้ง ทั้งหมด 2 รอบ โดยแต่ละครั้ง จะทำการเปลี่ยนตำแหน่งของวัตถุไปยังตำแหน่งใดๆ

ตารางที่ 4.4 ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 1

ครั้งที่	ตำแหน่ง	สำเร็จ / ไม่สำเร็จ
1	(0, 0)	✓
2	(68, 92)	✓
3	(123, 20)	✗
4	(20, 80)	✓
5	(46, 53)	✓
6	(79, 165)	✓
7	(92, 46)	✓
8	(79, 34)	✓
9	(170, 97)	✗
10	(39, 64)	✓
11	(20, 164)	✓
12	(69, 24)	✓
13	(139, 54)	✗

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 (ต่อ) ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 1

14	(39, 56)	✓
15	(59, 167)	✗
16	(43, 124)	✓
17	(54, 24)	✓
18	(89, 44)	✗
19	(9, 44)	✓
20	(69, 24)	✓
21	(39, 6)	✗
22	(139, 74)	✓
23	(39, 64)	✓
24	(39, 3)	✓
25	(9, 74)	✓
26	(39, 99)	✓
27	(4, 44)	✓
28	(19, 64)	✗
29	(79, 6)	✓
30	(39, 84)	✓
31	(39, 64)	✓
32	(56, 75)	✓
33	(149, 4)	✓
34	(73, 14)	✓
35	(39, 68)	✓
36	(6, 74)	✗
37	(38, 59)	✓
38	(73, 94)	✓
39	(9, 77)	✓
40	(37, 64)	✗
41	(36, 87	✓
42	(39, 63)	✓
43	(9., 8)	✓
44	(64, 94)	✓
45	(10, 66)	✗
46	(89, 67)	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 (ต่อ) ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 1

47	(35, 85)	✓
48	(39, 71)	✓
49	(37, 57)	✓
50	(36, 65)	✓

ตารางที่ 4.5 ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 2

ครั้งที่	ตำแหน่ง	สำเร็จ / ไม่สำเร็จ
1	(0, 0)	✓
2	(6, 2)	✓
3	(12, 120)	✓
4	(27 89)	✗
5	(65, 57)	✓
6	(9, 16)	✓
7	(29, 47)	✗
8	(7, 69)	✓
9	(170, 97)	✓
10	(54, 43)	✓
11	(6, 124)	✗
12	(69, 24)	✓
13	(17, 84)	✓
14	(63, 76)	✓
15	(95, 67)	✓
16	(49, 24)	✓
17	(51, 64)	✓
18	(23, 74)	✓
19	(9, 74)	✗
20	(9, 94)	✓
21	(39, 116)	✓
22	(139, 84)	✗
23	(39, 74)	✓
24	(31, 3)	✓
25	(9, 74)	✓
26	(39, 99)	✓

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 (ต่อ) ทดลองหยิบวัตถุที่ตำแหน่งใด ๆ ครั้งที่ 2

27	(4, 44)	✗
28	(19, 64)	✓
29	(77, 6)	✓
30	(39, 64)	✓
31	(3, 74)	✓
32	(5, 35)	✗
33	(117, 34)	✓
34	(73, 144)	✓
35	(39, 88)	✗
36	(6, 94)	✓
37	(38, 89)	✓
38	(73, 94)	✓
39	(9, 77)	✓
40	(37, 64)	✓
41	(37, 27	✓
42	(69, 43)	✓
43	(9., 8)	✓
44	(78, 94)	✓
45	(10, 136)	✓
46	(89, 67)	✗
47	(65, 78)	✓
48	(12, 98)	✓
49	(7, 97)	✓
50	(37, 135)	✓

จากการทดลองหยิบวัตถุที่ตำแหน่งใดๆ บนระนาบ ทั้งหมด 100 ครั้ง เป็นจำนวน 2 รอบ รอบละ 50 ครั้ง โดยแต่ละครั้ง จะทำการเปลี่ยนตำแหน่งของวัตถุไปยังตำแหน่งใดๆ พบว่าในการหยิบ 100 ครั้ง มีการหยิบวัตถุสำเร็จทั้งหมด 81 ครั้ง และเกิดความผิดพลาดในการหยิบ 19 ครั้ง จะได้เปอร์เซ็นต์ในการหยิบสำเร็จ 81 เปอร์เซ็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

