

การติดตามชิ้นงานของแขนกลระนาบสองมิติ
ด้วยวิธีการมองเห็นของเครื่องจักร



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมเครื่องกล
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2559

**MACHINE VISION FOR PLANAR ROBOT
PART TRACKING**



**A Thesis Submitted in Partial Fulfillment of the Requirements for the
Degree of Master of Engineering in Mechanical Engineering**

Suranaree University of Technology

Academic Year 2016

การติดตามชิ้นงานของแขนกลระนาบสองมิติด้วยวิธีการมองเห็นของเครื่องจักร

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นำวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษา
ตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์



(ผศ. ดร. กীরดี สุลักษณ์)

ประธานกรรมการ



(รศ. ร.อ. ดร. กนดรรชนี ชำนิประศาสน์)

กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)



(อ. ดร. พิจิตรา เอื่องไพโรจน์)

กรรมการ



(อ. ดร. โสรัฎา แจ็งการ)

กรรมการ



(ผศ. ดร. ชานินทร์ จูมิม)

กรรมการ



(ศ. ดร. สุขกิจ ลิ้มปิงานงค์)

รองอธิการบดีฝ่ายวิชาการและนวัตกรรม



(รศ. ร.อ. ดร. กนดรรชนี ชำนิประศาสน์)

คณบดีสำนักวิชาวิศวกรรมศาสตร์

ภูวรินทร์ สีนมะเริง : การติดตามชิ้นงานของแขนกลระนาบสองมิติด้วยวิธีการมองเห็น
ของเครื่องจักร (MACHINE VISION FOR PLANAR ROBOT PART TRACKING)
อาจารย์ที่ปรึกษา : รองศาสตราจารย์ เรืออากาศเอก ดร.กนต์ธร ชำนิประศาสน์, 98 หน้า.

งานวิจัยนี้เป็นการออกแบบและสร้างเครื่องต้นแบบระดับห้องปฏิบัติการเพื่อใช้ในการติดตามชิ้นงาน โดยเพิ่มความสามารถในการทำงานของหุ่นยนต์ด้วยการใช้ระบบการมองเห็นของเครื่องจักร และสามารถสร้างชิ้นงานเสมือนกับชิ้นงานต้นแบบ โดยอาศัยการทำงานของหุ่นยนต์ที่ได้สร้างขึ้น งานวิจัยนี้จะเริ่มต้นด้วยการออกแบบรูปต้นแบบ กล้องจะทำการถ่ายภาพและทำการประมวลผลภาพด้วยโปรแกรมที่ถูกสร้างขึ้นมาด้วยซอฟต์แวร์ MATLAB 2014b จากนั้นโปรแกรมประมวลผลภาพจะทำการสั่งให้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติทำงานด้วยการวาดรูปใหม่ที่เสมือนกับรูปต้นแบบออกมาผ่าน ไมโครคอนโทรลเลอร์ Arduino

งานวิจัยนี้จะเป็นการออกแบบ โปรแกรมประมวลผลภาพ สร้างเครื่องต้นแบบ และทำการรวมระบบประมวลผลภาพและเครื่องต้นแบบเข้าด้วยกัน ผลที่ได้จากงานวิจัยนี้คือแต่ละระบบสามารถทำงานได้ตามวัตถุประสงค์ที่วางไว้ และยังสามารถทำงานร่วมกันได้อย่างมีประสิทธิภาพ การทำงานร่วมกันของระบบประมวลผลภาพและเครื่องต้นแบบเมื่อทดสอบความผิดพลาดปรากฏว่าในงานวิจัยนี้มีเปอร์เซ็นต์ความผิดพลาดประมาณไม่เกิน 3% ซึ่งความผิดพลาดที่เกิดขึ้นเป็นความผิดพลาดที่เกิดจากระบบการประมวลผลภาพ

สาขาวิชา วิศวกรรมเครื่องกล
ปีการศึกษา 2559

ลายมือชื่อนักศึกษา ภูวรินทร์ สีนมะเริง
ลายมือชื่ออาจารย์ที่ปรึกษา ดร.กนต์ธร ชำนิประศาสน์

PHUWARIN SINMAROENG : MACHINE VISION FOR PLANAR
ROBOT PART TRACKING. THESIS ADVISOR : ASSOC. PROF.
FLT. LT. KONTORN CHAMNIPRASART, Ph.D., 98 PP.

MACHINE VISION / 2-R PLANAR ROBOT

This research is aimed to design and prototype a laboratory machine for part tracking. This research improves machine performance by adding machine vision system and duplicating a new part of robot prototyping. This research designed machine vision for image processing, built 2-R planar robot and combined both system together. This system starts by designing of 2D image. Next a webcam camera captures a prototype picture and processes the picture using image processing generated with MATLAB 2014b software. Coordinates and angle obtained from the image processing are used to control the prototype 2-R planar robot for drawing the duplicated picture. The stepper motors of 2-R planar robot are controlled by microcontrolled Arduino. As the results of the study, each system can work properly. The machine vision system and the 2-R planar robot also work together efficiently. This research has size error less than 3% occurred from machine vision system.

School of Mechanical Engineering

Academic Year 2016

Student's Signature Phuwarin Sinmaroeng

Advisor's Signature Kontorn Chamniprasart

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี ผู้วิจัยขอขอบพระคุณบุคคลและกลุ่มบุคคลต่าง ๆ ที่ให้คำปรึกษา ชี้แนะแนวทาง และช่วยเหลือเป็นอย่างดีมาโดยตลอด ทั้งด้านวิชาการ และการดำเนินงานวิจัย ดังนี้

มหาวิทยาลัยเทคโนโลยีสุรนารี รวมถึงบุคลากรประจำศูนย์เครื่องมือวิทยาศาสตร์และเทคโนโลยี ที่ได้ให้ความช่วยเหลือในการใช้อุปกรณ์ และเครื่องมือต่าง ๆ ในการทำวิจัยนี้

รองศาสตราจารย์ เรืออากาศเอก ดร.กนต์ธร ชำนิประศาสน์ คณบดีสำนักวิชาวิศวกรรมศาสตร์ และยังเป็นอาจารย์ที่ปรึกษาวิทยานิพนธ์ที่ได้ มอบโอกาสทางการศึกษาในระดับบัณฑิตศึกษา อีกทั้งยังให้ความรู้ด้านวิชาการ การดำเนินชีวิต อีกทั้งยังให้คำปรึกษาจนสามารถผ่านอุปสรรคต่าง ๆ ลุล่วงมาได้ด้วยดี นอกจากนี้ยังชี้แนะ และมอบโอกาสที่ดี ๆ เสมอมาที่

ผู้ช่วยศาสตราจารย์ ดร.กิริติ สุกฤษณ์ หัวหน้าสาขาวิชาวิศวกรรมการผลิต ผู้ซึ่งประสาทวิชาความรู้ ให้คำแนะนำ ทั้งในด้านวิชาการ การดำเนินชีวิต อีกทั้งยังให้กำลังใจจนสามารถผ่านอุปสรรคต่าง ๆ ลุล่วงมาได้ด้วยดี นอกจากนี้ยังชี้แนะแนวทางการทำงาน

อาจารย์ ดร.พิจิตรา เอื้องไพโรจน์ อาจารย์ ดร.โสทรฎา แข็งการ และ อาจารย์วิชัย ศรีสุลักษณ์ ที่ประสาทวิชาความรู้ ให้คำแนะนำ ทั้งในด้านวิชาการ การดำเนินชีวิต อีกทั้งยังให้กำลังใจจนสามารถผ่านอุปสรรคต่าง ๆ ลุล่วงมาได้ด้วยดี นอกจากนี้ยังชี้แนะแนวทางการทำงาน

นายศิวศิลป์ พรจำศิลป์ และ นายอภิสิทธิ์ หล่อนกลาง ที่ได้ช่วยแนะนำการเขียนโปรแกรม การออกแบบโปรแกรมเพื่อให้ได้ตามเป้าหมายที่ได้กำหนดไว้

นายพันศักดิ์ เนระแก, นายรัฐพล อินทชาติ และ นายชัยยุทธ คงเชื้อ ที่ได้ช่วยออกแบบระบบการทำงานของเครื่องจักร และได้ช่วยสร้างเครื่องจักรจนสำเร็จ

บริษัท ไอคอน เอนจิเนียริ่ง โซลูชั่น ที่ให้ความอนุเคราะห์วัสดุในการทำวิจัย รวมทั้ง ข้อมูลที่จำเป็นต่องานวิจัย ตลอดจนคำแนะนำต่าง ๆ

สุดท้ายนี้ คุณงามความดีอันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอมอบแต่บิดา มารดา อันเป็นที่เคารพยิ่ง และคณาจารย์ผู้ประสาทวิชาความรู้ ตลอดจนทุก ๆ ท่านที่ให้กำลังใจช่วยเหลือจนกระทั่งวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงลงด้วยดี

ภูวรินทร์ สีนมะเริง

สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย)	ก
บทคัดย่อ (ภาษาอังกฤษ).....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ	ง
สารบัญตาราง	ช
สารบัญรูป.....	ซ
บทที่	
1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์การวิจัย	2
1.3 ขอบเขตของการวิจัย	2
1.4 สถานที่ทำงานวิจัย	3
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	3
2 ปรัชญาวรรณกรรมและทฤษฎีที่เกี่ยวข้อง.....	5
2.1 กล่าวนำ.....	5
2.2 การประมวลผลภาพ และการมองเห็นเครื่องจักร	5
2.2.1 ภาพดิจิทัล (Digital Image).....	5
2.2.2 การแปลงภาพดิจิทัลเป็นพิกเซล	8
2.2.3 Spline Interpolation.....	8
2.2.4 ซอฟต์แวร์ MATLAB	12
2.3 ทฤษฎีหุ่นยนต์เบื้องต้น และ Planar Robot	13
2.3.1 การหา D-H Parameters.....	19
2.3.2 การหาสมการจลนศาสตร์ไปข้างหน้าของหุ่นยนต์.....	20
2.3.3 การหาสมการจลนศาสตร์ย้อนกลับของหุ่นยนต์.....	23
2.3.4 ไมโครคอนโทรลเลอร์ (Microcontroller).....	24

สารบัญ (ต่อ)

หน้า

2.3.4.1	ส่วนประกอบทั่วไปของไมโครคอนโทรลเลอร์.....	25
2.3.4.2	ประเภทของไมโครคอนโทรลเลอร์.....	26
2.3.5	การควบคุมการทำงานของสเตปเปอร์มอเตอร์ด้วย Microcontroller	27
3	วิธีดำเนินการวิจัย.....	29
3.1	กล่าวนำ.....	29
3.2	การสร้างระบบการประมวลผลภาพ.....	29
3.2.1	การออกแบบระบบการประมวลผลภาพ.....	29
3.2.2	ส่วนประกอบของระบบประมวลผลภาพ.....	30
3.3	การออกแบบและสร้างเครื่องต้นแบบ.....	34
3.3.1	การออกแบบชุดกลไกสำหรับเครื่องต้นแบบ.....	34
3.3.2	การเลือกใช้อุปกรณ์ต้นกำลัง.....	35
3.4	การออกแบบขั้นตอนการทำงานของระบบ.....	38
3.4.1	Flow Chart การทำงานของระบบ.....	38
3.4.2	การแปลงข้อมูลจากรูปภาพเป็นพิกเซล.....	39
3.4.3	การกำหนดตำแหน่งพิกัด (x,y).....	40
3.4.4	การ Curve Fitting ด้วยวิธี Spline Interpolation.....	42
3.4.5	ออกแบบและสร้างโปรแกรมประมวลผลภาพผ่านหน้าต่าง GUI.....	43
3.4.6	การตั้งค่าโปรแกรมประมวลผลภาพและการเริ่มทำงานเบื้องต้น.....	45
3.4.7	การส่งออกข้อมูลเพื่อเก็บเป็นฐานข้อมูล.....	47
3.4.8	การส่งออกข้อมูลเพื่อไปควบคุมการทำงานของหุ่นยนต์แขนกลระนาบ สองมิติ.....	48
3.4.9	หุ่นยนต์แขนกลระนาบสองมิติ.....	49
3.5	เวลาที่ใช้ในการทำงานวิจัย.....	50
3.5.1	ช่วงการออกแบบรูปต้นแบบ.....	50
3.5.2	ช่วงการประมวลผลภาพและทำการส่งข้อมูล.....	51
3.5.3	ช่วงการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.....	51

สารบัญ (ต่อ)

หน้า

4 ผลการทดลอง.....	52
4.1 การทำงานของระบบการประมวลผลด้วยภาพ	52
4.1.1 เริ่มการทดสอบ.....	52
4.1.2 ซิดจำกัดของโปรแกรมประมวลผลภาพ.....	55
4.2 การทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.....	58
4.3 การทำงานร่วมกันระหว่างระบบการประมวลผลภาพและเครื่องต้นแบบ หุ่นยนต์แขนกลระนาบสองมิติ	64
4.3.1 ผลการทำงานของระบบการประมวลผลภาพร่วมกับ การทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	65
4.3.2 ผลการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ โดยไม่ผ่านระบบการประมวลผลภาพ	72
4.4 ผลการทดสอบเวลาการทำงานของงานวิจัย.....	77
4.5 ผลการทดสอบความถูกต้องของงานวิจัย.....	79
4.5.1 การทดสอบความถูกต้องทั้งระบบ	79
4.5.2 การทดสอบความถูกต้องเฉพาะการทำงานของเครื่องต้นแบบ	80
4.6 สรุปผลการทดสอบ.....	81
5 สรุปงานวิจัยและข้อเสนอแนะ	83
5.1 สรุปผลการวิจัย	83
5.2 ข้อเสนอแนะ	83
รายการอ้างอิง	85
ภาคผนวก	
ภาคผนวก ก Drawing เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.....	87
ภาคผนวก ข บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่	92
ประวัติผู้เขียน	98

สารบัญตาราง

ตารางที่		หน้า
2.1	เงื่อนไขขอบ.....	11
2.2	D-H Parameter of the 2-R Robot	20
3.1	แสดงข้อมูลตำแหน่งพิกัด (x,y) ที่สามารถอ่านค่าได้จากรูปที่ 3.15	41
4.1	แสดง D-H Parameter สำหรับ 2-R Planar Robot	59
4.2	ค่าพารามิเตอร์ของสเตปเปอร์มอเตอร์.....	63
4.3	ค่าพารามิเตอร์ของสเตปเปอร์มอเตอร์ร่วมกับ Driver IM483	64
4.4	แสดงเวลาทำงานเฉลี่ยของงานวิจัย	78



สารบัญรูป

รูปที่		หน้า
2.1	แสดงการกำหนดตำแหน่งของพิกเซล	7
2.2	แสดงการแปลงพิกเซล Thresholding	8
2.3	แสดงกราฟของเทคนิคการ Interpolation	9
2.4	แสดงกราฟของเทคนิค Least Square Regression	10
2.5	กราฟ Cubic Splines	10
2.6	ตัวอย่าง The Cartesian Robot	16
2.7	ตัวอย่าง The Cylindrical Robot	16
2.8	ตัวอย่าง The Spherical Robot	17
2.9	ตัวอย่าง The Articulated Robot หรือ The Revolute Robot	17
2.10	ตัวอย่าง The SCARA Robot	18
2.11	ตัวอย่าง The Parallel Robot	18
2.12	แสดงหุ่นยนต์แขนกลระนาบสองมิติ (Planar Robot)	19
2.13	แสดงรายละเอียดของตำแหน่งตัวแปรต่าง ๆ ของ D-H Parameters	20
2.14	แสดงตัวอย่างแบบจำลองของหุ่นยนต์ 2 แขน	21
2.15	แสดงข้อศอกขึ้นและลงของหุ่นยนต์	23
3.1	แสดงการเชื่อมต่ออุปกรณ์ระบบการประมวลผลภาพ	30
3.2	แสดงพื้นที่ในการออกแบบ	31
3.3	แสดงการกำหนดสีของกระดาษและปากกาสำหรับรูปต้นแบบ	31
3.4	แสดงขอบเขตที่สามารถออกแบบรูปต้นแบบโดยไม่เกินขอบเขตของเครื่องทดสอบ	32
3.5	แสดงกล้อง WEBCAM CAMERA	32
3.6	แสดงคอมพิวเตอร์ที่ใช้ในการประมวลผลภาพ	33
3.7	แสดงหน้าต่างของซอฟต์แวร์ MATLAB เวอร์ชัน 2014b	34
3.8	แสดงเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	35
3.9	แสดงส่วนประกอบของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	35

สารบัญรูป (ต่อ)

รูปที่		หน้า
3.10	แสดง VEXTA รุ่น ASM34AK-H50.....	36
3.11	แสดง Linear Actuator Robot.....	37
3.12	แสดง Microcontroller Nano ATmega328.....	37
3.13	แผนผังการทำงานของระบบ	38
3.14	แสดงการแปลงข้อมูลรูปภาพเป็นค่าพิกเซล.....	39
3.15	แสดงการแปลงข้อมูลรูปภาพเป็นค่าพิกเซล ตัวอย่างที่ 2	40
3.16	แสดงตำแหน่งพิกัดของค่าพิกเซล	41
3.17	แสดงกราฟที่พล็อตด้วยวิธี Spline Interpolation.....	42
3.18	รูปต้นแบบตัวอย่างที่ 2	42
3.19	แสดงหน้าต่าง GUI ของ โปรแกรมประมวลผลภาพ	43
3.20	แสดงส่วนประกอบของหน้าต่างโปรแกรมประมวลผลภาพ	44
3.21	แสดงการเลือกกล้องที่ใช้ในโปรแกรมประมวลผลภาพ	45
3.22	แสดงการเลือกความละเอียดที่ใช้ในการประมวลผลภาพ	45
3.23	แสดงการทำงานของโปรแกรมประมวลผลภาพ.....	46
3.24	แสดงการกำหนดสีปากการของโปรแกรม.....	47
3.25	แสดงข้อมูลที่ถูกส่งออกมา.....	48
3.26	แสดงฟังก์ชันการส่งข้อมูลของซอฟต์แวร์ MATLAB เวอร์ชัน 2014b.....	49
3.27	แสดงฟังก์ชันการรับข้อมูลของไมโครคอนโทรลเลอร์ Arduino Nano ATmega328	49
3.28	แสดงเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	50
4.1	การออกแบบและสร้างเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.....	52
4.2	แสดงการทดสอบโปรแกรมประมวลผลภาพรูปเส้นโค้ง.....	53
4.3	แสดงการทดสอบโปรแกรมประมวลผลภาพรูปเส้นโค้งและหัวใจ.....	53
4.4	แสดงการทดสอบโปรแกรมประมวลผลภาพรูปดอกไม้.....	54
4.5	แสดงการทดสอบโปรแกรมประมวลผลภาพด้วยปากกาสองสี.....	55
4.6	แสดงขีดจำกัดข้อแรกของโปรแกรมประมวลผลภาพ.....	56
4.7	แสดงขีดจำกัดข้อที่สองของโปรแกรมประมวลผลภาพ.....	56

สารบัญรูป (ต่อ)

รูปที่		หน้า
4.8	แสดงการแก้ปัญหาเบื้องต้นของจิตจำกดข้อแรกของโปรแกรมประมวลผลภาพ.....	57
4.9	แสดงพื้นที่สำหรับใช้ในการออกแบบ.....	58
4.10	แสดง Model ของหุ่นยนต์ 2-R Planar Robot.....	59
4.11	แสดงท่าทางของแขนหุ่นยนต์แขนกลระนาบสองมิติ	61
4.12	แสดงพื้นที่การทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.....	62
4.13	แสดง VEXTA รุ่น ASM34AK-H50.....	62
4.14	แสดง Driver Stepping Motor : IM483	63
4.15	แสดงพื้นที่การทำงานร่วมกันระหว่างระบบประมวลผลภาพและเครื่องต้นแบบ ฯ	64
4.16	แสดงรูปต้นแบบ ก. ไม้ และภาพที่แสดงผ่านระบบประมวลผลภาพ	65
4.17	แสดงรูป ก. ไม้ จากการนำชุดข้อมูลตำแหน่งพิกัด ไปพล็อตเป็นรูป	66
4.18	แสดงรูป ก. ไม้ ที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	66
4.19	แสดงรูปต้นแบบเส้นโค้งกราฟไซน์ และภาพที่แสดงผ่านระบบประมวลผลภาพ	67
4.20	แสดงรูปเส้นโค้งกราฟไซน์ จากการนำชุดข้อมูลตำแหน่งพิกัด ไปพล็อตเป็นรูป	68
4.21	แสดงรูปเส้นโค้งกราฟไซน์ที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	68
4.22	แสดงรูปต้นแบบรูปวงกลม และภาพที่แสดงผ่านระบบประมวลผลภาพ.....	69
4.23	แสดงรูปวงกลม จากการนำชุดข้อมูลตำแหน่งพิกัด ไปพล็อตเป็นรูป.....	69
4.24	แสดงรูปวงกลมที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.....	70
4.25	แสดงรูปต้นแบบเส้นโค้งและเส้นตรง และภาพที่แสดงผ่านระบบประมวลผลภาพ	70
4.26	แสดงรูปเส้นโค้งและเส้นตรง จากการนำชุดข้อมูลตำแหน่งพิกัดไปพล็อตเป็นรูป	71
4.27	แสดงรูปเส้นโค้งและเส้นตรงที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	71
4.28	แสดงรูปสี่เหลี่ยมที่สร้างจากสมการ	73
4.29	แสดงรูปสี่เหลี่ยมที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.....	73
4.30	แสดงรูปเลขเก้าที่สร้างจากสมการ	74
4.31	แสดงรูปเลขเก้าที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	75
4.32	แสดงรูป ก. ไม้ ที่สร้างจากสมการ.....	75
4.33	แสดงรูป ก. ไม้ ที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ	76

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.34	แสดงรูปสามเหลี่ยมที่สร้างจากสมการ..... 76
4.35	แสดงรูปสามเหลี่ยมที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ 77
ก.1	แกนลำดับที่หนึ่งของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ 88
ก.2	แกนลำดับที่สองของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ 88
ก.3	คัปปี้งของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ 89
ก.4	ฐานรองสำหรับยึดกับ Linear Robot ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ.. 89
ก.5	Linear Robot ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ 90
ก.6	ฐานจับปากกาของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ..... 90
ก.7	สเตปเปอร์มอเตอร์ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ..... 91
ก.8	Assembly ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ..... 91

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันอุตสาหกรรมถือเป็นสิ่งหนึ่งที่สำคัญต่อมนุษย์อย่างยิ่ง ด้วยว่ามนุษย์ต้องพึ่งพาการผลิตสิ่งที่เป็นต่อชีวิตประจำวัน หรือเรียกรวมว่าปัจจัยสี่ โดยสิ่งที่สามารถผลิตปัจจัยสี่ให้ดี มีคุณภาพ และไม่ก่ออันตรายหรือก่ออันตรายให้กับร่างกายและทรัพย์สินน้อยที่สุดคือการผลิตจากอุตสาหกรรม ซึ่งในอุตสาหกรรมนั้นก็ประกอบไปด้วยกระบวนการย่อย ๆ หลายอย่าง อาทิเช่น การออกแบบ การสร้าง การประกอบ การตรวจสอบ เป็นต้น

กระบวนการออกแบบถือได้ว่าเป็นหนึ่งในกระบวนการที่มีความสำคัญอย่างมากในอุตสาหกรรม เพราะกระบวนการออกแบบเป็นจุดริเริ่มของการสร้างงาน โดยการออกแบบสามารถนำไปประยุกต์ใช้กับหุ่นยนต์และระบบการทำงานอัตโนมัติได้อย่างมาก หนึ่งในความท้าทายสำหรับนักออกแบบ คือ การนำแบบที่ตนเองคิดค้นได้หรือออกแบบมานั้นนำมาใช้งานได้จริง ซึ่งปัจจุบันจะเป็นการทำงานด้วยเครื่องจักรแทบจะทั้งสิ้น นักออกแบบนอกจากจะต้องมีความรู้ความสามารถทางการออกแบบจากจินตนาการแล้ว ยังต้องมีความเชี่ยวชาญในการออกแบบงานผ่านตัวโปรแกรมสำหรับออกแบบ ที่โปรแกรมเหล่านั้นจะนำไปประยุกต์กับเครื่องจักรต่อไป

หากการออกแบบด้วยโปรแกรมออกแบบอยู่ในเงื่อนไขรูปทรงพื้นฐานอาจจะไม่เป็นปัญหามากนัก รูปทรงพื้นฐานในที่นี้คือ รูปทรงเรขาคณิตทั้งสมมาตรและไม่สมมาตร แต่หากเป็นรูปในอุดมคติ รูปในจินตนาการก็จะเป็นเรื่องที่ยุ่ยากที่จะใช้โปรแกรมออกแบบในการออกแบบได้ เช่น เส้นโค้งเว้า รูปที่ไม่ใช่เรขาคณิต เป็นต้น

ปัญหาที่พบตามมา คือนักออกแบบหลายคนมีความรู้ความสามารถมากแต่ขาดประสบการณ์เกี่ยวกับโปรแกรมที่ใช้ออกแบบ โดยนักออกแบบเหล่านั้นจะเชี่ยวชาญการออกแบบด้วยมือ หรือที่รู้จักในชื่อว่า Free-hand Sketch โดยทำการออกแบบผ่านทางกระดาษและปากกา แต่การออกแบบด้วยวิธีดังกล่าวนี้จะทำให้ไม่สามารถดึงข้อมูลจากการออกแบบส่งไปให้เครื่องจักรประมวลผลและทำงานได้

อีกด้านหนึ่งก็มีผู้สังเกตเห็นความสำคัญของปัญหาดังกล่าว จึงมีนวัตกรรมต่าง ๆ ตามออกมา อาทิเช่น คอมพิวเตอร์สำหรับออกแบบ โดยอุปกรณ์จะมาพร้อมชุดจอสำหรับออกแบบ และปากกาที่ใช้เชื่อมต่อโดยตรง การแก้ปัญหาที่จุดนี้สามารถตอบโจทย์ของปัญหาข้างต้นได้ส่วนหนึ่ง แต่

คอมพิวเตอร์และปากกาที่ใช้ในการออกแบบนี้ก็ยังไม่สามารถทดแทนความรู้สึกของการวาดรูปด้วยมือได้ และด้วยการลงทุนที่สูงตามมาก็ยังทำให้หลาย ๆ โรงงานอุตสาหกรรมยังไม่เป็นที่พึงพอใจกับวิธีดังกล่าว

ดังนั้นจึงเป็นที่มาของงานวิจัยนี้ ที่จะใช้เทคโนโลยีอุตสาหกรรม ที่เป็นศาสตร์หนึ่งที่เกี่ยวข้องกับเทคโนโลยี การนำเทคโนโลยีไปประยุกต์ใช้กับงานด้านต่าง ๆ ในอุตสาหกรรม โดยผู้วิจัยมีแนวคิดในการออกแบบและสร้างเครื่องต้นแบบระดับห้องปฏิบัติการด้วยการใช้การมองเห็นของเครื่องจักรร่วมกับการทำงานของหุ่นยนต์ที่ได้สร้างขึ้น มีหลักการทำงาน คือ นักออกแบบทำการวาดรูปใด ๆ หลังจากนั้นระบบการมองเห็นของเครื่องจักรจะทำการประมวลผลภาพดังกล่าวด้วยโปรแกรม MATLAB 2014b โดยผ่านอัลกอริทึมที่ผู้วิจัยได้ค้นคว้ามา ซึ่งโปรแกรมดังกล่าวถูกออกแบบมาให้ใช้งานสะดวกและง่ายต่อการทำความเข้าใจ หลังจากนั้น โปรแกรมจะทำการส่งข้อมูลไปที่เครื่องต้นแบบระดับห้องปฏิบัติการ จำลองโดยใช้รูปแบบของหุ่นยนต์แขนกลระนาบสองมิติเคลื่อนที่โดยใช้มอเตอร์ควบคุมแต่ละแกนผ่านโปรแกรม Microcontroller Arduino ซึ่งเป็นการจำลองการทำงานของเครื่องจักรในโรงงานอุตสาหกรรม ผู้วิจัยคาดว่าจะสามารถนำงานวิจัยไปประยุกต์ใช้กับอุตสาหกรรมอื่น ๆ ได้ต่อไป

1.2 วัตถุประสงค์การวิจัย

สำหรับงานวิจัยนี้ มีวัตถุประสงค์ของงานวิจัยในด้านการออกแบบและสร้างเครื่องต้นแบบระดับห้องปฏิบัติการด้วยการใช้การมองเห็นของเครื่องจักรร่วมกับการทำงานของหุ่นยนต์ที่ได้สร้างขึ้น ตามเงื่อนไขที่กำหนดให้ ซึ่งวัตถุประสงค์หลักของงานวิจัยมีดังต่อไปนี้

1.2.1 ออกแบบอัลกอริทึมของระบบการมองเห็นเครื่องจักรเพื่อสร้างระบบการประมวลผลด้วยภาพสำหรับใช้ในการออกแบบ

1.2.2 จัดสร้างเครื่องต้นแบบระดับห้องปฏิบัติการ ที่สามารถทำงานตามที่ต้องการได้อย่างถูกต้อง

1.2.3 ออกแบบระบบการทำงานร่วมกันระหว่างระบบการมองเห็นเครื่องจักรและเครื่องต้นแบบระดับห้องปฏิบัติการ เพื่อให้ทั้งสองส่วนทำงานที่สอดคล้องกันได้อย่างถูกต้องและเต็มประสิทธิภาพ

1.3 ขอบเขตงานวิจัย

งานวิจัยนี้จะกำหนดขอบเขตการวิจัยเพื่อสามารถกำหนดเป้าหมายและวิธีการในการแก้ปัญหาได้อย่างเป็นระบบ ขอบเขตของงานวิจัยนี้ประกอบด้วย

