

เครื่องวัดขนาดวัตถุอัตโนมัติโดยใช้อุปกรณ์ขนาดเล็ก
AUTOMATIC MEASURE OBJECT DIMENSION TESTER



ณัฐวุฒิ คันธนาสินชัย
ภูริพัฒน์ จงเจริญนิรันดร์
ภูวรินทร์ กลิ่นส่ง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมแมคคาทรอนิกส์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

เครื่องวัดขนาดวัตถุอัตโนมัติโดยใช้อุปกรณ์ขนาดเล็ก

AUTOMATIC MEASURE OBJECT DIMENSION TESTER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมแมคคาทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AUTOMATIC MEASURE OBJECT DIMENSION TESTER



THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN MECHATRONICS ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2016

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

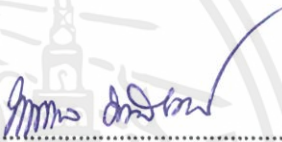
ปริญญาบัตรปีการศึกษา 2559

ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องวัดขนาดวัตถุอัตโนมัติโดยใช้อุปกรณ์ขนาดเล็ก
AUTOMATIC MEASURE OBJECT DIMENSION TESTER

ผู้จัดทำ นายณัฐวุฒิ คັນธนาสินชัย 56010435
นายภูริพัฒน์ จงเจริญรินทร์ 56010955
นายภูวรินทร์ กลิ่นสง 56011210


.....อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.นพดล มณีรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องวัดขนาดวัตถุอัตโนมัติโดยใช้อุปกรณ์ขนาดเล็ก

โดย

นายณัฐวุฒิ คันทนาสินชัย 56010435

นายภูริพัฒน์ จงเจริญรินทร์ 56010955

นายภูวรินทร์ กลิ่นสง 56011220

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ ดร.นพดล มณีรัตน์

ปีการศึกษา 2559

บทคัดย่อ

ปัจจุบันระบบการประมวลผลภาพมีการนำมาใช้ประโยชน์ในด้านต่างๆ มากมาย เช่น คอมพิวเตอร์ โทรศัพท์ โดยเฉพาะอย่างยิ่งในโรงงานอุตสาหกรรมที่ใช้ตรวจสอบวัตถุต่างๆ เป็นต้น การนำเอาการประมวลผลภาพมาใช้ในการวัดวัตถุ โดยวัตถุมีขนาดไม่เกิน 10 x 10 เซนติเมตร โดยเป็นรูปทรงเรขาคณิต และเป็น 2 มิติ โดยการนำตัวประมวลผลขนาดเล็กและราคาถูกคือ บอร์ด Raspberry Pi3 มาใช้งานเป็นตัวประมวลผลหลักทำงานร่วมกับกล้อง Webcam และอุปกรณ์ควบคุมอื่นๆ จะได้เป็นเครื่องวัดขนาดวัตถุ ราคาประหยัด ดังนั้นตัวเครื่องจะประกอบไปด้วยส่วนของ Hardware คือ ตัวกล้อง เซนเซอร์และเซอร์ไวโมเตอร์ที่ติดตั้งบนระบบสายพานลำเลียง ระบบจะทำงานอัตโนมัติ ตั้งแต่การลำเลียง นำวัตถุใส่ถาดเข้าสู่เครื่องวัดขนาด เมื่อโดนเซนเซอร์สายพานจะหยุดกล้องจะถ่ายรูป และ Raspberry Pi จะประมวลผล ถ้าขนาดผิดไปจากที่กำหนดไว้เซอร์ไวโจะทำงานทำให้วัตถุออกจากระบบสายพาน โดยการทำงานเป็นเช่นนี้ไปเรื่อยๆ ดังนั้นเมื่อโครงการนี้สามารถทำได้ตามเป้าหมายแล้ว ยังนำไปต่อยอดทางอุตสาหกรรมและสามารถสร้างรายได้ได้จริง

LINEAR INDUCTION MOTOR CONTROLLER

By

Mr. Natthawut Kantanasinchai 56010435

Mr. Phuriphat Jongcharoennirun 56010955

Mr. Phuwarin Klinsong 56011220

Advisor

Asst.Prof.Dr.Noppadon Maneerat

Academic Year 2016

ABSTRACT

Today's image processing systems are used in many areas, such as computers, telephones, especially in industrial plants. Used to check the various objects, etc. The authors propose to implement image processing to use. Measures the size of the object by an object up to 10 x 10 cm in geometric shape and in 2 D. Raspberry Pi3 is a small and inexpensive processor, the main processor. Works with webcams and other control devices. It will be a small size. So the machine consists of the hardware, the sensor, and the servo motor mounted on the conveyor system. The system will run automatically. Since conveying bring the object into the measuring device if the sensor detect object, belt will stops. The camera will take a picture and Raspberry Pi will process. If the size is wrong, the servo will work to remove the object from the belt system. When this project can achieve the target. It can also be used to increase the industry and make real money.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้และโครงการชิ้นนี้ดำเนินการสำเร็จลุล่วงไปได้ด้วยดีด้วยความช่วยเหลือจากอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.นพดล มณีรัตน์ ซึ่งท่านได้ให้ข้อคิดเห็น และคำแนะนำต่างๆ อันเป็นประโยชน์ในการดำเนินการทำโครงการนี้เป็นอย่างมาก อีกทั้งยังช่วยคิดแนวทางแก้ปัญหาต่างๆ ที่เกิดขึ้นระหว่างทำโครงการชิ้นนี้

ขอขอบคุณคณะอาจารย์และเจ้าหน้าที่ภาควิชาวิศวกรรมการวัดและควบคุม เพื่อนพ้อง ญาติสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง รวมถึงผู้ปกครองและครอบครัวที่ให้การสนับสนุนในด้านต่างๆ อีกทั้งยังคอยเป็นกำลังใจมาโดยตลอดจนสำเร็จการศึกษา คณะผู้จัดทำรู้สึกประทับใจและซาบซึ้งเป็นอย่างยิ่งสำหรับทุกความช่วยเหลือ

ผู้จัดทำ

ณัฐวุฒิ คันธนาสินชัย

ภูริพัฒน์ จงเจริญนิรันดร์

ศิริสิทธิ์ กลิ่นส่ง

สารบัญ

	หน้า
บทคัดย่อ	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ในการทำโครงการ	1
1.3 ขอบเขตของโครงการ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	3
2.1 Raspberry Pi 3 Model 1GB	3
2.1.1 คุณสมบัติบอร์ด Raspberry Pi 3 Model B	3
2.1.2 บอร์ด Raspberry Pi 3 Model B	4
2.2 การประมวลผลภาพ	5
2.2.1 Image Enhancement	6
2.2.2 Image Analysis	6
2.3 ความรู้พื้นฐานด้านการประมวลผลภาพดิจิทัล	6
2.3.1 ภาพสีเทา	7
2.4 การจัดแสงแบบไฟลอบเงา (Fill Light)	8
2.5 Python กับการประมวลผลภาพ	9
2.6 ไลบรารีในงานการประมวลผลภาพ	9
2.7 การหาค่าฮิสโตแกรมของวัตถุภาพ (Edge Detection Histogram)	11
2.8 วิธีการหาขอบภาพ (Edge Detection)	11
2.8.1 วิธี Gradient	11
2.8.2 วิธี Laplacian	12
2.8.3 Image Smoothing	14
2.8.4 Differentiation	14

สารบัญ (ต่อ)

	หน้า
2.8.5 Nonmaxima Suppression	14
2.8.6 Thresholding	14
2.9 ระบบสายพานลำเลียง (Conveyor Belt System)	15
2.10 ระบบการเคลื่อนที่	16
บทที่ 3 หลักการออกแบบและโครงสร้างของวงจรควบคุม	17
3.1 ภาพรวมของระบบ	17
3.2 ส่วนขยายของฮาร์ดแวร์	17
3.2.1 การออกแบบโครงสร้าง	18
3.2.2 อุปกรณ์ที่ใช้ในการทดลอง	19
3.2.3 ส่วนประกอบของชิ้นงาน	21
3.3 ส่วนขยายของซอฟต์แวร์ในการวิเคราะห์ภาพ	23
3.3.1 การถ่ายภาพวัตถุ	23
3.3.2 การทำให้ภาพเป็นสีเทา	23
3.3.3 การปรับปรุงภาพ	23
3.3.4 การตรวจหาขอบของวัตถุ	23
3.3.5 การหาระยะห่างจุดกึ่งกลางขอบทั้งสี่ด้านเพื่อที่จะนำมาคำนวณหาขนาด	23
3.4 ลำดับขั้นตอนการทำงานโดยรวมของโครงการ	23
3.4.1 ขั้นตอนการเตรียมสายพาน	23
3.4.2 ขั้นตอนการเตรียมไฟ	24
3.4.3 ขั้นตอนการตั้งกล้อง	24
3.4.4 ขั้นตอนการทำโปรแกรม	24
บทที่ 4 การทดลองและผลการทดลอง	25
4.1 ผลที่ได้จากการวิเคราะห์ภาพ จากโปรแกรม Python	25
4.1.1 ผลของฝาขวดน้ำ	25
4.1.2 ผลของบัตรเครดิต	29
4.1.3 ผลของเหรียญสิบบาท	33

สารบัญ (ต่อ)

	หน้า
บทที่ 5 บทวิจารณ์และสรุป	38
5.1 สรุปผลการทดลอง	38
5.2 ปัญหาและอุปสรรคที่พบ	38
5.3 แนวทางแก้ไข	38
5.4 ข้อเสนอแนะ	38
5.5 แนวทางการพัฒนา	39
เอกสารอ้างอิง	40
ภาคผนวก	41
ภาคผนวก ก	42
ภาคผนวก ข	45
ภาคผนวก ค	54
ประวัติผู้เขียน	59

สารบัญรูป

รูปที่	หน้า
2.1 บอร์ด Raspberry Pi3 Model B	4
2.2 ส่วนประกอบบอร์ด	4
2.3 คอนเน็คเตอร์สำหรับเชื่อมต่ออินพุตเอาต์พุต	5
2.4 การปรับปรุงคุณภาพภาพ	6
2.5 Image Analysis	6
2.6 การแปลงภาพเป็นภาพขาวดำ	7
2.7 โทนสีเทา	7
2.8 การจัดแสงในทฤษฎี	8
2.9 รูปจำลองการทดลอง	8
2.10 Logo Python	9
2.11 Logo Opencv	9
2.12 Logo Numpy	10
2.13 Logo Scipy	10
2.14 Logo Tkinter	10
2.15 ตัวอย่างการหาขอบด้วยวิธี Gradient	12
2.16 การกำหนดค่า Threshold	12
2.17 การหาขอบด้วยวิธี Lapacian	13
2.18 ขั้นตอนวิธีการหาขอบโดยวิธี Canny	13
2.18 สายพานและลูกกลิ้ง	15
2.19 มอเตอร์	16
3.1 รูปโครงสร้าง	18
3.2 กล้อง Logitech C170	19
3.3 Raspberry Pi 3 Model 1GB	19
3.4 Sensor แสง KY-033	20
3.5 Servo SG 90	20
3.6 แผ่นไม้รองสายงาน	21
3.7 ลูกกลิ้ง	21
3.8 ตูไฟ (1)	22
3.9 ตูไฟ (2)	22

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.10 เหล็กฉาก	22
3.11 ลำดับขั้นตอนการทำงานของโครงงาน	23
4.1 ภาพสีเทาของฝาขวดน้ำ	25
4.2 ภาพ GaussianBlur ฝาขวดน้ำ	25
4.3 ภาพ Canny ฝาขวดน้ำ	26
4.4 ภาพ Dialate ฝาขวดน้ำ	26
4.5 ภาพ Erode ฝาขวดน้ำ	27
4.6 ภาพ Contour และ Imutile ฝาขวดน้ำ	27
4.7 ภาพขนาดฝาขวดน้ำ	28
4.8 ภาพสีเทาของบัตรเครดิต	29
4.9 ภาพ GaussianBlur บัตรเครดิต	29
4.10 ภาพ Canny บัตรเครดิต	30
4.11 ภาพ Dialate บัตรเครดิต	30
4.12 ภาพ Erode บัตรเครดิต	31
4.13 ภาพ Contour และ Imutile บัตรเครดิต	31
4.14 ภาพขนาดบัตรเครดิต	32
4.15 ภาพสีเทาของเหรียญสิบบาท	33
4.16 ภาพ GaussianBlur เหรียญสิบบาท	33
4.17 ภาพ Canny เหรียญสิบบาท	34
4.18 ภาพ Dialate เหรียญสิบบาท	34
4.19 ภาพ Erode เหรียญสิบบาท	35
4.20 ภาพ Contour และ Imutile เหรียญสิบบาท	35
4.21 ภาพขนาดเหรียญสิบบาท	36

สารบัญตาราง

ตารางที่	หน้า
4.1 ขนาดของฝาขวดน้ำ	28
4.2 ขนาดของบัตรเครดิต	32
4.3 ขนาดของเหรียญสิบ	36



บทที่ 1

บทนำ

ปัจจุบันระบบการประมวลผลภาพมีการนำมาใช้ประโยชน์ในด้านต่างๆ มากมาย เช่น คอมพิวเตอร์ โทรศัพท์ โดยเฉพาะอย่างยิ่งในโรงงานอุตสาหกรรมที่ใช้ตรวจสอบวัตถุต่างๆ เป็นต้น การนำเอาการประมวลผลภาพมาใช้ในการวัดขนาดของวัตถุโดยวัตถุเป็นรูปทรงเรขาคณิตและเป็น 2 มิติ โดยการนำตัวประมวลผลขนาดเล็กและราคาถูก มาใช้งานเป็นตัวประมวลผลหลัก ทำงานร่วมกับกล้อง Webcam และอุปกรณ์ควบคุมอื่นๆ จะได้เป็นเครื่องวัดขนาดวัตถุราคาประหยัด เมื่อโครงการนี้สามารถทำได้ตามเป้าหมายแล้ว ยังสามารถนำไปต่อยอดทางอุตสาหกรรมและสามารถสร้างรายได้ได้จริง

1.1 ความสำคัญและที่มาของปริญญานิพนธ์

ปัจจุบันเทคโนโลยีการถ่ายภาพ ได้เข้ามามีบทบาทในชีวิตประจำวันของมนุษย์มากขึ้นเรื่อยๆ ไม่ว่าจะเป็นการใช้เทคโนโลยีถ่ายภาพมาใส่ไว้ในโทรศัพท์มือถือ คอมพิวเตอร์แบบพกพา การจดจำเจ้าของ การจำแนกบุคคลที่เข้าถึงอุปกรณ์ต่างๆ ในกลุ่มอุตสาหกรรมก็มีการนำมาใช้กันอย่างแพร่หลาย เพื่อเพิ่มประสิทธิภาพและความสะดวก ในการทำงานเรียกเทคโนโลยีนี้ว่า กระบวนการประมวลผลภาพ และยังสามารถประยุกต์ใช้ในหลายๆ งานที่เทคนิคอื่นทำได้ยาก โดยเฉพาะในด้านการตรวจสอบผลิตภัณฑ์ ดังนั้นจึงมีการประยุกต์ใช้งานในภาคอุตสาหกรรมกันอย่างหลากหลาย เช่น ขนาดของผลิตภัณฑ์ การใช้กระบวนการประมวลผลภาพในการคัดลอก โมเดลจากผลิตภัณฑ์ที่มีอยู่ การวัดขนาดของวัสดุ ประกอบกับในปัจจุบันโปรแกรมการประมวลผลด้านภาพนั้นสามารถสร้างใช้งานได้กับ Free Ware ด้วยเช่นกัน เช่น โปรแกรม OpenCV และ Python