ในการทดลองนี้ มีจุดประสงค์เพื่อสร้างหุ่นยนต์แขนกลเคลื่อนย้ายวัตถุที่สามารถจำแนก คัดแยก และเลือกหยิบวัตถุที่ต้องการในตำแหน่งใดๆ และนำไปวางในตำแหน่งที่ต้องการโดยใช้หลักการประมวลผลภาพร่วมกับการเขียนโปรแกรมเพื่อควบคุมการทำงานของหุ่นยนต์แขนกล ซึ่งจากผลการทดลองพบว่า หุ่นยนต์แขนกลสามารถหยิบวัตถุในตำแหน่งใดๆ ได้อย่างถูกต้อง และเมื่อเปลี่ยนตำแหน่งของวัตถุไปยังตำแหน่งใดๆ หุ่นยนต์แขนกลสามารถตามไปหยิบวัตถุในตำแหน่งใหม่ได้อย่างถูกต้องประมาณ 80 เปอร์เซ็นต์ และสามารถนำไปวางในตำแหน่งที่ต้องการได้ ดังนั้นหุ่นยนต์แขนกลที่สร้างขึ้นนี้สามารถคัดแยกและเคลื่อนย้ายวัตถุไปยังตำแหน่งต่างๆ ได้ตามต้องการ

5.2 ข้อเสนอแนะ

ในการพัฒนาหุ่นยนต์แขนกลเคลื่อนย้ายวัตถุที่สามารถจำแนกหรือเลือกหยิบวัตถุที่ต้องการในตำแหน่งใดๆ และนำวัตถุไปวางในตำแหน่งที่ต้องการโดยใช้หลักการประมวลผลภาพนั้น เนื่องจากมอเตอร์ที่ใช้เป็น Servo motor ซึ่งทำให้การเคลื่อนที่ไม่นิ่มนวลและรับน้ำหนักได้น้อย ซึ่งการแก้ไขปัญหาดังกล่าวสามารถเปลี่ยนมอเตอร์จาก Servo motor ไปใช้ DC motor แทน เพื่อให้การเคลื่อนที่ของแขนกลมีความนุ่มนวล และรองรับน้ำหนักในการยกวัตถุสูงขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ดร.มุฑิตา สงฆ์จันทร์, "การควบคุมหุ่นยนต์แขนกลโดยใช้วิธีการเรียนรู้แบบทำซ้ำชนิดค่าพารามิเตอร์ที่เหมาะสมที่สุด", [online] แหล่งที่มา:http://web.eng.nu.ac.th/eng2012/enmis/doc/project/fullpaper/mutita56-complete.pdf

[2] การใช้งานบอร์ด Arduino, [online] แหล่งที่มา:http://www.arduitronics.com/article/installing-driver-arduino-on-window.

[3] Andre Marion, "Introduction to Image Processing", Chapman and Hall in London, New York, 1991.

[4] Shameem Hameed, "Haar Cascade", [online] แหล่งที่มา: https://github.com/voidstellar/haar-cascade-files/blob/master/README.md

[5] James Payne, " Beginning python", Indianapolis : Wiley, 2010.

[6] จักฤษณ์ แสงแก้ว,"การเขียนโปรแกรมภาษาไพธอนด้วยตัวเอง", กรุงเทพ : สมาคมส่งเสริมเทคโนโลยี, 2550

[7] Learn how to train your own OpenCV Haar classifier, [online] แหล่งที่มา:https://github-.com/mrnugget/opencv-haar-classifier-training.

[8] M.K Dabhi, B.K Pancholi, "Face Detection System Based on Viola-Jones Algorithm" International Journal of Science and Research (IJSR)

[9] P.I.Corke, "Visual Control of Robots: high-performance visual servoing," Research Studies Press ltd., Somerset, England, 1996.

[10] T.Kawanishi and H.Murase, "Quick Object Search Using Color Histograms and Pan-Tilt-Zoom Camera - Dynamic Active Search -," Trans. IEICE D-II

[11] N. N. San, N. Anye, "Face and Eye Detection Using Haar Cascade Classifier and Symmetry Detection", International Journal of Information & Computer Science (IJITCS)

[12] T. Ball, Train your own OpenCV Haar classifier, [online] แหล่งที่มา : http://coding-robin.de/2013/07/22/train-your-own-opencv-haar-classifier.html.