1.3.1 ออกแบบการมองเห็นของเครื่องจักร เพื่อการสร้างระบบประมวลผลภาพด้วยกล้อง Webcam Camera เพียงตัวเดียว

1.3.2 สร้างเครื่องต้นแบบเพื่อสามารถทำการวาดรูปตามแบบที่ต้องการ โดยเครื่องต้นแบบนั้นจะสร้างบนพื้นฐานของหุ่นยนต์ระนาบสองแกน (2-R Planar Robot) ที่มีแกนหลักยาว 11 เซนติเมตร และแกนรองยาว 10 เซนติเมตร

1.3.3 การมองเห็นของเครื่องจักร ใช้โปรแกรม MATLAB เวอร์ชัน 2014b

1.3.4 การควบคุมการทำงานของเครื่องต้นแบบทำงานโดยใช้ Stepper Motor 2 ตัว สำหรับควบคุมแต่ละแกนของหุ่นยนต์ โดยควบคุมผ่านซอฟต์แวร์ Microcontroller Arduino และใช้บอร์ด Arduino Nano 2 ตัว

1.3.5 รูปต้นแบบที่จะนำไปประมวลผลจะต้องเป็นรูปที่มีความต่อเนื่อง ไม่สามารถวาดสองรูปในเวลาเดียวกันได้

1.3.6 รูปต้นแบบที่จะนำไปประมวลผลจะเป็นรูปกรอบนอกเท่านั้น หากมีรูปอื่นใดอยู่ภายในรูปกรอบที่เป็นรูปปิด จะไม่สามารถวาดได้

1.3.7 ในการออกแบบรูปต้นแบบนั้นจะต้องอยู่ในขอบเขตที่กำหนดเพื่อให้เครื่องต้นแบบสามารถทำงานได้อย่างมีประสิทธิภาพ

1.3.8 การออกแบบเครื่องจักรจะออกแบบบนพื้นฐานที่ไม่สามารถหาครุภัณฑ์ใหม่ได้ทั้งหมด จึงเป็นการออกแบบเพื่อให้เครื่องจักรที่จัดสร้างขึ้นสามารถทำงานได้ดีที่สุดตามอุปกรณ์ควบคุมที่สามารถจัดหาได้จากห้องปฏิบัติการเท่านั้น

1.4 สถานที่ทำงานวิจัย

อาคารเครื่องมือวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยเทคโนโลยีสุรนารี

1.5 ประโยชน์ที่คาดว่าจะได้รับ

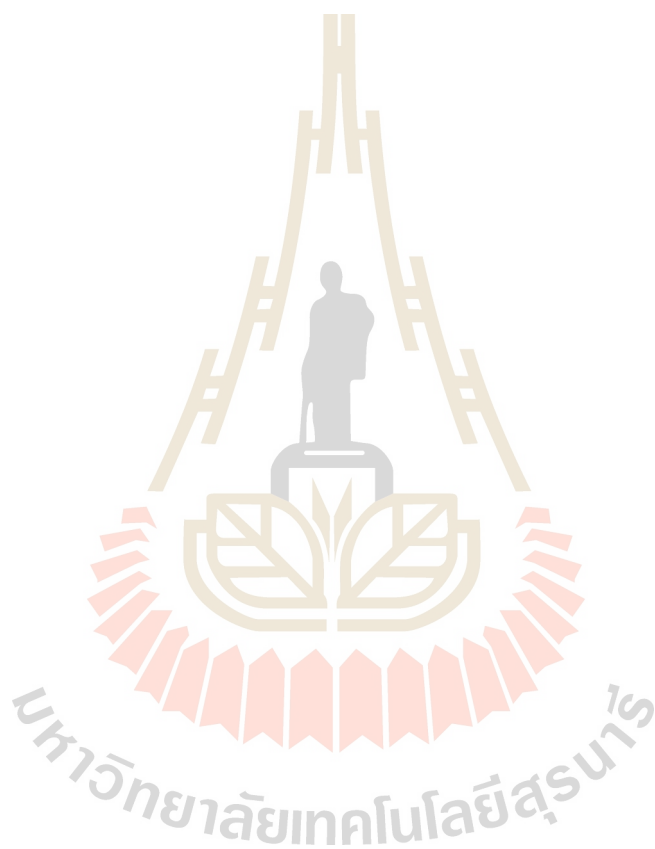
1.5.1 ช่วยเรื่องการออกแบบสำหรับนักออกแบบหรือบุคคลใดก็ตามที่ไม่มีความสามารถด้านการใช้โปรแกรมออกแบบ โดยที่จะทำให้ไม่ต้องออกแบบผ่านโปรแกรมออกแบบ ทำได้โดยออกแบบโดยใช้กระดาษและปากการ่วมกับโปรแกรมประมวลผลภาพที่จัดสร้างขึ้นมา

1.5.2 ได้ต้นแบบโปรแกรมการมองเห็นของเครื่องจักรเพื่อสร้างระบบการประมวลผลด้วยภาพสำหรับการออกแบบ

1.5.3 ได้เครื่องต้นแบบระดับห้องปฏิบัติการ ที่ใช้ในการติดตามชิ้นงาน โดยเพิ่มความสามารถในการทำงานของหุ่นยนต์ด้วยการใช้ระบบการมองเห็นของเครื่องจักร และสามารถ

สร้างชิ้นงานเสมือนกับชิ้นงานต้นแบบ โดยอาศัยการทำงานของหุ่นยนต์ที่ได้สร้างขึ้นได้อย่างเต็มประสิทธิภาพ

1.5.4 ได้แนวทางการนำโปรแกรมการมองเห็นของเครื่องจักรเพื่อสร้างระบบการประมวลผลด้วยภาพสำหรับการออกแบบเพื่อนำไปพัฒนาเพื่อใช้ร่วมกับเครื่องจักรอื่น ๆ ในอุตสาหกรรมต่อไป



บทที่ 2

ปริทัศน์วรรณกรรมและทฤษฎีที่เกี่ยวข้อง

2.1 กล่าวนำ

ในบทนี้จะเป็นการกล่าวถึงปริทัศน์วรรณกรรมที่เกี่ยวข้องกับงานวิจัยฉบับนี้ และยังคงกล่าวถึงความเป็นมาของการพัฒนาเทคโนโลยีในด้านของความรู้ทั่วไปเกี่ยวกับงานวิจัย โดยแบ่งทฤษฎีออกเป็นสองส่วนตามการทำงานของงานวิจัย

1. การประมวลผลภาพ และการมองเห็นเครื่องจักร
 - a. ภาพดิจิทัล
 - b. การแปลงภาพดิจิทัลเป็นพิกเซล
 - c. Spline interpolation
 - d. ซอฟต์แวร์ MATLAB
2. ทฤษฎีหุ่นยนต์เบื้องต้น และ Planar Robot
 - a. การหาสมการจลนศาสตร์ของหุ่นยนต์แขนกลระนาบสองมิติ
 - b. ไมโครคอนโทรลเลอร์ (Microcontroller)
 - c. การควบคุมการทำงานของสเตปเปอร์มอเตอร์ด้วย Microcontroller

ในภาคอุตสาหกรรมซึ่งมีการค้นคว้าและวิจัยในด้านนี้มาเป็นระยะเวลานานมาก และได้เริ่มเข้ามามีบทบาทแพร่หลายในภาคอุตสาหกรรมประกอบกับอุปกรณ์ด้านการมองเห็นของเครื่องจักร จึงได้รับการวิจัยอย่างต่อเนื่องเสมอมา

2.2 การประมวลผลภาพ และการมองเห็นเครื่องจักร

2.2.1 ภาพดิจิทัล (Digital Image)

Boyle, W.S., and Smith, G.E. (1970) ใน ขณะ ที่ ได้ ทำงาน วิจัย ใ้ กับ Bell Laboratories ประเทศสหรัฐอเมริกา ได้ออกแบบและสร้างอุปกรณ์เซนเซอร์ที่ทำหน้าที่รับแสงและให้ผลออกมาเป็นรูปได้ครั้งแรก โดยเซนเซอร์รับแสงแบบแรกของโลกเป็นแบบ Charge Coupled Device (CCD) หลังจากนั้นก็เป็นจุดเริ่มของการพัฒนากล้องดิจิทัล แม้ว่าในช่วงแรกของการสร้างกล้องดิจิทัล โลกตกอยู่ภายใต้สภาวะของสงครามเย็น ทำให้กล้องส่วนใหญ่ได้พัฒนามาเพื่อ

การทหารเป็นหลัก โดยกล้องดิจิตอลรุ่นแรก ๆ นั้นได้รับการออกแบบให้ทำงานในอวกาศ เพื่อติดตั้งบนดาวเทียม สอดแนมของประเทศสหรัฐอเมริกาอย่างไรก็ตามการพัฒนาของกล้องดิจิตอลก็เหมือนกับอุปกรณ์เครื่องใช้หลาย ๆ แบบที่พบเห็นทั่วไปทุกวันนี้ นั่นคือมีจุดกำเนิดและการพัฒนาเพื่อผลประโยชน์ทางการทหาร จากนั้นก็นำมาสู่การใช้งานในชีวิตประจำวัน และยังเข้าสู่ภาคประชาชนทั่วไปมากขึ้นก็จะพบว่ามีความก้าวหน้าของเทคโนโลยีสูงขึ้น ในขณะที่ราคาของอุปกรณ์นั้นลดลงเป็นอย่างมาก จะเห็นได้จากกล้องดิจิตอลทุกวันนี้มีให้พบเห็นได้ทั่วไป พร้อมทั้งให้รายละเอียดของภาพสูงมากแต่มีราคาต่ำเป็นอย่างมาก

S.B. Dworkin, T.J. Nye (2006) ได้นำเทคโนโลยีการมองเห็นเครื่องจักร machine vision มาใช้ในการวัดขนาดของชิ้นส่วนที่มีความร้อนในระหว่างการอัดขึ้นรูป ซึ่งทำได้ยากเนื่องจากทำในขณะที่ชิ้นงานยังมีความร้อนสูง รวมถึงมีปัญหาเรื่องความสว่างของสถานที่ทำงาน ดังนั้นเทคโนโลยีการมองเห็นเครื่องจักรจึงถูกนำมาใช้ในอุตสาหกรรมจำนวนมากเพื่อความรวดเร็วในกระบวนการวัดแบบไม่สัมผัสชิ้นงาน โดยมุ่งเน้นไปที่การทดลองควบคุมการประมวลผลภาพของการอัดขึ้นรูปแบบร้อน (hot forgings) โดยเฉพาะในการอัดขึ้นรูปด้วยแม่พิมพ์ชนิดเปิด (Open-Die Forging) ขนาดใหญ่ และวิธีการใช้งานของกล้อง CCD ราคาไม่แพงร่วมกับฟิลเตอร์อินฟราเรดในการประมวลผลภาพและวิเคราะห์รูปแบบงานสำหรับระบบการวัดการมองเห็นเครื่องจักร ผลการทดลองพบว่าการใช้กล้อง CCD ธรรมดาพร้อมกับฟิลเตอร์อินฟราเรด ให้ผลการวัดที่มีความถูกต้องและความน่าเชื่อถือสำหรับการวัดขนาดของชิ้นส่วนร้อน

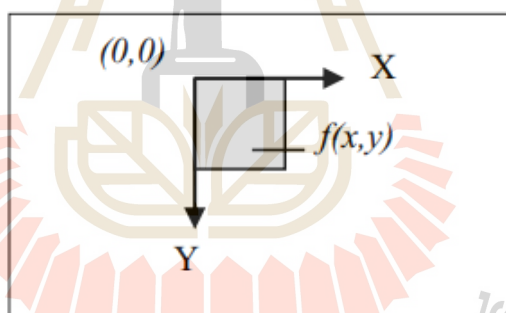
Pradyumna Deshpande (2012) งานวิจัยนี้ได้กล่าวไว้ว่าภาพดิจิตอลนั้นมีการเติบโตและพัฒนาอย่างรวดเร็ว จึงเป็นเรื่องที่ยากมากสำหรับผู้ชมที่จะสามารถตัดสินความถูกต้องของภาพนั้น ๆ ดังนั้นการโอนข้อมูลหรือถ่ายภาพดิจิตอลเพื่อใช้เป็นหลักฐานทางกฎหมายนั้น ปัญหาหลักคือจะต้องทำการตรวจสอบความถูกต้องของภาพ เทคนิคในการจัดหมวดหมู่ของการปลอมแปลงภาพจึงแบ่งเป็นสองประเภท ได้แก่ Copy-Move Forgery และ Fast-Copy-Move detection จากงานวิจัยได้ผลว่าวิธีที่สองมีประสิทธิภาพมากกว่าวิธีการแรก แม้จะต้องใช้เวลาในการแก้ปัญหา ทั้งการหมุนภาพการกำจัดตัวแปรภายนอกที่เสียเวลามากกว่า แต่ก็เพิ่มอัตราการตรวจจับมากขึ้น นอกจากนี้ สามปัจจัยสำคัญที่ส่งผลให้เกิดความผิดพลาดคือ การบีบอัดภาพ การตัดแปลงภาพด้วยเสียง

Muzamil Bhat (2014) งานวิจัยนี้ได้กล่าวไว้ว่าปัจจุบัน โลกทุกวันนี้มีการใช้ข้อมูลภาพมากขึ้น อาทิเช่น ภาพถ่ายจากดาวเทียม อุปกรณ์วัดต่าง ๆ เซนเซอร์ซึ่งเปรียบเสมือนตาที่มองมายังโลก ซึ่งรูปภาพเหล่านี้มีความสำคัญมากเพราะข้อมูลจะต้องถูกส่งมาประมวลผลที่สถานีภาคพื้น ดังนั้นการที่จับภาพและถ่ายโอนมาจะทำให้ง่ายและสะดวก แล้วยังลดค่าใช้จ่ายที่มีอยู่อย่างมากมากกว่าการใช้รูปแบบธรรมดาอีกด้วย ข้อดีคือ ความคล่องตัวในการทำซ้ำ และเก็บรักษาข้อมูลเดิม ความแม่นยำ การประมวลผลภาพได้เร็วขึ้น การส่งภาพดิจิตอลมีความสะดวกและง่ายมากเมื่อ

เทียบกับการส่งภาพแบบธรรมดา การประมวลผลภาพสามารถใช้งานได้ในระยะยาวยิ่งขึ้นเมื่อใช้ร่วมกับซอฟต์แวร์ที่รองรับระบบการประมวลผลภาพ

นิยามของภาพดิจิทัล (Definition of a Digital Image) ภาพดิจิทัลจะเป็นการจัดเรียงค่าต่อกันในสองมิติ (2-D Array) โดยค่าที่แสดงคือค่าความเข้มของแสงที่ตำแหน่งนั้น เนื่องจากงานวิจัยนี้เน้นไปที่กระบวนการประมวลผลภาพดิจิทัล (Image Processing) ดังนั้นคำว่า “ภาพ” (Image) ที่ใช้ในเอกสารนี้จะหมายถึง ภาพดิจิทัลเท่านั้นภาพจะเป็นฟังก์ชันของความเข้มของแสงที่เปลี่ยนไปตามตำแหน่งของแกน x และ y ในระนาบสองมิติ หรือเขียนเป็นคณิตศาสตร์จะเขียนได้ว่า ภาพคือฟังก์ชันที่เปลี่ยนแปลงไปตามตำแหน่ง x และ y หรือ $f(x, y)$

โดย $f(x, y)$ จะเป็นความสว่าง (Brightness) ของจุด (x, y) ซึ่ง x และ y จะแทนพิกัดในระนาบของส่วนเล็ก ๆ ของรูปภาพ ซึ่งจะใช้คำว่า Picture Element หรือเขียนย่อเป็นพิกเซล (Pixel) สำหรับข้อตกลงมาตรฐานจุดอ้างอิงของพิกเซลจะเริ่มด้วยพิกัด $(0,0)$ จะอยู่ที่ตำแหน่งบนซ้ายของรูปภาพและแกน x จะเป็นแกนในแนวนอน ค่าเพิ่มจากซ้ายไปขวา สำหรับค่า y จะเป็นแกนในแนวตั้งและจะมีค่าเพิ่มขึ้นจากบนลงล่าง ดังรูปที่ 2.1

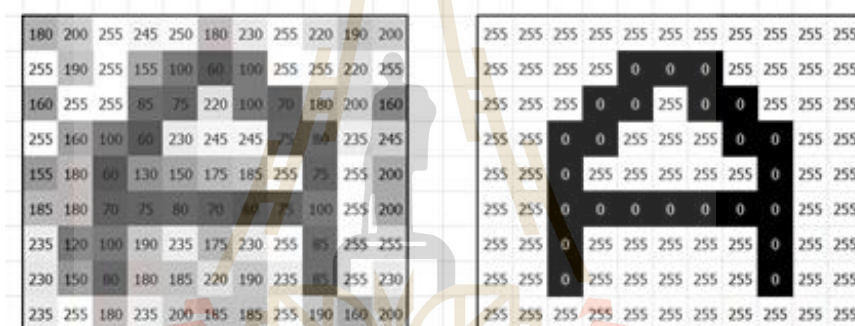


รูปที่ 2.1 แสดงการกำหนดตำแหน่งของพิกเซล

ในการประมวลผลภาพดิจิทัล (Digital Image Processing) เซนเซอร์รับภาพ (Image Sensor) จะทำหน้าที่เปลี่ยนรูปภาพตามธรรมชาติเป็นจำนวนเต็มหน่วย (Discrete Number) สำหรับเป็นค่าของในแต่ละพิกเซล โดยเซนเซอร์รับภาพ จะกำหนดค่าตัวเลขแสดงตำแหน่งให้กับแต่ละพิกเซลพร้อมทั้งกำหนดค่าความสว่างให้กับพิกเซลนั้นด้วย ซึ่งค่าความสว่างอาจจะเป็นระดับสีเทา (Gray Level) ในกรณีที่ภาพที่เก็บมาเป็นภาพขาวดำ และอาจกำหนดเป็นค่าระดับสี (Color Value) ในกรณีที่ภาพที่เก็บเป็นภาพสี

2.2.2 การแปลงภาพดิจิทัลเป็นพิกเซล

การแปลงภาพดิจิทัลเป็นพิกเซลสามารถทำได้โดยการปรับค่าพิกเซลของภาพด้วยการ Thresholding คือ ภาพที่ได้จากการผ่านวิธีการ smoothing แล้ว บางครั้งอาจมีการปรากฏขอบภาพซึ่งอาจจะไม่ใช่ขอบของภาพที่แท้จริง เนื่องจากการมีสัญญาณรบกวนหรือภาพดังกล่าวมีลักษณะของรายละเอียดค่อนข้างมาก ดังนั้นเพื่อลดปัญหาดังกล่าวจึงมีการกำหนดค่าขึ้นมาเพื่อแยกแยะลักษณะของขอบ ค่านั้นคือค่า threshold และจะแบ่งเป็น 2 ค่าคือ high threshold (T_1) และ low threshold (T_2) โดยพิกเซลที่มีค่ามากกว่า T_1 จะถูกปรับเป็น 255 (เป็นพิกเซลที่เป็นขอบ) แต่ถ้าน้อยกว่า T_2 จะถูกปรับเป็น 0 และค่าที่อยู่ระหว่างค่า T_1, T_2 จะปรับเป็นค่า 255 หรือ 0 นั้น ขึ้นอยู่กับพิกเซลที่อยู่รอบข้างหากพบว่าพิกเซลที่อยู่รอบข้างของพิกเซลที่เป็นขอบ (ค่า $> T_1$) มีค่ามากกว่า T_2 แล้วจะปรับค่าพิกเซลดังกล่าวให้มีค่าเป็น ดังรูปที่ 2.2



รูปที่ 2.2 แสดงการแปลงพิกเซล Thresholding

2.2.3 Spline Interpolation

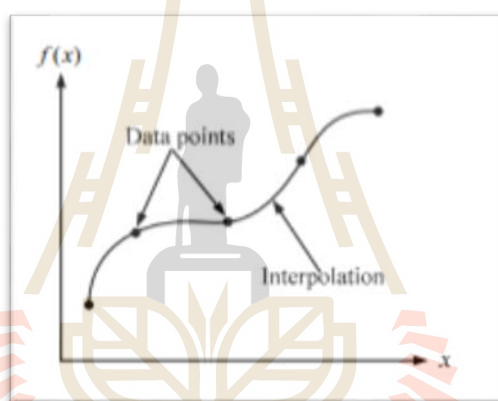
Shafeeque Ahmed (2015) งานวิจัยนี้ได้ทำการศึกษาเกี่ยวกับการคำนวณทางด้านการตลาดของหน่วยงานสำคัญ ๆ ต่าง ๆ โดยหุ้่นถูกนำมาใช้เป็นดัชนีในการคำนวณตลาด โดยงานวิจัยนี้เน้นไปที่ความรวดเร็วในการทำงานและความถูกต้องของการคำนวณ โดยได้ใช้ Cubic Splines เป็นเทคนิคที่สอดแทรกค่าต่าง ๆ ของตลาด โดยการนำ Cubic Spline มาคำนวณชุดข้อมูลที่เกิดขึ้นว่าจะมีแนวโน้มไปทิศทางใด และจำลองทิศทางความเป็นไปได้ของตลาดออกมา วิธีการ Cubic Splines นี้จะช่วยบรรลู่วัตถุประสงค์ของนักลงทุน โดยผลงานวิจัยนี้ได้แนวทางการลงทุนเพื่อแนะนำการลงทุนโดยงานวิจัยที่จำลองออกมานั้นทำบนพื้นฐานซอฟต์แวร์ MATLAB 2012b

เมื่อพิจารณาปัญหาทางวิศวกรรม รวมถึงการทดลองทางวิทยาศาสตร์ต่าง ๆ เราจะพบว่าข้อมูลที่เรารู้ได้โดยส่วนใหญ่ จะเป็นจุดไม่ต่อเนื่อง (discrete points) เราอาจมีความต้องการวาดเส้นกราฟต่อเนื่องผ่านจุดเหล่านั้น (to fit a curve over those data points) เพื่อเป็นการอธิบายข้อมูล

หรือต้องการทราบค่าที่อยู่ระหว่างจุดของข้อมูลที่เรามี ในทางวิศวกรรมจะอาศัยเทคนิคที่เรียกว่า curve fitting ซึ่งสามารถกระทำได้ 2 แนวทาง คือ

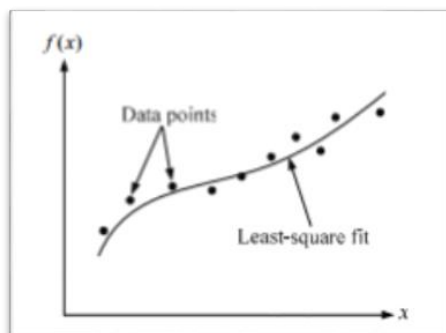
1. Interpolation
2. Regression

วิธีการ Interpolation เราจะลากเส้นกราฟ (ซึ่งก็คือ ฟังก์ชัน) ผ่านจุดทุกจุดของข้อมูลที่มี ดังรูปที่ 2.3 เทคนิคนี้เหมาะสำหรับจุดข้อมูลที่เราทราบ หรือแน่ใจว่ามีความถูกต้องสูง เช่น ข้อมูลที่ได้จากการคำนวณที่ซับซ้อน หรือได้จากการทดลองตามวิธีมาตรฐานด้วยเครื่องมือวัดที่มีความถูกต้องสูง (ซึ่งมีค่าใช้จ่ายในการวัดสูงด้วย ซึ่งเป็นสาเหตุหนึ่งที่ไม่สามารถวัดออกมาได้หลายๆ จุด) เป็นต้น โดยมีเทคนิคการ Interpolation หลายวิธี อาทิเช่น Newton's Interpolating Polynomials, Lagrange Interpolating Polynomials แต่ในงานวิจัยนี้จะเน้นไปที่วิธีการแบบ Splines



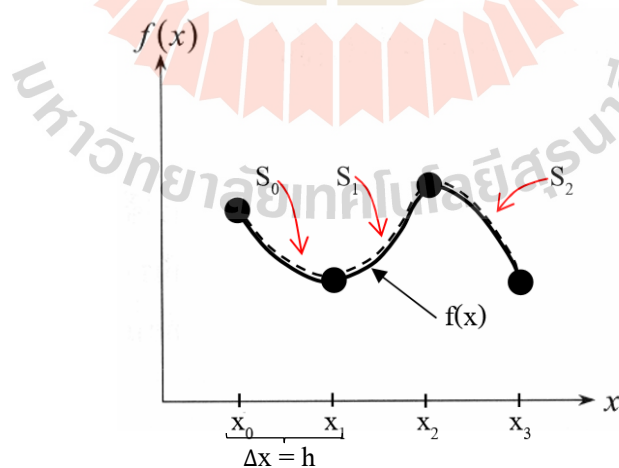
รูปที่ 2.3 แสดงกราฟของเทคนิคการ Interpolation

วิธีการ Regression เราจะสร้างเส้นกราฟ หรือลาก Curve ให้ผ่านจุดข้อมูลที่เรามีให้มากที่สุด โดยเส้นกราฟที่สร้างขึ้นจะเป็นตัวแทนของข้อมูล บอกว่าข้อมูลมีความสัมพันธ์แบบใด หรือมีแนวโน้มจะเปลี่ยนแปลงไปอย่างไรเมื่อปริมาณในแกนหนึ่งเพิ่มขึ้น เทคนิคนี้จะเหมาะกับข้อมูลที่ได้จากการทดลอง (experimental data) โดยเฉพาะข้อมูลจากการทดลองภาคสนาม (ซึ่งมีความแปรปรวนได้มาก) เราอาจรู้จักเทคนิคนี้ในชื่อ Least Square Regression โดยเทคนิคนี้แสดงในรูปที่ 2.4



รูปที่ 2.4 แสดงกราฟของเทคนิค Least Square Regression

งานวิจัยนี้ขอกล่าวถึงวิธีการ Cubic Splines เป็นการประมาณค่าในช่วงด้วยเส้นโค้ง (ประเภท Splines) ความต่อเนื่องที่ node แบบต่าง ๆ หลักการของ Splines ก็คือ เราจะแทนจุดข้อมูล $n + 1$ จุด ด้วย Splines จำนวน n เส้น ซึ่งเป็นการประมาณค่าที่ให้ความโค้งของเส้นกราฟมีความต่อเนื่องและความใกล้เคียงกับสมการหรือฟังก์ชัน $f(x)$ จริงมากที่สุด เมื่อเทียบกับ Linear Splines และ Quadratic Splines เนื่องด้วยเป็นการประมาณค่าที่ใช้อนุพันธ์ลำดับที่สอง (Second derivatives) ดังรูปที่ 2.5 โดยในการทำการประมาณค่าในช่วงนั้นจะประกอบไปด้วยจุดข้อมูลที่ไม่ต่อเนื่องกัน $N+1$ จุด และเส้นโค้ง Spline ทั้งหมด N เส้น โดยในการหาสมการของเส้นหาได้ดังสมการที่ 2.1



รูปที่ 2.5 กราฟ Cubic Splines

$$S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3 \quad (2.1)$$

โดยที่ $0 \leq j \leq N-1$, $S_j(x)$ เป็น Splines ของช่วง (x_j, x_{j+1}) และ a_j, b_j, c_j และ d_j เป็นตัวแปรไม่ทราบค่า

วิธีการหาตัวแปรไม่ทราบค่ามีขั้นตอนดังนี้

1. กำหนดให้สัมประสิทธิ์ $a_j = y_j$

2. คำนวณหา c_j ได้จากสมการที่ 2.2 และการกำหนดเงื่อนไขขอบ (Boundary condition) เป็นลักษณะปล่อยอิสระ (Free or natural boundary) หรือจับยึด (Clamped boundary) ซึ่งในแต่ละเงื่อนไขใช้ลักษณะการคำนวณที่ต่างกัน ตารางที่ 2.1

$$[A]\{c_j\} = \{b\} \quad (2.2)$$

ตารางที่ 2.1 เงื่อนไขขอบ

Natural spline	Clamped spline
$A = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & \dots & 0 \\ 0 & h_1 & 2(h_1+h_2) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \dots & 0 & 0 & 1 \end{bmatrix}$	$A = \begin{bmatrix} 2h_0 & h_0 & 0 & \dots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & \dots & 0 \\ 0 & h_1 & 2(h_1+h_2) & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \dots & 0 & h_{n-1} & 2h_{n-1} \end{bmatrix}$
$b = \begin{bmatrix} 0 \\ \frac{3}{h_1}(a_2 - a_1) - \frac{3}{h_0}(a_1 - a_0) \\ \vdots \\ \frac{3}{h_{n-1}}(a_n - a_{n-1}) - \frac{3}{h_{n-2}}(a_{n-1} - a_{n-2}) \\ 0 \end{bmatrix}$	$b = \begin{bmatrix} \frac{3}{h_0}(a_1 - a_0) - 3f'(a) \\ \frac{3}{h_1}(a_2 - a_1) - \frac{3}{h_0}(a_1 - a_0) \\ \vdots \\ \frac{3}{h_{n-1}}(a_n - a_{n-1}) - \frac{3}{h_{n-2}}(a_{n-1} - a_{n-2}) \\ 3f'(b) - \frac{3}{h_{n-1}}(a_n - a_{n-1}) \end{bmatrix}$

3. คำนวณหาสัมประสิทธิ์ b_j จากสมการที่ 2.3

$$b_j = \frac{c_{j+1} - c_j}{3h_j} \quad (2.3)$$

4. คำนวณหาสัมประสิทธิ์ d_j จากสมการที่ 2.4

$$d_j = \frac{a_j - a_{j-1}}{h_{j-1}} + \frac{h_{j-1}}{3}(c_{j-1} + 2c_j) \quad (2.4)$$