1.2 วัตถุประสงค์ของปริญญานิพนธ์

1. เพื่อศึกษาการประมวลผลภาพ
2. เพื่อศึกษาการเขียนโปรแกรมบน Raspberry Pi3 Model B
3. เพื่อเพิ่มทักษะการออกแบบให้ประมวลผลได้เร็วขึ้น
4. เพื่อศึกษาการเขียนโปรแกรม Python

1.3 ขอบเขตของโครงการ

1. ขนาดของวัตถุที่นำมาศึกษาไม่เกิน 10 x 10 cm
2. ขนาดของวัตถุต้องสูงไม่เกิน 5 cm
3. ผิวของวัตถุต้องเป็นผิวด้าน
4. วัตถุต้องเป็นรูปทรงเรขาคณิต

1.4 ผลที่คาดว่าจะได้รับ

1. ปริญญาโทสามารถเป็นต้นแบบการพัฒนาต่อยอดการประมวลผลภาพในภาคอุตสาหกรรม
2. ปริญญาโทสามารถเป็นต้นแบบการใช้เทคนิคประมวลผลภาพร่วมกับระบบสายพาน
3. ได้รับความรู้ความเข้าใจเกี่ยวกับการเลือกใช้อุปกรณ์



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 Raspberry Pi 3 Model 1GB

2.1.1 คุณสมบัติบอร์ด Raspberry Pi 3 Model B

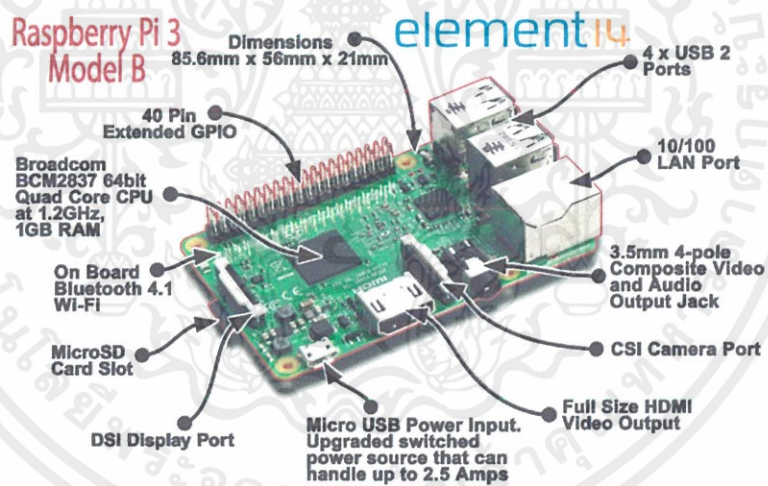
- ใช้ซีพียู Broadcom BCM2837 64-Bit
- หน่วยประมวลผลกลาง (CPU) : Quad-core 1.2 GHz ARM Cortex-A53 แบบ 64 Bits
- หน่วยประมวลผลภาพ (GPU) : Broadcom VideoCore IV @ 400 MHz
- หน่วยความจำ (Memory) ขนาด 1 GB (LPDDR2-900 SDRAM)
- หน่วยความจุแบบ MicroSD
- ขั้วต่อ USB 2.0 จำนวน 4 พอร์ต
- 802.11n Wireless LAN
- Bluetooth 4.1
- รองรับ HDMI/Composite ผ่านทาง RCA Jack
- รองรับสัญญาณเสียงผ่านทางแจ๊ค 3.5 mm
- คอนเน็คเตอร์สำหรับเชื่อมต่ออินพุตเอาต์พุต (GPIO), SPI, I²C, I²S (GPIO ต่างๆ ไม่สามารถรับแรงดันอินพุต 5 VDC ได้สูงสุดแค่ 3.3 VDC)
- ใช้ไฟเลี้ยงบอร์ด 5 VDC กระแสอย่างน้อย 2.5A
- ขั้วต่อ LAN 10/100 Mbps
- ขนาดของบอร์ด 85.0 x 56.0 mm

2.1.2 บอร์ด Raspberry Pi 3 Model B ดังรูปที่ 2.1



รูปที่ 2.1 บอร์ด Raspberry Pi3 Model B

1. คอนเน็คเตอร์ MICRO SD CARD จะติดตั้งอยู่ที่บอร์ด
2. LED แสดงสถานะต่างๆ ของบอร์ด Raspberry Pi ดังรูปที่ 2.2



รูปที่ 2.2 ส่วนประกอบบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนเน็คเตอร์สำหรับเชื่อมต่ออินพุตเอาต์พุต (GPIO), SPI, I²C, I²S และ UART โดยตำแหน่งขาต่างๆ แสดงในรูปที่ 2.3

Pin#	NAME		NAME	Pin#
01	3.3v DC Power	⬢	DC Power 5v	02
03	GPIO02 (SDA1 , I ² C)	⬢	DC Power 5v	04
05	GPIO03 (SCL1 , I ² C)	⬢	Ground	06
07	GPIO04 (GPIO_GCLK)	⬢	(TXD0) GPIO14	08
09	Ground	⬢	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	⬢	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	⬢	Ground	14
15	GPIO22 (GPIO_GEN3)	⬢	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	⬢	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	⬢	Ground	20
21	GPIO 9 (SPI_MISO)	⬢	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	⬢	(SPI_CE0_N) GPIO 08	24
25	Ground	⬢	(SPI_CE1_N) GPIO 07	26
27	ID_SD (I ² C ID EEPROM)	⬢	(I ² C ID EEPROM) ID_SC	28
29	GPIO 5	⬢	Ground	30
31	GPIO 6	⬢	GPIO12	32
33	GPIO13	⬢	Ground	34
35	GPIO19	⬢	GPIO16	36
37	GPIO26	⬢	GPIO20	38
39	Ground	⬢	GPIO21	40

Rev. 2
29/02/2016

www.element14.com/RaspberryPi

รูปที่ 2.3 คอนเน็คเตอร์สำหรับเชื่อมต่ออินพุตเอาต์พุต

2.2 การประมวลผลภาพ

การประมวลผลภาพคือ การนำเสนอรายละเอียดของวัตถุ (Object) โดยการจับเอาหรือถ่ายเท (Manipulation) รายละเอียดของภาพที่ได้จากวัตถุเพื่อนำรายละเอียดของภาพเหล่านั้นไปทำประโยชน์ให้มากยิ่งขึ้น วิธีการทางการประมวลผลภาพจะต่างกับวิธีการทางคอมพิวเตอร์กราฟฟิก กล่าวคือ วิธีการทางคอมพิวเตอร์กราฟฟิก ตัวคอมพิวเตอร์เองที่จะเป็นตัวที่สร้างภาพ แต่เทคนิคทาง Image Processing นั้นใช้คอมพิวเตอร์สำหรับจัดเก็บข้อมูล

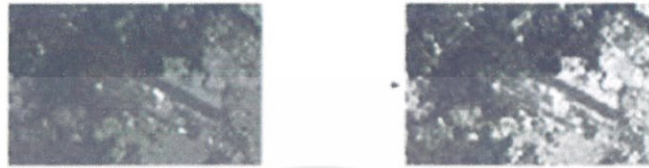
- นำภาพมาทำการประมวลผลด้วยคอมพิวเตอร์ เพื่อให้ภาพที่ได้มีความชัดเจนขึ้น
- ภาพอาจได้มาจากภาพถ่าย เทปวิดีโอ फिल्मจากการแพทย์
- เริ่มต้นจะต้องทำการเปลี่ยนข้อมูลให้อยู่ในรูปของไบนารี (Binary) ให้คอมพิวเตอร์สามารถ

เข้าใจข้อมูลภาพได้

- จากนั้นทำการจัดการกับภาพ เช่น ปรับสี ความคมชัด เป็นต้น

2.2.1 Image Enhancement

คือ การปรับปรุงภาพให้ดีขึ้น โดยทำให้รายละเอียดที่ไม่ชัดเจนให้มีความชัดเจนขึ้น หรือทำให้คุณลักษณะที่สำคัญเด่นขึ้น ดังรูปที่ 2.4



รูปที่ 2.4 การปรับปรุงคุณภาพภาพ

2.2.2 Image Analysis

เริ่มต้น คือ ถ่ายภาพวัตถุที่ต้องการ จากนั้นทำการปรับเพิ่มความถูกต้องให้ภาพ ดังรูปที่ 2.5



รูปที่ 2.5 Image Analysis

2.3 ความรู้พื้นฐานด้านการประมวลผลภาพดิจิทัล

การประมวลผลภาพดิจิทัล จะเกี่ยวข้องกับการแปลงข้อมูลภาพให้อยู่ในรูปแบบข้อมูลดิจิทัล เพื่อที่จะสามารถนำเอาข้อมูลนี้ไปผ่านกระบวนการต่างๆ ด้วยคอมพิวเตอร์ได้ ซึ่งการทำงานของคอมพิวเตอร์ ระบบการรับข้อมูลเข้าหรือส่งข้อมูลออกจะอยู่ในรูปแบบดิจิทัลเท่านั้น ในการประมวลผลภาพดิจิทัล เมื่อระบบได้รับข้อมูลภาพเข้าไปแล้วจะทำการคำนวณและส่งออกมาเป็น

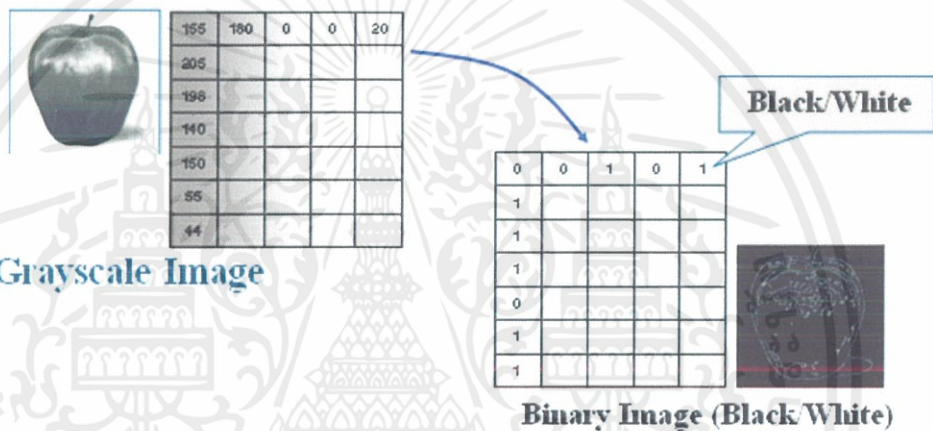
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลที่ใช้แทนข้อมูลภาพดิจิทัลเหล่านั้น การเก็บข้อมูลภาพลงหน่วยความจำของคอมพิวเตอร์สามารถทำได้โดยการจองหน่วยความจำของเครื่องไว้ในรูปของตัวแปรอาร์เรย์ โดยค่าในแต่ละช่องจะแสดงถึงคุณสมบัติของจุดภาพ และตำแหน่งของช่อง Array เป็นตัวกำหนดตำแหน่งของจุดภาพ

2.3.1 ภาพสีเทา

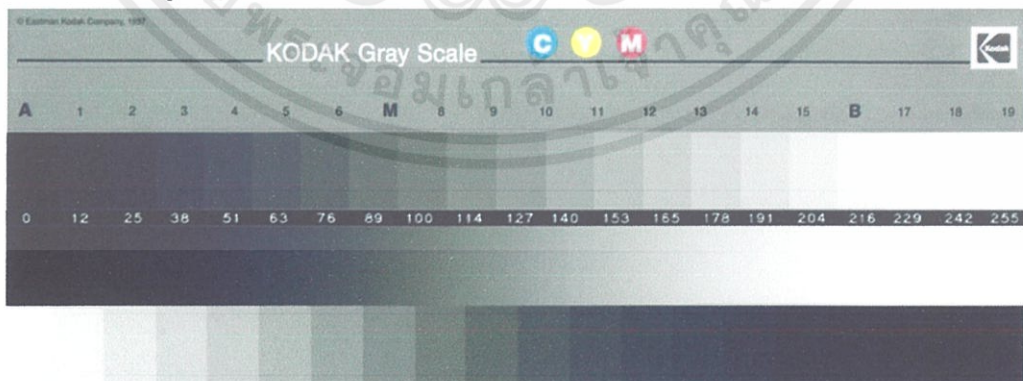
การแปลงภาพเป็นภาพขาวดำมีขั้นตอนคือ แปลงภาพสีเป็นภาพสีเทา แล้วแปลงภาพสีเทา เป็นภาพขาวดำอีกที การแปลงภาพสีเทา เป็นภาพขาวดำ จำเป็นต้องกำหนดค่า Threshold เพื่อให้ได้ภาพขาวดำที่เหมาะสม ค่า Threshold มีไว้สำหรับกำหนดว่าค่าความสว่างของรูปในแต่ละจุดภาพ เมื่อแปลงเป็นภาพขาวดำแล้วจุดภาพ ดังกล่าวควรจะเป็น ดังรูปที่ 2.6

- ♦ Binary can be obtained from Grayscale
- ♦ Binary represents contours, edges, etc.