[13] ทีมงานสมารทเลิรนนิ่ง, เริ่มตนเรียนรูไมโครคอนโทรลเลอร MCS-51 ดวยภาษา C, พิมพครั้งที่ 1, หางหุนสวนสามัญสมารทเลิรนนิ่ง, 2555

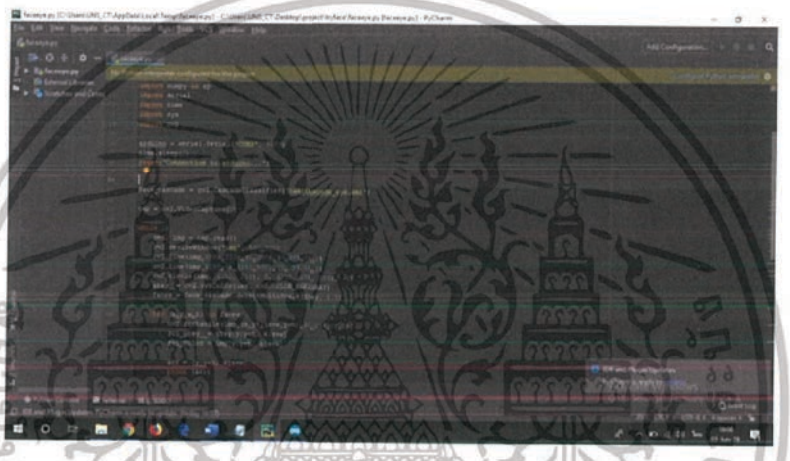
[14] อดุม รานอก, ภาษา C สำหรับงานควบคุมไมโครคอนโทรลเลอร MCS-51, พิมพครั้งที่ 1, บริษัท ไอดี ซี อินโฟ ดิสทริบิวเตอร เซ็นเตอร จำกัด, 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก
โปรแกรมที่ใช้ในโครงงาน

ในบทนี้จะอธิบายถึงส่วนของโปรแกรมต่างๆที่ใช้ในโครงงาน ซึ่งประกอบด้วย 2 ส่วน คือส่วนประมวลผลภาพซึ่งทำการประมวลภาพผ่านทางคอมพิวเตอร์และอีกส่วนคือส่วนควบคุมการทำงานของแขนกล ซึ่งทั้งสองส่วนจะทำการสื่อสารกันผ่านทางพอร์ต serial usb ระหว่างคอมพิวเตอร์กับบอร์ดไมโครคอนโทรลเลอร์

1. โปรแกรมที่ใช้ในส่วนประมวลผลภาพ



รูปที่ ผ.1 หน้าต่างโปรแกรมที่ใช้ภาษาไพธอนในการเขียน

ในส่วนนี้จะใช้ภาษาไพธอนในการเขียน เพื่อให้รับภาพจากกล้องและนำมาประมวลผล ในการติดตามและหาตำแหน่งของวัตถุ เพื่อที่จะส่งค่าพิกัดตำแหน่งของวัตถุ ไปยังไมโครคอนโทรลเลอร์เพื่อใช้ในการควบคุมการเคลื่อนที่ของแขนกลต่อไป

```
import numpy as np
import serial
import time
import sys
import cv2

arduino = serial.Serial('COM3', 9600)
time.sleep(2)
print("Connection to arduino...")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
face_cascade = cv2.CascadeClassifier('haarcascade_eye.xml')
cap = cv2.VideoCapture(0)
while 1:
    ret, img = cap.read()
    cv2.resizeWindow('img', 500,500)
    cv2.line(img,(500,250),(0,250),(0,255,0),1)
    cv2.line(img,(250,0),(250,500),(0,255,0),1)
    cv2.circle(img, (250, 250), 5, (255, 255, 255), -1)
    gray  = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, 1.3)

    for (x,y,w,h) in faces:
        cv2.rectangle(img,(x,y),(x+w,y+h),(0,255,0),5)
        roi_gray  = gray[y:y+h, x:x+w]
        roi_color = img[y:y+h, x:x+w]

        arr = {y:y+h, x:x+w}
        print (arr)
         print ('X :' +str(x))
        print ('Y :'+str(y))
        print ('x+w :' +str(x+w))
        print ('y+h :' +str(y+h))

        xx = int(x+(x+h))/2
        yy = int(y+(y+w))/2

        print (xx)
        print (yy)

        center = (xx,yy)

        print("Center of Rectangle is :", center)
        data = "X{0:d}Y{1:d}Z".format(xx, yy)
        print ("output = '" +data+ "'")
        arduino.write(data)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    cv2.imshow('img',img)

    k = cv2.waitKey(30) & 0xff
    if k == 27:
      break
```