ในงานวิจัยต่าง ๆ ได้มีการนำวิธีการนี้ไปใช้ในการสร้างรูปร่างต่าง ๆ และมีฟังก์ชัน spline สร้างเส้นกราฟใน โปรแกรม MATLAB ด้วย ดังตัวอย่างบทความของ Ticiano Monte Lúcioda Silva, Mateus Bonamigo Zupiroli, Rafael Pereira and Antônio C.P. Brasil Junior (2007) ได้นำเสนอการหาพารามิเตอร์สำหรับการหารูปร่างที่มีค่าทางอากาศพลศาสตร์ดีที่สุดของแพนอากาศ ซึ่งใช้วิธีการดังกล่าวในการสร้างรูปร่างแพนอากาศเพื่อนำมาเปรียบเทียบในกระบวนการหารูปร่างที่ดีที่สุด เป็นต้น

โดยนิยามของความต่อเนื่องกล่าวได้ ดังนี้

1. ในแบบแรกเส้นตรงสองเส้นวิ่งมาบรรจบกันที่จุดเดียวกัน หรือปลายเส้นทับกัน โดยเส้นทั้งสองอาจมีความชัน (Slope) ไม่เท่ากันที่จุดนั้นก็ได้ เรียกว่า Continuous at point หรือต่อเนื่องที่จุดแบบ 0th Derivatives

2. แบบที่สองนอกจากปลายเส้นทับกันแล้ว ความชันของทั้งสองเส้นที่วิ่งมาพบกันยังเท่ากันอีกแบบนี้เรียกว่า Continuous at slope หรือต่อเนื่องที่ 1st Derivatives ซึ่งในความเป็นจริงแล้ว แคนี่ในสายตามนุษย์ก็มองว่าเป็นเส้นแบบ smooth ใช้ได้แล้ว

3. แบบที่สามกล่าวคือนอกจากปลายเส้นทับกัน และมีความชันที่จุดปลายเท่ากันแล้ว อัตราการเปลี่ยนแปลงความชันของแต่ละเส้นที่วิ่งเข้าหา node ยังเท่ากันอีก จึงกล่าวได้ว่าเป็นความต่อเนื่องที่ smooth มาก ๆ เรียกว่า continuous of 2nd Derivatives

2.2.4 ซอฟต์แวร์ MATLAB

Frans W. Cornelissen (2002) งานวิจัยนี้ได้กล่าวถึงฟังก์ชันในการติดตามชิ้นงาน ฟังก์ชันหนึ่งในซอฟต์แวร์ MATLAB โดยจะกล่าวถึงสองฟังก์ชันหลักได้แก่ กล้องเครื่องมือ Psychophysics และกล้องเครื่องมือ Eyelink โดยกล้องเครื่องมือ Psychophysics คือการเขียนโปรแกรมแสดงผลภาพด้วยภาษาที่มีความสามารถในการอินเตอร์เฟสกับโปรแกรมอื่น ๆ ได้ ส่วนกล้องเครื่องมือ Eyelink คือการติดตามด้วยวิดีโอ เป็นการเคลื่อนไหวของอุปกรณ์จับภาพตามการดำเนินงานที่กำลังปฏิบัติ โดยผลของงานวิจัยนี้ก็เป็นที่คาดหวังไว้ว่าประสิทธิภาพการทำงานด้วยกล้องเครื่องมือ Psychophysics และ Eyelink มีประสิทธิภาพและรวดเร็วการทำงานด้วยตาของมนุษย์

Bhavana C. Bendale (2015) งานวิจัยนี้ได้อธิบายเกี่ยวกับการติดตามวัตถุที่เคลื่อนที่ด้วยการลำดับจากวิดีโอ วิธีการนี้จะดำเนินโดยใช้ทฤษฎีการไหลของแสง (Horn-Schunck) ด้วยซอฟต์แวร์ MATLAB โดยความหลากหลายของการใช้งานอยู่ที่การปฏิสัมพันธ์ระหว่างมนุษย์และคอมพิวเตอร์ สามารถนำความรู้จากการติดตามวัตถุเคลื่อนไหวผ่านวิดีโอไปใช้ในงานต่าง ๆ ได้ อาทิเช่น ความปลอดภัย การเฝ้าระวัง การสื่อสารผ่านวิดีโอ การควบคุมจราจร การถ่ายภาพทางการแพทย์ เป็นต้น โดยผลสรุปกล่าวไว้ว่า การติดตามผ่านวิดีโอ นั้นสามารถทำได้โดยไม่ต้องทำการตัดภาพพื้นหลังออก เพราะการตัดภาพพื้นหลังออกจากเฟรมภาพนั้นอาจจะก่อให้เกิด Thresholding ได้ ซึ่งจะก่อให้เกิดความสับสนของภาพได้ โดยในงานวิจัยนี้หวังว่าอนาคตจะพัฒนาไปจนถึงระดับการทราบถึงความแตกต่างของวัตถุแบบ Real-time ซึ่งจะสามารรถใช้ในการดำเนินการหลีกเลี่ยงอุปสรรคต่าง ๆ ในอนาคตได้

2.3 ทฤษฎีหุ่นยนต์เบื้องต้น และ Planar Robot

Rajkumar Jain (2008) งานวิจัยนี้อธิบายถึงการนำเสนอการวางแผนเส้นทางวิถีปลายของหุ่นยนต์พื้นที่ระนาบ โดยไม่ต้องใช้อุปกรณ์บังคับวิถีช่วย เพื่อให้ส่วนฐานไม่ต้องรับแรงใด ๆ จากส่วนอื่น โดยงานวิจัยนี้ได้เลือกกราฟบอนด์มาใช้ในการสร้างแบบจำลอง และพัฒนาการควบคุมต่าง ๆ โดยปรับเปลี่ยนกระบวนการทางกายภาพ แรงจูงใจในงานวิจัยนี้คือ การพัฒนาเส้นทางวิถีปลายของหุ่นยนต์ให้มีค่าน้อยที่สุด เพื่อไปถึงเป้าหมายได้เร็วที่สุด งานวิจัยนี้ได้เลือกใช้ wheel เข้ามาช่วยคำนวณเพราะการคำนวณแบบนี้ทำให้ได้ค่าที่สามารถกำหนดได้ และวิธีนี้ยังพัฒนาที่จะย้ายปลายจากตำแหน่งเริ่มต้นเพื่อกำหนดเป้าหมายใหม่ ๆ ต่อไป ผลคืองานวิจัยนี้สามารถสร้างเส้นทางวิถีปลายที่สั้นและสามารถขยายขอบเขตแนวคิดสำหรับประยุกต์ใช้กับหุ่นยนต์ที่มีพื้นที่การทำงานมากกว่าสององศาอิสระได้

Mahmoud Gouasmi (2012) ทำการศึกษาเกี่ยวกับการจำลองการทำงานของหุ่นยนต์แขนกลระนาบสองมิติ โดยใช้ซอฟต์แวร์ MATLAB ในการจำลองการทำงานขึ้น ร่วมกับหาพื้นที่การทำงานและเส้นทางการทำงานของหุ่นยนต์แขนกลระนาบสองมิตินี้ขึ้นมา ซึ่งก่อนจะสามารถทราบถึงพื้นที่การทำงานและเส้นทางการทำงานของหุ่นยนต์ชนิดนี้ จะต้องทราบถึงสมการจลนศาสตร์แบบไปข้างหน้าและย้อนกลับของหุ่นยนต์ โดยในงานวิจัยอธิบายวิธีการหาสมการอย่างละเอียด โดยผลจากงานวิจัยทำให้สามารถทราบถึงการนำไปใช้งานของหุ่นยนต์ชนิดนี้ได้โดยไม่ต้องสร้างหุ่นยนต์ต้นแบบจริงขึ้นมา สามารถแสดงผลการทำงานผ่านหน้าจอการจำลองการทำงานของหุ่นยนต์แขนกลระนาบสองมิติได้

ประวัติความเป็นมาของหุ่นยนต์

หุ่นยนต์เป็นเทคโนโลยีที่มีใช้กันอย่างแพร่หลายในโรงงานอุตสาหกรรมการผลิต เพราะความสามารถที่ทำงานได้คล้ายกับมนุษย์และไม่รู้จักเหน็ดเหนื่อย ในปัจจุบันนั้นได้มีการพัฒนาหุ่นยนต์กันอย่างต่อเนื่องจนสามารถตอบสนองความต้องการของมนุษย์พร้อมกับแสดงปฏิกิริยาตามโปรแกรมที่ป้อนไว้เพื่อให้มนุษย์เข้าใจได้ หุ่นยนต์หรือที่เรียกกันว่าโรบอท (Robot) เริ่มต้นในปี ค.ศ. 1760 ได้มีการนำเอาเครื่องจักรไอน้ำเข้ามาทำงานแทนคนงานในโรงงานและผลที่ได้คือเครื่องจักรทำงานได้ดีกว่ามนุษย์จากนั้นเรื่องก็เงียบมาจนถึงปี ค.ศ. 1920 ได้มีนักเขียนละครชาวเชคโกสโลวาเกียชื่อ Karel Capek เขียนบทละครเรื่อง Rossum's Universal Robots : R.U.R โดยคำว่า Robot เป็นคำแผลงมาจากคำว่า Robota ในภาษาเช็กที่แปลว่า Worker หรือคนงานต่อมาในปี ค.ศ. 1942 ได้มีนักประพันธ์ทางด้านวิทยาศาสตร์ Isaac Asimov ได้บัญญัติกฎของหุ่นยนต์ขึ้น 3 ข้อคือ

1. หุ่นยนต์ห้ามทำร้ายมนุษย์หรือปล่อยให้มนุษย์ได้รับอันตราย
2. หุ่นยนต์ต้องเชื่อฟังคำสั่งของมนุษย์เว้นแต่จะขัดแย้งกับกฎข้อ 1
3. หุ่นยนต์ต้องป้องกันตัวเองโดยไม่ขัดแย้งกับกฎข้อหนึ่งและข้อ 2

ถัดมาในปี ค.ศ. 1954 ได้มีการสร้างหุ่นยนต์ตัวแรกของโลกโดย George C. Devol วิศวกรชาวอเมริกาพร้อมกับจดสิทธิบัตรและจากนั้นก็ได้มีการพัฒนาเทคโนโลยีหุ่นยนต์เรื่อย ๆ จนถึงปัจจุบัน

โครงสร้างของหุ่นยนต์

โครงสร้างของหุ่นยนต์สามารถแบ่งออกเป็น 2 ส่วนสำคัญคือส่วนลำตัวและแขน (Body-and - Arm) และส่วนข้อมือ (Wrist) โดยมากแล้วส่วนลำตัวและแขนจะมีระดับขั้นความเสรี 3 ระดับและในส่วนข้อมือจะมีระดับขั้นของความเสรีอยู่ 2 – 3 ระดับที่ปลายของข้อมือจะเป็นวัตถุซึ่งมีความสัมพันธ์กับงานที่หุ่นยนต์ต้องทำ ตัวอย่างเช่น วัตถุอาจจะเป็นชิ้นงานที่ต้องการโหลดเข้าเครื่องจักรหรืออาจจะเป็นเครื่องมือที่หุ่นยนต์ต้องใช้ในการกระบวนการผลิตบางอย่าง เป็นต้น ส่วนลำตัวและแขนของหุ่นยนต์จะใช้ในการจัดตำแหน่งที่ถูกต้องของวัตถุและส่วนข้อมือของหุ่นยนต์จะใช้สำหรับจัดทิศทางการวางตัวที่เหมาะสมให้แก่วัตถุเพื่อที่จะวางตำแหน่งของวัตถุให้เหมาะสม ส่วนลำตัวและแขนของหุ่นยนต์จะต้องสามารถเคลื่อนวัตถุได้ทิศทางทั้งสามดังต่อไปนี้คือ

1. การเคลื่อนที่ในแนวตั้ง (การเคลื่อนที่ในแนวแกน Z)
2. การเคลื่อนที่ในแนวรัศมี (เข้าหรือออกหรือการเคลื่อนที่ในแนวแกน Y)
3. การเคลื่อนที่จากซ้ายไปขวา (การเคลื่อนที่ในแนวแกน X)

การที่จะทำให้หุ่นยนต์สามารถเคลื่อนที่ในลักษณะข้างต้นได้มีหลายวิธีทั้งนี้ขึ้นอยู่กับชนิดของข้อต่อที่ใช้ในการสร้างลำตัวและแขนของหุ่นยนต์ซึ่งจะกล่าวถึงรายละเอียดต่อไป เพื่อที่จะทำให้เกิดการจัดทิศทางเหมาะสมของวัตถุเราสามารถกำหนดระดับขั้นความเสรี 3 ระดับ สำหรับข้อมือของหุ่นยนต์ ตัวอย่างต่อไปนี้เป็นหนึ่งรูปแบบในการประกอบข้อมือของหุ่นยนต์เพื่อทำให้เกิดเป็น 3 ลำดับขั้นความเสรี (3 Degree of Freedom)

1. Roll ลำดับขั้นความเสรีลักษณะนี้ทำได้โดยใช้ข้อต่อแบบ T เพื่อที่จะหมุนวัตถุรอบแกนของแขน

2. Pitch เกี่ยวกับการหมุนขึ้นลงของวัตถุซึ่งทำได้โดยข้อต่อแบบ R (Rotational Joint)

3. Yaw เกี่ยวกับการหมุนซ้ายขวาของวัตถุซึ่งทำได้โดยใช้ข้อต่อแบบ R (Rotational Joint)

ส่วนประกอบของหุ่นยนต์

1. Actuator เป็นอุปกรณ์หรือชุดขับเคลื่อนเพื่อให้แขนกลหรือหุ่นยนต์เคลื่อนไหวได้แก่ มอเตอร์และกระบอกสูบ เป็นต้น

2. Manipulators เป็นส่วนประกอบเพื่อช่วยเรื่องระยะการเคลื่อนที่ ซึ่งประกอบด้วย แขน (Links) และข้อต่อ (Joints)

3. End Effectors เป็นส่วนท้ายสุดของแขนกลไว้ใช้ในการทำงานต่าง ๆ เช่น มือจับ สิ่งของและอุปกรณ์ที่ใช้ในทางเทคนิคประเภทงานกัดกลึงเชื่อม เป็นต้น

4. Sensor เป็นอุปกรณ์ที่ใช้ตรวจสอบสถานะว่าตำแหน่งหรือข้อมูลที่ต้องการได้หรือยัง หรือมีค่าเท่าไรเทียบได้กับประสาทสัมผัสของมนุษย์

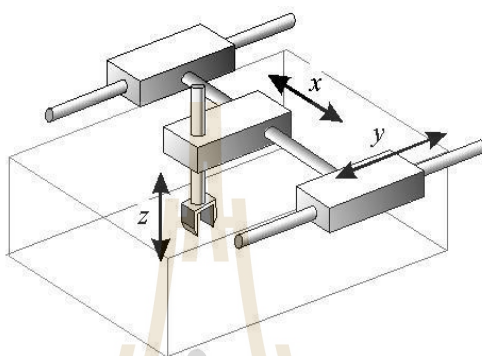
5. Power Supply เป็นแหล่งจ่ายพลังงานให้กับอุปกรณ์ต่าง ๆ ของแขนกลไม่ว่าจะเป็นกระแสไฟฟ้าหรือวาล์วที่ใช้ควบคุมกระบอกสูบ

6. Controller เปรียบเสมือนสมองของมนุษย์ที่คอยควบคุมขั้นตอนต่าง ๆ ในการทำงานของระบบแขนกล โดยสั่งการไปยังชุดขับเคลื่อนเพื่อให้เป็นไปตามโปรแกรมที่ตั้งไว้

7. Program เป็นการกำหนดขั้นตอนต่าง ๆ ของการทำงานเพื่อให้ได้งานตามที่ต้องการ

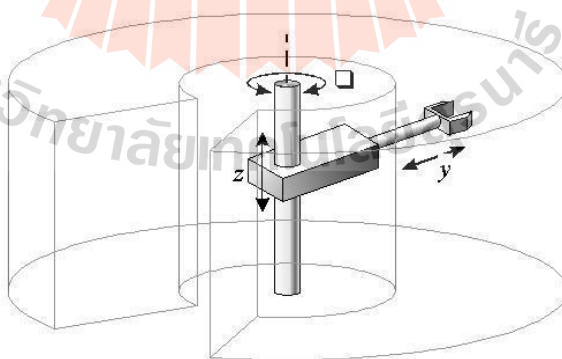
ชนิดของหุ่นยนต์

1. The Cartesian Robot เป็นหุ่นยนต์ที่มีการเคลื่อนที่เป็นแบบเชิงเส้นทั้ง 3 แกนมีลักษณะการทำงานที่ไม่ซับซ้อนสามารถทำความเข้าใจได้ง่าย แต่หุ่นยนต์ชนิดนี้มีขีดจำกัดเรื่องของพื้นที่ที่สามารถทำงานได้เพราะอาณาเขตการเคลื่อนที่เป็นแบบกล่องสี่เหลี่ยม ส่วนมากหุ่นยนต์ประเภทนี้จะนำไปใช้ในลักษณะเคลื่อนย้ายของหนักซึ่งหุ่นยนต์ประเภทนี้จะมีลักษณะดังรูปที่ 2.6



รูปที่ 2.6 ตัวอย่าง The Cartesian Robot

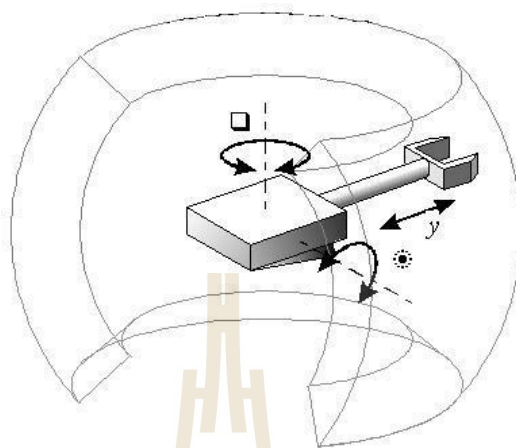
2. The Cylindrical Robot เป็นหุ่นยนต์ที่มีลักษณะของการทำงานแบบทรงกระบอก โดยการเคลื่อนที่นั้นจะเป็นเชิงเส้น 2 แกนและเคลื่อนที่เชิงมุม อีกแกนสามารถนำหุ่นยนต์ชนิดนี้ไปใช้ในลักษณะงานจับวางได้ ซึ่งหุ่นยนต์ประเภทนี้มีลักษณะดังรูปที่ 2.7



รูปที่ 2.7 ตัวอย่าง The Cylindrical Robot

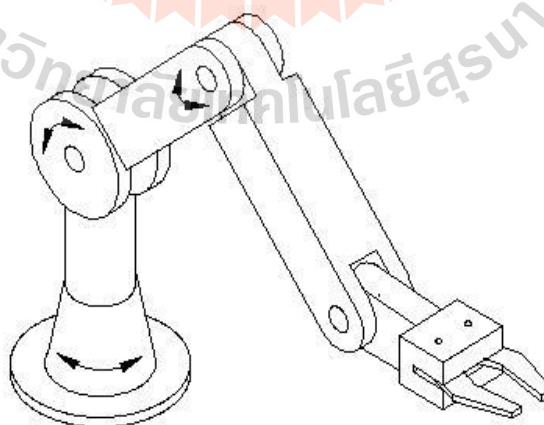
3. The Spherical Robot หุ่นยนต์ประเภทนี้เป็นหุ่นยนต์ที่มีลักษณะการเคลื่อนที่แบบเชิงเส้น 1 แกนกับการเคลื่อนที่เชิงมุมอีก 2 แกน ซึ่งสามารถเข้าถึงพื้นที่ที่มีความซับซ้อนได้ใน

ระดับหนึ่งแต่หุ่นยนต์ชนิดนี้จะไม่สามารถรับวัตถุหรือสิ่งของที่มีน้ำหนักมากได้ โดยหุ่นยนต์ประเภทนี้มีลักษณะดังรูปที่ 2.8



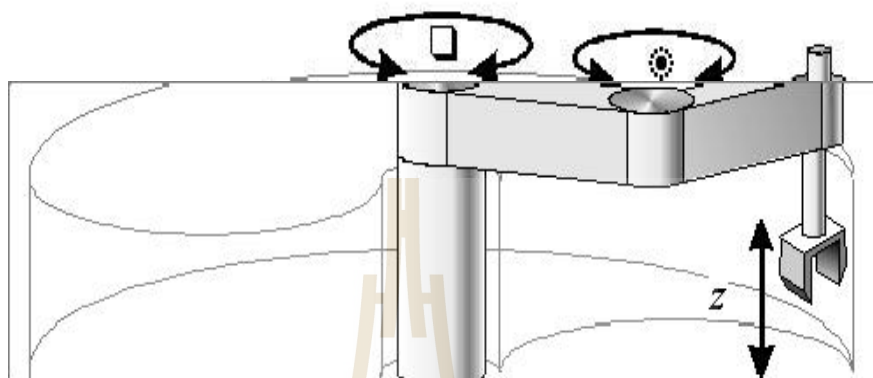
รูปที่ 2.8 ตัวอย่าง The Spherical Robot

4. The Articulated Robot หรือ The Revolute Robot เป็นหุ่นยนต์ที่มีลักษณะการเคลื่อนที่แบบเชิงมุมทั้ง 3 แกนและมีพื้นที่การทำงานรอบตัว โดยหุ่นยนต์ชนิดนี้จะสามารถเข้าถึงพื้นที่ที่ยากต่อการเข้าถึงได้ โดยทั่วไปจะมี 2 แขนกับอีก 1 ฐาน ซึ่งหุ่นยนต์ชนิดนี้สามารถประยุกต์ใช้งานได้อีกมากมาย เช่น การเพิ่มแขนและข้อต่อเข้าไปอีกก็จะสามารถทำงานในจุดที่มนุษย์ไม่สามารถเข้าได้โดยหุ่นยนต์ชนิดนี้มีลักษณะดังรูปที่ 2.9



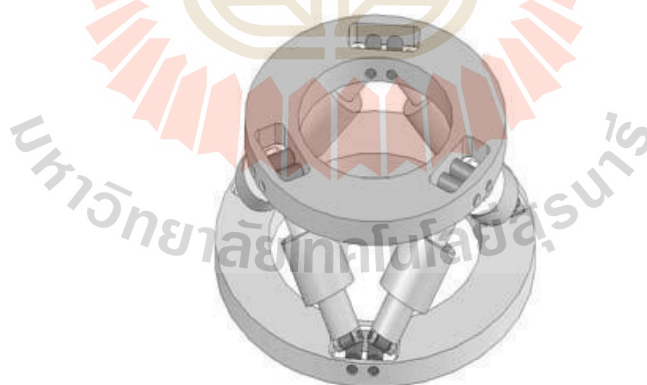
รูปที่ 2.9 ตัวอย่าง The Articulated Robot หรือ The Revolute Robot

5. The SCARA Robot เป็นหุ่นยนต์ที่มีลักษณะการเคลื่อนที่แบบเชิงเส้น 1 แกนและแบบเชิงมุมอีก 2 แกน โดยส่วนมากจะใช้ในการป้อนชิ้นงานตามรางขนส่ง (Conveyer) หรือประกอบชิ้นงานที่มีขนาดไม่ใหญ่นัก ซึ่งหุ่นยนต์ชนิดนี้มีลักษณะดังรูปที่ 2.10



รูปที่ 2.10 ตัวอย่าง The SCARA Robot

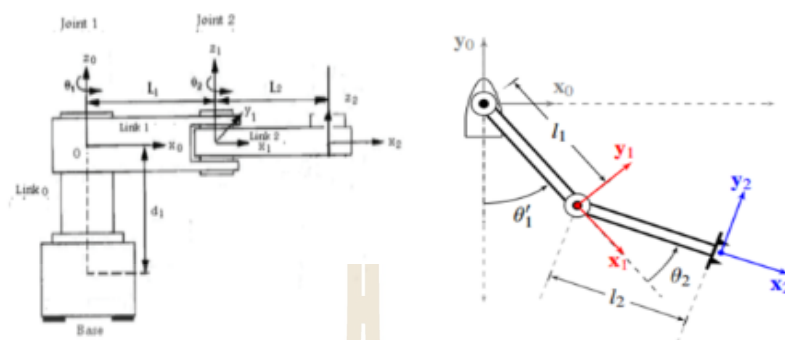
6. The Parallel Robot เป็นหุ่นยนต์อีกแบบหนึ่งที่มีลักษณะเป็นระนาบ 2 ระนาบควบคุมโดยขาแต่ละขา ซึ่งสามารถนำไปประยุกต์ใช้งานได้อีกมากดังรูปที่ 2.11



รูปที่ 2.11 ตัวอย่าง The Parallel Robot

โดยหุ่นยนต์ที่ใช้ในการศึกษาในงานวิจัยนี้ได้แก่ หุ่นยนต์แขนกลระนาบสองมิติ หรือรู้จักในชื่อ Planar Robot ดังรูปที่ 2.12 ซึ่งหุ่นยนต์ชนิดนี้เป็นหุ่นยนต์ที่มีการลักษณะการทำงานแบบหมุนรอบแกนสองจุด ทำงานในสองมิติ โดยหุ่นยนต์ Planar Robot จะจัดอยู่ในประเภท Articulated

Robot ประเภทหนึ่ง โดยทำการลดขนาดของข้อต่อลงมาเหลือเพียง 2 ข้อต่อแบบหมุน เพื่อให้สะดวก และง่ายต่องานวิจัย



รูปที่ 2.12 แสดงหุ่นยนต์แขนกลระนาบสองมิติ (Planar Robot)

2.3.1 การหา D-H Parameters

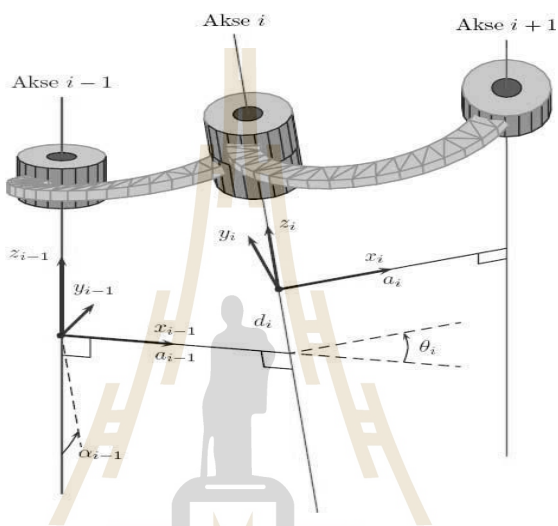
D-H Parameters ย่อมาจากคำว่า Denavit - Hartenberg Parameters เป็นการสร้างตัวแปรเพื่อนำมาใช้ในการหาสมการจลนศาสตร์การเคลื่อนที่ของหุ่นยนต์ซึ่งมีตัวแปรทั้งหมด 4 ตัวได้แก่

- 2.3.1.1 Link Length (a_i) คือระยะทางของ x_i ที่ตั้งฉากระหว่างแกน z_i ถึงแกน z_{i+1}
- 2.3.1.2 Link Twist (α_i) คือมุมระหว่าง z_{i-1} และ z_i ที่มองจากหัวลูกศร x_{i-1} เข้าไป
- 2.3.1.3 Link Offset (d_i) คือระยะทางระหว่างแกน a_{i-1} ที่ตัดบนแกน z_i กับแกน a_i ที่ตัดบนแกน z_i
- 2.3.1.4 Joint Angel (θ_i) คือมุมระหว่างแกน x_{i-1} กับแกน x_i โดยพิจารณาบนแกน z_i

จากคำอธิบายตัวแปรทั้ง 4 ตัวนั้นสามารถแสดงได้ดังรูปที่ 2.13 โดยเมื่อพิจารณาจากข้อต่อและจุดหมุนของหุ่นยนต์แขนกลระนาบสองมิติ ทำให้ได้ตารางค่า D-H Parameter ดังนี้

ตารางที่ 2.2 D-H Parameter of the 2-R Robot

Link	a_i	(α_i)	(d_i)	(θ_i)
1	l_1	0	0	θ_1
2	l_2	0	0	θ_2



รูปที่ 2.13 แสดงรายละเอียดของตำแหน่งตัวแปรต่างๆ ของ D-H Parameters

2.3.2 การหาสมการจลนศาสตร์ไปข้างหน้าของหุ่นยนต์

เมื่อทราบว่ามุมที่ต้องการของการเคลื่อนที่มีค่าเท่าใด เราจะทำนายจุดที่แขนของหุ่นยนต์ไปถึงได้จากจลนศาสตร์ไปข้างหน้า โดยหลังจากที่ได้ค่า D-H Parameters แล้วต่อไปทำการหาค่าทรานฟอร์มเมชันเมตริกซ์ (Transformation Matrix : T) ซึ่งเป็นเมตริกซ์ขนาด 4 x 4 ซึ่งภายในเมตริกซ์นี้ประกอบไปด้วยเมตริกซ์การหมุนและเวกเตอร์การเคลื่อนที่โดยการระบุค่าต่างๆของ D-H Parameters ลงในสมการที่ 2.5 โดย

$$T_i^{i-1} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) & a_i \cos(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) & a_i \sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

T_i^{i-1} คือ ทรานส์ฟอร์มเมชันเมตริกซ์ (Transformation Matrix)

i คือ จำนวนแขน (Link)

a_i คือ Link Length

θ_i คือ Link Twist

d_i คือ Link Offset

α_i คือ Joint Angel

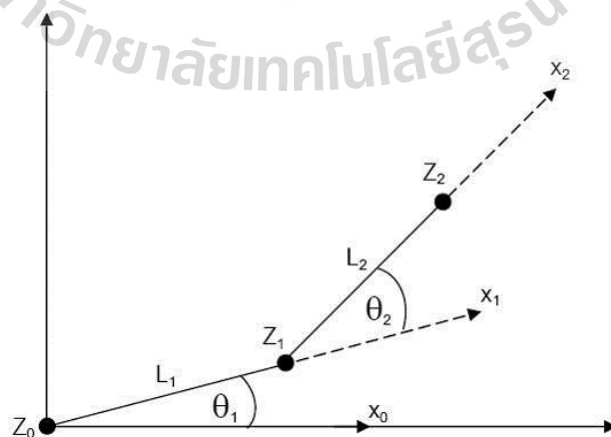
จากสมการที่ 2.5 จะได้เมตริกซ์การหมุนและเวกเตอร์การเคลื่อนที่ดังสมการที่ 2.6