รูปที่ 2.6 การแปลงภาพเป็นภาพขาวดำ

ระบบสีของภาพสีเทา คือ ระบบที่มีค่าของสีแดง สีเขียว และสีน้ำเงินเท่ากัน ภาพจึงออกมาในโทนสีขาวดำ ดังรูปที่ 2.7



รูปที่ 2.7 โทนสีเทา

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

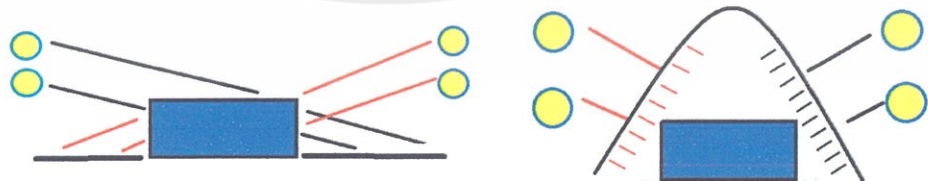
2.4 การจัดแสงแบบไฟลอบเงา (Fill Light)

ผลกระทบของทิศทางและความเข้มของแสงมีผลต่อภาพถ่ายอย่างมาก เพราะจากการทำการทดลองพบว่าเมื่อไม่มีการจัดแสงจะทำให้เกิดเงา และตัวเงานี้เวลาคำนวณมันจะนำไปคิดด้วยซึ่งเป็นค่าที่ผิดพลาด ฉะนั้นควรทำการจัดแสงให้เหมาะสมกับการทดลอง

แสงเป็นส่วนที่สำคัญที่สุดเพราะเป็นต้นกำเนิดที่ทำให้เกิดภาพที่ตาสามารถมองเห็น แสงที่เห็นเป็นสี คุณภาพของแหล่งกำเนิดแสงมีความสำคัญมาก แหล่งกำเนิดแสงที่ดูมีขนาดเล็ก เช่น หลอดไฟหรือดวงอาทิตย์ (เนื่องจากอยู่ไกลจากโลก) เมื่อตกกระทบลงบนผิวของวัตถุจะปรากฏจำนวนชั้นความแตกต่างของความเข้มของแสงมีน้อย กล่าวคือ ที่สว่างก็สว่างมาก ที่มีดกก็มีดกมาก ส่วนบริเวณที่สว่างกลางๆ มีน้อย (High Contrast) ทำให้ภาพของวัตถุดูแข็ง ขาดความนุ่มนวล หากแหล่งกำเนิดแสงมีขนาดกว้างใหญ่ เช่น ช่วงที่พระอาทิตย์อยู่หลังก้อนเมฆ ท้องฟ้าบริเวณนั้นเปรียบเสมือนแผ่นแสงขนาดใหญ่ที่ส่องลงมา หรือช่องหน้าต่างขนาดใหญ่ที่ปูด้วยกระจกฝ้า/พลาสติกขาวขุ่นและมีแสงส่องจากอีกด้านหนึ่ง เมื่อแสงดังกล่าวกระทบลงบนวัตถุจะทำให้ภาพของวัตถุดูนุ่มนวล ถือเป็นแสงที่มีคุณภาพ ในสตูดิโอถ่ายภาพนั้น ไฟแฟลชถือเป็นแหล่งกำเนิดแสงที่ด้อยคุณภาพเช่นเดียวกับแสงจากหลอดไฟ แต่เมื่อหุ้มด้วยกระโจมกระจายแสง (Soft Box) ลักษณะแสงที่ได้จะเป็นแผ่น ทำให้แสงตกกระทบบนวัตถุดูนุ่มขึ้น ดังรูปที่ 2.8 และดังรูปที่ 2.9



รูปที่ 2.8 การจัดแสงในทฤษฎี



รูปที่ 2.9 การจำลองการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 Python กับการประมวลผลภาพ

Python เป็นโปรแกรมขนาดเล็กไม่กินสเปค Raspberry Pi3 Model B เหมาะแก่การนำมาพัฒนาในการประมวลผลภาพ และมี Library ที่มีให้เลือกมากมายในการนำมาประยุกต์ใช้กับโครงการนี้ และยังเป็นโปรแกรมที่ช่วยในการคิดคำนวณทางคณิตศาสตร์ จะนำเอา Python มาใช้กับการประมวลผลภาพรูปภาพ จะต้องเป็นภาพดิจิทัลที่มีการเก็บค่าตัวเลข เมื่ออ่านภาพเข้าไปในโปรแกรมแล้ว โปรแกรมจะเก็บค่าตัวเลขไว้ในรูปแบบของเมตริกซ์ ดังรูปที่ 2.10



รูปที่ 2.10 Logo Python

2.6 ไลบรารีในงานการประมวลผลภาพ

Opencv ใช้ในการประมวลผลภาพทางคอมพิวเตอร์ โดยพัฒนาโปรแกรมร่วมกับภาษา Python ซึ่งจะมุ่งเน้นที่ความเข้าใจง่ายของรูปแบบภาษา เมื่อนำไปใช้งานโค้ดที่เขียนจะกระชับ เข้าใจง่ายกว่าให้ผลลัพธ์ได้ตามที่ต้องการ ดังรูปที่ 2.11



รูปที่ 2.11 Logo Opencv

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมดูล NumPy เป็นโมดูลส่วนเสริมของภาษา Python สำหรับใช้คำนวณทางคณิตศาสตร์และวิทยาศาสตร์โดยมีคำสั่งพร้อมใช้งานจำนวนมากและโมดูลนี้สามารถสร้างข้อมูลชนิดอาร์เรย์ (ที่ภาษา Python ไม่มี) และคำนวณอาร์เรย์ของตัวเลข สตริงและวัตถุได้ ดังรูปที่ 2.12



รูปที่ 2.12 Logo Numpy

SciPy ไลบรารี Python แบบโอเพนซอร์สที่ใช้สำหรับการคำนวณทางวิทยาศาสตร์และการคำนวณทางเทคนิค ดังรูปที่ 2.13

SciPy มีโมดูลสำหรับการเพิ่มประสิทธิภาพพีชคณิตเชิงเส้นการรวมการแก้ไขฟังก์ชันพิเศษ FFT การประมวลผลสัญญาณและภาพการแก้ปัญหา ODE และงานอื่น ๆ ที่พบในวิทยาศาสตร์และวิศวกรรม



รูปที่ 2.13 Logo Scipy

Tkinter เป็นไลบรารีสำหรับสร้างยูสเซอร์อินเตอร์เฟซหรือหน้าต่างของโปรแกรมนั้นเอง ซึ่งเขียนในโปรแกรม Python สามารถนำโคดการประมวลผลภาพมาเชื่อมต่อกันได้เลย ดังรูปที่ 2.14



รูปที่ 2.14 Logo Tkinter

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 การหาค่าฮิสโตแกรมของวัตถุภาพ (Edge Detection Histogram)

เทคนิคการหาขอบของวัตถุภาพเป็นคุณสมบัติหนึ่งที่สามารถเป็นตัวแทนของวัตถุในภาพได้อย่างชัดเจน ด้วยคุณสมบัติของขอบที่ไม่ขึ้นกับการเปลี่ยนตำแหน่ง การหมุน และการเปลี่ยนขนาด เมื่อนำมาสร้างฮิสโตแกรมแล้วสามารถได้ลักษณะเด่น สำหรับการเปรียบเทียบข้อมูลได้อย่างชัดเจน ซึ่งการหาค่าฮิสโตแกรมของขอบวัตถุภาพนี้ ได้นำเอาเทคนิคการหาขอบภาพโดยวิธีการ Canny มาประยุกต์ใช้ โดยการดัดแปลงขั้นตอนจากวิธีการหาขอบของ Canny หลังจากการคำนวณค่า Gradient แล้วจึงคำนวณพิกเซลภาพตามทิศทางค่ามุมของแต่ละกลุ่มทิศทาง ซึ่งสามารถแสดงทิศทางของจุดขอบบนพิกเซลภาพได้ ในงานวิจัยของ Wang และ Zhang ได้ใช้วิธีการหาฮิสโตแกรมของวัตถุภาพ จำนวนทั้งหมด 37 กลุ่มทิศทาง โดยกลุ่มทิศทางที่ 36 แรกมาจากการนับจำนวน ของจุดขอบภาพที่สามารถเกิดทิศทางบนวัตถุภาพ เมื่อทำการควอนไทซ์ ครั้งละ 10 องศา และกลุ่มทิศทางสุดท้ายเป็นการนับจำนวนจุดขอบทั้งหมดที่ไม่มีทิศทางมุม คำนวณได้จากสมการที่ (2.1)

$$H(i) = h(i) / M \quad (2.1)$$

เมื่อ $H(i)$ คือ จำนวนจุดขอบในแต่ละทิศทางที่เกิดขึ้น

M คือ จำนวนพิกเซลทั้งหมดที่เกิดขึ้นบนวัตถุภาพ

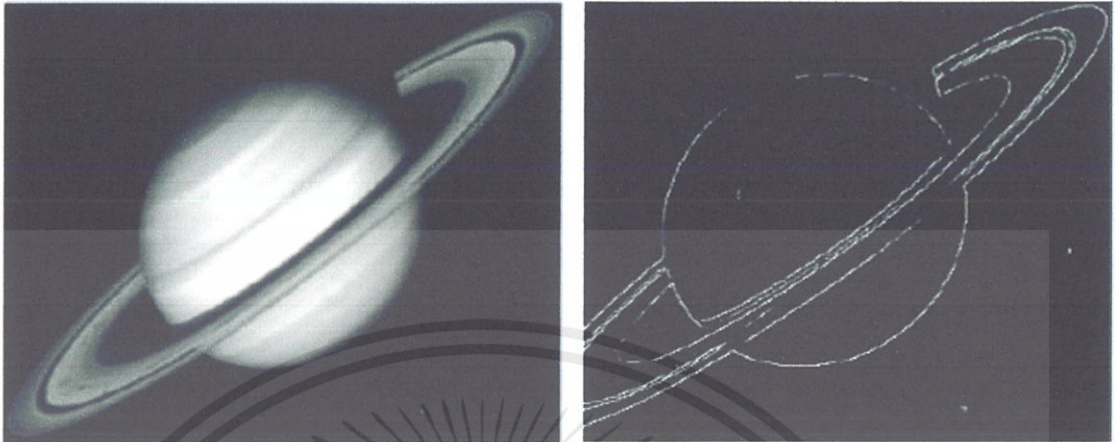
2.8 วิธีการหาขอบภาพ (Edge Detection)

การหาขอบภาพ (Edge Detection) เป็นการหาเส้นรอบวัตถุที่อยู่ในภาพ เมื่อทราบเส้นรอบวัตถุจะสามารถคำนวณหาพื้นที่ (ขนาด) หรือรู้จำนวนชนิดของวัตถุนั้นได้ อย่างไรก็ตามการหาขอบภาพที่ถูกต้องสมบูรณ์ไม่ใช่เป็นเรื่องที่ง่าย โดยเฉพาะอย่างยิ่งการหาขอบของภาพที่มีคุณภาพต่ำ มีความแตกต่างระหว่างพื้นหน้ากับพื้นหลังน้อย หรือมีความสว่างไม่สม่ำเสมอทั่วภาพขอบภาพเกิดจากความแตกต่างของความเข้มแสงจากจุดหนึ่งไปยังอีกจุดหนึ่ง หากต่างนี้มีค่ามากขอบภาพก็จะเห็นได้ชัด ถ้าความแตกต่างมีค่าน้อย ขอบภาพก็จะไม่ชัดเจนสามารถแบ่งได้เป็น 2 กลุ่มหลักคือ วิธี Gradient และวิธี Laplacian โดยในแต่ละวิธีมีรายละเอียดดังต่อไปนี้

2.8.1 วิธี Gradient

เป็นความคิดแรกที่ใช้วิเคราะห์หาขอบภาพ โดยใช้วิธีการวัดความเป็นแปลงความเข้ม (Gray Level) ของจุดภาพที่กำลังพิจารณากับจุดภาพที่อยู่ข้างเคียง เพื่อการตัดสินใจต่อไปว่าเป็นขอบภาพหรือไม่ โดยวิธีนี้จะหาขอบด้วยการหาจุดต่ำสุดและจุดสูงสุดในรูปของอนุพันธ์อันดับหนึ่งของภาพ โดย

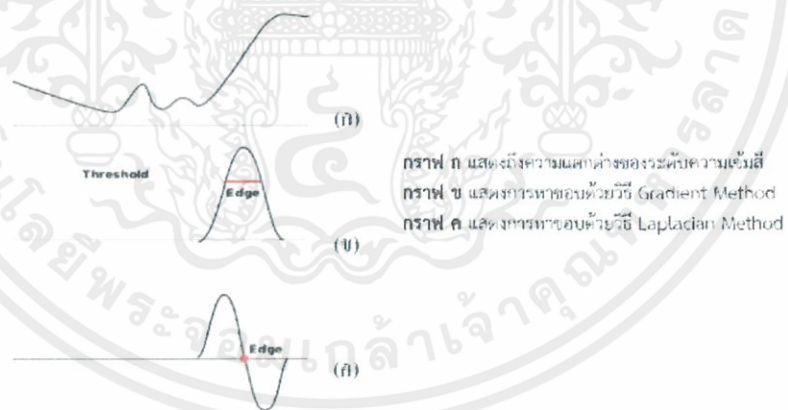
จุดที่เป็นขอบจะอยู่ในส่วนที่เหนือค่า Threshold จึงอาจทำให้เส้นขอบที่ได้มีลักษณะหนา ตัวอย่างวิธีการหาขอบของกลุ่มนี้ เช่น Roberts, Prewitt และ Canny เป็นต้น ดังรูปที่ 2.15



รูปที่ 2.15 ตัวอย่างการหาขอบด้วยวิธี Gradient

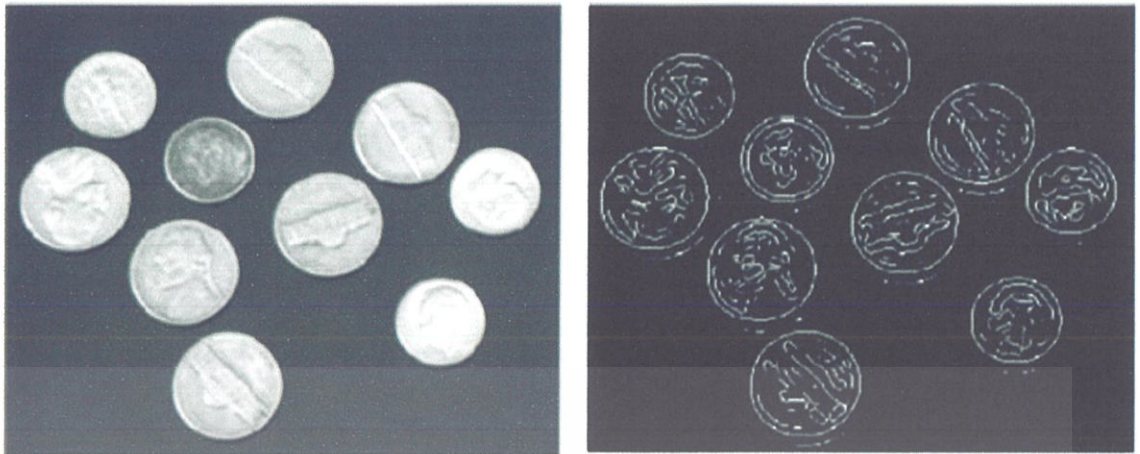
2.8.2 วิธี Laplacian

วิธีนี้จะหาขอบโดยใช้อนุพันธ์อันดับ 2 โดยใช้จุดที่ค่า เป็น "0" ซึ่งวิธีนี้จะใช้เวลาในการคำนวณมากกว่าวิธี Gradient สามารถใช้กรองความถี่ต่ำได้ขอบของโครงภาพ ตัวอย่างวิธีการหาขอบของกลุ่มนี้ เช่น Laplacian of Gaussian และ Marrs-Hildreth เป็นต้น ดังรูปที่ 2.16 และรูปที่ 2.17



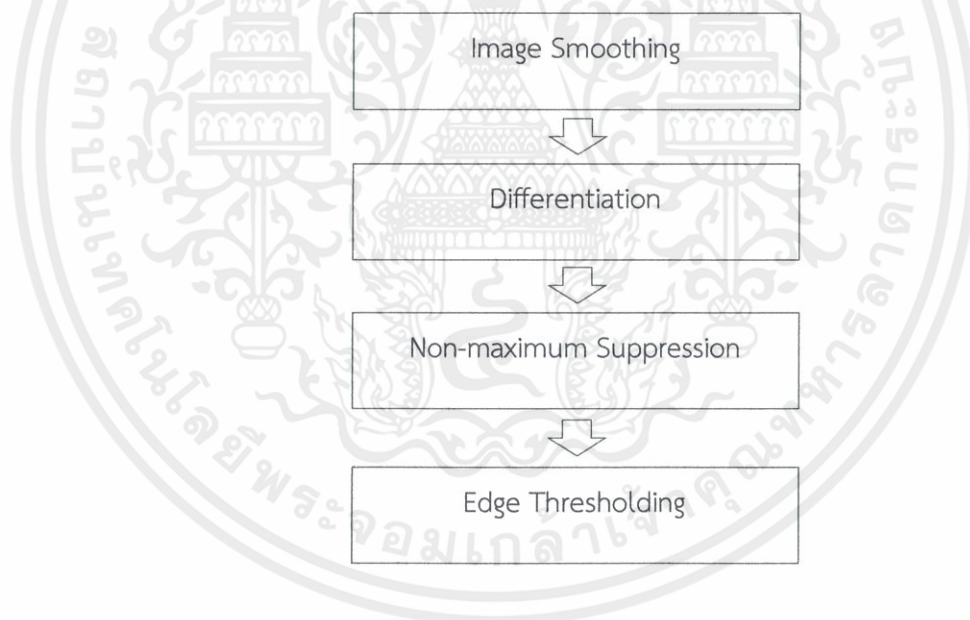
รูปที่ 2.16 การกำหนดค่า Threshold

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 การหาขอบด้วยวิธี Laplacian

ในปริศยานี้เลือกใช่วิธีหาขอบด้วยวิธี Canny ซึ่งเป็นวิธีการหาขอบที่สามารถเลือกทิศทางการเกิดขอบภาพได้ และได้ผลดีในการหาขอบในภาพซึ่งอยู่ในประเภทวิธีการหาขอบแบบ Gradient ขั้นตอนวิธีการหาขอบโดยวิธีของ Canny ดังรูปที่ 2.18