2. โปรแกรมในส่วนคัดแยกสี

```
import cv2
import numpy as np

# capturing video through webcam
cap = cv2.VideoCapture(0)

while (1):
    _, img = cap.read()

    # converting frame(img i.e BGR) to HSV (hue-saturation-value)

    hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)

    # definig the range of red color
    red_lower = np.array([136, 87, 111], np.uint8)
    red_upper = np.array([180, 255, 255], np.uint8)

    # defining the Range of Blue color
    blue_lower = np.array([99, 115, 150], np.uint8)
    blue_upper = np.array([110, 255, 255], np.uint8)

    # defining the Range of yellow color
    yellow_lower = np.array([22, 60, 200], np.uint8)
    yellow_upper = np.array([60, 255, 255], np.uint8)

    # finding the range of red,blue and yellow color in the image
    red = cv2.inRange(hsv, red_lower, red_upper)
    blue = cv2.inRange(hsv, blue_lower, blue_upper)
    yellow = cv2.inRange(hsv, yellow_lower, yellow_upper)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    # Morphological transformation, Dilation
    kernal = np.ones((5, 5), "uint8")

    red = cv2.dilate(red, kernal)
    res = cv2.bitwise_and(img, img, mask=red)

    blue = cv2.dilate(blue, kernal)
    res1 = cv2.bitwise_and(img, img, mask=blue)

    yellow = cv2.dilate(yellow, kernal)
    res2 = cv2.bitwise_and(img, img, mask=yellow)

    # Tracking the Red Color

    (contours,hierarchy)                                                    =
cv2.findContours(red,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)

    for pic, contour in enumerate(contours):
        area = cv2.contourArea(contour)
        if (area > 300):
            x, y, w, h = cv2.boundingRect(contour)
            img = cv2.rectangle(img, (x, y), (x + w, y + h), (0, 0, 255), 2)
            cv2.putText(img, "RED color", (x, y), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255))

    # Tracking the Blue Color
    (contours , hierarchy) = cv2.findContours (blue, cv2.RETR_TREE ,
cv2.CHAIN_APPROX_SIMPLE )
    for pic, contour in enumerate(contours):
        area = cv2.contourArea(contour)
        if (area > 300):
            x, y, w, h = cv2.boundingRect(contour)
            img = cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 2)
            cv2.putText(img, "Blue color", (x, y), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255, 0, 0))

    # Tracking the yellow Color
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    (contours , hierarchy) = cv2.findContours ( yellow, cv2.RETR_TREE ,
cv2.CHAIN_APPROX_SIMPLE )
    for pic, contour in enumerate(contours):
        area = cv2.contourArea(contour)
        if (area > 300):
            x, y, w, h = cv2.boundingRect(contour)
            img = cv2.rectangle(img, (x, y), (x + w, y + h), (0, 255, 0), 2)
            cv2.putText(img, "yellow  color", (x, y), cv2.FONT_HERSHEY_SIMPLEX, 1.0, (0, 255,
0))

    # cv2.imshow("Redcolour",red)
    cv2.imshow("Color Tracking", img)
    # cv2.imshow("red",res)
    if cv2.waitKey(10) & 0xFF == ord('q'):
        cap.release()
        cv2.destroyAllWindows()
        break
```