และ 2.7

$$\text{สมการการหมุน} = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i)\cos(\alpha_i) & \sin(\theta_i)\sin(\alpha_i) \\ \sin(\theta_i) & \cos(\theta_i)\cos(\alpha_i) & -\cos(\theta_i)\sin(\alpha_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \quad (2.6)$$

$$\text{เวกเตอร์การเคลื่อนที่} = \begin{bmatrix} a_i \cos(\theta_i) \\ a_i \sin(\theta_i) \\ d_i \\ 1 \end{bmatrix} \quad (2.7)$$

ตามลำดับหากหุ่นยนต์นั้นมีจำนวนแขน i แขน (Link) จะต้องนำค่าทรานส์ฟอร์มเมชันเมตริกซ์ (T) มาคูณกันทางเมตริกซ์ เช่น หุ่นยนต์มีแขน 2 แขน ดังรูปที่ 2.14



รูปที่ 2.14 แสดงตัวอย่างแบบจำลองของหุ่นยนต์ 2 แขน

จากรูปที่ 2.14 สามารถหาค่า D-H Parameters โดยพิจารณาทีละแขนได้แขน l_1 มี D-H Parameters 4 ตัวดังนี้

a_i มีค่าเท่ากับ l_1

θ_i มีค่าเท่ากับ $\theta_{1(t)}$

d_i มีค่าเท่ากับ 0

α_i มีค่าเท่ากับ 0

แขนของ l_2 มี DH Parameter 4 ตัว ดังนี้

a_i มีค่าเท่ากับ l_2

θ_i มีค่าเท่ากับ $\theta_{2(t)}$

d_i มีค่าเท่ากับ 0

α_i มีค่าเท่ากับ 0

หลังจากหาค่า D-H Parameters เสร็จแล้วต่อไปเป็นการค่าหาทรานฟอร์มเมชันเมตริกซ์ (Transformation Matrix) ได้ดังนี้

$$T_1^0 = \begin{bmatrix} c_1 & -s_1 & 0 & l_1 c_1 \\ s_1 & c_1 & 0 & l_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

$$T_2^1 = \begin{bmatrix} c_2 & -s_2 & 0 & l_2 c_2 \\ s_2 & c_2 & 0 & l_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.9)$$

$$T_2^0 = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1 c_1 + l_2 c_{12} \\ s_{12} & c_{12} & 0 & l_1 s_1 + l_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.10)$$

จากสมการที่ 2.10 จะได้เวกเตอร์การเคลื่อนที่ (Translation vector) ของหุ่นยนต์ตามแนวพิกัด xyz ได้ดังนี้

$$1. \text{ แกน } x \text{ มีเวกเตอร์การเคลื่อนที่คือ } p_x = l_1 c_1 + l_2 c_{12}$$

$$2. \text{ แกน } y \text{ มีเวกเตอร์การเคลื่อนที่คือ } p_y = l_1 s_1 + l_2 s_{12}$$

$$3. \text{ แกน } z \text{ มีเวกเตอร์การเคลื่อนที่คือ } 0 \text{ (ไม่มีการเคลื่อนที่ในแนวแกน } z \text{)}$$

เท่านี้จะได้เวกเตอร์การเคลื่อนที่ของแกนต่าง ๆ ของหุ่นยนต์ในหัวข้อต่อไปจะกล่าวถึงทฤษฎีเกี่ยวกับแรงเสียดทานเพื่อใช้ในการวิเคราะห์ถึงการสั่นไหวของวัตถุและการทำให้วัตถุไม่เกิดการสั่นไหว

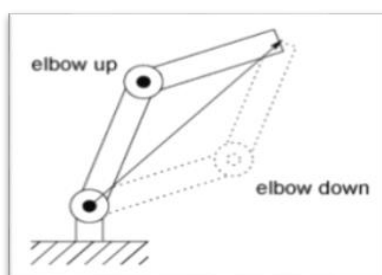
2.3.3 การหาสมการจลนศาสตร์ย้อนกลับของหุ่นยนต์

ในกรณีที่ไม่ทราบว่าแต่ละแขนของหุ่นยนต์จะต้องทำมุมเท่าใดในการเคลื่อนที่ แต่ทราบว่าตำแหน่งที่แขนหุ่นยนต์ไปถึงนั้นมีค่าเท่าใด ก็สามารถใช่วิธีการจลนศาสตร์ย้อนกลับของหุ่นยนต์เพื่อที่จะสามารถหามุมของหุ่นยนต์ที่แต่ละแขนจะเคลื่อนที่ไปได้ โดยเราทราบแล้วว่าสมการที่กำหนดตำแหน่งได้แก่สมการที่ 2.11 และ 2.12

$$p_x = l_1 c_1 + l_2 c_{12} \quad (2.11)$$

$$p_y = l_1 s_1 + l_2 s_{12} \quad (2.12)$$

เมื่อนำสมการหาตำแหน่ง 2.11 และ 2.12 มาทำย้อนกลับ จะทำให้ได้สมการสำหรับหาค่ามุมทั้งสองของแขนหุ่นยนต์ออกมา โดยแต่ละมุมที่ได้ออกมาจะมีค่าเป็นสองค่า คือ มีค่าเป็นมุมบวกและลบ เนื่องจากการจะไปถึงตำแหน่งสุดท้ายของแขนนั้น ในแต่ละจุดสามารถเคลื่อนที่ในลักษณะสองแบบคือ ข้อศอกขึ้นและลง ดังรูปที่ 2.9



รูปที่ 2.15 แสดงข้อศอกขึ้นและลงของหุ่นยนต์

ดังนั้นจะทำให้ได้สมการจลนศาสตร์ย้อนกลับของหุ่นยนต์ดังนี้ คือ

1. กรณีข้อศอกขึ้น

$$\theta_1 = \text{atan} \frac{+p_x l_2 s_2 + p_y (l_1 + l_2 c_2)}{p_x (l_1 + l_2 c_2) - p_y (l_2 s_2)} \quad (2.13)$$

$$\theta_2 = -\text{atan} \frac{s_2}{c_2} \quad (2.14)$$

2. กรณีข้อศอกลง

$$\theta_1 = \text{atan} \frac{-p_x l_2 s_2 + p_y (l_1 + l_2 c_2)}{p_x (l_1 + l_2 c_2) + p_y (l_2 s_2)} \quad (2.15)$$

$$\theta_2 = +\text{atan} \frac{s_2}{c_2} \quad (2.16)$$

2.3.4 ไมโครคอนโทรลเลอร์ (Microcontroller)

R.Hari Sudhan (2015) งานวิจัยนี้ได้อธิบายว่าปัจจุบันได้มีการใช้งานอุปกรณ์ที่ใช้ในการควบคุมการทำงานขนาดเล็กที่รู้จักในชื่อ Microcontroller โดยที่จะกล่าวถึงในงานวิจัยนี้คือ Microcontroller ยี่ห้อ Arduino รุ่น ATmega-328 ว่ามีลักษณะเช่นใด การทำงานเป็นแบบไหน ส่วนประกอบสำคัญที่นำมาใช้ประกอบกันเป็นบอร์ดตัวนี้ได้แก่ ช่องรับสัญญาณเข้าและออก 14 ช่อง ทั้งแบบดิจิทัลและอนาล็อก ใช้ไฟต่ำในการควบคุมการทำงานสามารถจ่ายไฟออกมาได้ 5 VDC-12 VDC ด้วยกระแส 40 mA โดยงานวิจัยนี้ได้สรุปไว้ว่าในอนาคต microcontroller นี้รวมถึงรุ่นอื่น ๆ จะมีบทบาทสำคัญในอุตสาหกรรมเนื่องด้วยขนาดที่กะทัดรัดและราคาต่ำ ทำให้สามารถมาแทนที่วงจรควบคุมต่าง ๆ เช่น RS232, PLC ได้ เป็นต้น

ไมโครคอนโทรลเลอร์ คือ อุปกรณ์ที่สามารถสร้างระบบควบคุมได้ โดยอุปกรณ์นี้มีขนาดเล็ก และเป็นอุปกรณ์ประเภทสารกึ่งตัวนำที่มีการรวมเอาฟังก์ชันการทำงานต่าง ๆ ไว้ในตัวมันเอง ซึ่งมีลักษณะคล้ายกับคอมพิวเตอร์ ซึ่งในที่นี้หมายถึงอุปกรณ์ภายในที่ประกอบด้วยหน่วยประมวลผลกลางและพอร์ตในการเชื่อมต่อแบบต่าง ๆ

2.3.4.1 ส่วนประกอบทั่วไปของไมโครคอนโทรลเลอร์

1. หน่วยประมวลผลกลาง (Control Processing Unit)
2. หน่วยความจำ ซึ่งประกอบด้วย RAM (Random Access Memory) และ EPROM/ EEPROM/PROM/ROM (Erasable Programmable Read Only Memory)
3. หน่วยรับและแสดงผลข้อมูล (Input/Output) ซึ่งมีพอร์ตขยายแบบขนาน (Parallel) และอนุกรม (Serial)

4. ตัวนับเวลา (Timer)

5. หน่วยควบคุมการอินเทอร์รัปต์ (Interrupt Controller)

ส่วนประกอบเหล่านี้เป็นเพียงส่วนประกอบพื้นฐานของไมโครคอนโทรลเลอร์ ซึ่งไมโครคอนโทรลเลอร์นั้นยังมีส่วนประกอบอย่างอื่นอีก เพื่อเพิ่มความสามารถขึ้นอยู่กับไมโครคอนโทรลเลอร์แต่ละเบอร์ด้วย เช่น

- ADC (Analog to Digital) ส่วนภาครับสัญญาณอนาล็อกแปลงไปเป็นสัญญาณดิจิทัล

- DAC (Digital to Analog) ส่วนภาคส่งสัญญาณดิจิทัลแปลงไปเป็นสัญญาณอนาล็อก

- I2C (Inter Integrate Circuit Bus) เป็นการสื่อสารอนุกรม แบบซิงโครนัส (Synchronous) เพื่อใช้ติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์ (MCU) กับอุปกรณ์ภายนอก ซึ่งถูกพัฒนาขึ้นโดยบริษัท Philips Semiconductors โดยใช้สายสัญญาณเพียง 2 เส้นเท่านั้น คือ Serial Data (SDA) และสาย Serial Clock (SCL) ซึ่งสามารถเชื่อมต่ออุปกรณ์จำนวนหลาย ๆ ตัวเข้าด้วยกันได้ ทำให้ MCU ใช้พอร์ตเพียง 2 พอร์ตเท่านั้น

- SPI (Serial Peripheral Interface) เป็นการเชื่อมต่อกับอุปกรณ์เพื่อรับส่งข้อมูลแบบซิงโครนัส (Synchronize) มีสัญญาณนาฬิกาเข้ามาเกี่ยวข้องกับระหว่างไมโครคอนโทรลเลอร์ (Microcontroller) หรือจะเป็นอุปกรณ์ภายนอกที่มีการรับส่งข้อมูลแบบ SPI อุปกรณ์ที่ทำหน้าที่เป็นมาสเตอร์ (Master) โดยปกติแล้วจะเป็นไมโครคอนโทรลเลอร์ หรืออาจกล่าวได้ว่าอุปกรณ์ Master จะต้องควบคุมอุปกรณ์ Slave ได้ โดยปกติตัว Slave มักจะเป็นไอซี (IC) หน้าที่พิเศษต่าง ๆ เช่น ไอซีอุณหภูมิ, ไอซีฐานเวลานาฬิกาจริง (Real-Time Clock) หรืออาจเป็นไมโครคอนโทรลเลอร์ ที่ทำหน้าที่ในโหมด Slave ก็ได้เช่นกัน

- PWM (Pulse Width Modulation) การสร้างสัญญาณพัลส์แบบสแควร์เวฟ ที่สามารถปรับเปลี่ยนความถี่และ Duty Cycle ได้เพื่อนำไปควบคุมอุปกรณ์ต่าง ๆ เช่น มอเตอร์ เป็นต้น

- UART (Universal Asynchronous Receiver Transmitter) ทำหน้าที่รับส่งข้อมูลแบบอะซิงโครนัสสำหรับมาตรฐานการรับส่งข้อมูลแบบ RS-232

2.3.4.2 ประเภทของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ มีด้วยกันหลายประเภทแบ่งตามสถาปัตยกรรม (การผลิตและกระบวนการทำงานระบบการประมวลผล) ที่มีใช้ในปัจจุบันยกตัวอย่างดังนี้

1. ไมโครคอนโทรลเลอร์ตระกูล PIC (บริษัทผู้ผลิต Microchip)
2. ไมโครคอนโทรลเลอร์ตระกูล MCS51 (บริษัทผู้ผลิต Atmel, Philips)
3. ไมโครคอนโทรลเลอร์ตระกูล AVR (บริษัทผู้ผลิต Atmel)
4. ไมโครคอนโทรลเลอร์ Arduino (บริษัทผู้ผลิต Atmel) เป็นการนำ AVR มาเขียน Firmware Boot Loader คำสั่งใช้งานใหม่ ซึ่งข้อแตกต่างสำหรับบอร์ด Arduino กับบอร์ดทดลองทั่วไป คือเป็น Open Source ซึ่งหมายถึง การเปิดเผยวิธีการสร้างทั้งในส่วนของ Hardware ไม่ว่าจะเป็นอย่างใด ๆ ภายในบอร์ด และตัว Tool Software ที่จะใช้เป็นการเขียนคำสั่งต่าง ๆ ลงบอร์ด ล้วนเป็น Open Source ทั้งหมด ส่งผลให้นักวิจัยทุกคนมีสิทธิ์ที่จะดาวน์โหลดโปรแกรมต่าง ๆ ที่เกี่ยวกับการพัฒนา Arduino ไปใช้ฟรี และสามารถสร้างวงจรเองจากต้นแบบใช้ฟรีได้เช่นกัน ซึ่งข้อดี คือ เป็นการสร้างมาตรฐานในการพัฒนาที่จะเป็นไปในทิศทางเดียวกัน เพราะนักพัฒนาทั่วโลกนิยมใช้งานบอร์ดชนิดนี้กันมากขึ้น ดังนั้นไม่ว่าจะเป็นในเรื่องของตัวอย่าง Source Code ตลอดจนการพูดคุยปัญหาเรื่องการใช้งาน และตัวอย่างการต่อวงจรเพื่อการใช้งานในรูปแบบต่าง ๆ จะถูกพัฒนาภายใต้ Open Source ของบอร์ดนี้

5. ไมโครคอนโทรลเลอร์ตระกูล ARM7, ARM9 (บริษัทผู้ผลิต Atmel, Philips, Analog Device, Samsung, STMicroelectronics)

6. ไมโครคอนโทรลเลอร์ตระกูล Basic Stamp (บริษัทผู้ผลิต Parallax)

7. ไมโครคอนโทรลเลอร์ตระกูล PSOC (บริษัทผู้ผลิต CYPRESS)

8. ไมโครคอนโทรลเลอร์ตระกูล MSP (บริษัทผู้ผลิต Texas Instruments)

9. ไมโครคอนโทรลเลอร์ตระกูล 68HC (บริษัทผู้ผลิต MOTOROLA)

10. ไมโครคอนโทรลเลอร์ตระกูล H8 (บริษัทผู้ผลิต Renesas)

11. ไมโครคอนโทรลเลอร์ตระกูล RABBIT (บริษัทผู้ผลิต RABBIT SEMICONDUCTOR)

12. ไมโครคอนโทรลเลอร์ตระกูล Z80 (บริษัทผู้ผลิต Zilog)

2.3.4.3 ภาษาที่ใช้เขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์

1. ภาษา Assembly

2. ภาษา Basic

3. ภาษา C หรือ C++

4. ภาษา Pascal

ภาษาดังกล่าวที่กล่าวในเบื้องต้น ไมโครคอนโทรลเลอร์บางตระกูลจะใช้ได้ครบทุกภาษา แต่บางตระกูลจะใช้ได้บางภาษา ขึ้นอยู่กับบริษัทผู้ผลิต Software (โดยทั่วไปจะเรียกว่า Editor and Compiler) ที่ใช้เขียนภาษาไมโครคอนโทรลเลอร์นั้นจะผลิตออกมาให้ Support หรือไม่

2.3.5 การควบคุมการทำงานของสเตปเปอร์มอเตอร์ด้วย Microcontroller

Ivan Virgala (2015) งานวิจัยนี้ได้อธิบายถึงการควบคุมการทำงานของสเตปเปอร์มอเตอร์ด้วย Microcontroller ในงานวิจัยนี้ได้ใช้ Arduino ATmega-328 โดยขอบเขตทำการวิจัยด้วยการทำงานแบบ Full phase และ half phase ด้วยมอเตอร์ที่สามารถทำงานด้วย 1.8 deg/step ผลที่ได้จากการจำลองด้วยซอฟต์แวร์ MATLAB/Simulink ได้ผลว่าทั้งสองแบบทำงานได้แตกต่างกัน โดยแบบ Full-Phase ได้รับความเร็วเชิงมุมที่สูงกว่า ในขณะที่ Half-Phase มีความเสถียรที่น้อยกว่า และในการวัดค่าการกินกระแสไฟฟ้าพบว่า Full Phase จะกินกระแสมากกว่าในกรณีที่ได้แรงบิดเท่ากัน

Shrddha Baldha (2015) งานวิจัยนี้ได้อธิบายถึงการออกแบบและจำลองการทำงานของสเตปเปอร์มอเตอร์ด้วยวิธี HSM หรือ Hybrid Stepper Motor ซึ่งในการควบคุมการทำงานของมอเตอร์นั้นความแม่นยำในการไปถึงพื้นที่เป้าหมายมีความสำคัญมากเพราะการทำงานนั้นไม่ใช่การทำงานแบบเชิงเส้น โดยในการวิเคราะห์จะใช้ขอบเขตการทำงานของ full phase และ half phase ด้วยซอฟต์แวร์ MATLAB ในการจำลองการทำงานเป็นงานอยู่ในการควบคุมแบบ bang bang ซึ่งมีการรักษาขีดจำกัดทั้งด้านความเร็วคงที่ ตำแหน่งของมอเตอร์ และความสามารถที่เหมาะสมของมอเตอร์อย่างต่อเนื่อง ผลที่ได้รับคือวิธีการ HSM สามารถจำลองได้และวิธีนี้ช่วยให้มอเตอร์มีประสิทธิภาพการทำงานที่ดีขึ้น แต่ข้อเสียคือข้อจำกัดในการเปลี่ยนแปลงความเร็ว ดังนั้นการควบคุมด้วยวิธีนี้จึงไม่สามารถใช้ในความเร็วสูงหรือช่วงที่มีการแกว่งที่มาก

Barhoumi EM (2011) งานวิจัยนี้ได้อธิบายเกี่ยวกับวิธีการใหม่ที่จะนำมาใช้ปรับปรุงความละเอียดในการหมุนแม่เหล็กถาวรของสเตปเปอร์มอเตอร์ โดยข้อจำกัดของเทคนิคนี้คือแรงบิดคงที่และตัวแปรอื่น ๆ ไม่อยู่ในขั้นสูงเกินไป เป็นการควบคุมแบบเปิดโดยส่วนแรกจะกล่าวถึง pulse ส่วนที่สองเป็นการออกแบบวิธีการใหม่บนพื้นฐานการใช้เครือข่าย neural ผลการจำลองการปรับปรุงความละเอียดการหมุนแม่เหล็กถาวรของสเตปเปอร์มอเตอร์มีประสิทธิภาพที่ดีขึ้น แต่ขึ้นอยู่กับทักษะความชำนาญ

Pratiksha N.Balai and Jimit A.Talati (2013) งานวิจัยนี้ได้อธิบายถึงการควบคุมมอเตอร์แบบ HSM หรือ Hybrid stepper motor เพราะวิธีการควบคุมแบบ HSM นั้นเป็นที่นิยมอย่างกว้างขวางเนื่องจากเป็นวิธีการที่มีความแม่นยำของตำแหน่งที่ละเอียดสูง แต่ในการควบคุมเป็นแบบวงจรถูกเปิดยังไม่ดี เพราะการตอบสนองต่อเวลาไม่ดีและมีโอเวอร์ชูทที่แกว่งมาก ซึ่งการใช้งานจริงนั้นมอเตอร์จะต้องตอบสนองต่อการกระตุ้นทุกครั้ง เมื่อเกิดการเปลี่ยนแปลงการกระตุ้นที่เร็วเกินไป มอเตอร์อาจเกิดความผิดพลาด เพราะไม่สามารถย้ายโรเตอร์ใหม่ทุกครั้งได้ ซึ่งจะก่อให้เกิดความผิดพลาดสะสม ดังนั้นเนื่องด้วยข้อจำกัดนี้ มอเตอร์จึงไม่สามารถใช้งานได้โดยตรง ต้องต่อเซนเซอร์และออกแบบการทำงานให้เป็นแบบวงจรถูกปิด เพื่อให้มีประสิทธิภาพที่สูงและแม่นยำ



บทที่ 3

วิธีการดำเนินการวิจัย

3.1 กล่าวนำ

งานวิจัยนี้จะเป็นการออกแบบและสร้างเครื่องต้นแบบระดับห้องปฏิบัติการด้วยการใช้การมองเห็นของเครื่องจักรร่วมกับการทำงานของหุ่นยนต์ที่ได้สร้างขึ้น โดยเริ่มต้นด้วยการศึกษาระบบการมองเห็นของเครื่องจักรเพื่อทำการออกแบบ โปรแกรมสำหรับการประมวลผลรูปภาพ จากนั้นจึงส่งข้อมูลที่จำเป็นไปให้ไมโครคอนโทรลเลอร์เพื่อควบคุมการทำงานของเครื่องต้นแบบต่อไป ทั้งนี้โปรแกรมสำหรับใช้ในการประมวลผลรูปภาพนั้นจะต้องสามารถทำการสร้างข้อมูลภาพเสมือนจริงของรูปภาพต้นแบบให้ได้มากที่สุดเพื่อที่จะนำไปควบคุมเครื่องต้นแบบให้ทำงานได้อย่างเต็มประสิทธิภาพที่สุด

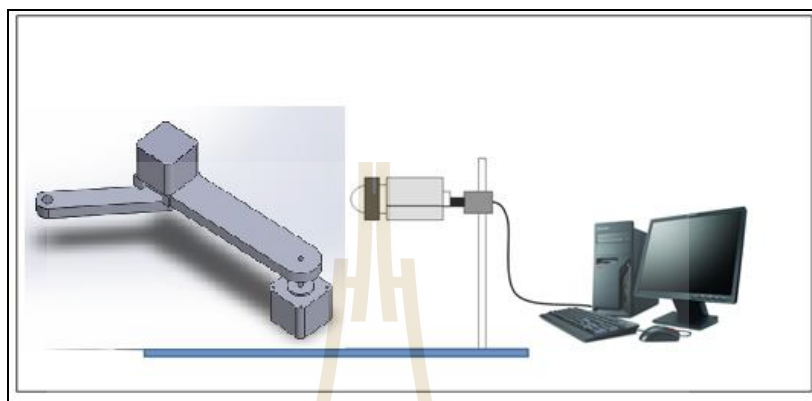
3.2 การสร้างระบบการประมวลผลภาพ

3.2.1 การออกแบบระบบการประมวลผลภาพ

เบื้องต้นงานวิจัยนี้ต้องการที่จะสร้างระบบการประมวลผลภาพ เพื่อช่วยในการออกแบบของนักออกแบบ จากการศึกษาข้อมูลของภาพต้นแบบที่ต้องการประมวลผลแล้วนั้นทางผู้วิจัยจึงเลือกที่จะสร้างระบบประมวลผลภาพโดยการใช้ซอฟต์แวร์ MATLAB เวอร์ชัน 2014b ซึ่งเป็นโปรแกรมที่มีความนิยมกันมากในวิศวกรรมเพราะรูปแบบภาษาที่ใช้ออกแบบโปรแกรมนั้นง่ายต่อความเข้าใจ นอกจากนี้ซอฟต์แวร์ตัวนี้ยังสามารถนำไปประยุกต์ใช้ได้กับงานและอุปกรณ์ประเภทต่าง ๆ โดยผู้วิจัยได้สร้างหน้าต่าง GUI หรือ Graphical User Interface ขึ้นมาจากฟังก์ชันที่มีในโปรแกรม ซึ่งหน้าต่าง GUI นี้เป็นส่วนที่ใช้ติดต่อกับผู้ใช้งานด้วยระบบกราฟฟิก ซึ่งเป็นการนำรูปภาพและสัญลักษณ์เข้ามาใช้งานในโปรแกรมคอมพิวเตอร์ เนื่องจากการใช้งาน โปรแกรมคอมพิวเตอร์ในยุคแรก ผู้ใช้จะต้องจำรูปแบบคำสั่งที่ใช้งานซึ่งทำให้ไม่สะดวกต่อการใช้งาน จึงมีสัญลักษณ์แทนเพื่อให้ง่ายต่อการจดจำและนำไปใช้งาน โดยรูปแบบโปรแกรมจะทำการถ่ายภาพรูปภาพต้นแบบเพื่อนำมาประมวลผลโดยรูปภาพที่ถูกถ่าย จะใช้วิธีการแปลงพิกเซลให้เป็นรูปภาพ โดยแต่ละพิกเซลจะมีพิกัดตำแหน่ง (x,y) ของตัวเองและจะทำการอ่านค่าพิกัดตำแหน่ง และใช้ฟังก์ชันการสร้างรูปภาพด้วยวิธีการ Curve fitting แบบ Spline ซึ่งจะทำได้ภาพที่มีลักษณะเหมือนจริงที่สุด และโปรแกรมจะนำพิกัดตำแหน่งใหม่ที่ได้มาทำการคำนวณด้วยสมการทางคณิตศาสตร์เพื่อนำไปควบคุมการทำงานของมอเตอร์เพื่อให้เครื่องต้นแบบทำงานต่อไป

3.2.2 ส่วนประกอบของระบบประมวลผลภาพ

ในส่วนของระบบการประมวลผลภาพนั้นมีอุปกรณ์ที่ใช้ ดังรูปที่ 3.1 ได้แก่ กล้องจะถูกต่อเข้ากับคอมพิวเตอร์ โดยการประมวลผลภาพจะผ่านซอฟต์แวร์ MATLAB เวอร์ชัน 2014b จากนั้นจึงจะไปสั่งให้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติทำงานต่อไป



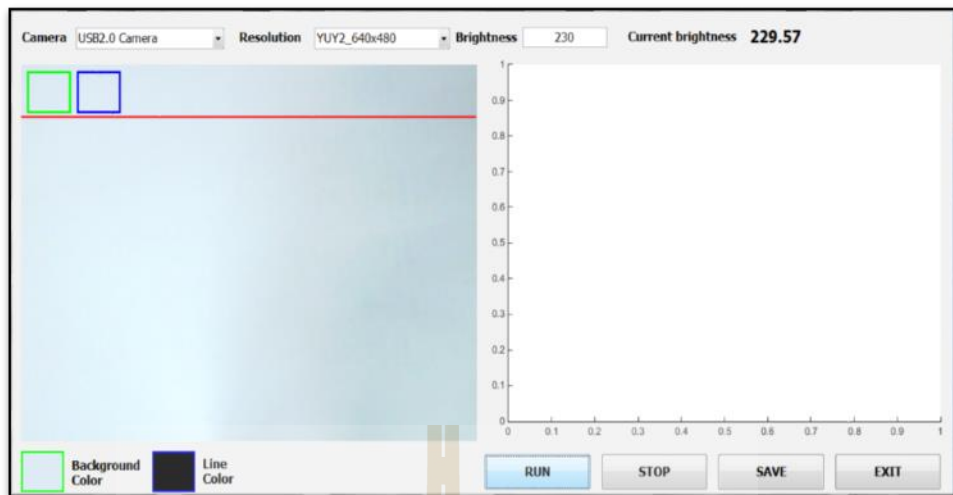
รูปที่ 3.1 แสดงการเชื่อมต่ออุปกรณ์ระบบการประมวลผลภาพ

โดยมีรายละเอียดดังนี้

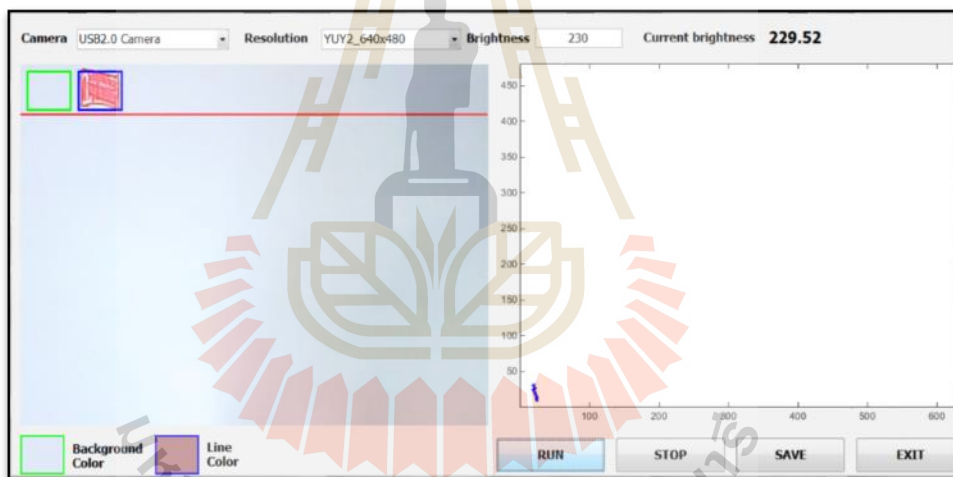
1. รูปภาพต้นแบบ

งานวิจัยนี้จะเริ่มต้นจากการสร้างรูปภาพต้นแบบ โดยรูปภาพต้นแบบจะทำการออกแบบบนกระดาษ A4 ด้วยดินสอ หรือปากกา โดยรูปภาพต้นแบบนี้จะต้องเป็นรูปสองมิติ (2D) หลังจากนั้นรูปภาพต้นแบบที่ถูกสร้างจะถูกแสดงในหน้าต่าง GUI ของโปรแกรมที่แสดงพื้นที่ใช้ในการออกแบบ ดังรูปที่ 3.2