รูปที่ 2.18 ขั้นตอนวิธีการหาขอบโดยวิธี Canny

การทำงานของ Canny Edge Detection นั้นเริ่มต้นจากการปรับภาพให้เรียบ (Smoothing) ด้วยตัวกรองเกาส์เซียน (Gaussian Filter) เพื่อกำจัดสัญญาณรบกวน หลังจากนั้นคำนวณค่าขนาด (Magnitude) และทิศทาง (Orientation) ของ Gradient โดยการใช้การหาอนุพันธ์อันดับหนึ่ง ในถัดมา

จึงใช้ Nonmaxima Suppression กับ Gradient Magnitude เพื่อให้ได้ขอบที่บางลง และในขั้นตอนสุดท้ายใช้ Double Thresholding Algorithm เพื่อระบุพิกเซลที่เป็นขอบและช่วยเชื่อมต่อขอบ (Green, Bill 2002 ; ION528 - Image Processing Algorithms 2005 ; Canny Operator Links 2005 ; Rubino, Matthew 2005) โดยในแต่ละขั้นตอนมีรายละเอียดดังต่อไปนี้

2.8.3 Image Smoothing

ในขั้นตอนแรกของการหาขอบโดยอัลกอริทึมนี้จะต้องกำจัดสัญญาณรบกวนออกก่อนโดยใช้ Gaussian Filter ซึ่งสามารถคำนวณได้จากการใช้กรอบ (Mask) ขนาดเล็ก ขนาดของ Gaussian Mask นี้หากมีขนาดกว้างจะมีผลทำให้ลดสัญญาณรบกวนได้มาก แต่ถ้ากว้างมากเกินไปจะมีผลทำให้ขอบย่อยๆ ที่เป็นส่วนรายละเอียดนั้นหายไป สำหรับการคำนวณหาภาพที่ได้จากการใช้ Gaussian Filter

2.8.4 Differentiation

ในขั้นแรกนำ Smoothing Image $S[i, j]$ มาสร้าง x, y Partial Derivatives $P[i, j]$ และ $Q[i, j]$ ตามลำดับ

$$P[i, j] \approx (S[i, j+1] - S[i, j] + S[i+1, j+1] - S[i+1, j]) / 2$$

$$Q[i, j] \approx (S[i, j] - S[i+1, j] + S[i, j+1] - S[i+1, j+1]) / 2$$

หลังจากนั้นนำค่า x, y Partial Derivatives มาคำนวณด้วยสูตรมาตรฐานสำหรับการแปลงรูปแบบจาก Rectangular ไปเป็น Polar (Rectangular-to-polar Conversion) เพื่อหาขนาดและทิศทางของ Gradient

2.8.5 Nonmaxima Suppression

สำหรับการหาขอบโดย Canny Method จุดที่ถือเป็นเส้นขอบได้นั้นต้องเป็นจุดที่ให้ค่าสูงสุดเฉพาะที่และเป็นทิศทางเดียวกับ Gradient ด้วย ซึ่งด้วยวิธีดังกล่าวนี้ทำให้ได้ขอบที่บางเพียง 1 พิกเซล ภาพที่ได้หลังการทำ Nonmaxima Suppression จะให้ค่าเป็นศูนย์ในทุกจุดยกเว้นจุดที่เป็น Local Maxima Points ซึ่งจะยังคงค่าเดิมไว้

2.8.6 Thresholding

แม้ว่าภาพจะผ่านการ Smoothing ในขั้นตอนแรกแล้วก็ตาม ภาพที่ได้อาจยังมีเส้นขอบที่ไม่ใช่ขอบที่แท้จริงปรากฏอยู่ อันเนื่องจากสัญญาณรบกวนหรือลักษณะของวัตถุในภาพเป็นพื้นผิวที่มีสลาดหลายหรือมีรายละเอียดภายในมาก ดังนั้นเพื่อลดปัญหาดังกล่าวจึงได้มีการกำหนดค่า Threshold

บทที่ 3

หลักการออกแบบและโครงสร้างของวงจรควบคุม

3.1 ภาพรวมของระบบ

ในการเริ่มต้นนั้นจะเริ่มจากการออกแบบโครงสร้างของชิ้นงานหรือในส่วนของฮาร์ดแวร์ก่อน โดยเริ่มจากฐานซึ่งเป็นโครงในการใส่สายพานและทำโครงกล่องสำหรับถ่ายภาพวัตถุในการวัดขนาด ซึ่งมีลักษณะมิดชิดกันสิ่งรบกวนจากภายนอก เช่น แสง ฝุ่น เป็นต้น จึงมีการทำให้กล่องดังกล่าวเป็นเหมือนสตูดิโอขนาดเล็กเพื่อลบเงาของวัตถุ ทำให้ได้ความละเอียดที่มากขึ้น โดยการทำให้กล่องนั้นต้องคำนวณระยะเวลาการทำงานของกล่องก่อน เพื่อให้เมื่อติดตั้งแล้วได้คุณภาพการทำงานอย่างสูงสุด และเมื่อประกอบชิ้นส่วนทั้งสองเข้ากันแล้วก็หาจุดศูนย์กลางของกล่องเพื่อติดตั้งกล่อง ต่อมาเป็นการทำงานในส่วนของซอฟต์แวร์ ในส่วนนี้จะเป็นการในด้านการประมวลผลของภาพเพื่อหาขนาดของวัตถุ แล้วนำไปเปรียบเทียบกับชิ้นงานต้นแบบเปรียบเสมือนทำหน้าที่เป็นผู้ตรวจชิ้นงาน โดยหากของที่ผลิตออกมามีความคลาดเคลื่อนหรือผิดไปจากต้นแบบจะมีการนำออกจากสายการผลิต โดยชุดประมวลผลนั้นได้ใช้เป็นบอร์ด Raspberry Pi ใช้โปรแกรม Python ในการเขียนและควบคุมกับไลบรารี OpenCV ในการเขียนโค้ด โดย Raspberry Pi จะมีหน้าที่ในการควบคุมการหยุดของสายพาน การทำงานของกล่องเพื่อวัดขนาดและเซอร์โวในการนำของออกจากสายการผลิต โดยการทำงานทั้งหมดจะมีการแสดงค่าข้อมูลต่างๆ ไปที่จอแสดงผลเพื่อให้ง่ายต่อการควบคุมและการใช้งาน

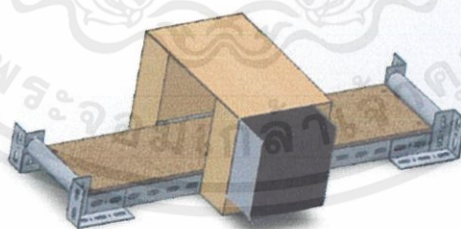
3.2 ส่วนขยายของฮาร์ดแวร์

ลักษณะการทำงานของเครื่องวัดขนาดวัตถุ เป็นการทำงานแบบอัตโนมัติ โดยในการทำงานผู้ใช้มีหน้าที่สั่งเริ่มต้นการทำงานหรือจบการทำงานของระบบ ในส่วนของการทำงานจะมีการทำงานโดยมีสายพานลำเลียงวัตถุ โดยจะผ่านการทำงานต่างๆ เริ่มจากวัดขนาดที่วางตัวในลักษณะที่คร่อมกับสายการผลิต ในส่วนนี้เป็นการป้อนข้อมูลเข้าไปในระบบผ่านกล่องเพื่อทำการเอาภาพที่ถ่ายได้ไปวัดขนาด โดยจะมีเซนเซอร์ในการตรวจจับตำแหน่งของวัตถุ หากวัตถุอยู่ในตำแหน่งที่เหมาะสมจะทำการหยุดสายพานเพื่อถ่ายภาพแล้วนำไปประมวลผล ถ้าหากวัตถุที่นำไปหาขนาดมีขนาดอยู่ในช่วงที่ยอมรับได้ก็จะถูกลำเลียงต่อไปบนสายพาน แต่หากวัตถุมีขนาดที่ผิดเพี้ยนไปจากค่าที่ยอมรับจะมีการนำออกจากสายการผลิตด้วยเซอร์โวที่วางตัวอยู่ข้างกับสายการผลิต โดยจะต่อกับไม้กั้นเพื่อนำของออกโดยระยะห่างของวัตถุ

แต่ละชั้นที่อยู่บนสายการผลิตจะห่างกันประมาณ 20 cm ซึ่งเมื่อวัตถุชิ้นใหม่เข้ากระบวนการถ่ายภาพ สายพานจะหยุด และวัตถุที่วัดขนาดแล้วจะหยุดตรงตำแหน่งที่อยู่หน้าเซอร์โวพอด ซึ่งเซอร์โวจะทำการนำของที่ไม่ได้มาตรฐานออกจากระบบ โดยการทำงานจะซ้ำกันในลักษณะนี้ไปเรื่อยๆ โดยสามารถใช้งานได้จริง, ใช้งานง่าย, และเมื่อหากเกิดปัญหาก็สามารถแก้ไขได้โดยง่าย เนื่องจากไม่มีความซับซ้อนมากนักและสามารถซ่อมได้ในราคาที่เหมาะสม เนื่องจากวัสดุที่ได้นำมาใช้เป็นวัสดุที่เรียบง่ายและหาได้ทั่วไปอีกทั้งยังสามารถนำไปพัฒนาในการใช้งานได้เองตามความเหมาะสมของงานนั้นๆ อีกด้วย

3.2.1 การออกแบบโครงสร้าง

ในการออกแบบโครงสร้างเนื่องจากอยากให้การทำงานนี้ทำงานควบคู่กับสายการผลิต จึงได้ออกแบบในลักษณะเป็นสายพานลำเลียง เพื่อที่จะทำงานควบคู่ไปกับการผลิตเพื่อลดทรัพยากรต่างๆ ได้ โครงสร้างดังกล่าวจึงใช้เหล็กฉากซึ่งเป็นวัสดุที่มีความแข็งแรง, เบา และสามารถหาทั่วไปได้จึงทำให้ราคาไม่แพงจนเกินไปอีก ทั้งยังสามารถซบภาระจากการทำงานของสายพานได้เป็นอย่างดี โดยจากการออกแบบตัวฐานได้มีขนาดความกว้าง 20 cm. ความยาว 90 cm. ความสูง 4 cm. (ขนาดดังกล่าวเป็นขนาดของตัวฐานเท่านั้นไม่รวมขนาดชุดลูกกลิ้งสายพานและชุดการถ่ายภาพ) โดยการออกแบบนี้ได้คำนวณจากคุณภาพของรูปที่กล้องสามารถถ่ายได้ และให้ความแม่นยำสูงสุด ในส่วนของลูกกลิ้งมีขนาดความยาว 23 cm. เส้นผ่าศูนย์กลาง 3 cm. และในส่วนของการถ่ายภาพได้ออกแบบให้มีลักษณะเป็นกล่องทึบ เพื่อไม่ให้มีปัจจัยภายนอกเข้ามารบกวนการทำงานของกล้องได้ออกแบบให้มีขนาดความกว้าง 20 cm. ความยาว 28 cm. ความสูง 23 cm. โดยมีการติดแผ่นไม้ในภายหลังเพื่อให้กลายเป็นกล่องทึบแสงทำให้มีขนาดความกว้าง 20 cm. ความยาว 31 cm. ความสูง 24 cm. ดังรูปที่ 3.1



รูปที่ 3.1 การออกแบบโครงสร้าง

3.2.2 อุปกรณ์ที่ใช้ในการทดลอง

- Webcam Logitech C170

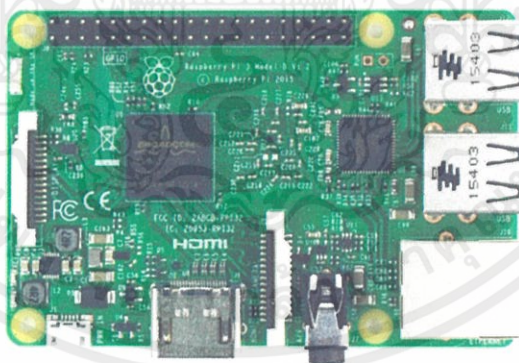
กล้องที่เลือกนั้นมีความละเอียดที่เพียงพอต่อการทำงานนั้นคือ ขนาด 480 x 640 Pixels เพราะวัตถุที่นำมาทำการทดสอบนั้นไม่ได้มีความละเอียดมาก ดังรูปที่ 3.2



รูปที่ 3.2 กล้อง Logitech C170

- Raspberry Pi 3 Model 1GB

การเลือกใช้หน่วยประมวลผลจะต้องสอดคล้องกับการทดลอง การเลือก Raspberry Pi มาใช้นั้นมีความเหมาะสมทุกประการในเรื่องความเร็วในการประมวลผล อายุการใช้งาน และราคา ดังรูปที่ 3.3

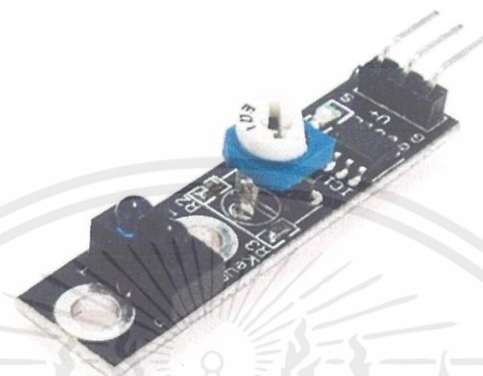


รูปที่ 3.3 Raspberry Pi 3 Model 1GB

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Senor แสง KY-033

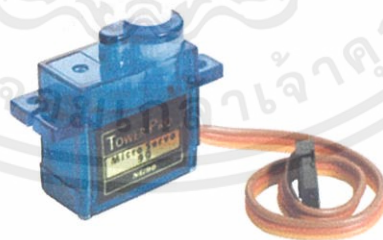
เลือกเซนเซอร์ตัวนี้เพราะราคาถูกและเหมาะกับงานที่ใช้ เพราะระยะที่ใช้ในการตรวจจับมีระยะใกล้ ซึ่งไม่เกินสเปคของเซนเซอร์ตัวนี้ ดังรูปที่ 3.4



รูปที่ 3.4 Senor แสง KY-033

- Servo SG 90

เลือกเซอร์โวลิตัวนี้เพราะวัสดุที่ใช้ในการทดลองมีขนาดและน้ำหนักที่เซอร์โวลิตัวนี้นั้นสามารถรับได้ ดังรูปที่ 3.5