3. โปรแกรมส่วนคัดแยกสี

```
#include <Servo.h>
Servo servo1;
Servo servo2;
Servo servo3;
Servo servo4;
Servo servo5;
Servo servo6;

int i = 0;
void setup() {
  servo1.attach(5);
  servo2.attach(6);
  servo3.attach(7);
  servo4.attach(8);
  servo5.attach(9);
  servo6.attach(10);
  pinMode(2,INPUT);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  pinMode(3,INPUT);
  pinMode(4,INPUT);
}
 void loop() {
  int red = digitalRead(2);
  int blue  = digitalRead(3);
  int green = digitalRead(4);
   for (i = 70; i < 150; i++) {
    servo1.write(i);
    delay(50);
  }
   for (i = 90; i < 100; i++) {
    servo2.write(i);
    delay(15);
  }
 for (i = 0; i < 90; i++) {
    servo5.write(i);
    delay(30);
  }
 for (i = 70; i > 20; i--) {
 servo3.write(i);
delay(10);
  }
 for (i = 30; i < 80; i++) {
   servo4.write(i);
   delay(50);
 }
  for (i = 90; i < 120; i++) {
   servo2.write(i);
   delay(15);
 }
 for (i = 40; i > 0; i--) {
    servo6.write(i);
    delay(30);
 }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    for (i = 20    ; i < 70; i++) {
    servo3.write(i);
    delay(10);
  }
   for (i = 80; i < 160; i++) {
 servo4.write(i);
  delay(50);
   }
    for (i = 120; i > 90  ; i--) {
     servo2.write(i);
    delay(15);
   }
     if (blue == HIGH){
  for (i = 150; i > 10; i--) {
     servo1.write(i);
     delay(50);
 } for (i = 90; i < 100; i++) {
     servo2.write(i);
     delay(15);
  }
  for (i = 70; i > 20; i--) {
     servo3.write(i);
     delay(10);                }
  for (i = 30; i < 80; i++) {
     servo4.write(i);
     delay(50);
  }
   for (i = 90; i < 120; i++) {
    servo2.write(i);
    delay(15);
  }
   for (i = 90; i > 0; i--) {
    servo5.write(i);
    delay(30);
 }for (i = 0; i < 30; i++) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    servo6.write(i);
    delay(30);
  }
  }else if (red == HIGH){
  for (i = 150; i > 40; i--) {
  servo1.write(i);
  delay(50);
  } for (i = 90; i < 100; i++) {
    servo2.write(i);
    delay(15);
  }
 for (i = 70; i > 20; i--) {
    servo3.write(i);
    delay(10);
 }
 for (i = 30; i < 80; i++) {
    servo4.write(i);
    delay(50);
 }
  for (i = 90; i < 120; i++) {
    servo2.write(i);
    delay(15);
 }
   for (i = 90; i > 0; i--) {
    servo5.write(i);
   delay(30);
 }for (i = 0; i < 30; i++) {
   servo6.write(i);
  delay(30);
 }
 }else if (green == HIGH){

 for (i = 150; i > 70; i--) {
   servo1.write(i);
delay(50);
```

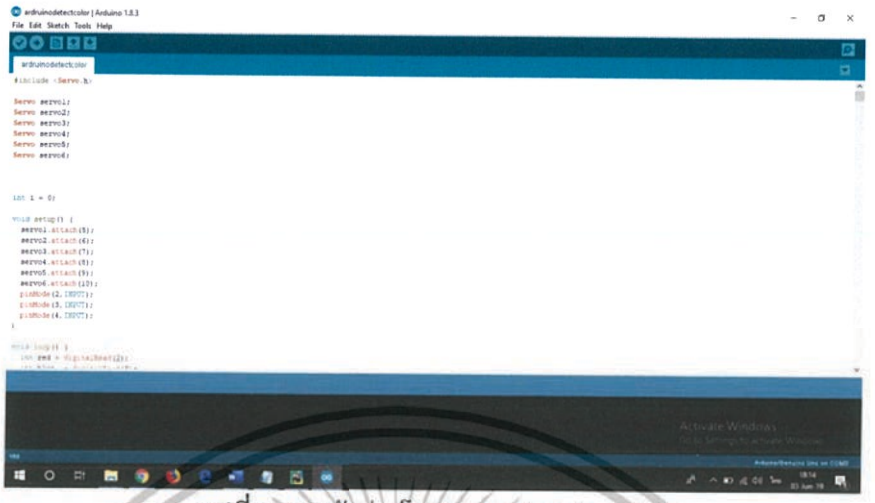
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
} for (i = 90; i < 100; i++) {
 servo2.write(i);
 delay(15);
  }
  for (i = 70; i > 20; i--) {
  servo3.write(i);
  delay(10);
  }
  for (i = 30; i < 80; i++) {
  servo4.write(i);
  delay(50);
  }
   for (i = 90; i < 120; i++) {
    servo2.write(i);

    delay(15);
  }for (i = 0; i < 30; i++) {
    servo6.write(i);
    delay(30);
  }
     for (i = 90; i > 0; i--) {
     servo5.write(i);
  delay(30);
  }
  for (i = 70; i > 0; i--) {
   servo1.write(i);
   delay(50);
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. โปรแกรมในส่วนควบคุมแขนกล