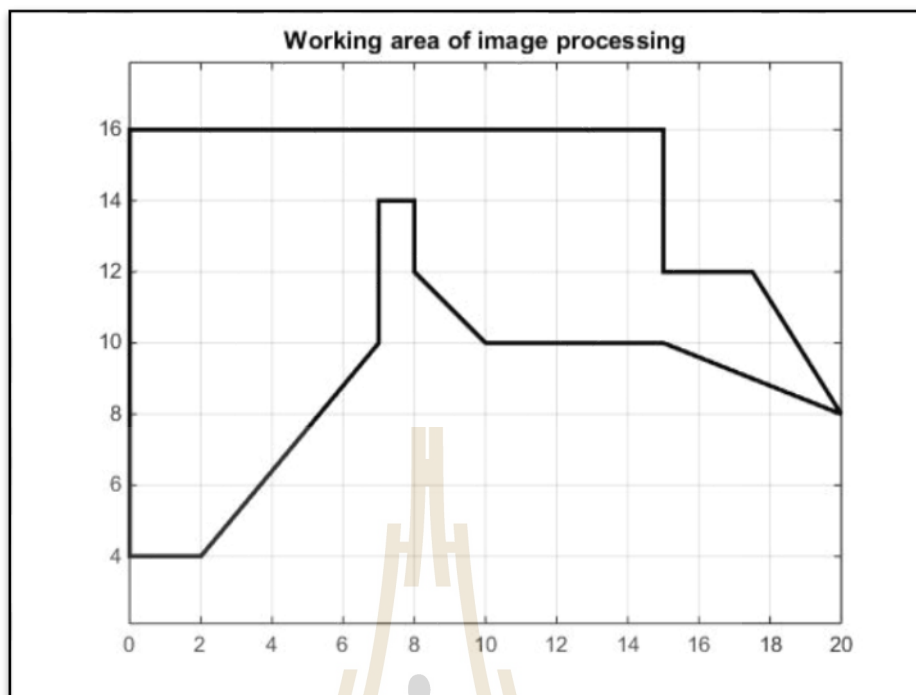
ต่อมาจำเป็นต้องทำการกำหนดสีของกระดาษและปากกาให้ชัดเจน ตามที่โปรแกรมกำหนด ดังรูปที่ 3.3 ซึ่งในงานวิจัยนี้ได้ทดลองออกแบบโดยใช้ปากกาสีแดง โดยผู้ทดลองจะต้องระบายสีในช่องสีน้ำเงินด้านบนเพื่อกำหนดสีของปากกา และในการวาดรูปต้องทำในพื้นที่ที่กำหนดดังรูปที่ 3.4 ซึ่งพื้นที่ดังกล่าวจะเป็นขอบเขตที่เครื่องต้นแบบสามารถทำงานได้ หากทำการออกแบบนอกพื้นที่แล้วนั้น โปรแกรมจะเกิดความผิดพลาดเนื่องมาจากเกินขีดความสามารถของเครื่องทดสอบ ซึ่งจะอธิบายวิธีการทำงานในส่วนถัดไป



รูปที่ 3.2 แสดงพื้นที่ใช้ในการออกแบบ



รูปที่ 3.3 แสดงการกำหนดสีของกระดาษและปากกาสำหรับรูปต้นแบบ



รูปที่ 3.4 แสดงขอบเขตที่สามารถออกแบบรูปต้นแบบโดยไม่เกินขอบเขตของเครื่องทดสอบ

2. กล้องสำหรับถ่ายภาพ

กล้องที่ใช้ในงานวิจัยนี้ ได้แก่ กล้อง WEBCAM CAMERA ยี่ห้อ WEBCAM ANITECH รุ่น IR160 ความละเอียด 16 MEGAPIXEL ดังรูปที่ 3.5 เชื่อมต่อคอมพิวเตอร์ด้วย Universal Serial Bus (USB) สามารถปรับระยะโฟกัสได้ ใช้สำหรับจับภาพรูปภาพต้นแบบ โดยตัวกล้องจะติดบริเวณพื้นที่สำหรับออกแบบรูปภาพต้นแบบ โดยติดตั้งให้สูงเหนือรูปภาพต้นแบบ 50 เซนติเมตร เพื่อให้รูปภาพที่ถูกถ่ายออกมามีขนาดเท่ากับรูปภาพต้นแบบ



รูปที่ 3.5 แสดงกล้อง WEBCAM CAMERA

3. คอมพิวเตอร์สำหรับประมวลผลภาพ ดังรูปที่ 3.6

สำหรับในส่วนของคอมพิวเตอร์ส่วนประกอบหลัก ๆ ได้แก่

- หน้าจอคอมพิวเตอร์

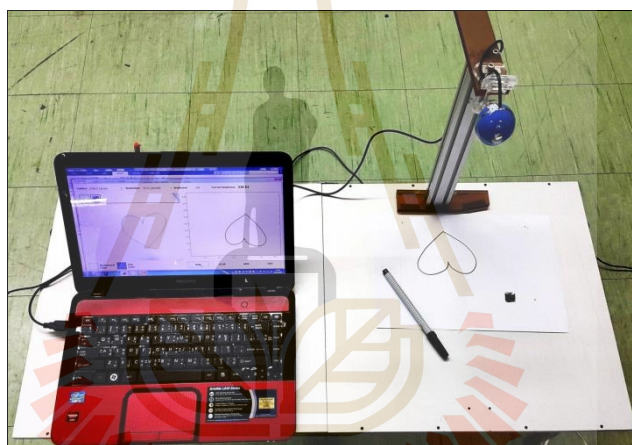
หน้าจอคอมพิวเตอร์ใช้สำหรับการแสดงผลขณะโปรแกรมกำลังประมวลผลภาพ เพื่อให้ผู้วิจัยได้รู้ถึงสถานการณ์ทำงานของโปรแกรม

- KEYBOARD

ใช้สำหรับป้อนค่าและแก้ไขชุดคำสั่งต่าง ๆ

- เมาส์

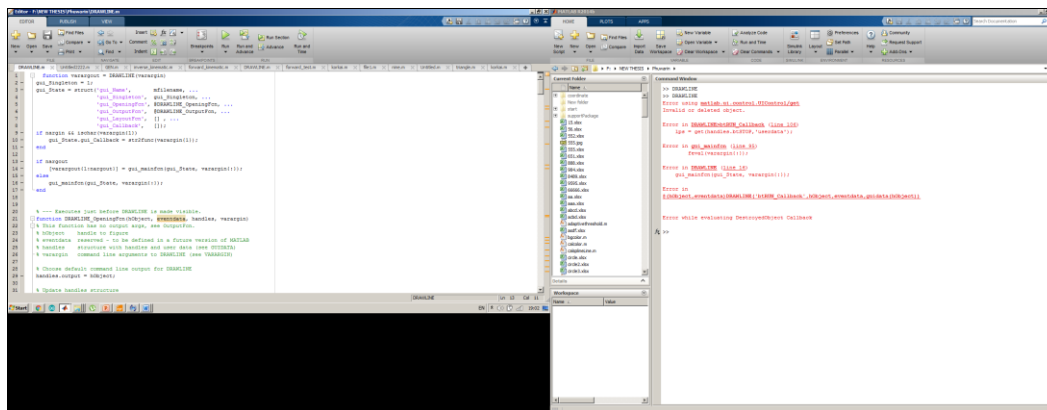
ใช้สำหรับการกดปุ่มคำสั่งต่างของโปรแกรม เช่น RUN, SAVE หรือ STOP หลังจากพนักงานพิมพ์รหัสพนักงานเสร็จ เพื่อให้กล้องเริ่มการทำงาน



รูปที่ 3.6 แสดงคอมพิวเตอร์ที่ใช้ในการประมวลผลภาพ

4. ซอฟต์แวร์สำหรับการประมวลผล

การประมวลผลภาพใช้ซอฟต์แวร์ MATLAB เวอร์ชัน 2014b ดังรูปที่ 3.7 โดยผู้วิจัยได้ทำการเขียนฟังก์ชันการทำงาน ซึ่งจะอธิบายการตั้งค่าและการทำงานในส่วนถัดไป



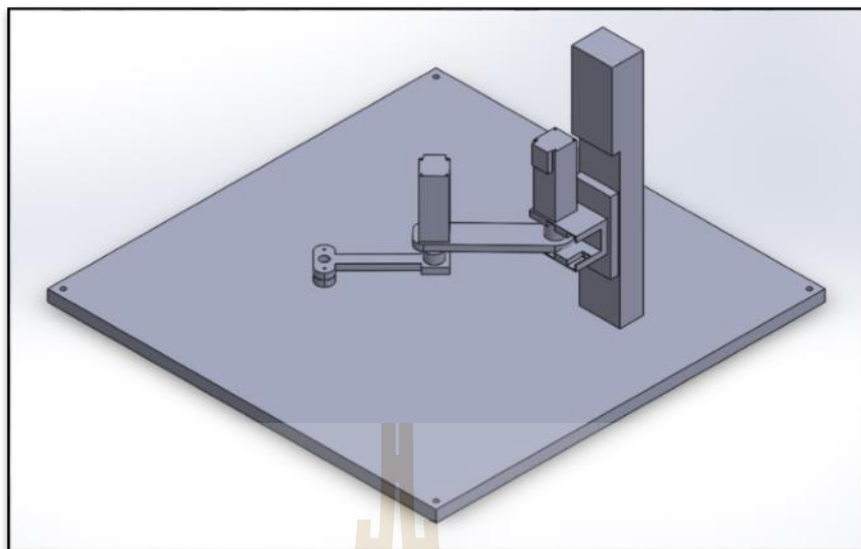
รูปที่ 3.7 แสดงหน้าต่างของซอฟต์แวร์ MATLAB เวอร์ชัน 2014b

3.3 การออกแบบและสร้างเครื่องต้นแบบ

3.3.1 การออกแบบชุดกลไกสำหรับเครื่องต้นแบบ

เนื่องจากในงานวิจัยนี้ต้องการสร้างเครื่องต้นแบบเพื่อใช้จำลองการทำงานจริงของอุตสาหกรรมเพื่อนำมาใช้ร่วมกับโปรแกรมประมวลผลภาพที่ทำการออกแบบขึ้น โดยในส่วนนี้ผู้วิจัยได้เลือกใช้หุ่นยนต์แขนกลระนาบสองมิติ หรือรู้จักกันในชื่อ 2-R Planar Robot โดยหุ่นยนต์นี้จะสามารถเคลื่อนที่ในแกนระนาบแบบเชิงมุมได้สองแกน

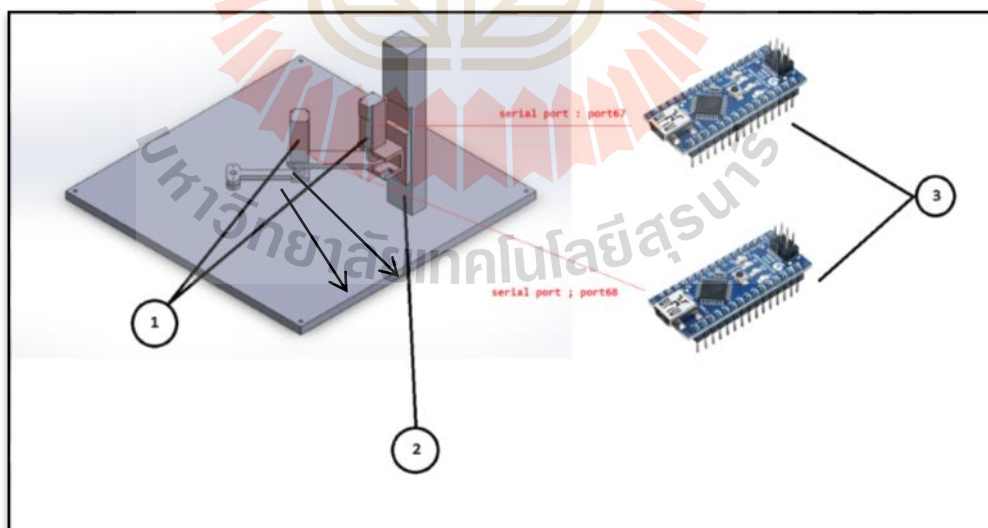
กระบวนการแรกของการจัดทำเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองแกนนี้ ทางผู้วิจัยได้ออกแบบชุดจับยึดที่สามารถจับปากกาเพื่อที่จะสามารถวาดรูปได้ตามรูปต้นแบบ และในเบื้องต้นทางผู้วิจัยได้ออกแบบโดยการใช้โปรแกรม Solidworks เพื่อจำลองตำแหน่ง การเคลื่อนที่ รวมถึงการเลือกขนาดอุปกรณ์ ดังรูปที่ 3.8



รูปที่ 3.8 แสดงเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

3.3.2 การเลือกใช้อุปกรณ์ต้นกำลัง

อุปกรณ์ต้นกำลังที่เลือกใช้เป็นส่วนประกอบในการควบคุมการเคลื่อนที่ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



รูปที่ 3.9 แสดงส่วนประกอบของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

ดังแสดงรายละเอียดต่อไปนี้

1. มอเตอร์สเตปเปอร์ (Stepper Motor)

เนื่องจากเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติจะต้องทำการเคลื่อนที่ในเชิงมุม จึงจำเป็นจะต้องมีการใช้มอเตอร์เป็นต้นกำลังในการเคลื่อนที่ โดยการเคลื่อนที่นี้จะต้องเคลื่อนที่ไป และหยุดในตำแหน่งที่ต้องการด้วยความแม่นยำที่ค่อนข้างสูง ทางผู้วิจัยจึงเลือกใช้สเตปเปอร์ มอเตอร์ของบริษัท VEXTA รุ่น ASM34AK-H50 ดังรูปที่ 3.10 ซึ่งเป็น สเตปเปอร์ มอเตอร์ที่มีเฟืองทดรอบเป็น 1:50 รอบสำหรับควบคุมการหมุน และยังมี ENCODER ติดที่ส่วนปลายของมอเตอร์ช่วยให้มีการป้อนสัญญาณย้อนกลับ (Feedback) กลับมาเพื่อยืนยันพิกัดของมอเตอร์อีกด้วย ซึ่งมอเตอร์ที่มีเฟืองทดรอบนั้นเพื่อช่วยในการลดความเร็วรอบของมอเตอร์ที่ออกมา และเพิ่มแรงบิดให้เหมาะสมกับการใช้งาน นอกจากนี้ยังช่วยให้มอเตอร์มีความละเอียดจาก $1.8^\circ/\text{step}$ ละเอียดเพิ่มขึ้นอีก 50 เท่าจากเฟืองทดรอบ เป็นละเอียด $0.036^\circ/\text{step}$ ทำให้แม่นยำขึ้นอีกด้วย



รูปที่ 3.10 แสดง VEXTA รุ่น ASM34AK-H50

2. หุ่นยนต์เคลื่อนที่เชิงเส้น (Linear Actuator Robot)

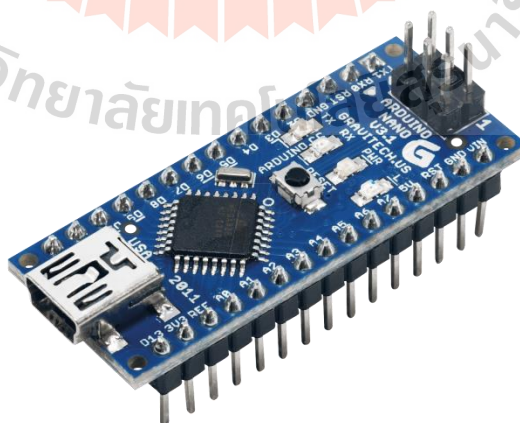
ความเป็นจริงเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิตินั้นไม่จำเป็นต้องมีหุ่นยนต์เคลื่อนที่เชิงเส้น แต่เพื่อความสมจริงและเป็นขั้นเป็นตอนของการทำงานจริง ผู้วิจัยจึงได้ทำการเพิ่มหุ่นยนต์เคลื่อนที่เชิงเส้นตัวนี้เข้าไปหนึ่งแกน เพื่อทำการช่วยในการยกและวางหุ่นยนต์แขนกลระนาบสองมิติ เพื่อให้สามารถยกและวางปากกาได้อย่างเหมือนจริงโดยจะเคลื่อนที่ในแนวเชิงเส้น ขึ้นลงในแนวแกน Z โดยจะสามารถเคลื่อนที่ได้ 50 มิลลิเมตร (Workspace) ดังรูปที่ 3.11



รูปที่ 3.11 แสดง Linear Actuator Robot

3. การควบคุมการทำงานของหุ่นยนต์แขนกลขนาดสองมิติ

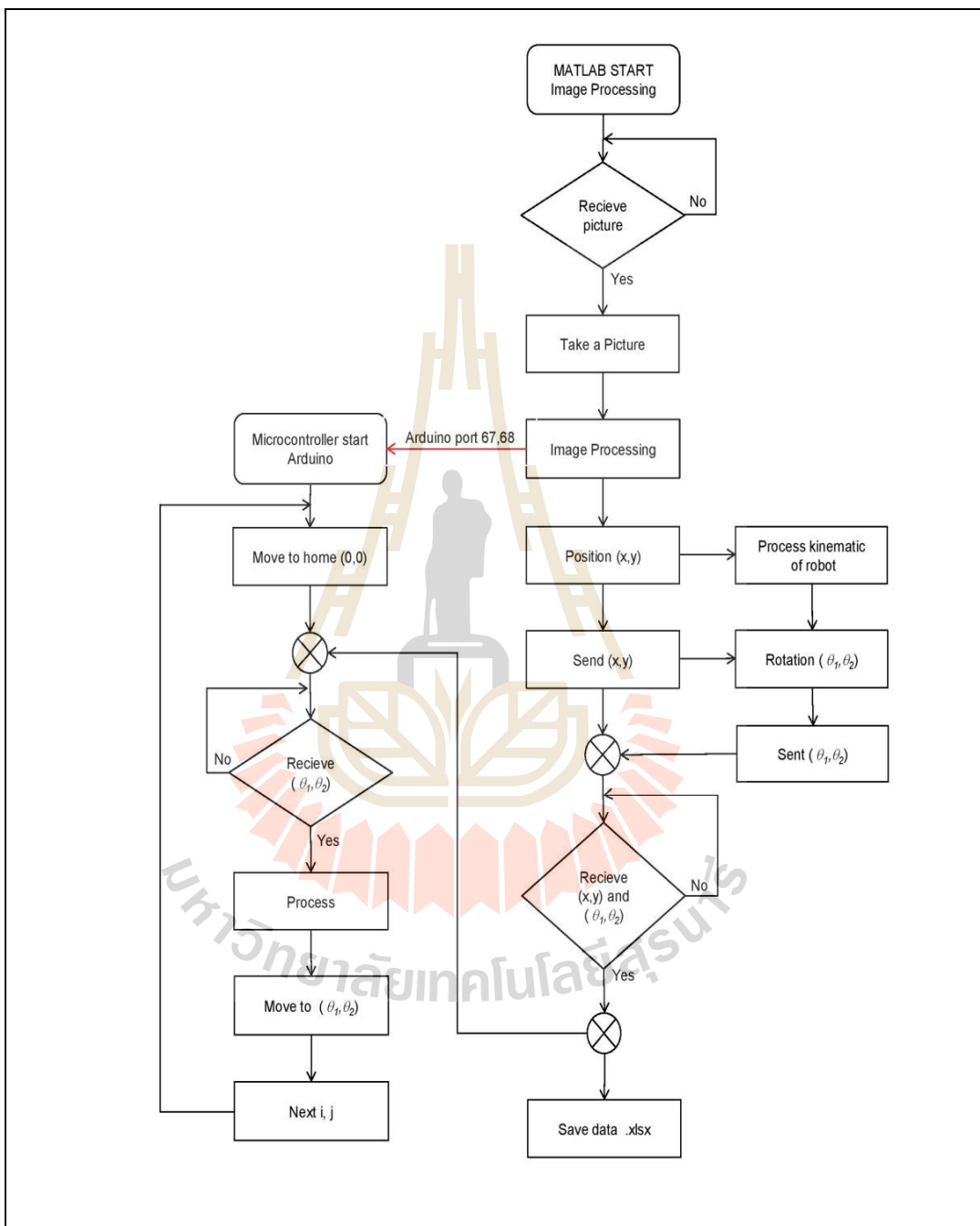
สำหรับการควบคุมการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลสองมิตินี้ จะใช้ไมโครคอนโทรลเลอร์ ในการควบคุมการทำงานของมอเตอร์เพื่อสั่งให้มอเตอร์สเตปเปอร์หมุนไปตามองศาที่ต้องการ โดยเลือกใช้เป็นไมโครคอนโทรลเลอร์ ยี่ห้อ Arduino รุ่น Nano ATmega328 ซึ่งมีขนาดเล็กกะทัดรัดมีขนาด 18 x 45 มิลลิเมตร น้ำหนัก 7 กรัม ใช้งานสะดวก หาซื้อได้ทั่วไปในราคาถูก โดยมี Digital I/O 22 ช่อง ซึ่งเพียงพอกับความต้องการของผู้วิจัย ดังรูปที่ 3.12 โดย Arduino จะทำการรับค่าข้อมูลที่ต้องการจากโปรแกรมประมวลผลภาพ MATLAB จากนั้นจึงสั่งให้มอเตอร์ทำงานตามที่ต้องการต่อไป



รูปที่ 3.12 แสดง Microcontroller Nano ATmega328

3.4 การออกแบบขั้นตอนการทำงานของระบบ

3.4.1 Flow Chart การทำงานของระบบ

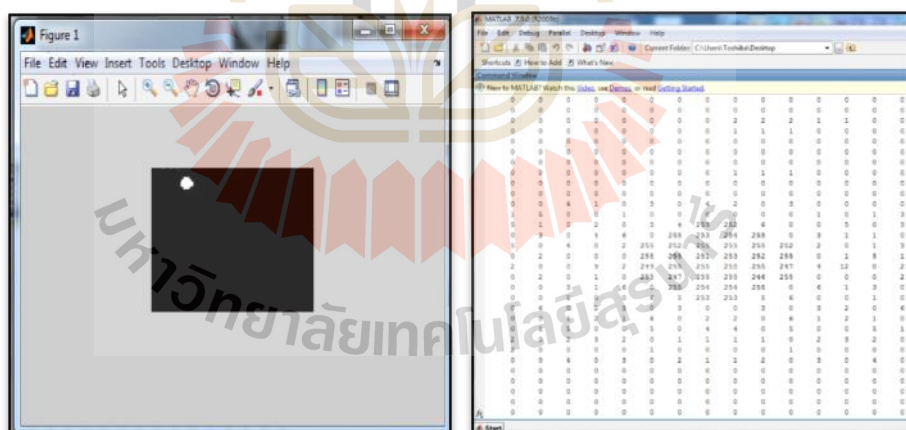


รูปที่ 3.13 แผนผังการทำงานของระบบ

จาก Flow Chart ในรูปที่ 3.13 จะเห็นว่าการทำงานของงานวิจัยนี้ประกอบไปด้วย 2 ส่วนหลักได้แก่ ส่วนแรกส่วนของโปรแกรมประมวลผลภาพ และส่วนที่สองส่วนของการนำข้อมูลจากส่วนแรกเพื่อนำไปควบคุมการทำงานของเครื่องต้นแบบแขนกลระนาบสองมิติ โดยผู้วิจัยจะทำการอธิบายงานวิจัยนี้โดยละเอียดเป็นส่วนๆดังต่อไปนี้

3.4.2 การแปลงข้อมูลจากรูปภาพเป็นพิกเซล

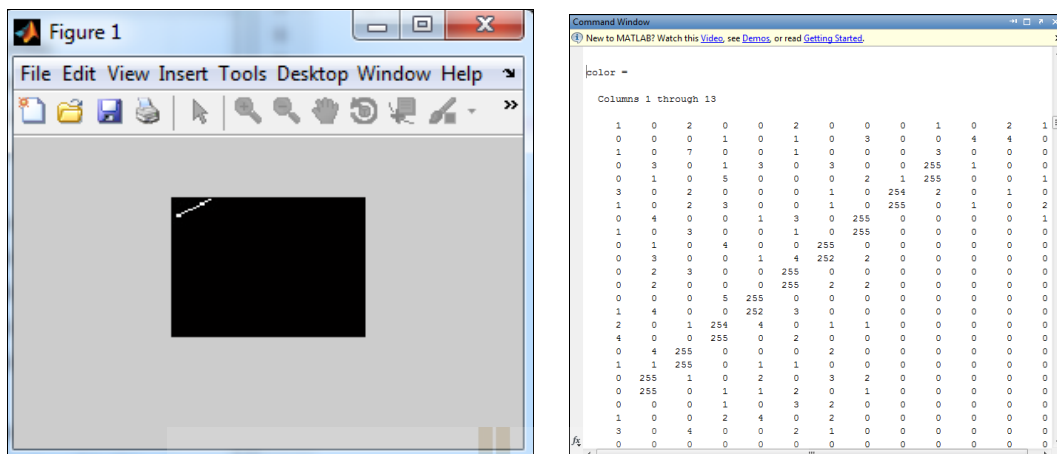
เนื่องด้วยเริ่มต้นในงานวิจัยนี้ต้องการที่จะช่วยในการออกแบบ โดยเมื่อนักออกแบบทำการออกแบบใด ๆ ขึ้นมานั้น ตัวโปรแกรมประมวลผลภาพนั้นจะทำการถ่ายรูปรูปนั้น ๆ แล้วนำการแปลงข้อมูลจากรูปภาพให้อยู่ในลักษณะพิกเซล แล้วทำการอ่านค่าตำแหน่งพิกัด (x,y) ของพิกเซลนั้น ๆ ออกมา เพื่อให้ง่ายในการศึกษาและวิจัยจะเริ่มต้นที่ภาพแบบขาวดำ ซึ่งค่าพิกเซลที่อ่านออกมาได้นั้นจะมีค่าสองค่า คือ 1. เมื่อพิกเซลมีค่าเท่ากับ “ศูนย์” แสดงว่า ณ ตำแหน่งพิกัดนั้น ๆ จากรูปก่อนจะแปลงเป็นค่าพิกเซล จุด ๆ นั้นเป็น “สีดำ” และ 2. เมื่อพิกเซลมีค่าเท่ากับ “สองร้อยห้าสิบห้า” แสดงว่า ณ ตำแหน่งพิกัดนั้น ๆ จากรูปภาพจริงก่อนจะแปลงเป็นค่าพิกเซล จุด ๆ นั้นเป็น “สีขาว” โดยผู้วิจัยได้ทำการทดลองฟังก์ชันคำสั่งในโปรแกรม MATLAB เวอร์ชัน 2014b ด้วยคำสั่ง “`impixel(i,j)`” ซึ่งคำสั่งนี้จะแปลงข้อมูลกลับค่าจากภาพต้นแบบเป็นค่าพิกเซล ดังรูปที่ 3.14



`impixel(i,j)`: the command returns the value of the pixel (i,j)

รูปที่ 3.14 แสดงการแปลงข้อมูลรูปภาพเป็นค่าพิกเซล

จากรูปที่ 3.14 นั้นทำให้เราเห็นว่าสมมติฐานที่เราตั้งไว้สำหรับนำไปใช้ในงานวิจัยจริงสามารถทำได้ เพื่อความมั่นใจผู้วิจัยจึงทดลองกับรูปต้นแบบอื่น โดยที่นี้ยกตัวอย่างเป็นรูปเส้นตรง ดังรูปที่ 3.15



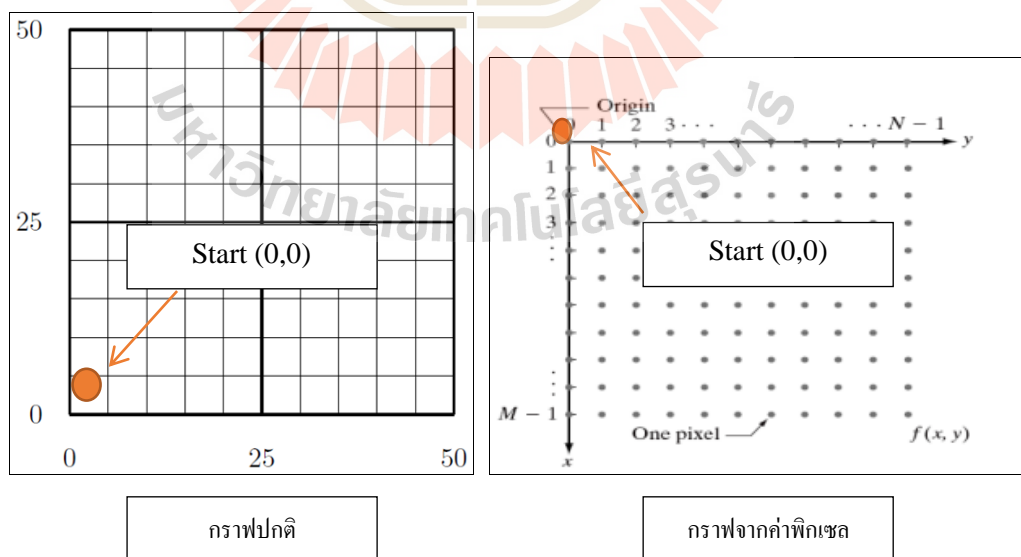
รูปที่ 3.15 แสดงการแปลงข้อมูลรูปภาพเป็นค่าพิกเซล ตัวอย่างที่ 2