รูปที่ 3.5 Servo SG 90

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 ส่วนประกอบของชิ้นงาน

- แผ่นไม้รองสายพาน

แผ่นไม้เนื้อดีเพื่อไม่ให้น้ำหนักของสายพาน ทำให้สายพานหย่อนจนเกิดการโค้งงอของตัวสายพานเอง โดยสาเหตุที่ใช้ไม้อัดเนื่องจากมีราคาถูกหาซื้อได้ง่าย และสามารถนำมาใช้งานได้โดยง่ายโดยมีขนาดเท่ากับ 20 x 85 x 0.5 cm ดังรูปที่ 3.6



รูปที่ 3.6 แผ่นไม้รองสายพาน

- ลูกกลิ้ง

ลูกกลิ้งที่นำมาใช้เป็นลูกกลิ้งที่เอาไว้สำหรับใช้กับสายพานโดยเฉพาะ โดยเมื่อนำมาติดตั้งจะต่างกับลูกกลิ้งที่ทำเองตรงเซนเซอร์การหมุนจะได้ตำแหน่งมากกว่าทำให้ใช้งานง่าย และสะดวกในการติดตั้งโดยมีขนาดหน้ากว้าง 23 cm เส้นผ่าศูนย์กลาง 3 cm ดังรูปที่ 3.7

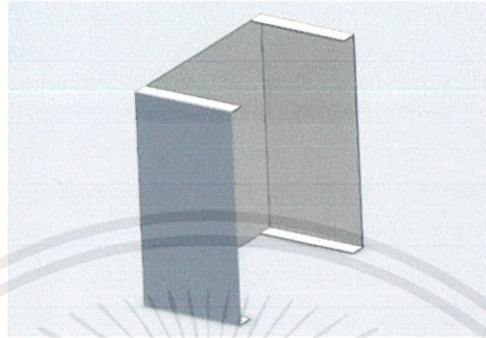


รูปที่ 3.7 ลูกกลิ้ง

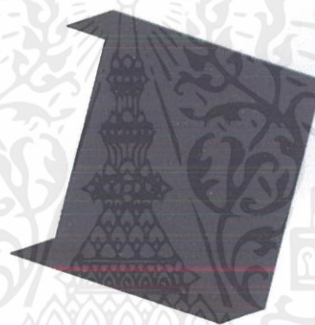
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ตูไฟ

ตูไฟที่นำมาใช้เป็นตู้เนกประสงค์โดยเมื่อประกอบแล้วจะมีขนาด $9 \times 21 \times 25$ cm ภายในบรรจุอุปกรณ์ที่ใช้ในการควบคุมต่างๆ ดังรูปที่ 3.8 และรูปที่ 3.9



รูปที่ 3.8 ตู้ไฟ (1)



รูปที่ 3.9 ตู้ไฟ (2)

- เหล็กฉาก

เหล็กฉากเป็นเหล็กที่สามารถนำมาตัดหรือประกอบอะไรได้ง่าย เนื่องจากมีขนาดบอกรายละเอียดชัดเจน ดังรูปที่ 3.10



รูปที่ 3.10 เหล็กฉาก

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนขยายของซอฟต์แวร์ในการวิเคราะห์ภาพ

หลักการของการทำงานของโปรแกรมดังนี้คือ

3.3.1 การถ่ายภาพวัตถุ

การถ่ายรูปรูปร่างตามที่ต้องการทราบขนาดโดยใช้กล้องที่เตรียมไว้กับโปรแกรมที่ได้เขียนโค้ด เพื่อนำไปใช้ในกระบวนการต่อไป

3.3.2 การทำให้ภาพเป็นสีเทา

การนำภาพที่ได้จากกล้องที่เป็นภาพสีต่างๆ ทำให้เป็นสีเทาขาวเพื่อที่จะนำไปหาขอบของรูปภาพต่อไป

3.3.3 การปรับปรุงภาพ

การทำภาพให้ดูซอฟหรือนุ่มนวลขึ้น มากขึ้นในที่นี้เลือกวิธีการ GaussianBlur

3.3.4 การตรวจหาขอบของวัตถุ

เป็นการหาเส้นรอบวัตถุที่อยู่ในภาพ เมื่อทราบเส้นรอบวัตถุ จะสามารถคำนวณหาพื้นที่ (ขนาด) หรือรู้ชนิดของวัตถุนั้นได้ อย่างไรก็ตามการหาขอบภาพที่ถูกต้องสมบูรณ์ต้องมีความแตกต่างระหว่างพื้นผิววัตถุกับพื้นหลังมากๆ

3.3.5 การหาระยะห่างจุดกึ่งกลางขอบทั้งสี่ด้านเพื่อที่จะนำมาคำนวณหาขนาดของวัตถุ

3.4 ลำดับขั้นตอนการทำงานโดยรวมของโครงการ ดังรูปที่ 3.11



รูปที่ 3.11 ลำดับขั้นตอนการทำงานของโครงการ

3.4.1 ขั้นตอนการเตรียมสายพาน

สายพานที่นำมาทำจะต้องเซตให้ตรงกันทั้งสองฝั่งเพื่อไม่ให้สายพานเกิดการเคลื่อนออกจากลูกกลิ้ง มิฉะนั้นจะทำให้สายพานหลุดออกจากลูกกลิ้งทำให้ระบบหยุดการทำงาน

3.4.2 ขั้นตอนการเตรียมไฟ

เนื่องจากได้ทำการเซตไฟที่จุดกึ่งกลางภาพพอดี วัตถุที่จะนำมาวัดขนาดนั้นจะต้องอยู่บนจุดที่มาร์คไว้เพื่อให้เกิดเงาสะท้อนน้อยที่สุด

3.4.3 ขั้นตอนการตั้งกล้อง

เนื่องจากกล้องที่นำมาใช้นั้นไม่ได้มีความละเอียดมากเพียงพอ จึงต้องตั้งกล้องให้ใกล้วัตถุเพื่อจะได้วัดได้มีความใกล้เคียงของจริงมากที่สุด

3.4.4 ขั้นตอนการทำโปรแกรม

การทำโปรแกรมนั้นต้องนำทุกๆ ขั้นตอนมารวมกันเพื่อแก้ไขปัญหาของระบบ ที่อาจจะเกิดขึ้นได้ไม่ว่าจะเป็นการหาขนาดที่คลาดเคลื่อนหรือไฟที่มากไปทำให้ไม่สามารถถ่ายรูปได้



บทที่ 4

การทดลองและผลการทดลอง

การทดลองได้ทำการทดลองผ่าน Raspberry Pi โดยการเขียนลงในโปรแกรม Python และใช้ภาษา Python รวมกับการประยุกต์ใช้ไลบรารี OpenCV เพื่อทำการตรวจสอบขนาดวัตถุ โดยดูที่พื้นที่และรูปทรง

4.1 ผลที่ได้จากการวิเคราะห์ภาพจากโปรแกรม Python

ผลที่ได้จากการวิเคราะห์ภาพจากโปรแกรม Python จะเรียงลำดับขั้นตอนการประมวลผลของโปรแกรมดังนี้

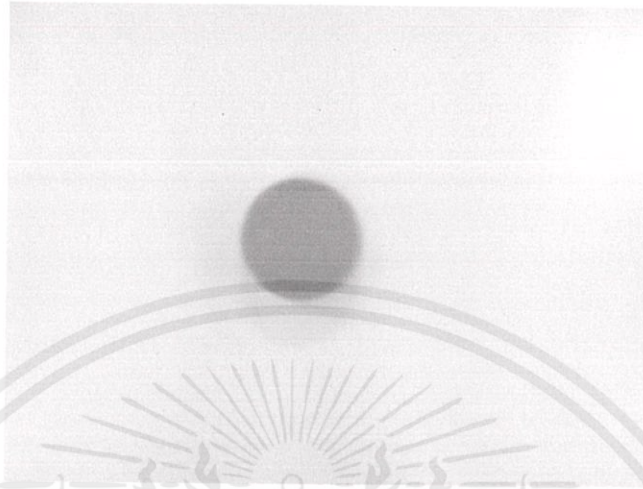
4.1.1 ผลของฝาชวดน้ำ

ขั้นตอนที่ 1 : ทำการจับภาพของวัตถุและทำให้ภาพสีเป็นภาพสีเทา เนื่องจากการปรับภาพและการหาขนาดภาพโปรแกรมยึดค่าสีเทาเป็นหลัก ดังรูปที่ 4.1



รูปที่ 4.1 ภาพสีเทาของฝาชวดน้ำ

ขั้นตอนที่ 2 : ทำการปรับภาพแบบ GaussianBlur เป็นการปรับปรุงภาพให้ดีขึ้น โดยปรับปรุงจากตรงกลางภาพก่อนแล้วไปจนถึงขอบภาพ สังเกตได้จากขอบของภาพเป็นโทนสีเดียวกันดังรูปที่ 4.2



รูปที่ 4.2 ภาพ GaussianBlur ฝาชวดน้ำ

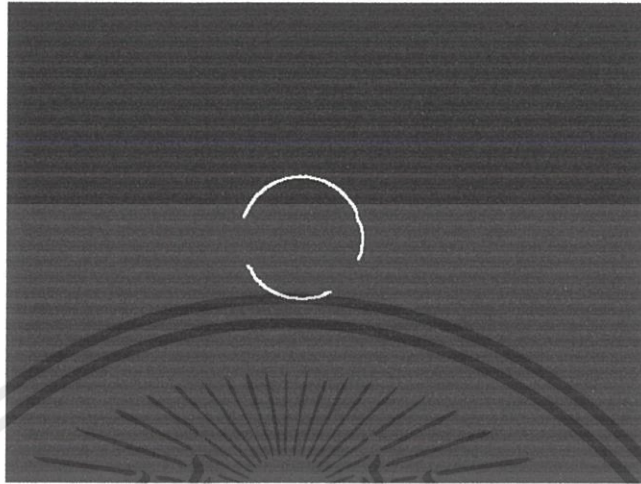
ขั้นตอนที่ 3 : ใช้วิธี Canny เพราะช่วยกำจัดสัญญาณรบกวนได้ด้วย และโปรแกรมจะทำการเก็บค่าของจุดกึ่งกลางของวัตถุเพื่อทำการคำนวณในขั้นตอนต่อไป ดังรูปที่ 4.3



รูปที่ 4.3 ภาพ Canny ฝาชวดน้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 : ใช้วิธี Dialate เพื่อเป็นการขยายภาพให้ใหญ่ขึ้น เพื่อเพิ่มสีให้กับวัตถุที่แสดงผล
ดังรูปที่ 4.4



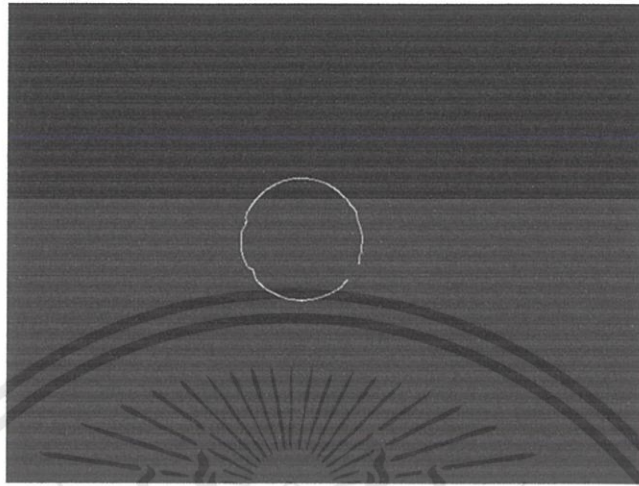
รูปที่ 4.4 ภาพ Dialate ฝาขวดน้ำ

ขั้นตอนที่ 5 : ใช้วิธี Erode เพื่อเป็นการกร่อนขนาดบริเวณขอบของวัตถุ ซึ่งการกร่อนมีวิธีคล้ายกับการขยายคือ สร้างส่วนประกอบโครงสร้างขึ้นมาแล้วนำไปกราดตามข้อมูลภาพ ดังรูปที่ 4.5



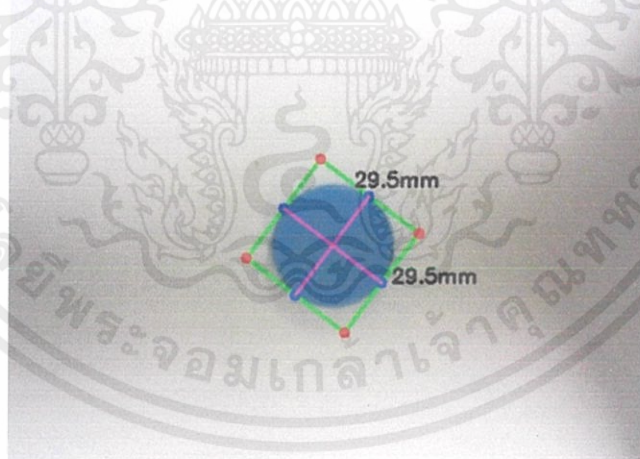
รูปที่ 4.5 ภาพ Erode ฝาขวดน้ำ

ขั้นตอนที่ 6 : ใช้วิธี Contour, Imutile เพื่อเป็นการหามุมต่างๆ ของภาพและจุดกึ่งกลางของภาพ เพื่อตรวจสอบความถูกต้องในการเก็บค่าจุดกึ่งกลางของวิธี Candy ดังรูปที่ 4.6



รูปที่ 4.6 ภาพ Contour และ Imutile ฝาชวดน้ำ

ขั้นตอนที่ 7 : การหาระยะและแสดงผล หลังจากผ่านขั้นตอนที่ 1-6 จะนำการคำนวณที่ได้มาแสดง ในภาพแรกที่ได้จากการถ่ายของกล้อง โดยที่ยังไม่ได้ผ่านขั้นตอนการปรับภาพเป็นสีเทา ดังรูปที่ 4.7



รูปที่ 4.7 ภาพขนาดฝาชวดน้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 ขนาดของฝาขวดน้ำ

การตรวจวัด	ความกว้าง	ความยาว	การตรวจวัด	ความกว้าง	ความยาว
1	29.5	29.5	6	29.6	29.6
2	29.6	29.6	7	29.5	29.5
3	29.5	29.5	8	29.6	29.6
4	29.4	29.4	9	29.7	29.7
5	29.6	29.6	10	29.5	29.5

ขนาดมาตรฐานของฝาขวดคือ 30 x 30 มิลลิเมตร เปอร์เซ็นต์ความผิดพลาดทั้งด้านกว้างและด้านยาวคือ 1.5 เปอร์เซ็นต์

4.1.2 ผลของบัตรเครดิต

ขั้นตอนที่ 1 : ทำการจับภาพของวัตถุและทำให้ภาพสีเป็นภาพสีเทา เนื่องจากการปรับภาพและการหาขนาดภาพโปรแกรมยึดค่าสีเทาเป็นหลัก ดังรูปที่ 4.8



รูปที่ 4.8 ภาพสีเทาของบัตรเครดิต

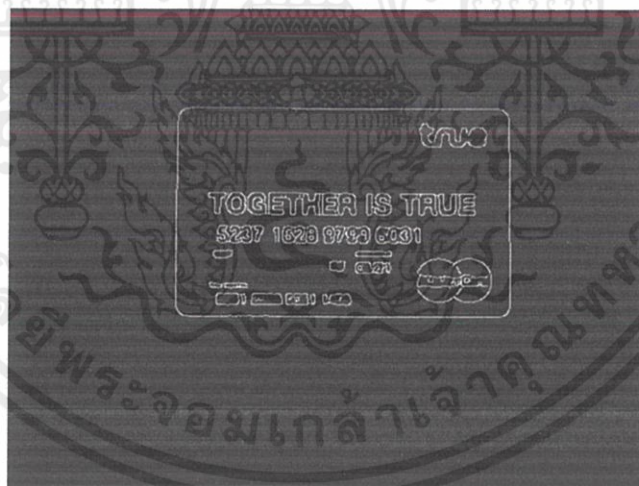
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 : ทำการปรับภาพแบบ GaussianBlur เป็นการปรับปรุงภาพให้ดีขึ้น โดยปรับปรุงจากตรงกลางภาพก่อนแล้วไปจนถึงขอบภาพ สังเกตได้จากขอบของภาพเป็นโทนสีเดียวกัน ดังรูปที่ 4.9