รูปที่ ผ.2 หน้าต่างโปรแกรม Arduino IDE

ในส่วนนี้ใช้โปรแกรม Arduino IDE ในการเขียนโปรแกรมควบคุมการทำงานของแขนกล โดยจะรับค่าพิกัดมาจากส่วนประมวลผลภาพ ในคอมพิวเตอร์ผ่านทางพอร์ต usb serial และนำค่าพิกัดที่ได้มาควบคุมการหมุนของเซอร์โวมอเตอร์ ในแกน x และ y

```
#include<Servo.h>
Servo servoVer; //Vertical Servo
Servo servoHor; //Horizontal Servo
int x;
int y;
int prevX;
int prevY;
void setup()
{
  Serial.begin(9600);
  servoVer.attach(5);
  servoHor.attach(6);
  servoVer.write(90);
  servoHor.write(90);
   pinMode(2,OUTPUT);
}
void Pos()
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  if(prevX != x || prevY != y)
  {
    int servoX = map(x, 600, 0, 70, 179);
    int servoY = map(y, 450, 0, 179, 95);
    servoX = min(servoX, 179);
    servoX = max(servoX, 70);
    servoY = min(servoY, 179);
    servoY = max(servoY, 95);

    servoHor.write(servoX);
   // delay(1000);

    servoVer.write(servoY);
   // delay(1000);

     digitalWrite(2,HIGH);
      delay(2000);
      digitalWrite(2,LOW);
  }
}
void loop()
{
 if(Serial.available() > 0)
 {
  if(Serial.read() == 'X')
  {
    x = Serial.parseInt();
    if(Serial.read() == 'Y')
    {
      y = Serial.parseInt();
     Pos();
    }
  }
  while(Serial.available() > 0)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Serial.read();
    }
  }
}
```

5. โปรแกรมในส่วนแขนกล

ในส่วนนี้จะเป็นการควบคุมแขนกลในการก้มลงไปหยิบวัตถุ โดยจะทำงานสัมพันธ์กับส่วนที่ตามวัตถุตามแกน x และ y

```
#include <Servo.h>
Servo servo1;
Servo servo2;
Servo servo3;
Servo servo4;
Servo servo5;

int i = 0;
int arduino2 =0;

void setup() {
 servo1.attach(5);
 servo2.attach(6);
 servo3.attach(7);
 servo4.attach(8);
 servo5.attach(9);
 pinMode(2,INPUT);
}
void loop() {
 arduino2 = digitalRead(2);
 if (arduino2)
 {
  for (i = 90; i > 45; i--) {
  servo1.write(i);
  delay(20);
 }
 for (i = 0; i < 70; i++) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      servo4.write(i);
      delay(20);
    }
    for (i = 45; i < 130; i++) {
      servo1.write(i);
      delay(30);
    }
     for (i = 200; i > 0; i--) {
     servo3.write(i);
     delay(30);
   }
   for (i = 70; i > 0; i--) {
 servo4.write(i);
 delay(50);
 }
  for (i = 0; i < 200; i++) {
     servo3.write(i);
     delay(30);
  }
   for (i = 130; i > 90; i--) {
    servo1.write(i);
    delay(30);
  }
 }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IEEE THAILAND SECTION

ICA SP-CON 2019

ECTI Association

This is to certify that

Surakiat Trakarntanasakul, Supasin Popone

has successfully completed the presentation of the project entitled

Robotic arm pick up object via image processing

at the Tenth Instrumentation, Control, and Automation Senior Project Conference (ICA SP-CON 2019) on April 30, 2019 at Sirindhorn International Institute of Technology, Thammasat University, Thailand.

.....

Assoc. Prof. Dr. Waree Kongprawechnon
SIIT, Thammasat University
ICA SP-CON 2019 General Chair

.....

Asst. Prof. Dr. Itthisek Nilkhamhang
SIIT, Thammasat University
ICA SP-CON 2019 Technical Program Chair