3.4.3 การกำหนดตำแหน่งพิกัด (x,y)

จากการที่ผู้วิจัยสามารถแปลงข้อมูลรูปภาพต้นแบบให้อยู่ในรูปของค่าพิกเซลได้แล้วนั้น ต่อไปจะทำการอ่านค่าพิกเซลนั้น ๆ ออกมา เพราะว่าค่าพิกเซลแต่ละค่าจะถูกอ่านออกมาจากตำแหน่งพิกัดของตัวเอง ดังนั้นจากรูปตัวอย่างที่ 2 ที่แสดงในรูปที่ 3.15 ข้างต้นนั้น ผู้วิจัยจะสนใจวิเคราะห์ในส่วนของเส้นที่เป็นเส้นสีขาว ซึ่งเมื่ออ่านค่าพิกเซลออกมาจะมีค่า “สองร้อยห้าสิบห้า” โดยจะไม่สนใจค่าพิกเซลที่มีค่าเท่ากับศูนย์ เมื่อเรียกใช้ค่าพิกเซลที่มีค่า “สองร้อยห้าสิบห้า” นั้น ๆ ออกมา จะได้ค่าตำแหน่งพิกัดออกมาด้วย ดังแสดงในตารางที่ 3.1 แต่พิกัดที่อ่านออกมาได้นั้นจะมีลักษณะกลับหัวกับรูปต้นแบบ เพราะในชีวิตจริงเราจะกำหนดตำแหน่งพิกัดโดยเริ่มจากพิกัด (0,0) ที่อยู่ “ด้านล่างซ้าย” ของตารางพิกัด แต่การกำหนดพิกเซลนั้นพิกเซลแรกอยู่ในตำแหน่ง “ด้านซ้ายบน” ของตารางพิกัด ดังรูปที่ 3.16 ดังนั้นจึงทำให้รูปที่ผ่าน โปรแกรมประมวลผลออกมานั้นจะกลับหัวเหมือนกระจก แต่จะกลับด้านในทิศทางแกน y ซึ่งผู้วิจัยไม่คำนึงกับปัญหาดังกล่าว เพราะในการใช้งานจริงสามารถแก้ปัญหานี้ได้ง่ายโดยการแก้แบบให้กลับด้าน การแก้ด้วยวิธีนี้สะดวกกว่าการแก้โปรแกรมประมวลผลเพราะอาจจะมีปัญหาตามมาในอนาคตต่อไป

ตารางที่ 3.1 แสดงข้อมูลตำแหน่งพิกัด (x,y) ที่สามารถอ่านค่าได้จากรูปที่ 3.15

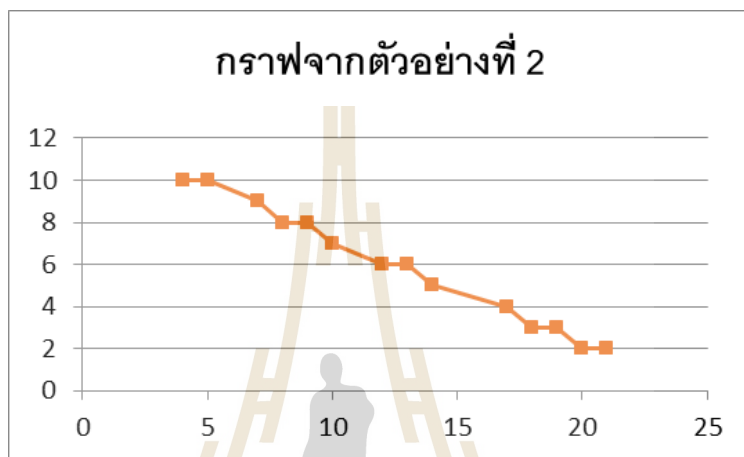
พิกัดในแนวแกน X	พิกัดในแนวแกน Y
4	10
5	10
7	9
8	8
9	8
10	7
12	6
13	6
14	5
17	4
18	3
19	3
20	2
21	2



รูป 3.16 แสดงตำแหน่งพิกัดของค่าพิกเซล

3.4.4 การ Curve Fitting ด้วยวิธี Spline Interpolation

ในส่วนของวิธีว่าเหตุใด ผู้วิจัยจึงเลือกใช้วิธีการ Curve fitting แบบ Spline Interpolation นั้น ได้กล่าวไว้ในบทปริทัศน์วรรณกรรมข้างต้น จะยกตัวอย่างจากตัวอย่างที่ 2 ซึ่งคือ รูปที่ 3.15 เมื่อผ่านการแปลงข้อมูลรูปภาพเป็นค่าพิกเซลแล้วทำให้ได้พิกัดดังแสดงในตารางที่ 3.1 เมื่อนำมาพล็อตกราฟด้วยวิธี Spline Interpolation จะทำให้ได้ ดังรูปที่ 3.17



รูปที่ 3.17 แสดงกราฟที่พล็อตด้วยวิธี Spline Interpolation

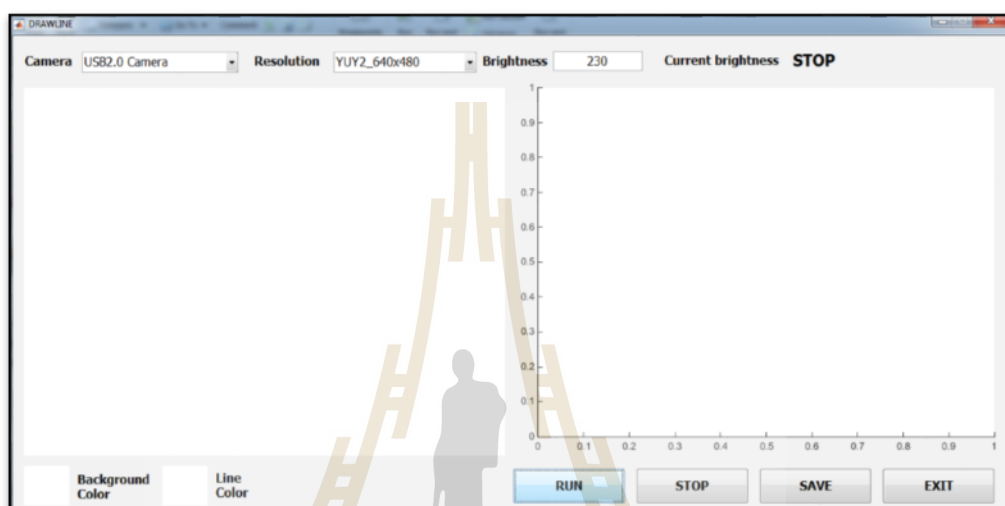
เมื่อนำมาเปรียบเทียบกับรูปต้นแบบ ดังรูปที่ 3.15 แล้วนั้น ก็ตรงตามที่ได้การทดลองมา โดยที่รูปต้นแบบมีลักษณะเป็นเส้นเฉียงลากจากซ้ายขึ้นไปขวา แต่รูปที่อ่านค่าตำแหน่งพิกัดแล้วพล็อตด้วยวิธี Spline Interpolation นั้น มีลักษณะเป็นเส้นเฉียงเช่นกัน แต่พล็อตจากซ้ายลงมาทางขวา ซึ่งตรงตามทฤษฎีที่ รูปที่ออกมาจะกลับด้านในแนวแกน y



รูปที่ 3.18 รูปต้นแบบตัวอย่างที่ 2

3.4.5 ออกแบบและสร้างโปรแกรมประมวลผลภาพผ่านหน้าต่าง GUI

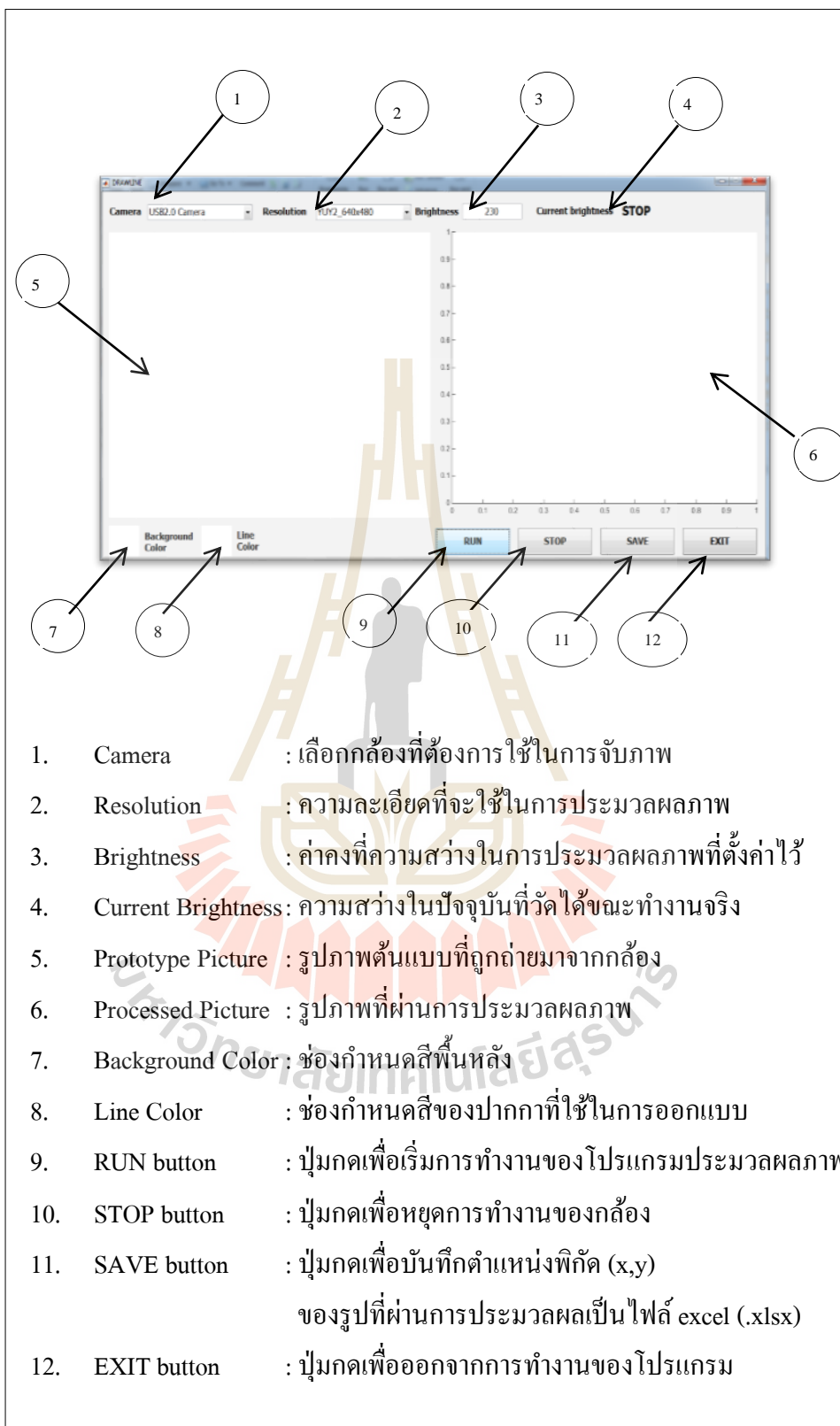
เพื่อความสะดวกและง่ายต่อการใช้โปรแกรม ผู้วิจัยได้ทำการออกแบบหน้าต่างของโปรแกรม โดยใช้ฟังก์ชัน GUI (Graphical User Interface) ของซอฟต์แวร์ MATLAB เวอร์ชัน 2014b ทำให้ได้หน้าต่างใช้งานติดต่อกับผู้ใช้ ดังรูปที่ 3.19 โดยใช้หลักการการสร้างขึ้นมาจากที่ทำการทดลองข้างต้น



รูปที่ 3.19 แสดงหน้าต่าง GUI ของโปรแกรมประมวลผลภาพ

โดยแบ่งแต่ละส่วนดังแสดงในรูปที่ 3.20

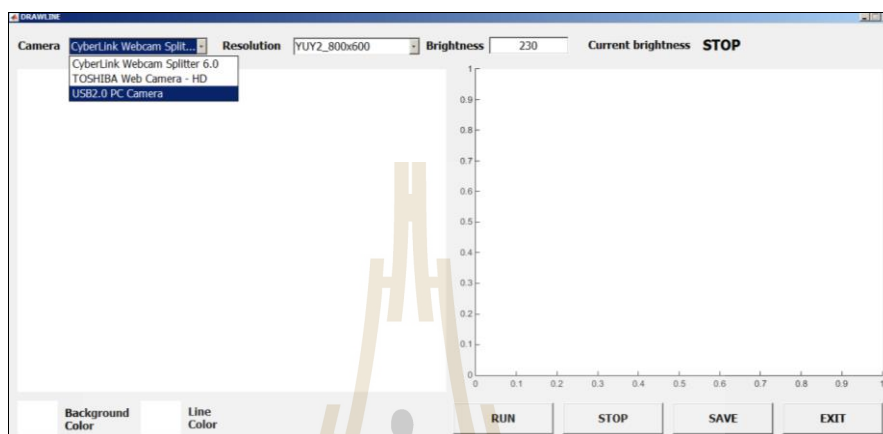
มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 3.20 แสดงส่วนประกอบของหน้าต่างโปรแกรมประมวลผลภาพ

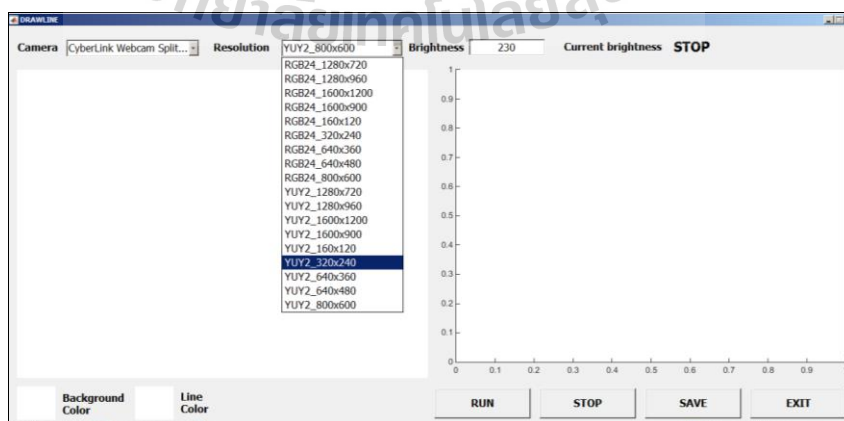
3.4.6 การตั้งค่าโปรแกรมประมวลผลภาพและการเริ่มทำงานเบื้องต้น

ก่อนจะทำการกดปุ่ม RUN นั้น ผู้ใช้จะต้องทำการเลือกว่าจะใช้กล้องตัวไหนทำการเชื่อมต่อกับโปรแกรมประมวลผลภาพ ในที่นี้ผู้วิจัยได้เชื่อมต่อกับกล้อง WEBCAM CAMERA : USB 2.0 PC CAMERA ดังรูปที่ 3.21



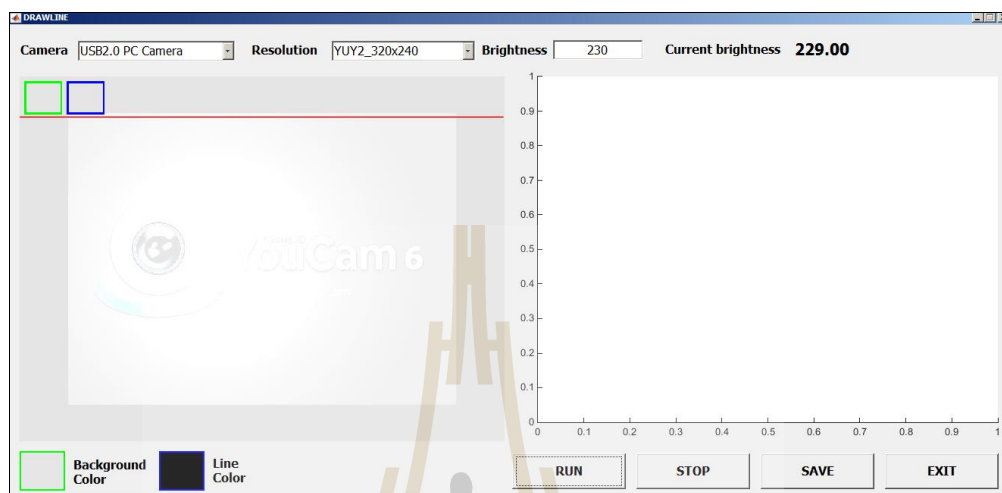
รูปที่ 3.21 แสดงการเลือกกล้องที่ใช้ในโปรแกรมประมวลผลภาพ

ต่อมาหลังจากทำการเลือกกล้องที่ใช้เรียบร้อยแล้ว ก็มาทำการเลือกความละเอียดของกล้องที่ใช้ในโปรแกรม ในที่นี้ผู้วิจัยจะทำการเลือกที่ความละเอียด 320 x 240 (หน่วย) เนื่องจากเป็นความละเอียดที่เหมาะสม โดยความละเอียดที่ใช้จะเลือกโดยคำนึงมาจากการตั้งค่ากล้องและระยะห่างระหว่างกล้องและภาพต้นแบบด้วย ซึ่งความละเอียด 320 x 240 นี้ เมื่อประมวลผลภาพออกมาแล้ว ทำให้ตำแหน่งพิกัด (x,y) ออกมามีขนาด 1:1 กับภาพต้นแบบ ดังรูปที่ 3.22



รูปที่ 3.22 แสดงการเลือกความละเอียดที่ใช้ในโปรแกรมประมวลผลภาพ

หลังจากนั้นเมื่อทำการเลือกกล้องและความละเอียดที่เหมาะสมแล้ว ก็จะเริ่มการทำงานของโปรแกรมประมวลผลภาพ โดยกดปุ่ม RUN หลังจากนั้นโปรแกรมจะเริ่มต้นการทำงาน ดังรูปที่ 3.23

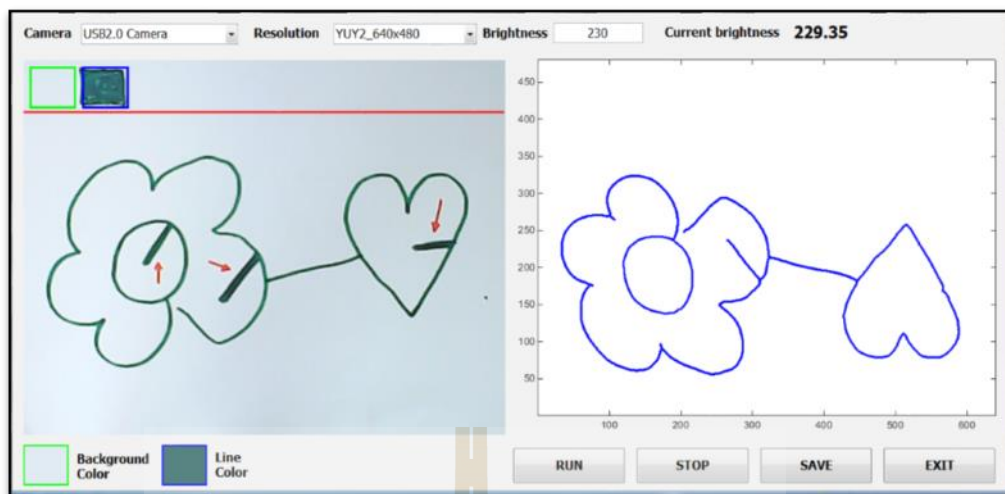


รูปที่ 3.23 แสดงการทำงานของโปรแกรมประมวลผลภาพ

เมื่อโปรแกรมทำงานผู้ใช้จะต้องมาทำการตั้งค่าส่วนสุดท้าย ซึ่งก็คือการกำหนดสีของกระดาษและปากกาที่จะใช้ในกระบวนการประมวลผลภาพ ซึ่งเมื่อโปรแกรมทำงานจะสังเกตเห็นได้ว่าในส่วนของ Prototype Picture นั้นจะปรากฏกล่องสีเขียว และกล่องสีน้ำเงิน

โดยเมื่อใดก็ตามที่ผู้ใช้ระบายสีใด ๆ ลงไปในกล่องสีเขียวนั้นจะเป็นการกำหนดว่า สีของพื้นหลังจะเป็นไปตามที่ผู้ใช้ระบายไป เพื่อให้ง่ายที่ในการทำงานผู้ใช้จึงใช้กระดาษ A4 สีขาวในการเป็นพื้น โดยไม่ต้องระบายสีใด ๆ ลงไปในกล่องสีเขียว ก็จะเป็นการใช้สีของกระดาษนั้นในการกำหนดสีพื้นหลังไปเอง ตามที่ได้ทดลองมาในหัวข้อที่ 3.4.2 การแปลงรูปภาพเป็นค่าพิกเซลที่กำหนดพื้นหลังเป็นสีดำ และรูปภาพที่เป็นเส้นมีสีขาว ในพื้นที่ที่กำหนดสีพื้นหลังก็ทำงานตามนั้น

เมื่อผู้ใช้กำหนดสีของปากกาแล้ว ก็จะต้องใช้ปากกาแท่งเดิม หรือปากกาที่มีสีเหมือนกันในการวาดและออกแบบ มิเช่นนั้น โปรแกรมประมวลผลจะไม่ประมวลผลสีอื่น ๆ ที่อยู่นอกเหนือการกำหนดที่เกิดขึ้น ดังรูปที่ 3.24

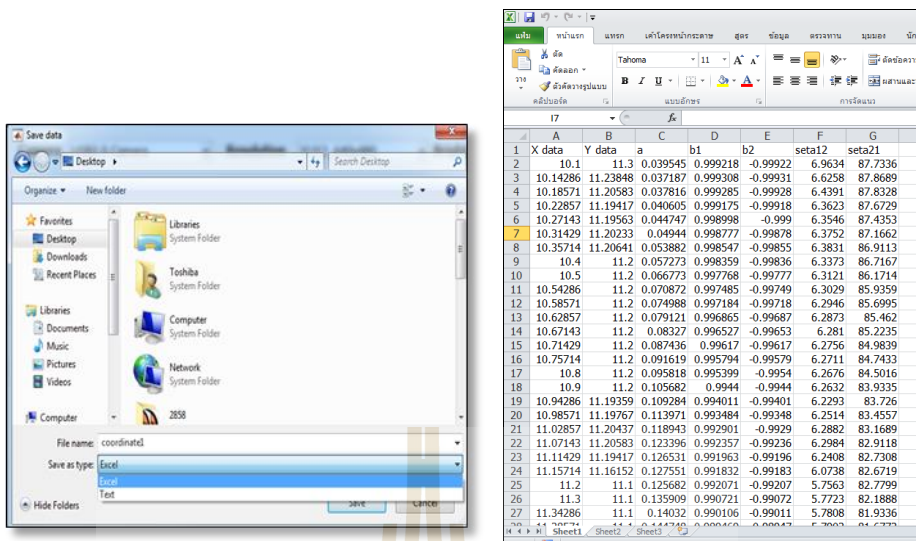


รูปที่ 3.24 แสดงการกำหนดสีปากกาของโปรแกรม

ดังนั้นเมื่อเราทำการกำหนดสีของกระดาษและปากกาแล้ว โปรแกรมประมวลผลภาพก็จะทำการประมวลผลเฉพาะสีที่ได้กำหนดเท่านั้น ส่วนสีอื่น ๆ ระบบจะไม่นำมาประมวลผล ดังแสดงในรูปข้างต้น รูปที่ 3.24 ซึ่งกำหนดสีปากกาเป็นสีเขียวเข้ม ดังนั้นเมื่อทำการวาดใด ๆ ด้วยปากกาสีแดงจึงไม่มีการแสดงผล เพราะ โปรแกรมไม่นำสีแดงมาประมวลผล

3.4.7 การส่งออกข้อมูลเพื่อเก็บเป็นฐานข้อมูล

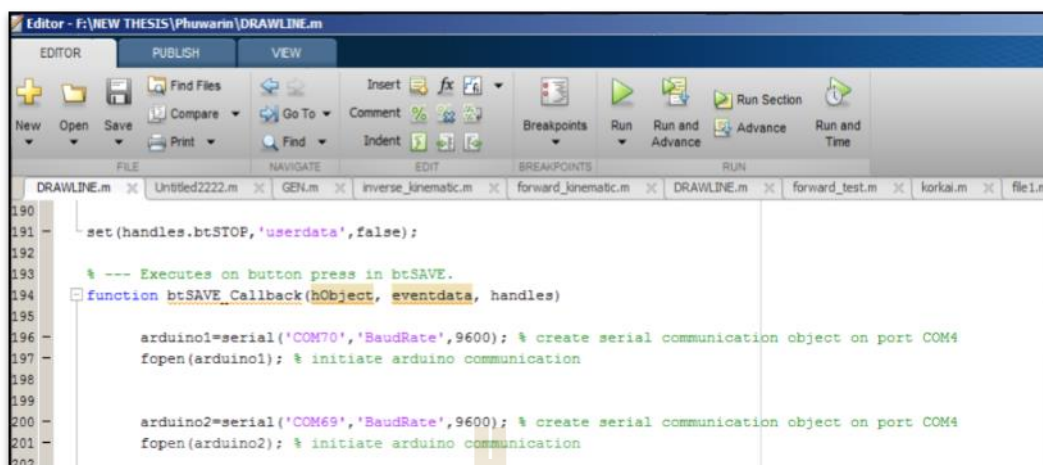
หลังจากที่นักออกแบบทำการออกแบบ และทำการนำแบบนั้นเข้าสู่ขั้นตอนของการประมวลผลภาพ เมื่อเสร็จสิ้นจะได้ตำแหน่งพิกัด (x,y) ใหม่ที่ได้จากรูปต้นแบบผ่านกระบวนการรูปต้นแบบเป็นค่าพิกเซล และนำค่าตำแหน่งพิกัด (x,y) ของพิกเซลนั้นมาจัดเรียงข้อมูลด้วยวิธี Spline Interpolation แล้วทำให้ได้ชุดข้อมูลใหม่ออกมา ซึ่งชุดข้อมูลใหม่จะถูกนำไปคำนวณด้วยสมการทางคณิตศาสตร์ของหุ่นยนต์เพื่อหามุมที่มอเตอร์ควรจะหมุนต่อไป โดยชุดคำสั่งใหม่เหล่านี้จะถูกส่งออกมา เมื่อทำการกดปุ่ม SAVE บนหน้าต่าง GUI ซึ่งอยู่บริเวณด้านซ้ายล่างของหน้าต่าง GUI ซึ่งข้อมูลทั้งหมดจะถูกบันทึกข้อมูลส่งออกมาเป็นไฟล์ Microsoft Excel (.xlsx) ดังรูปที่ 3.25 เพื่อที่ผู้วิจัยจะสามารถนำค่าเหล่านี้กลับมาตรวจสอบ และแก้ไขได้ในภายหลัง



รูปที่ 3.25 แสดงข้อมูลที่ถูกส่งออกมา

3.4.8 การส่งออกข้อมูลเพื่อไปควบคุมการทำงานของหุ่นยนต์แขนกลระนาบสองมิติ

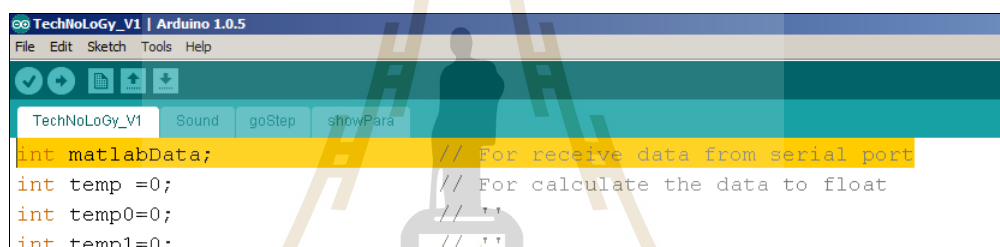
นอกจากการบันทึกข้อมูลข้างต้น ยังต้องมีการส่งออกของข้อมูลไปอีกที่หนึ่ง นั่นก็คือส่งไปให้กับหุ่นยนต์แขนกลระนาบสองมิติ โดยหุ่นยนต์จะสามารถทำงานได้นั้นจะต้องทำงานผ่านต้นกำลัง โดยต้นกำลังหลักทั้งสองตัวนั้นคือ สเตปเปอร์มอเตอร์ 2 ตัว แต่ละตัวควบคุมการหมุนของแกนที่ 1 และ 2 ตามลำดับ โดยการส่งข้อมูลออกไปนั้นจะต้องให้ซอฟต์แวร์ MATLAB เวอร์ชัน 2014b และไมโครคอนโทรลเลอร์ Arduino nano ATmega328 เชื่อมต่อกัน โดยผ่านการเชื่อมต่อที่เรียกว่า Serial Port ผู้วิจัยจึงทำการสร้างฟังก์ชันในซอฟต์แวร์ MATLAB เวอร์ชัน 2014b และไมโครคอนโทรลเลอร์ Arduino Nano ATmega328 ทั้งคู่ โดยแสดงฟังก์ชันของซอฟต์แวร์ MATLAB เวอร์ชัน 2014b ดังรูปที่ 3.26 และแสดงคำสั่งไมโครคอนโทรลเลอร์ Arduino Nano ATmega328 ในรูปที่ 3.27



```

190 set(handles.btSTOP,'userdata',false);
191
192
193 % --- Executes on button press in btSAVE.
194 function btSAVE_Callback(hObject, eventdata, handles)
195
196     arduino1=serial('COM70','BaudRate',9600); % create serial communication object on port COM4
197     fopen(arduino1); % initiate arduino communication
198
199
200     arduino2=serial('COM69','BaudRate',9600); % create serial communication object on port COM4
201     fopen(arduino2); % initiate arduino communication
202
  