รูปที่ 4.9 ภาพ GaussianBlur บัตรเครดิต

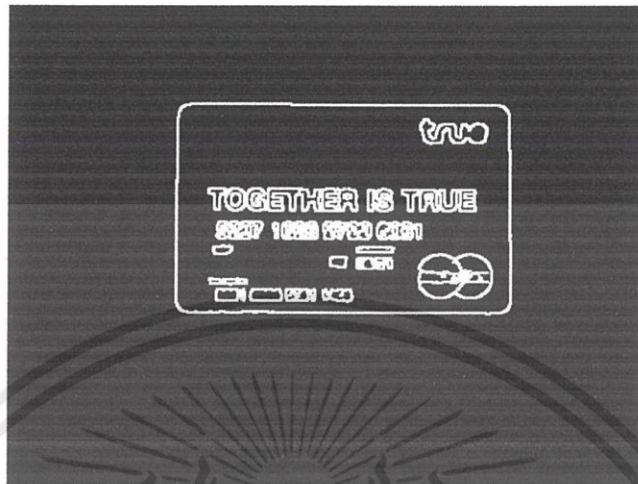
ขั้นตอนที่ 3 : ใช้วิธี Canny เพราะช่วยกำจัดสัญญาณรบกวนได้ด้วย และโปรแกรมจะทำการเก็บค่าของจุดกึ่งกลางของวัตถุเพื่อทำการคำนวณในขั้นตอนต่อไป ดังรูปที่ 4.10



รูปที่ 4.10 ภาพ Canny บัตรเครดิต

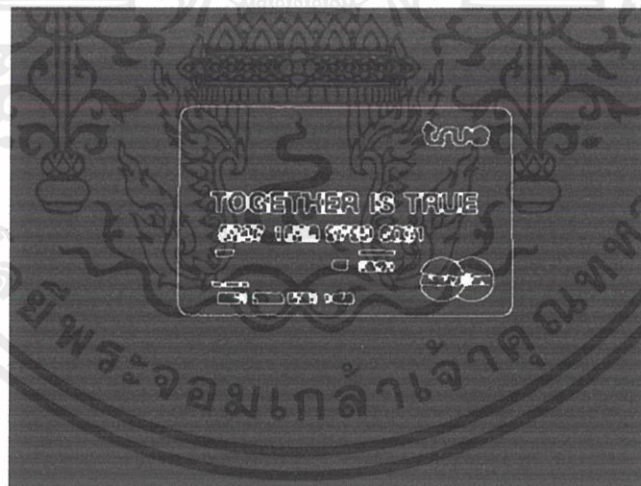
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 : ใช้วิธี Dialate เพื่อเป็นการขยายภาพให้ใหญ่ขึ้น เพื่อเพิ่มสีให้กับวัตถุที่แสดงผล
ดังรูปที่ 4.11



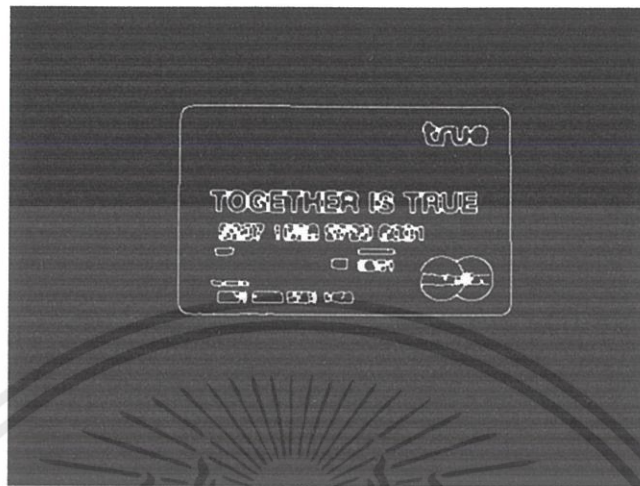
รูปที่ 4.11 ภาพ Dialate บัตรเครดิต

ขั้นตอนที่ 5 : ใช้วิธี Erode เพื่อเป็นการกร่อนขนาดบริเวณขอบของวัตถุ ซึ่งการกร่อนมีวิธีคล้ายกับการขยายคือ สร้างส่วนประกอบโครงสร้างขึ้นมาแล้วนำไปกราดตามข้อมูลภาพ ดังรูปที่ 4.12



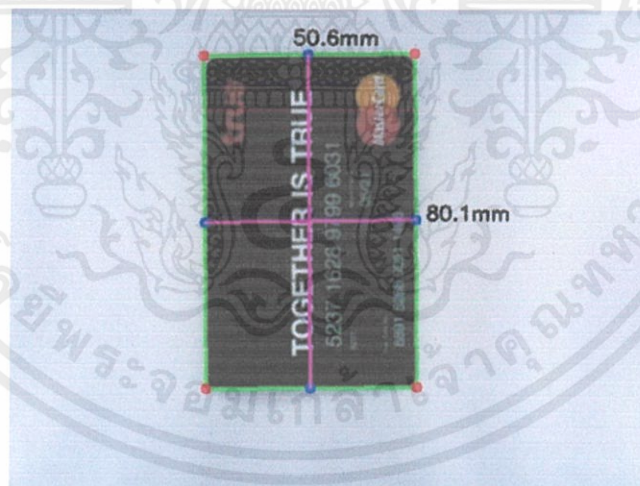
รูปที่ 4.12 ภาพ Erode บัตรเครดิต

ขั้นตอนที่ 6 : ใช้วิธี Contour, Imutile เพื่อเป็นการหามุมต่างๆ ของภาพและจุดกึ่งกลางของภาพ เพื่อตรวจสอบความถูกต้องในการเก็บค่าจุดกึ่งกลางของวิธี Candy ดังรูปที่ 4.13



รูปที่ 4.13 ภาพ Contour และ Imutile บัตรเครดิต

ขั้นตอนที่ 7 : การหาระยะและแสดงผล หลังจากผ่านขั้นตอนที่ 1-6 จะนำการคำนวณที่ได้มาแสดง ในภาพแรกที่ได้จากการถ่ายของกล้อง โดยที่ยังไม่ได้ผ่านขั้นตอนการปรับภาพเป็นสี่เหลี่ยม ดังรูปที่ 4.14



รูปที่ 4.14 ภาพขนาดบัตรเครดิต

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 ขนาดของบัตรเครดิต

การตรวจวัด	ความกว้าง	ความยาว	การตรวจวัด	ความกว้าง	ความยาว
1	50.6	80.1	6	50.9	80.2
2	51.0	80.0	7	50.8	80.1
3	50.8	79.9	8	51.0	80.1
4	50.7	80.0	9	50.6	80.2
5	51.0	80.0	10	50.2	80.0

ขนาดมาตรฐานของบัตรเครดิตคือ 54 x 85 มิลลิเมตร เปอร์เซ็นต์ความผิดพลาดด้านกว้างคือ 6.0 เปอร์เซ็นต์ และเปอร์เซ็นต์ความผิดพลาดด้านยาวคือ 6.9 เปอร์เซ็นต์

4.1.3 ผลของเหรียญสิบบาท

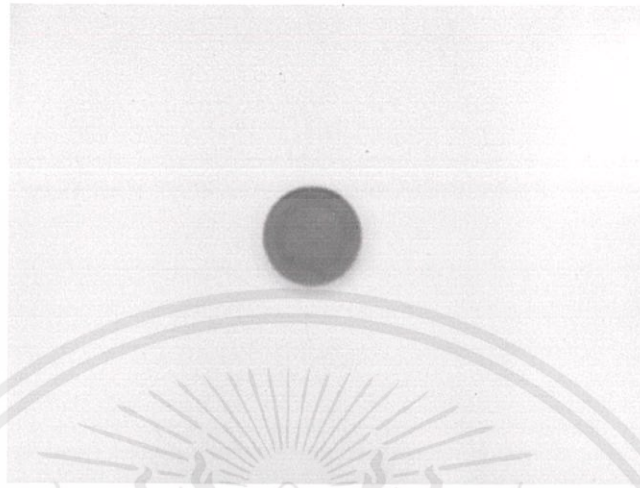
ขั้นตอนที่ 1 : ทำการจับภาพของวัตถุและทำให้ภาพสีเป็นภาพสีเทา เนื่องจากการปรับภาพและการหาขนาดภาพโปรแกรมยึดค่าสีเทาเป็นหลัก ดังรูปที่ 4.15



รูปที่ 4.15 ภาพสีเทาของเหรียญสิบบาท

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 : ทำการปรับภาพแบบ GaussianBlur เป็นการปรับปรุงภาพให้ดีขึ้น โดยปรับปรุงจากตรงกลางภาพก่อนแล้วไปจนถึงขอบภาพ สังเกตได้จากขอบของภาพเป็นโทนสีเดียวกัน ดังรูปที่ 4.16



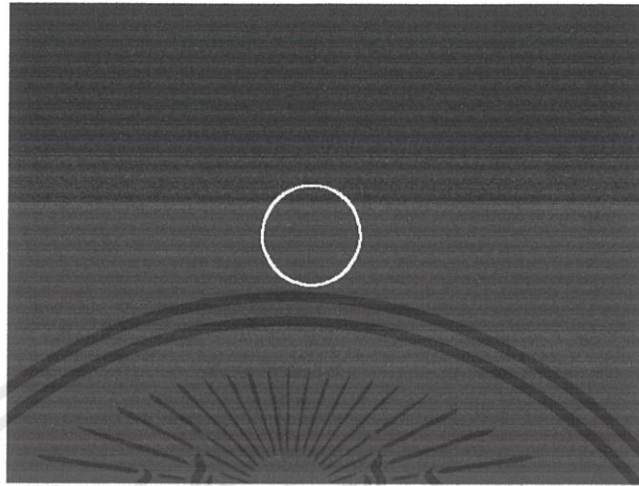
รูปที่ 4.16 ภาพ GaussianBlur เหยี่ยวลึบภาพ

ขั้นตอนที่ 3 : ใช้วิธี Canny เพราะช่วยกำจัดสัญญาณรบกวนได้ด้วย และโปรแกรมจะทำการเก็บค่าของจุดกึ่งกลางของวัตถุเพื่อทำการคำนวณในขั้นตอนต่อไป ดังรูปที่ 4.17



รูปที่ 4.17 ภาพ Canny เหยี่ยวลึบภาพ

ขั้นตอนที่ 4 : ใช้วิธี Dialate เพื่อเป็นการขยายภาพให้ใหญ่ขึ้น เพื่อเพิ่มสีให้กับวัตถุที่แสดงผล
ดังรูปที่ 4.18



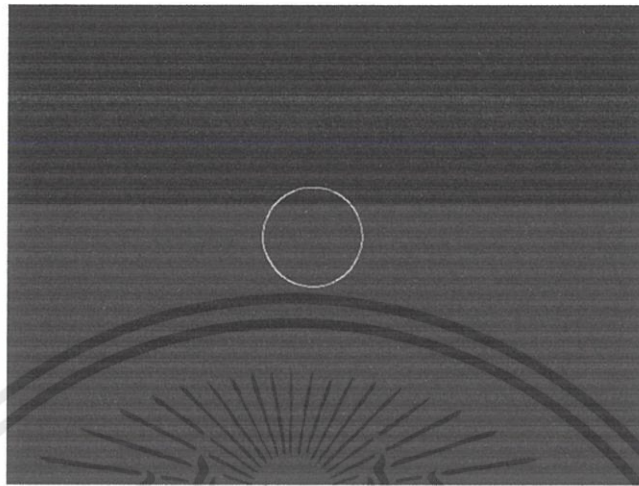
รูปที่ 4.18 ภาพ Dialate เหยี่ยวลึบบาท

ขั้นตอนที่ 5 : ใช้วิธี Erode เพื่อเป็นการกร่อนขนาดบริเวณขอบของวัตถุ ซึ่งการกร่อนมีวิธีคล้ายกับการขยายคือ สร้างส่วนประกอบโครงสร้างขึ้นมาแล้วนำไปกราดตามข้อมูลภาพ ดังรูปที่ 4.19



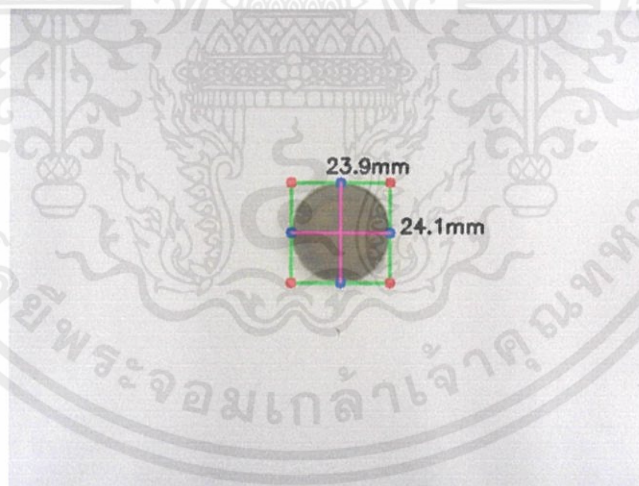
รูปที่ 4.19 ภาพ Erode เหยี่ยวลึบบาท

ขั้นตอนที่ 6 : ใช้วิธี Contour, Imutile (เหรียญสิบบาท) เพื่อเป็นการหามุมต่างๆ ของภาพและจุดกึ่งกลางของภาพเพื่อตรวจสอบความถูกต้องในการเก็บค่าจุดกึ่งกลางของวิธี Candy ดังรูปที่ 4.20



รูปที่ 4.20 ภาพ Contour และ Imutile เหรียญสิบบาท

ขั้นตอนที่ 7 : การหาระยะและแสดงผล หลังจากผ่านขั้นตอนที่ 1-6 จะนำการคำนวณที่ได้มาแสดงในภาพแรกที่ได้จากการถ่ายของกล้อง โดยที่ยังไม่ได้ผ่านขั้นตอนการปรับภาพเป็นสีเทา ดังรูปที่ 4.21



รูปที่ 4.21 ภาพขนาดเหรียญสิบบาท

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ขนาดของเหรียญสิบบาท

การตรวจวัด	ความกว้าง	ความยาว	การตรวจวัด	ความกว้าง	ความยาว
1	23.9	24.1	6	24.3	24.0
2	24.0	24.0	7	24.5	24.3
3	24.5	24.3	8	24.3	24.3
4	24.8	24.5	9	24.4	24.4
5	24.5	24.3	10	24.3	24.3

ขนาดมาตรฐานของเหรียญสิบบาทคือ 26 x 26 มิลลิเมตร เปอร์เซ็นต์ความผิดพลาดด้านกว้างคือ 6.3 เปอร์เซ็นต์ และเปอร์เซ็นต์ความผิดพลาดด้านยาวคือ 6.7 เปอร์เซ็นต์

บทที่ 5

บทวิจารณ์และสรุป

5.1 สรุปผลการทดลอง

จากการทดลองการหาขนาดของวัตถุที่มีความสูงที่ต่างกันจะส่งผลต่อการคำนวณ เพราะค่า Pixel/Metric จะมีค่าที่ต่างกันทำให้เกิดเปอร์เซ็นต์ความผิดพลาดที่มากขึ้น ถ้าความสูงมีขนาดเท่าๆ กัน อย่างเช่น ฝาขวดน้ำที่ต่างยี่ห้อกันจะต่างกันที่เส้นผ่าศูนย์กลางจะได้ขนาดที่มีค่าเปอร์เซ็นต์ความผิดพลาดน้อย เพราะฉะนั้นความสูงของวัตถุมีผลต่อการวัดขนาดมากๆ

5.2 ปัญหาและอุปสรรคที่พบ

1. ปัญหา เนื่องจากวัตถุมีขนาดความสูงที่ต่างกันทำให้ค่าที่วัดออกมาไม่ตรงกับความเป็นจริง
2. ปัญหา เนื่องจากการวางวัตถุจะต้องอยู่ตรงกลางภาพเสมอ ถ้าอยู่ขอบภาพจะทำให้เกิดเงาหรือจะเห็นส่วนของความหนาของวัตถุทำให้ค่าที่วัดออกไม่ผิดพลาดได้
3. ปัญหา เนื่องจากการจะวัดขนาดวัตถุนั้นต้องทำให้สายพานหยุดก่อนถึงจะวัดค่าได้ทำให้เสียเวลาในการวัดขนาด

5.3 แนวทางแก้ไข

1. เขียนโปรแกรมให้สอดคล้องกับวัตถุที่จะนำมาวัดและสมการในการคำนวณที่มากขึ้น
2. ศึกษาการใช้โปรแกรม Python ให้มากขึ้นเพื่อความยืดหยุ่นในการทำงาน
3. ศึกษาการใช้วีดีโอในการคำนวณเพื่อที่สายพานจะได้ไม่ต้องหยุด

5.4 ข้อเสนอแนะ

1. เพิ่มความละเอียดของกล้อง เพื่อจะวัดขนาดวัตถุได้มีความละเอียดที่มากขึ้น
2. เปลี่ยนสายพานเป็นแบบพื้นสีดำ เพื่อจะได้ไม่ต้องใส่ถาดสีขาวในการวัดขนาดวัตถุ

5.5 แนวทางการพัฒนา

1. ศึกษาภาษา Python และไลบรารีให้มากขึ้น เพื่อการประมวลผลที่รวดเร็วและไม่ต้องให้สายพานหยุด
2. เพิ่มกล้องอีก 1 ตัว เพื่อจะได้วัดขนาดเป็น 3 มิติ



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] “OpenCV 2 computer vision application programming cookbook” , Laganiere Robert, ISBN: 978-1-8495-1324-1 Year:2011
- [2] “Image processing : theory, algorithms, and architectures/Maher A. Sid-Ahmed” , Sid-Ahmed, Maher A ISBN: 0-07-057240-2Year: 1995
- [3] “Beginning Python From Novice to Professional” , Magnus Lie Hetland , 2nd Edition
- [4] “Image Processing” [online.] Available:
<https://www.engineersgarage.com/articles/image-processing-tutorial-applications>
- [5] “Image Smoothing” [online.] Available:
<http://www.cse.psu.edu/~rtc12/CSE486/lecture04.pdf>



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วิธีการใช้งานของระบบ

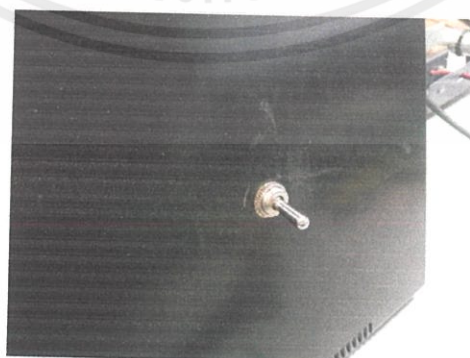
ขั้นตอนที่ 1 ทำการวางวัตถุไว้บนถาดที่เตรียมไว้



ขั้นตอนที่ 2 นำถาดไปวางบนสายพาน

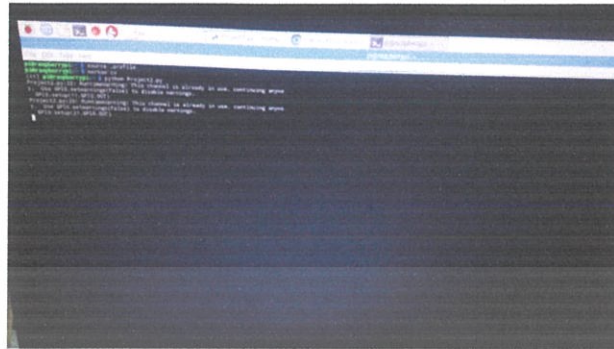


ขั้นตอนที่ 3 เปิดระบบโดยโยกสวิตช์เพื่อทำการจ่ายไฟให้กับระบบทั้งหมด

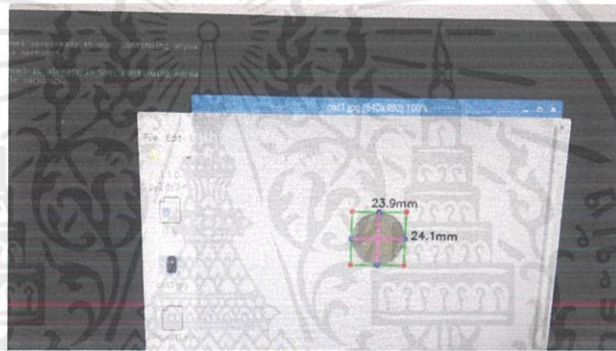


เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 รันโปรแกรม

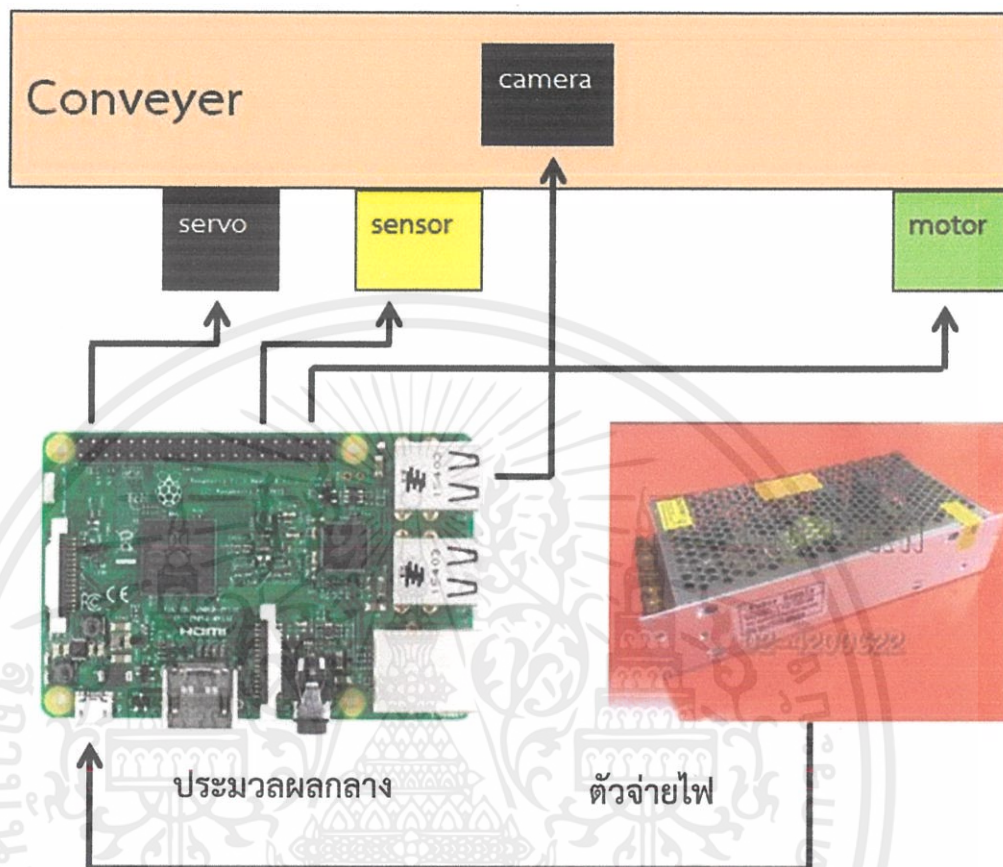


ขั้นตอนที่ 5 รูปที่วัดได้จะปรากฏบนจอภาพ



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการทำงาน

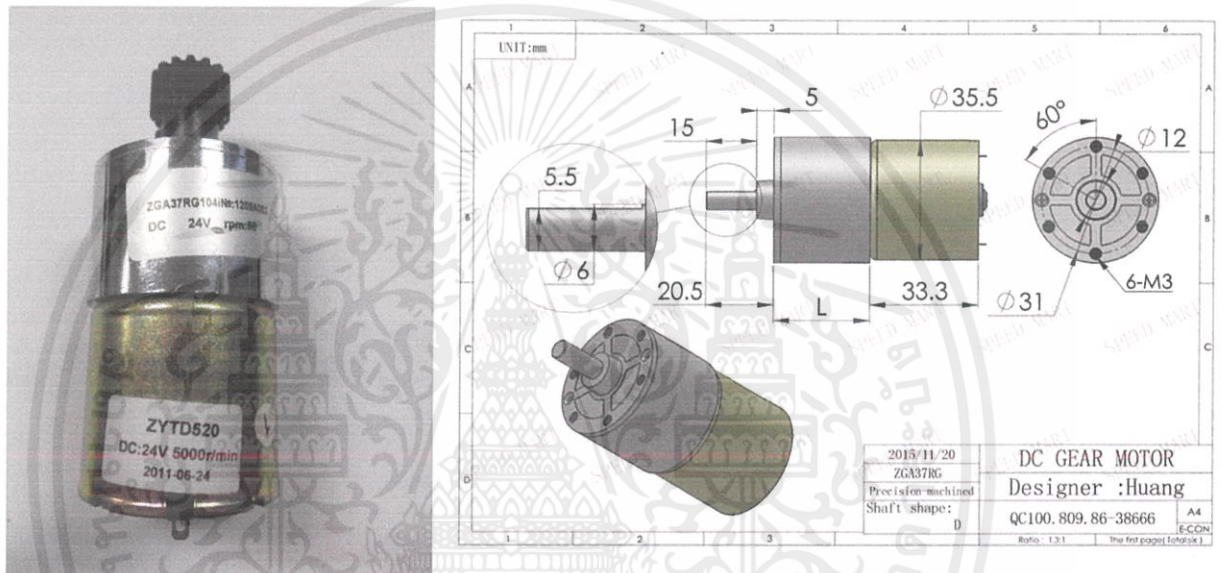


เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

Data Sheet

ZGA37RG104 dc gear motor ZYTD520 motor eccentric
shaft diameter 37mm 24V 50R/Min



■ 技术参数 Technical Data

电压 Voltage(VDC)	12V	12V	12V	12V	12V	24V	24V	24V	24V	24V
马达型号 Motor Model	17G	50G	50G	50G	35G	35G	50G	50G	50G	50G
减速比 Reduction Ratio	1/267	1/242	1/188	1/104	1/34.5	1/686	1/242	1/188	1/104	1/40
齿轮箱长度 Gearbox Length(mm)	29.5	29.5	29.5	26.5	24.5	30.5	29.5	29.5	26.5	24.5
空载转速 No-Load Speed(Rpm)	5	20	30	50	100	5	20	30	50	120
额定转速 Rated Speed(Rpm)	3.5	14	21	35	70	3.5	14	21	35	84
额定力矩 Rated Torque(Kg.cm)	7.18	15	14	8.8	4.5	15	15	10.5	6.3	3.95
额定电流 Rated Current(Amp)	0.16	1.1	1.1	1.1	1.1	0.25	0.36	0.36	0.36	0.36

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Webcam Logitech (C170)

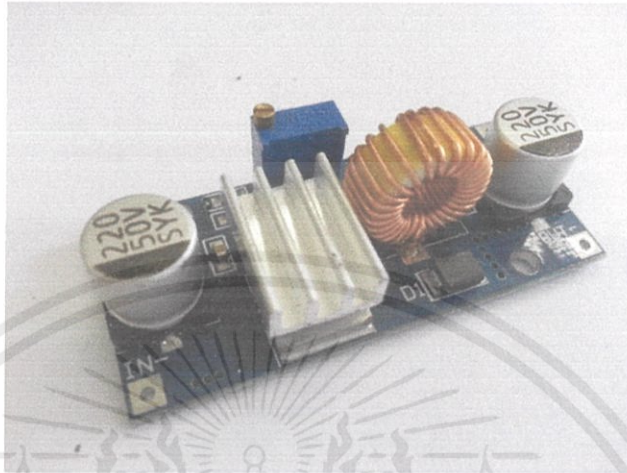


Camera Specifications:	
Available Image(s)	Click for [Front Image]
Connection Type	USB
USB Type	USB 2.0
USB VID_PID	082B
UVC Support	Yes
Microphone	Yes
Microphone Type	Mono
Lens and Sensor Type	Plastic
Focus Type	Fixed
Optical Resolution	True 640x480, 11.3MP, interpolated
Diagonal Field of View (FOV)	58°
Focal Length	2.3 mm
Image Capture (4:3 SD)	640x480, 1.3MP, 3MR, 5MP
Image Capture (16:9 W)	320x180, 360P
Video Capture (4:3 SD)	320x240, 640x480, 1024x768
Video Capture (16:9 W)	320x180, 360P
Frame Rate (max)	640x480@30
Right Light	N/A
Video Effects (VFX)	N/A
Buttons	N/A
Indicator Lights (LED)	Yes
Privacy Shade	No
Tripod Mounting Option	N/A
Universal Clip Adjustability (range)	71mm
Cable Length	4 Feet, 121.92 CM

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Step down DC-DC converter Buck Module

4-38V to 1.25-36V 5A



Specifications

This module can step-down applications is higher than the output voltage when the input voltage, such as battery, power transformer, DIY adjustable regulated power supply, 24V vehicle notebook power supply, industrial equipment, buck, 12V 3.3V, 12V 5V, 24V 5V, 24V 12V, 36V 24V etc.

Module name : 5A step down module.

Module property : Non-isolated step down module(BUCK).

Input voltage : 4V-38V

Output voltage : 1.25V-36V (adjustable)

Output current : 0A-5A,for safe working current, please under 4.5A.

Output power : it is recommended to use in the 75W, more than 50W Please add heat sink.