```

รูปที่ 3.26 แสดงฟังก์ชันการส่งข้อมูลของซอฟต์แวร์ MATLAB เวอร์ชัน 2014b



```

TechNoLoGy_V1 | Arduino 1.0.5
File Edit Sketch Tools Help
TechNoLoGy_V1 Sound goStep showPara
int matlabData; // For receive data from serial port
int temp =0; // For calculate the data to float
int temp0=0; // '
int temp1=0; // '
  
```

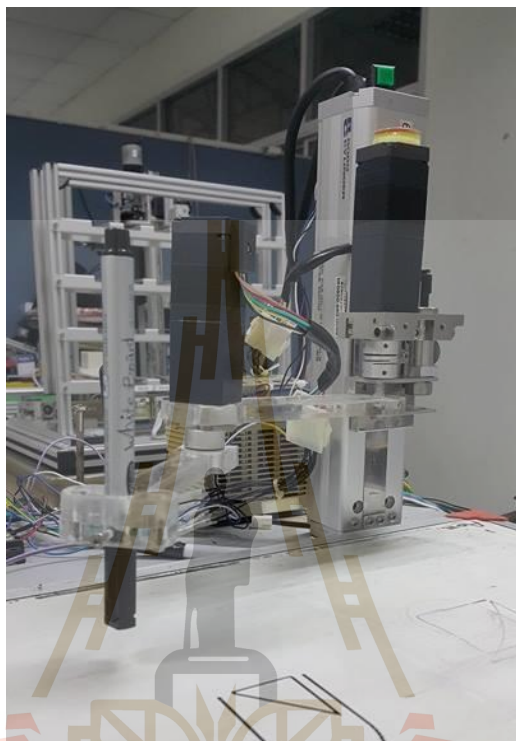
รูปที่ 3.27 แสดงฟังก์ชันการรับข้อมูลของไมโครคอนโทรลเลอร์ Arduino Nano ATmega328

จากข้างต้นผู้วิจัยได้กำหนดให้ไมโครคอนโทรลเลอร์ Arduino Nano ATmega328 ตัวแรกที่ควบคุมการทำงานของมอเตอร์ตัวที่หนึ่งเชื่อมต่อรับส่งข้อมูลกับซอฟต์แวร์ MATLAB เวอร์ชัน 2014b ผ่านช่องทาง Serial com port ช่องที่ 67 (COM67) และ ไมโครคอนโทรลเลอร์ Arduino Nano ATmega328 ตัวที่สองที่ควบคุมการทำงานของมอเตอร์ตัวที่สองเชื่อมต่อรับส่งข้อมูลกับซอฟต์แวร์ MATLAB เวอร์ชัน 2014b ผ่านช่องทาง Serial com port ช่องที่ 68 (COM68)

3.4.9 หุ่นยนต์แขนกลระนาบสองมิติ

ส่วนสุดท้ายของขั้นตอนวิธีการดำเนินงานนี้นั้น กล่าวถึงเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติที่ได้ทำขึ้นจริง ซึ่งเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติที่ได้สร้างขึ้นมานั้นดังรูปที่ 3.28 มีความแตกต่างจากแบบที่ทำการออกแบบมาบ้าง เนื่องจากว่าผู้วิจัยได้ทำการทดลองแล้วว่าการตัดบางส่วนออกไปจะช่วยลดต้นทุนการผลิต ลดน้ำหนัก และเพิ่มประสิทธิภาพด้านความคล่องตัวให้เพิ่มมากขึ้น ในส่วนของแขนของหุ่นยนต์นั้นผู้วิจัยได้เลือกใช้วัสดุคือ

อะคริลิกหนา 8 มิลลิเมตร เพราะมีน้ำหนักเบาและมีความแข็งแรงในระดับที่เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติสามารถยอมรับได้ โดยผ่านการทดสอบด้วยโปรแกรมจำลองการรับแรง ซึ่ง จะแสดงในส่วนของผลวิจัยต่อไป



รูปที่ 3.28 แสดงเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

3.5 เวลาที่ใช้ในการทำงานวิจัย

ในงานวิจัยนี้เพื่อให้ทราบว่าขั้นตอนการทำงานแต่ละขั้นตอนจะใช้เวลามากหรือน้อยในการทำงาน ผู้วิจัยจึงได้ทำการทดลอง เพื่อหาว่าในแต่ละขั้นตอนใช้เวลาเท่าใด โดยการทดลองจับเวลาจะทำโดยใช้นาฬิกาจับเวลา และแบ่งการจับเวลาออกเป็นสามส่วน ได้แก่

3.5.1 ช่วงการออกแบบรูปต้นแบบ

ช่วงนี้จะเป็นช่วงที่เริ่มตั้งแต่การทำงานทั้งหมด โดยนักออกแบบจะทำการออกแบบรูปต้นแบบก่อนที่จะเริ่มการประมวลผลภาพ โดยจะใช้เวลามากหรือน้อย ขึ้นอยู่กับว่าภาพที่ออกแบบมานั้น จะมีขนาดและความละเอียดเพียงใด

3.5.2 ช่วงการประมวลผลภาพและทำการส่งข้อมูล

ช่วงนี้จะเริ่มทำการจับเวลาตั้งแต่ เริ่มทำการประมวลผลภาพ (กดปุ่ม RUN) ซึ่งระบบ จะทำการประมวลผลภาพ และส่งข้อมูลจากซอฟต์แวร์ MATLAB เวอร์ชัน 2014b ไปให้ เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ โดยทำการส่งข้อมูลผ่านทาง Serial Port เพื่อไปให้ ตัวรับคือ ไมโครคอนโทรลเลอร์ Arduino ATmega328 ซึ่งในส่วนนี้จะเป็นส่วนที่ใช้เวลามาก เพราะ ข้อมูลที่รับได้นั้นจะมาทีละชุดข้อมูล และรอนกว่าจะครบ ถึงจะทำงานในส่วนต่อไป ซึ่งหากรูป ต้นแบบมีจำนวนข้อมูลทีมากก็จะใช้เวลาในการส่งที่นาน และจะจับเวลาจนกระทั่งโปรแกรม ประมวลผลภาพบอกว่าข้อมูลที่ส่งไปครบแล้ว

3.5.3 ช่วงการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

ช่วงสุดท้ายนี้จะเริ่มจับเวลาต่อเนื่องจากส่วนที่สอง คือเมื่อโปรแกรมประมวลผลภาพ ส่งข้อมูลจนครบจะเริ่มจับเวลาต่อทันที ทำการจับเวลาจนกระทั่งหุ่นยนต์วาดรูปแล้วเสร็จ และกลับไปสู่ตำแหน่งเริ่มต้น (Home Position)



บทที่ 4

ผลการทดลอง

สำหรับบทนี้จะนำเสนอผลการดำเนินการออกแบบและสร้างเครื่องต้นแบบระดับห้องปฏิบัติการด้วยการใช้การมองเห็นของเครื่องจักรร่วมกับการทำงานของหุ่นยนต์ที่ได้สร้างขึ้น ดังรูปที่ 4.1 โดยกระบวนการวิจัยนี้ได้แบ่งการทำงานออกเป็น 3 ระบบ คือ 1.ระบบการประมวลผลด้วยภาพ 2. ระบบการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ และ 3.ระบบการทำงานร่วมกันระหว่างระบบการประมวลผลภาพและเครื่องต้นแบบระดับห้องปฏิบัติการ จึงต้องทำการทดสอบระบบทั้งสองระบบให้สมบูรณ์ก่อนที่จะนำมาทำงานร่วมกัน เพื่อการทำงานถูกต้องและแม่นยำ

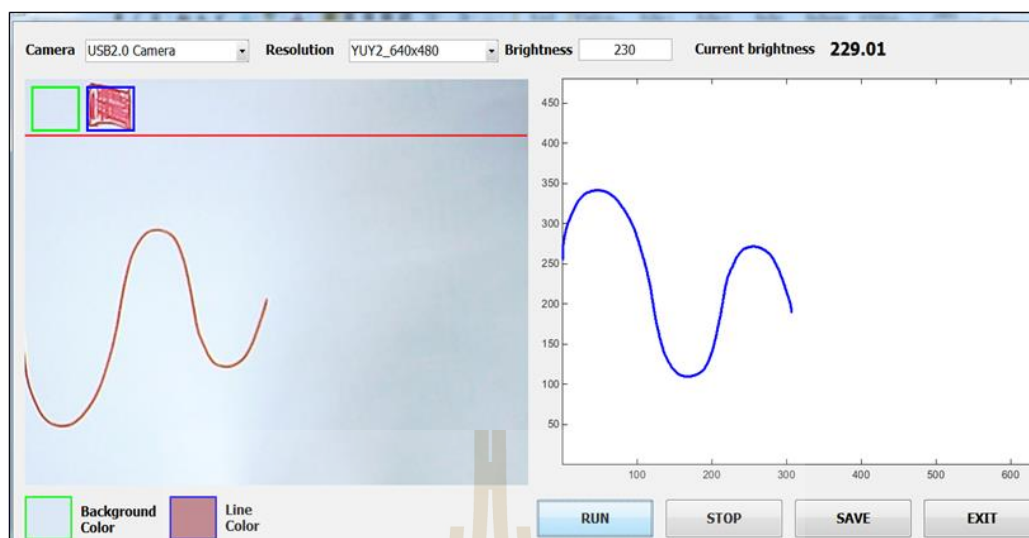


รูปที่ 4.1 การออกแบบและสร้างเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

4.1 การทำงานของระบบการประมวลผลด้วยภาพ

4.1.1 เริ่มทดสอบ

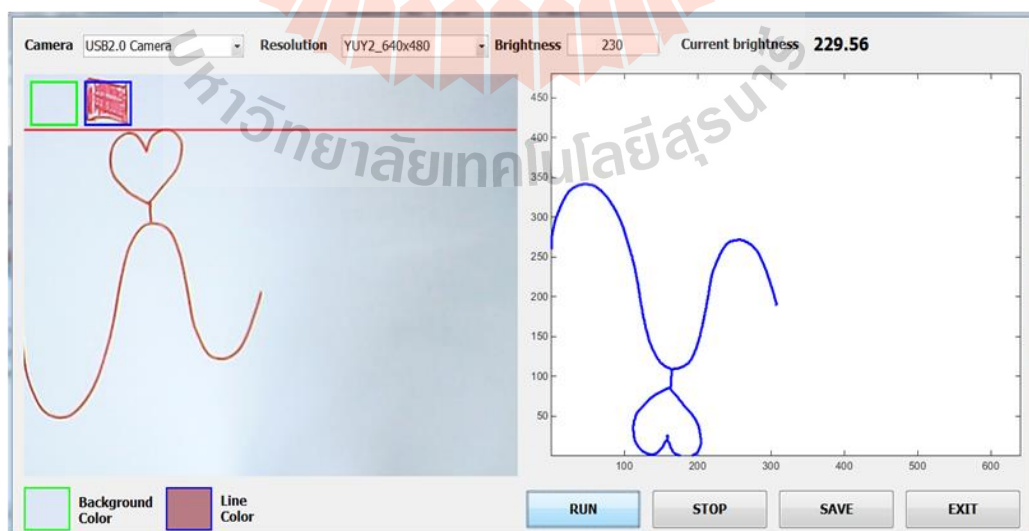
หลังจากทำการตั้งค่าโปรแกรมประมวลผลภาพตามบทที่ 3 แล้วนั้น ก็สามารถเริ่มการทำงานด้วยการที่ผู้วิจัยทำการสร้างรูปต้นแบบ กำหนดให้ใช้ความละเอียด 640 x 480 พิกเซล และปากกาที่ใช้ออกแบบเป็นสีแดง ดังรูปที่ 4.2



รูปที่ 4.2 แสดงการทดสอบโปรแกรมประมวลผลภาพรูปเส้นโค้ง

ผลการทดสอบโปรแกรมประมวลผลภาพจากรูปที่ 4.2 เป็นจริงตามที่ผู้วิจัยคาดหวังไว้ โดยโปรแกรมประมวลผลภาพสามารถตรวจพบเส้นสีแดง และสร้างรูปเส้นโค้งเสมือนที่เกิดจากการนำตำแหน่งพิกัด (x,y) มาสร้างภาพเสมือนด้วยวิธีการ Spline Interpolation

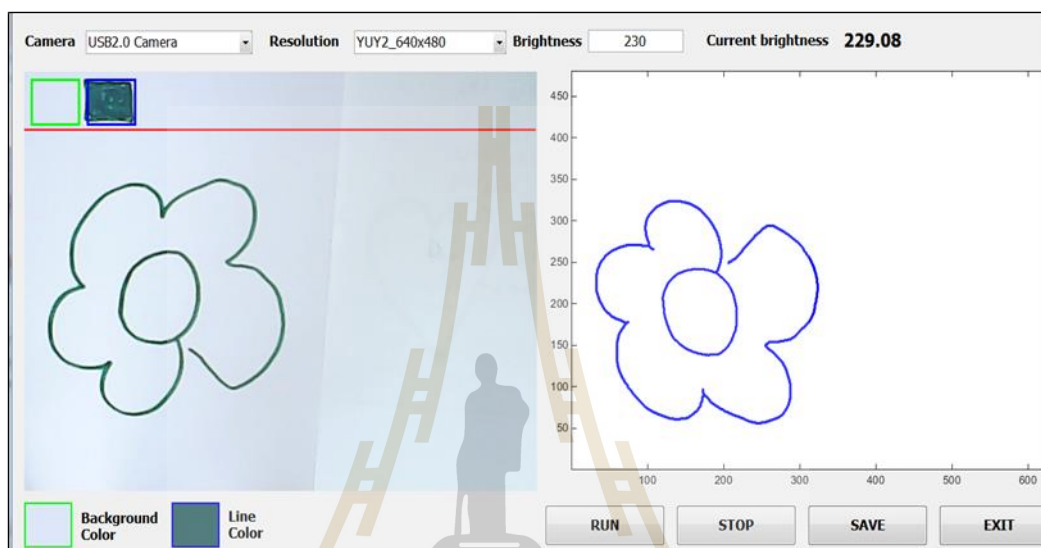
เมื่อผู้วิจัยลองออกแบบเพิ่มเติมจากรูปที่ 4.2 โดยลองวาดรูปหัวใจเพิ่มเข้าไป ดังรูปที่ 4.3



รูปที่ 4.3 แสดงการทดสอบโปรแกรมประมวลผลภาพรูปเส้นโค้งและหัวใจ

ผลการทดสอบ โปรแกรมประมวลผลภาพจากรูปที่ 4.3 ก็ยังได้ผลลัพธ์จากโปรแกรมประมวลผลภาพเป็นที่น่าพอใจเช่นเดิม

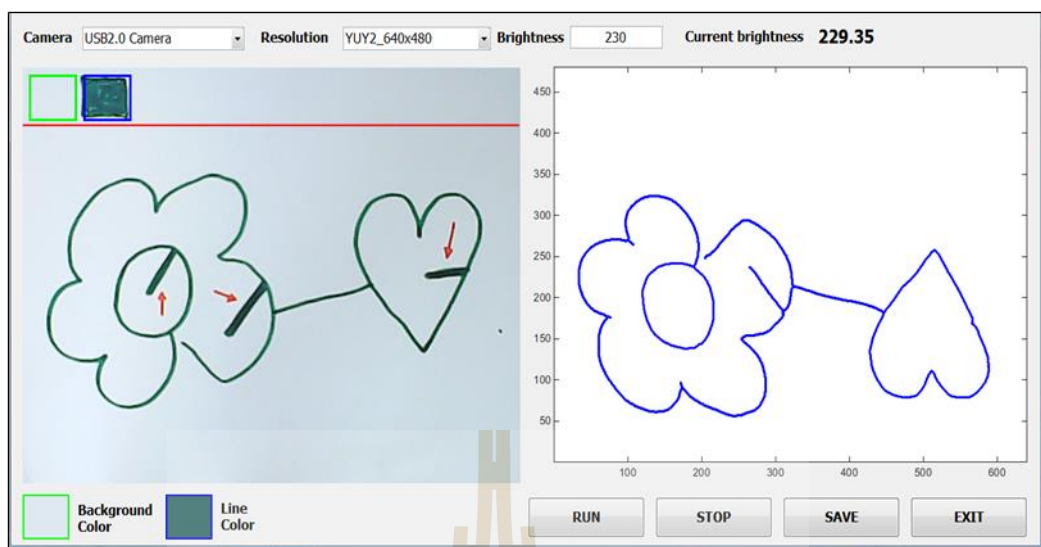
ต่อมาผู้วิจัยได้ทำการทดสอบการเพื่อหาผลลัพธ์ว่าเมื่อทดลองทำการเปลี่ยนสีของปากกา โปรแกรมประมวลผลยังสามารถทำงานได้เป็นที่พึงพอใจเช่นเดิมหรือไม่ จึงทดลองเปลี่ยนสีปากกาเป็นสีเขียวเข้ม และวาดรูปต้นแบบใหม่เป็นรูปดอกไม้ ดังรูปที่ 4.4



รูปที่ 4.4 แสดงการทดสอบ โปรแกรมประมวลผลภาพรูปดอกไม้

ผลการทดสอบ โปรแกรมประมวลผลภาพจากรูปที่ 4.4 ก็ยังได้ผลลัพธ์จากโปรแกรมประมวลผลภาพเป็นที่น่าพอใจเช่นเดิม คือ สามารถตรวจจับเส้นสีเขียวเข้มได้ดี เช่นเดียวกับเส้นที่แดงในตัวอย่างที่แล้ว

สุดท้ายการทดสอบ โปรแกรมประมวลผลภาพ ผู้วิจัยต้องการยืนยันความถูกต้องของโปรแกรมประมวลผลภาพว่าหากมีการใช้ปากกาสองสีในการวาดรูปแล้วนั้น จะเกิดเหตุการณ์อย่างไรในการประมวลผลรูปภาพ ดังรูปที่ 4.5



รูปที่ 4.5 แสดงการทดสอบ โปรแกรมประมวลผลภาพด้วยปากกาสองสี

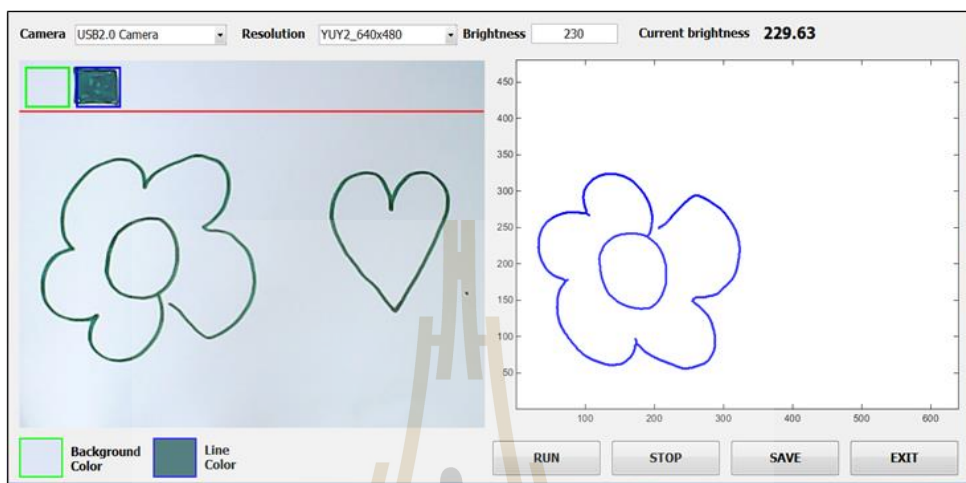
จากผลการทดสอบ โปรแกรมประมวลผลภาพจากรูปที่ 4.5 ข้างต้นซึ่งผู้วิจัยกำหนดให้โปรแกรมประมวลผลภาพตรวจเฉพาะเส้นสีเขียวเข้ม เมื่อวาดรูปต้นแบบ ผู้วิจัยได้ทดลองเขียนเส้นสีแดงแฝงเข้าไปในรูปต้นแบบ ปรากฏว่าโปรแกรมประมวลผลก็ยังคงตรวจเฉพาะเส้นที่เป็นสีเขียวเข้มเช่นเดิม โดยในรูปจะเห็นว่ามีเส้นสีแดง ซึ่งโปรแกรมประมวลผลภาพจะไม่นำเส้นสีแดงมาคิด ทำให้ผู้วิจัยสามารถยืนยันได้ 100% ว่าไม่ว่าจะใช้ปากกาสีใดก็ตาม หากว่าปากกาที่ใช้ในการออกแบบเป็นสีเขียวก็เหมือนกับปากกาที่ใช้ในการกำหนดในพื้นที่ช่อง Line Color โปรแกรมประมวลผลภาพสามารถทำงานได้อย่างถูกต้องสมบูรณ์แบบ

4.1.2 ซิดจำกัดของโปรแกรมประมวลผลภาพ

จากการใช้ความรู้ด้านต่าง ๆ ในการออกแบบและสร้างโปรแกรมประมวลผลภาพ ในการสร้างรูปขึ้นมาจากตำแหน่งพิกัด (x,y) ที่ได้มาจากการแปลงรูปภาพต้นแบบเป็นค่าพิกเซล และได้ตำแหน่งพิกัด (x,y) ใหม่ออกมา และได้ใช้วิธีการ Spline Interpolation สร้างข้อมูลแต่ละค่าที่มีความเสมือนจริงกับข้อมูลต้นแบบออกมา วิธีการนี้ทำให้เกิดซิดจำกัดของโปรแกรมประมวลผลภาพขึ้นมา 3 ข้อ ได้แก่

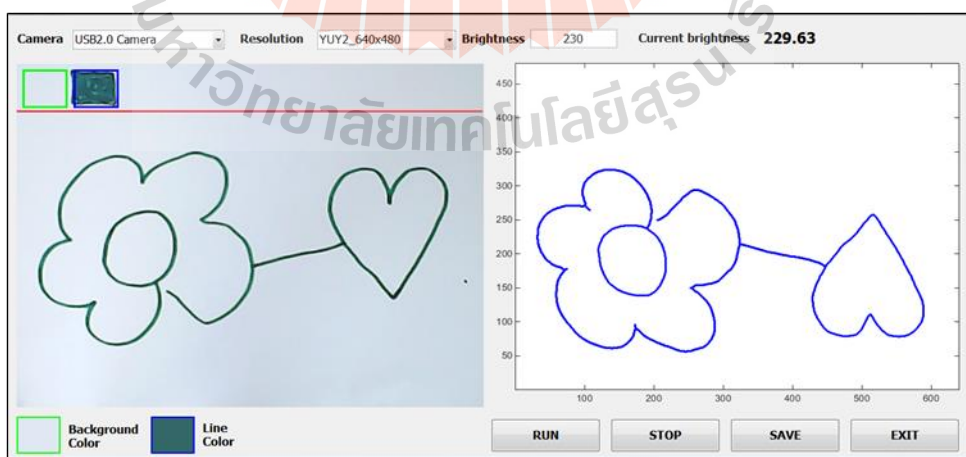
4.1.2.1 รูปภาพต้นแบบที่ออกแบบมานั้นจะต้องเป็นรูปภาพที่มีความต่อเนื่อง คือ จะไม่สามารถวาดรูปต้นแบบที่ภายในรูปต้นแบบมีหลาย ๆ รูปย่อยได้ เนื่องมาจากการพล็อตจุดแบบ Curve fitting ด้วยวิธีการ Interpolation เป็นการพล็อตจุดแบบจากจุดหนึ่งไปอีกจุดหนึ่ง โดย

จะต้องผ่านจุดทุกจุด ทำให้เมื่อมีรูปย่อยสองรูปอยู่ในรูปภาพต้นแบบเดียวกัน การพล็อตจุดจากรูปทั้งสองไม่ต่อเนื่อง โปรแกรมประมวลผลภาพไม่รู้ว่าจุดไหนจะไปต่อจึงทำให้ไม่สามารถทำงานตามรูปต้นแบบได้ ดังรูปที่ 4.6 ที่มีรูปต้นแบบคือ รูปดอกไม้และรูปหัวใจอยู่ในภาพเดียวกัน



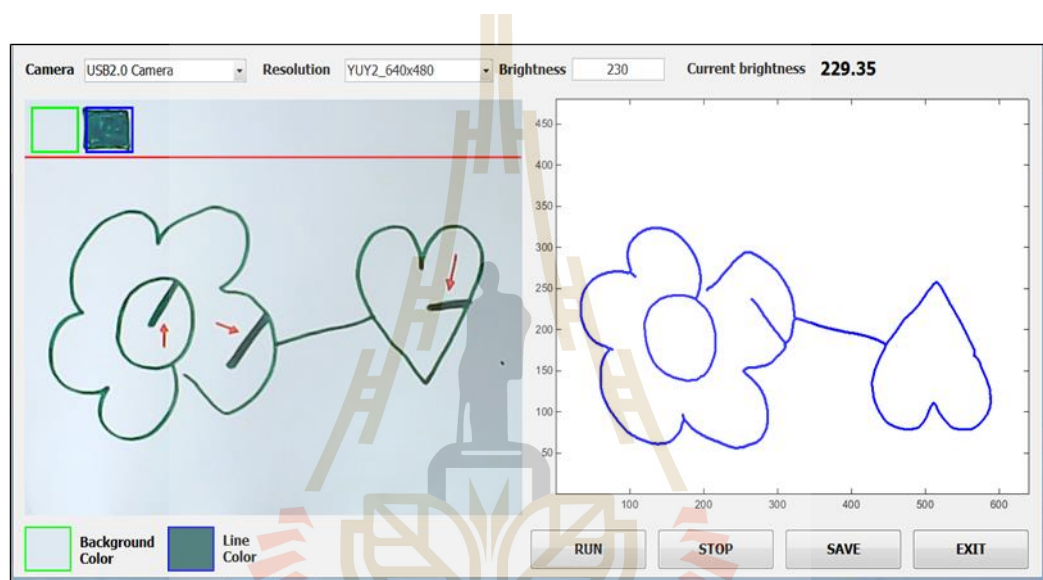
รูปที่ 4.6 แสดงขีดจำกัดข้อแรกของโปรแกรมประมวลผลภาพ

หากต้องการจะแก้ปัญหาลักษณะขีดจำกัดเบื้องต้นของโปรแกรมประมวลผลภาพ โดยต้องการให้แสดงทั้งสองรูปย่อย คือ ทั้งรูปดอกไม้ และรูปหัวใจ จะต้องสร้างความต่อเนื่องให้กับรูปต้นแบบ โดยลากเส้นหากัน ดังรูปที่ 4.7



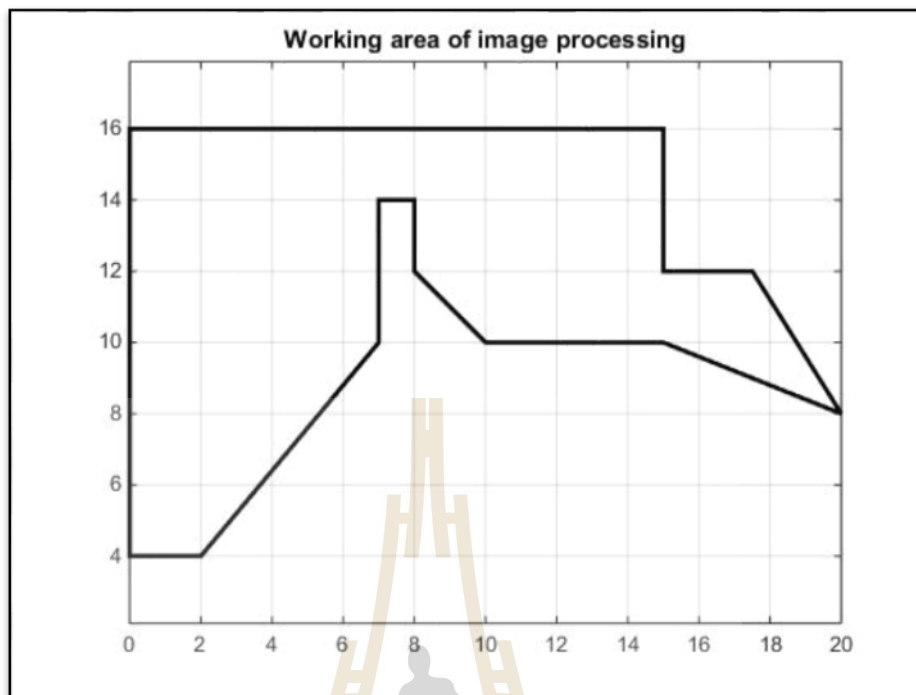
รูปที่ 4.7 แสดงขีดจำกัดข้อที่สองของโปรแกรมประมวลผลภาพ

4.1.2.2 รูปใหม่ที่สร้างจากกระบวนการประมวลผลนั้น จะไม่สามารถแสดงผลรูปใดก็ตามที่อยู่ภายในพื้นที่ปิด ดังรูปที่ 4.8 ซึ่งจากรูปต้นแบบสังเกตว่าในรูปดอกไม้ มีทั้งส่วนที่เป็นพื้นที่เปิด และพื้นที่ปิด พื้นที่ปิดก็คือบริเวณวงกลมด้านในของรูปดอกไม้และหัวใจ ผู้วิจัยทดลองวาดเส้นด้านในที่มีลูกศรสีแดงชี้ เมื่อ โปรแกรมประมวลผลภาพออกมาใหม่ ภาพที่ได้จะไม่แสดงเส้นที่อยู่ด้านในในพื้นที่ปิด ส่วนอีกจุดหนึ่งคือ บริเวณพื้นที่เปิดในส่วนของกลีบดอกไม้ ซึ่งจะเห็นว่าผู้วิจัยไม่ได้วาดให้เป็นพื้นที่ปิดเพราะเหลือที่ว่างเล็กน้อย ทำให้ส่วนนั้นเป็นพื้นที่เปิด จึงแสดงผลเส้นที่วาดในพื้นที่เปิดออกมาได้