Working temperature : -40 degree~+85degree

Working frequency : 180 KHz

Transfer efficiency : 96% (max)

Load regulation rate : $S(I) \leq 0.8\%$

Voltage regulation rate : $S(U) \leq 0.8\%$

Power indicator : yes

Short circuit protection : (current limiting 8A)

Over temperature protection: Yes (automatic shutdown output ultra temperature)

Short circuit protection : Yes

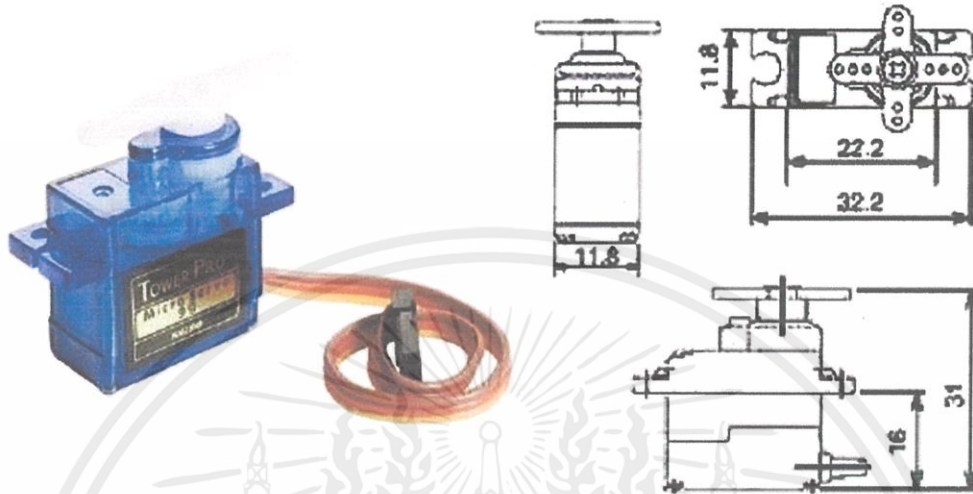
Input reverse connect protection: NO

Output reverse connect protection: No

PCB size : 54x24x12mm

Servo SG 90

SG90 9 g Micro Servo

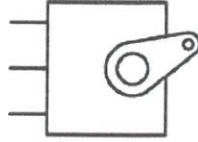
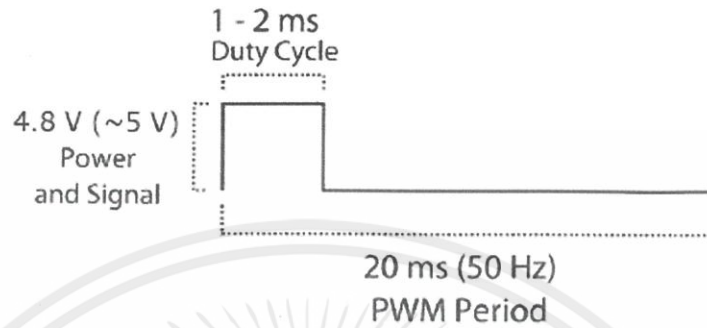


Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but *smaller*. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.

Specifications

- Weight: 9 g
- Dimension: 22.2 x 11.8 x 31 mm approx.
- Stall torque: 1.8 kgf·cm
- Operating speed: 0.1 s/60 degree
- Operating voltage: 4.8 V (~5V)
- Dead band width: 10 μ s
- Temperature range: 0 °C – 55 °C

PWM=Orange (⏏)
 Vcc = Red (+)
 Ground=Brown (-)

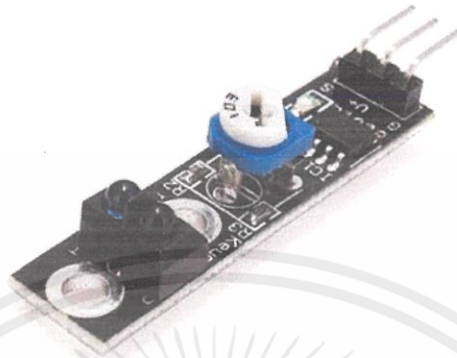



Position "0" (1.5 ms pulse) is middle, "90" (~2 ms pulse) is all the way to the right, "-90" (~1 ms pulse) is all the way to the left.



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Sensor KY-033



Specifications:

- Working Voltage : 3.3V - 5V DC
- Working Current : ≥ 20 mA
- Operating Temperature : $-10^{\circ}\text{C} \sim +50^{\circ}\text{C}$
- Detection distance : 2-40 cm
- IO Interface : 4-Wire Interfaces (-/+/S/EN)
- Output Signal : TTL Level (Low level when detecting objects/ high level when no objects/0 or 1 decides if objects exist)
- Effective angle : 35°
- Size : 28mm X 33mm



Raspberry Pi

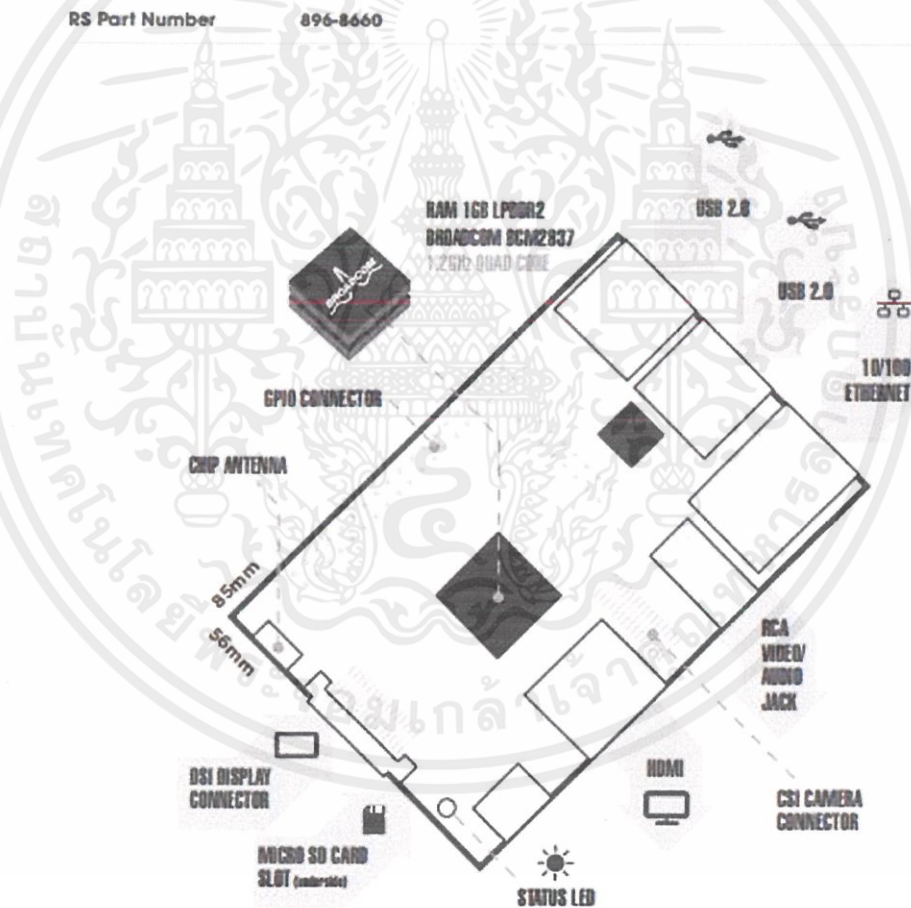


Raspberry Pi 3 Model B

Product Name Raspberry Pi 3

Product Description The Raspberry Pi 3 Model B is the third generation Raspberry Pi. This powerful credit-card sized single board computer can be used for many applications and supersedes the original Raspberry Pi Model B+ and Raspberry Pi 2 Model B. Whilst maintaining the popular board format the Raspberry Pi 3 Model B brings you a more powerful processor, 10x faster than the first generation Raspberry Pi. Additionally it adds wireless LAN & Bluetooth connectivity making it the ideal solution for powerful connected designs.

RS Part Number 896-8660



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Raspberry Pi 3 Model B

Specifications

Processor	Broadcom BCM2387 chipset. 1.2GHz Quad-Core ARM Cortex-A53 802.11 b/g/n Wireless LAN and Bluetooth 4.1 (Bluetooth Classic and LE)
GPU	Dual Core VideoCore IV® Multimedia Co-Processor. Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile decode. Capable of 1Gpixel/s, 1.5Gpixel/s or 24GFLOPs with texture filtering and DMA infrastructure
Memory	1GB LPDDR2
Operating System	Boots from Micro SD card, running a version of the Linux operating system or Windows 10 IoT
Dimensions	85 x 56 x 17mm
Power	Micro USB socket 5V1, 2.5A
Connectors:	
Ethernet	10/100 BaseT Ethernet socket
Video Output	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
Audio Output	Audio Output 3.5mm Jack, HDMI USB 4 x USB 2.0 Connector
GPIO Connector	40-pin 2.54 mm (100 mil) expansion header; 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
Camera Connector	15-pin MIPI Camera Serial Interface (CSI-2)
Display Connector	Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane
Memory Card Slot	Push/pull Micro SDIO

Key Benefits

- Low cost
- 10x faster processing
- Consistent board format
- Added connectivity

Key Applications

- Low cost PC/tablet/laptop
- Media centre
- Industrial/Home automation
- Print server
- Web camera
- Wireless access point
- Environmental sensing/monitoring (e.g. weather station)
- IoT applications
- Robotics
- Server/cloud server
- Security monitoring
- Gaming



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

Source Code

```

__author__ = Natthawut , Phuriphat , Phuwarin
# import เท่าที่จำเป็น
from scipy.spatial import distance as dist
from imutils import perspective
from imutils import contours
import numpy as np
import argparse
import imutils
import cv2
import multiprocessing
import time
import RPi.GPIO as GPIO
# Set up ขาต่างๆ
GPIO.setmode(GPIO.BOARD)
GPIO.setup(7,GPIO.IN)
GPIO.setup(11,GPIO.OUT)
GPIO.setup(21,GPIO.OUT)
p=GPIO.PWM(21,50)

# ประกาศฟังก์ชันเซอร์โว
def servo():
    p.start(7.5)
    p.ChangeDutyCycle(4.0)
    time.sleep(2)
    p.ChangeDutyCycle(7.0)
    time.sleep(1)
    print('servo finished')

```

ประกาศฟังก์ชันจุดกึ่งกลาง

```
def midpoint(ptA, ptB):
```

```
    return ((ptA[0] + ptB[0]) * 0.5, (ptA[1] + ptB[1]) * 0.5)
```

ถ่ายรูป , ทำรูปเป็นสีเทา และ ปรับปรุงภาพ

```
def cam(A,B):
```

```
    cam = cv2.VideoCapture(0) # 1 -> index of camera
```

```
    s, image = cam.read()
```

```
    gray = cv2.cvtColor(image,cv2.COLOR_BGR2GRAY)
```

```
    gray = cv2.GaussianBlur(gray, (7, 7), 0)
```

หาขอบ ขยายภาพและกร่อนภาพ

```
    edged = cv2.Canny(gray, 50, 100)
```

```
    edged = cv2.dilate(edged, None, iterations=1)
```

```
    edged = cv2.erode(edged, None, iterations=1)
```

หาขอบขอบวัตถุ

```
    cnts = cv2.findContours(edged.copy(), cv2.RETR_EXTERNAL,
                           cv2.CHAIN_APPROX_SIMPLE)
```

```
    cnts = cnts[0] if imutils.is_cv2() else cnts[1]
```

```
    (cnts, _) = contours.sort_contours(cnts)
```

```
    for c in cnts:
```

```
        if cv2.contourArea(c) < 100:
```

```
            continue
```

```
        orig = image.copy()
```

```
        box = cv2.minAreaRect(c)
```

```
        box = cv2.cv.BoxPoints(box) if imutils.is_cv2() else cv2.boxPoints(box)
```

```
        box = np.array(box, dtype="int")
```

หาจุดมุมต่างๆของภาพ

```
        box = perspective.order_points(box)
```

```
        cv2.drawContours(orig, [box.astype("int")], -1, (0, 255, 0), 2)
```

```
        for (x, y) in box:
```

```
            cv2.circle(orig, (int(x), int(y)), 5, (0, 0, 255), -1)
```

```
        (tl, tr, br, bl) = box
```

```
        (tltrX, tltrY) = midpoint(tl, tr)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(blbrX, blbrY) = midpoint(bl, br)
# หาจุดกึ่งกลางของด้านต่างๆของภาพ
(tlblX, tlblY) = midpoint(tl, bl)
(trbrX, trbrY) = midpoint(tr, br)
# แสดงจุดกึ่งกลางของด้านต่างๆของภาพ
cv2.circle(orig, (int(tltrX), int(tltrY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(blbrX), int(blbrY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(tlblX), int(tlblY)), 5, (255, 0, 0), -1)
cv2.circle(orig, (int(trbrX), int(trbrY)), 5, (255, 0, 0), -1)
# เชื่อมจุดกับจุดเข้าด้วยกันเป็นสี่เหลี่ยม
cv2.line(orig, (int(tltrX), int(tltrY)), (int(blbrX), int(blbrY)),
          (255, 0, 255), 2)
cv2.line(orig, (int(tlblX), int(tlblY)), (int(trbrX), int(trbrY)),
          (255, 0, 255), 2)
# หาระยะจุดศูนย์กลาง
dA = dist.euclidean((tltrX, tltrY), (blbrX, blbrY))
dB = dist.euclidean((tlblX, tlblY), (trbrX, trbrY))
# คำนวณด้านกว้างกับด้านยาว
dimA = dA / 4.15
dimB = dB / 4.15
A.value = dA / 4.15
B.value = dB / 4.15
# โชว์ขนาดภาพ
cv2.putText(orig, "{:.1f}mm".format(dimB),
            (int(tltrX - 15), int(tltrY - 10)), cv2.FONT_HERSHEY_SIMPLEX,
            0.65, (0, 0, 0), 2)
cv2.putText(orig, "{:.1f}mm".format(dimA),
            (int(trbrX + 10), int(trbrY)), cv2.FONT_HERSHEY_SIMPLEX,
            0.65, (0, 0, 0), 2)
# โชว์รูปที่ได้จากการคำนวณ
cv2.imwrite('oat3.jpg', orig)
cv2.imshow("Image", orig)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

cv2.waitKey(0)
time.sleep(0.5)
cv2.destroyAllWindows()

```

ฟังก์ชันการทำงานของระบบ

```

if __name__=='__main__':
    A=multiprocessing.Value('d')
    B=multiprocessing.Value('d')
    while True:
        #print(GPIO.input(7))
        if GPIO.input(7)==0 and s==1:
            print('cam start')
            GPIO.output(11, GPIO.HIGH)
            time.sleep(0.3)
            c=multiprocessing.Process(target=cam, args=(A,B,))
            c.start()
            c.join()
            time.sleep(1)
            #print(A.value)
            if (A.value<29 or A.value>30) or (B.value<29 or B.value>30):
                GPIO.output(11, GPIO.LOW)
                time.sleep(4.5)
                vo=multiprocessing.Process(target=servo)
                vo.start()
                vo.join()
                print('Incorrect')
                time.sleep(1)
            else :
                s=2
                print('Correct')
            s=2
        elif GPIO.input(7)==0 and s==2:
            GPIO.output(11, GPIO.LOW)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
time.sleep(0.3)
else :
    GPIO.output(11, GPIO.LOW)
    s=1
```



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ชื่อ - สกุล ภุริพัฒน์ จงเจริญนิรันดร์
วัน/เดือน/ปีเกิด 13 กรกฎาคม 2537
สาขาวิชา หลักสูตรวิศวกรรม ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ที่อยู่ 555/18 หมู่บ้าน อนุสาสิริ ถ.เทพารักษ์ ต.บางพลีใหญ่ อ.บางพลี จ.สมุทรปราการ
 10540
เบอร์โทรศัพท์ 0851112162
E-mail phuriphat_01@hotmail.com





ชื่อ - สกุล ภูวรินทร์ กลิ่นสง
 วัน/เดือน/ปีเกิด 17 สิงหาคม 2537
 สาขาวิชา หลักสูตรวิศวกรรม ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ที่อยู่ 134/169 ซ.14 ถ.ประชาราษฎร์ 1 เขต บางซื่อ กรุงเทพฯ 10800
 เบอร์โทรศัพท์ 0870704766
 E-mail Sirasit131@gmail.com



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้