รูปที่ 4.8 แสดงการแก้ปัญหาเบื้องต้นของขีดจำกัดข้อแรกของโปรแกรมประมวลผลภาพ

4.1.2.3 ขีดจำกัดข้อสุดท้าย คือ นักออกแบบจะไม่สามารถวาดรูปในพื้นที่สำหรับวาดรูปทั้งหมดได้ เนื่องจากว่า เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติมีขีดจำกัดในการทำงาน ดังนั้นหากวาดนอกเหนือขอบเขตที่หุ่นยนต์สามารถทำงานได้แล้วนั้น ถึงแม้ไม่มีผลต่อการประมวลผลภาพใด ๆ แต่จะทำให้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ ไม่สามารถทำงานได้ ดังที่สรุปในบทที่ 3 ขั้นตอนการวิจัย และพื้นที่สำหรับการใช้ในการออกแบบ ดังรูปที่ 4.9

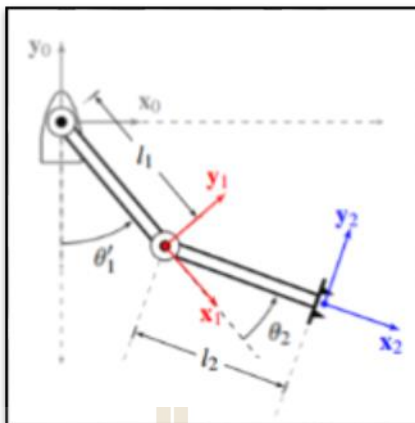


รูปที่ 4.9 แสดงพื้นที่สำหรับการออกแบบ

ดังนั้นการออกแบบรูปต้นแบบนั้น ควรจะอยู่ในขอบเขตที่เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติสามารถทำงานได้ โดยผู้วิจัยจะทำการออกแบบเฉพาะรูปเส้นขอบนอกเท่านั้น ต้องเป็นรูปที่ต่อเนื่อง และอยู่ในขอบเขตที่หุ่นยนต์สามารถทำงานได้

4.2 การทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

เครื่องต้นแบบที่ใช้ในงานวิจัยนี้ได้สร้างขึ้นโดยใช้หลักการและโครงสร้างของหุ่นยนต์แขนกลระนาบสองมิติ หรือ รู้จักในชื่อ 2-R Planar Robot โดยจะเป็นหุ่นยนต์ที่มี 2 แขน 2 ข้อต่อ โดยข้อต่อแต่ละข้อต่อจะสามารถทำงานแบบหมุนรอบแกน ดังรูปที่ 4.10



รูปที่ 4.10 แสดง Model ของหุ่นยนต์ 2-R Planar Robot

เพื่อให้เข้าใจการทำงาน และสร้างสมการจลนศาสตร์แบบไปข้างหน้า และจลนศาสตร์แบบผกผันสำหรับการทำงานของหุ่นยนต์แขนกลระนาบสองมิติ ผู้วิจัยจึงต้องทำการหาค่า D-H parameter ดังแสดงในตารางที่ 4.1 สำหรับใช้ในการแทนค่าลงในสมการจลนศาสตร์ไปข้างหน้า และจลนศาสตร์ผกผัน

ตารางที่ 4.1 แสดง D-H Parameter สำหรับ 2-R Planar Robot

Link	a_i	(α_i)	(d_i)	(θ_i)
1	l_1	0	0	θ_1
2	l_2	0	0	θ_2

จากตารางค่า D-H Parameter ได้ทำการกำหนดค่าคงที่แทนตัวแปรขึ้นมา โดยค่านี้จะเป็นค่าพื้นฐานสำหรับออกแบบและสร้างเครื่องต้นแบบแขนกลระนาบสองมิติต่อไป

กำหนดให้

l_1 คือแขนหลักของหุ่นยนต์แขนกลระนาบสองแกนมีความยาว 11 เซนติเมตร

l_2 คือแขนรองของหุ่นยนต์แขนกลระนาบสองแกนมีความยาว 10 เซนติเมตร

θ_1 คือมุมที่แขนหลักสามารถเคลื่อนที่ไปได้ โดยแขนหลักสามารถเคลื่อนที่ตั้งแต่มุม 0° ไปจนถึง 90°

θ_2 คือมุมที่แขนหลักสามารถเคลื่อนที่ไปได้ โดยแขนหลักสามารถเคลื่อนที่ตั้งแต่มุม 0° ไปจนถึง 150°

จากรูปแบบพื้นฐานของหุ่นยนต์แขนกลระนาบสองมิติ จะมีสมการจลนศาสตร์ไปข้างหน้าที่มีรูปทั่วไป ดังแสดงในสมการที่ 4.1

$$T_H^R = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

เมื่อนำค่าที่กำหนดแทนใน D-H parameter แล้วนำมาแทนค่าในสมการรูปแบบพื้นฐานของหุ่นยนต์แขนกลระนาบสองมิติ ทำให้ได้สมการของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติที่ใช้ในงานวิจัย แสดงดังสมการที่ 4.2

$$T_2^0 = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1c_1 + l_2c_{12} \\ s_{12} & c_{12} & 0 & l_1s_1 + l_2s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

ทำให้ตอนนี้เราสามารถทราบถึงสมการที่ใช้ในการหาพิกัดตำแหน่ง (x,y) ของเครื่องต้นแบบหุ่นยนต์ระนาบสองมิติ คือ

$$p_x = l_1c_1 + l_2c_{12} \quad (4.3)$$

$$p_y = l_1s_1 + l_2s_{12} \quad (4.4)$$

เมื่อเราสามารถทราบถึงสมการจลนศาสตร์แบบไปข้างหน้าของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติแล้ว จะทำให้เราไปแก้สมการเพื่อหาสมการจลนศาสตร์แบบย้อนกลับ เนื่องจากในงานวิจัยนี้ เมื่อผ่านกระบวนการประมวลผลภาพจะได้ตำแหน่งพิกัด (x,y) ออกมา เราจะนำมาข้อมูลตำแหน่งพิกัด (x,y) ชุดนั้นมาแทนค่ากลับในสมการจลนศาสตร์แบบย้อนกลับ เพื่อหามุมที่จะ

ไปควบคุมมอเตอร์ของหุ่นยนต์ให้ทำงานต่อไป โดยสมการจลนศาสตร์ย้อนกลับของเครื่องต้นแบบ หุ่นยนต์ระนาบสองมิติแสดงในสมการที่ 4.5 ถึง 4.8

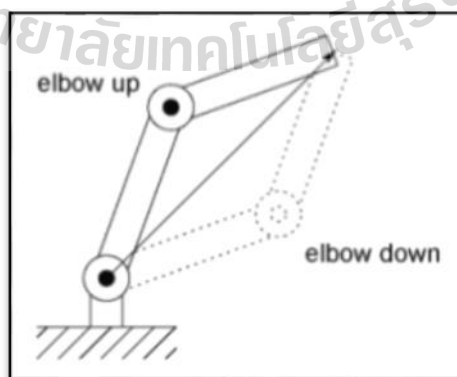
$$\theta_2 = -\text{atan} \frac{s_2}{c_2} \quad (4.5)$$

$$\theta_1 = \text{atan} \frac{+p_x l_2 s_2 + p_y (l_1 + l_2 c_2)}{p_x (l_1 + l_2 c_2) - p_y (l_2 s_2)} \quad (4.6)$$

$$\theta_2 = +\text{atan} \frac{s_2}{c_2} \quad (4.7)$$

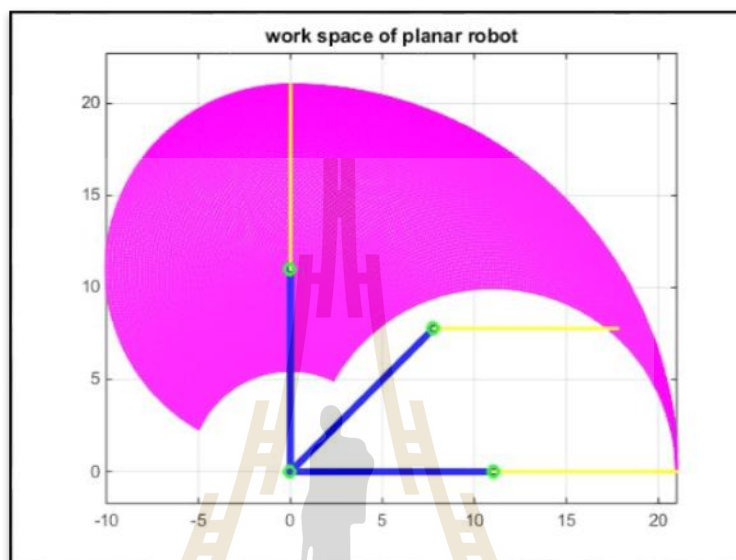
$$\theta_1 = \text{atan} \frac{-p_x l_2 s_2 + p_y (l_1 + l_2 c_2)}{p_x (l_1 + l_2 c_2) + p_y (l_2 s_2)} \quad (4.8)$$

จากสมการที่ 4.5 ถึง 4.8 สังเกตเห็นว่าเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติจะมีสมการคำนวณหามุมของแต่ละแกนจำนวนสองสมการ เนื่องจากว่าหุ่นยนต์สามารถเคลื่อนที่ไปหาตำแหน่งพิกัด (x,y) นั้น ๆ ได้ในสองทิศทาง คือทิศทางแขนแบบข้อศอกด้านบน และข้อศอกด้านล่าง ดังรูปที่ 4.11 โดยทิศทางแขนแบบข้อศอกด้านบนจะใช้สมการที่ 4.5 และ 4.6 และโดยทิศทางแขนแบบข้อศอกด้านล่างจะใช้สมการที่ 4.7 และ 4.8 โดยในงานวิจัยนี้ผู้วิจัยได้เลือกใช้ทิศทางแขนแบบข้อศอกด้านล่างเนื่องจากขีดจำกัดของหุ่นยนต์ที่ได้ตั้งไว้ว่าที่แต่ละมุมของแต่ละแกนจะต้องมีมุมที่เป็นบวก ไม่ติดลบนั่นเอง



รูปที่ 4.11 แสดงท่าทางของแขนหุ่นยนต์แขนกลระนาบสองมิติ

หลังจากทราบสมการจลนศาสตร์แบบไปข้างหน้าและย้อนกลับของเครื่องต้นแบบหุ่นยนต์ 2 มิติแล้วนั้น ทำให้ผู้วิจัยสามารถหาพื้นที่การทำงานของเครื่องต้นแบบหุ่นยนต์ 2 มิติได้ ดังรูปที่ 4.12 โดยพื้นที่สีชมพูเป็นขอบเขตที่เครื่องต้นแบบหุ่นยนต์ 2 มิติ สามารถทำงานได้ จะไม่สามารถทำงานออกเหนือจากพื้นที่นี้ได้



รูปที่ 4.12 แสดงพื้นที่การทำงานของเครื่องต้นแบบหุ่นยนต์ 2 มิติ

นอกจากแกนกลของหุ่นยนต์นั้น มอเตอร์ก็เป็นส่วนที่สำคัญไม่น้อยกว่าอื่น ๆ โดยมอเตอร์ที่ผู้วิจัยเลือกใช้ คือ สเต็ปเปอร์มอเตอร์ของบริษัท VEXTA รุ่น ASM34AK-H50 ดังรูปที่ 4.13 จำนวนสองตัว



รูปที่ 4.13 แสดง VEXTA รุ่น ASM34AK-H50

การควบคุมการหมุนของสเตปเปอร์มอเตอร์นั้นผู้วิจัยไม่สามารถกำหนดค่าพารามิเตอร์เองได้ เนื่องจากเป็นขีดจำกัดของบริษัทผู้ผลิต

ตารางที่ 4.2 ค่าพารามิเตอร์ของสเตปเปอร์มอเตอร์

Gear Ratio (x:1)	50:1
Pulse/Revolution	0.036° / pulse
Power Input	12-24 VDC

แต่เนื่องด้วยยังไม่เป็นที่พึงพอใจของผู้วิจัย เนื่องด้วยผู้วิจัยต้องการให้สเตปเปอร์มอเตอร์ละเอียดกว่าค่าพารามิเตอร์พื้นฐาน ซึ่งจากเดิมการจะหมุนครบหนึ่งรอบ หรือ 360° นั้นจะต้องใช้สัญญาณ PWM จำนวน 2000 ครั้ง ดังนั้นผู้วิจัยเลือกเลือกใช้ชุดควบคุม หรือ Driver Stepping Motor มาช่วยในการควบคุมการหมุนของมอเตอร์อีกทางหนึ่ง โดยผู้วิจัยเลือกใช้ Driver รุ่น IM483 ดังรูปที่ 4.14



รูปที่ 4.14 แสดง Driver Stepping Motor : IM483

โดย Driver ตัวนี้จะช่วยให้สเตปเปอร์มอเตอร์มีความละเอียดมากยิ่งขึ้นผ่านการสั่งการผ่านไมโครคอนโทรลเลอร์ Arduino Nano megaAT328 เมื่อใช้ Driver IM483 ร่วมกับ สเตปเปอร์มอเตอร์จะทำมอเตอร์มีค่าพารามิเตอร์ที่เปลี่ยนไป ดังแสดงในตารางที่ 4.3

ตารางที่ 4.3 ค่าพารามิเตอร์ของสเตปเปอร์มอเตอร์ร่วมกับ Driver IM483

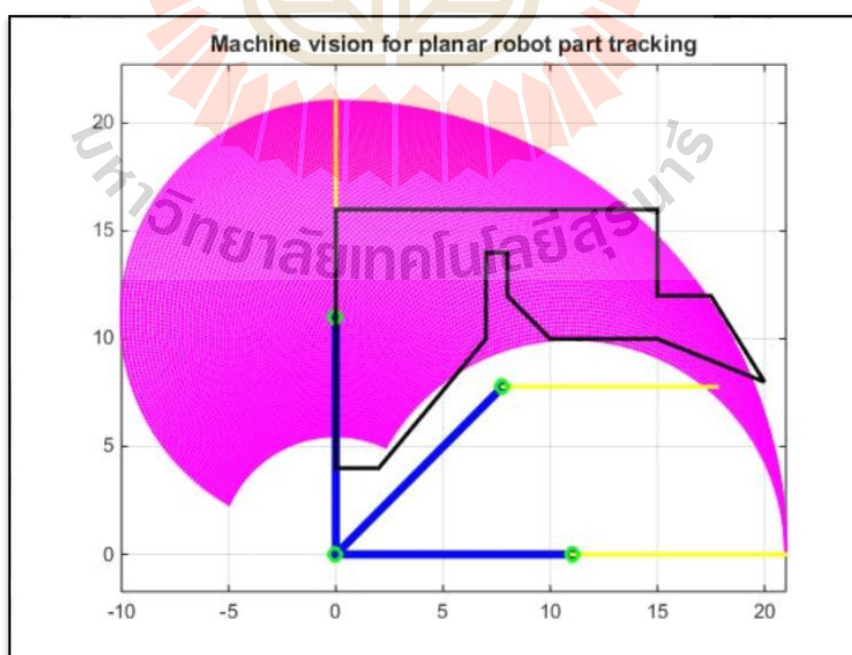
Gear Ratio (x:1)	50:1
Pulse/Revolution	0.00225° / pulse
Power Input	12-24 VDC

จากเดิมหมุนครบหนึ่งรอบ หรือ 360° นั้นจะต้องใช้สัญญาณ PWM จำนวน 2000 ครั้ง แต่เมื่อใช้ Driver IM483 มาควบคุมการหมุนของสเตปเปอร์มอเตอร์แล้ว ทำให้ละเอียดขึ้นมากกว่าเดิมอีก 80 เท่าเป็นหมุนครบหนึ่งรอบ หรือ 360° นั้นจะต้องใช้สัญญาณ PWM จำนวน 160000 ครั้ง

4.3 การทำงานร่วมกันระหว่างระบบการประมวลผลภาพและเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

งานวิจัยนี้จะสำเร็จได้ต่อเมื่อสามารถนำส่วนของระบบการประมวลผลภาพ และส่วนของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติมาสร้างระบบการทำงานร่วมกันอย่างมีประสิทธิภาพได้

ลำดับแรกจะต้องทราบถึงพื้นที่การทำงานของระบบประมวลผลภาพ และพื้นที่การทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติที่ใช้ร่วมกัน ดังรูปที่ 4.15



รูปที่ 4.15 แสดงพื้นที่การทำงานร่วมกันระหว่างระบบประมวลผลภาพและเครื่องต้นแบบ ๓

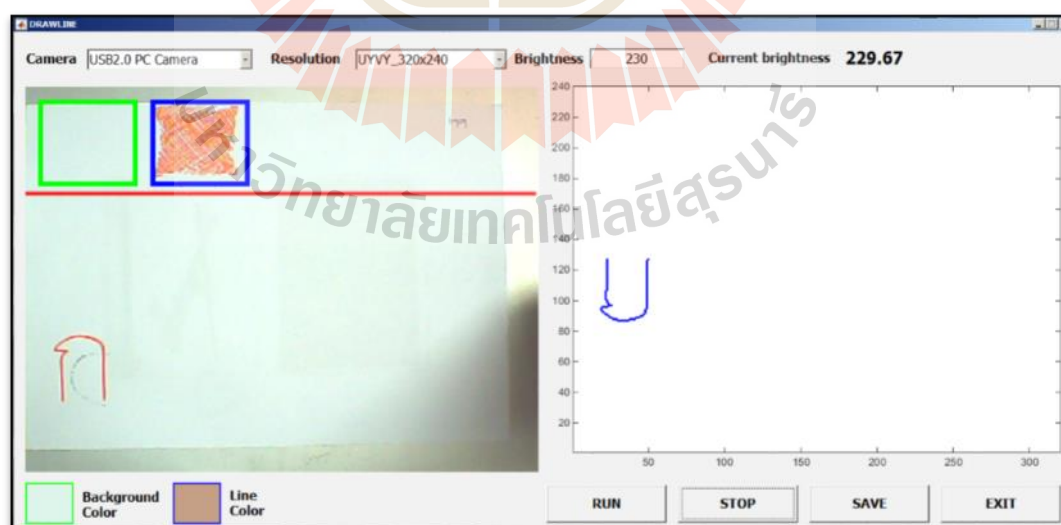
จากรูปที่ 4.15 นั้นแสดงให้เห็นว่าแม้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติจะสามารถทำงานได้ในขอบเขตที่แสดงในพื้นที่สี่มุมที่พบโดยแขนหลักของหุ่นยนต์แขนกลระนาบสองแกน (L_1) มีความยาว 11 เซนติเมตร แทนด้วยเส้นสีน้ำเงิน, แขนรองของหุ่นยนต์แขนกลระนาบสองแกน (L_2) มีความยาว 10 เซนติเมตรแทนด้วยเส้นสีเหลือง, มุมที่แขนหลัก (θ_1) สามารถเคลื่อนที่ไปได้ โดยแขนหลักสามารถเคลื่อนที่ตั้งแต่มุม 0° ไปจนถึง 90° และมุมที่แขนรอง (θ_2) สามารถเคลื่อนที่ไปได้ โดยแขนรองสามารถเคลื่อนที่ตั้งแต่มุม 0° ไปจนถึง 150°

ส่วนเส้นสีดำจะเป็นพื้นที่ที่ระบบประมวลผลภาพสามารถทำงานได้อย่างมีประสิทธิภาพ โดยถ้าหากเราทำการออกแบบในพื้นที่สีดำนั้น จะสามารถทำให้งานวิจัยนี้แสดงผลที่ทำงานตามจุดประสงค์ได้อย่างถูกต้องและแม่นยำ

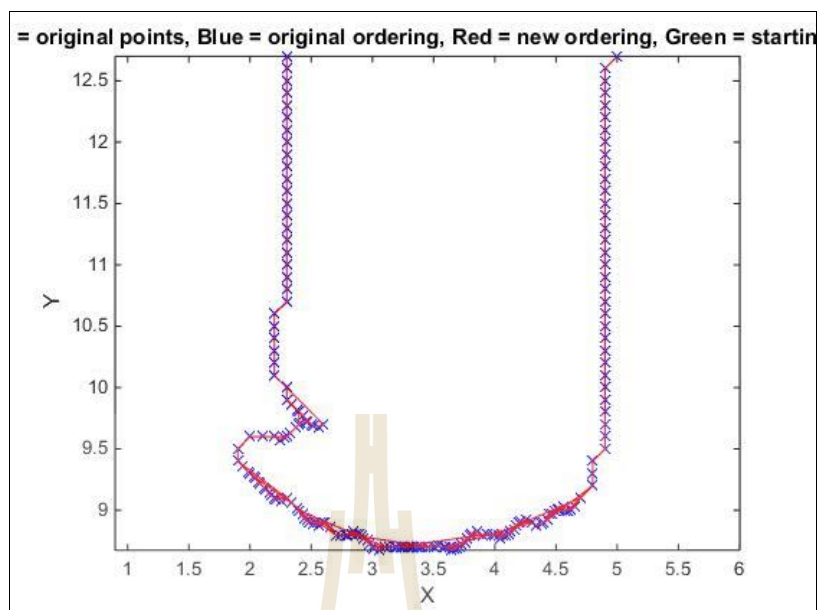
ดังนั้นในงานวิจัยนี้จึงถูกจำกัดที่ขอบเขตของเส้นสีดำ ซึ่งจะจำกัดพื้นที่การออกแบบและการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ ถึงแม้ความสามารถของเครื่องต้นแบบจะสามารถทำงานได้มากกว่านั้น แต่เพื่อความถูกต้อง และแม่นยำที่สุดจึงจะทำงานภายใต้ขอบเขตเฉพาะเส้นสีดำ

4.3.1 ผลการทำงานของระบบการประมวลผลภาพพร้อมกับการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

ตัวอย่างที่ 1 : ทดสอบการวาดรูปตัวอักษร ก.ไก่



รูปที่ 4.16 แสดงรูปต้นแบบ ก.ไก่ และภาพที่แสดงผ่านระบบประมวลผลภาพ



รูปที่ 4.17 แสดงรูป ก.ไก่ จากการนำชุดข้อมูลตำแหน่งพิกัดไปพล็อตเป็นรูป



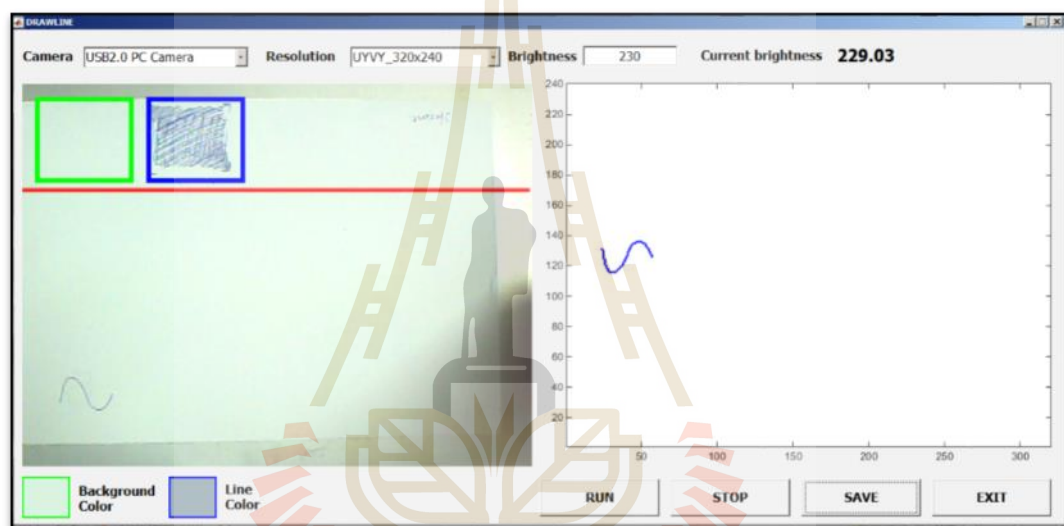
รูปที่ 4.18 แสดงรูป ก.ไก่ ที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

จากตัวอย่างที่ 1 : ทดสอบวาดรูป ก.ไก่ เมื่อแสดงผลการทำงานออกมาจะเห็นว่ารูปต้นแบบที่วาดได้นั้น จากรูปที่ 4.16 นั้นออกแบบเป็นรูป ก.ไก่ ที่สวยงาม แต่เมื่อเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติวาดรูปออกมาแล้ว จะเห็นว่ารูปที่วาดได้ออกมาไม่มีความราบเรียบ ดังเช่นรูปจริง ดังรูปที่ 4.18 ผู้วิจัยจึงได้พยายามหาคำตอบว่าเกิดจากสิ่งใด

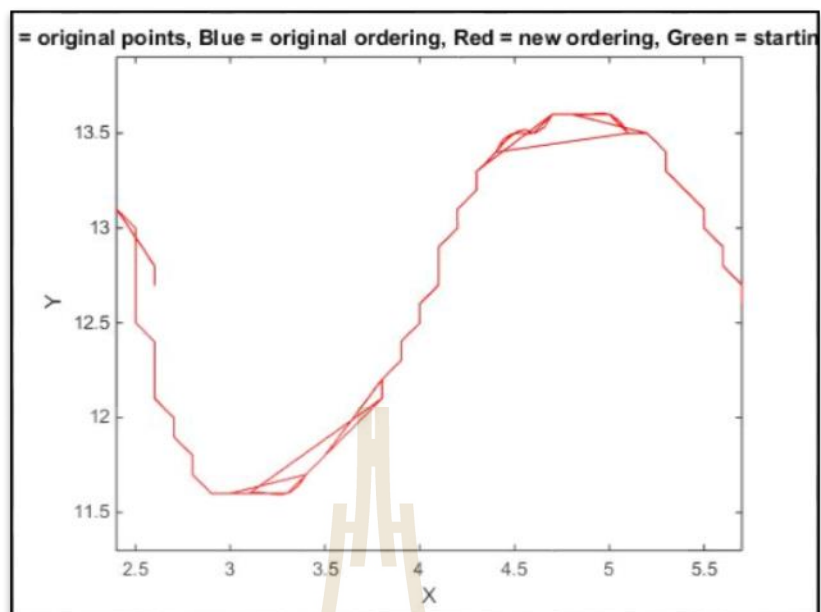
เมื่อผู้วิจัยได้นำตำแหน่งพิกัด (x,y) ใหม่ที่ได้มาจากการประมวลผลผ่านระบบประมวลผลภาพมาทำการพล็อตกราฟเพื่อสร้างรูปขึ้นมา ดังรูปที่ 4.17 เพื่อตรวจสอบ พบว่ารูปที่ได้จากการวาดด้วยเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติมีความเหมือนกับรูปที่ทดสอบโดยนำตำแหน่งพิกัด (x,y) มาพล็อตกราฟ

ทำให้ผู้วิจัยสามารถสรุปได้ว่า กล้องที่ใช้ในการประมวลผลภาพมีความสำคัญอย่างมากเพราะกล้องที่ผู้วิจัยใช้นั้น มีความละเอียดที่ไม่สูงมาก เมื่อทำการถ่ายรูปทำให้รูปที่ได้ออกมาเป็นเส้นที่ไม่คมชัด ราบเรียบดังรูปต้นแบบ

ตัวอย่างที่ 2 : ทดสอบการวาดรูปเส้น โค้งกราฟไซน์ (Sine Curve)



รูปที่ 4.19 แสดงรูปต้นแบบเส้น โค้งกราฟไซน์ และภาพที่แสดงผ่านระบบประมวลผลภาพ



รูปที่ 4.20 แสดงรูปเส้นโค้งกราฟฟิซันจากการนำชุดข้อมูลตำแหน่งพิกัดไปพล็อตเป็นรูป

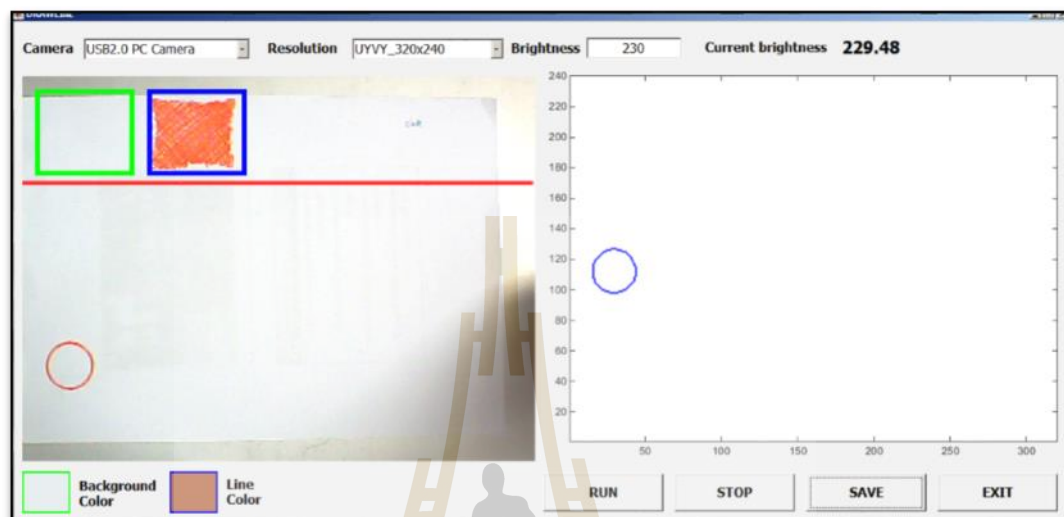


รูปที่ 4.21 แสดงรูปเส้นโค้งกราฟฟิซันที่วัดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

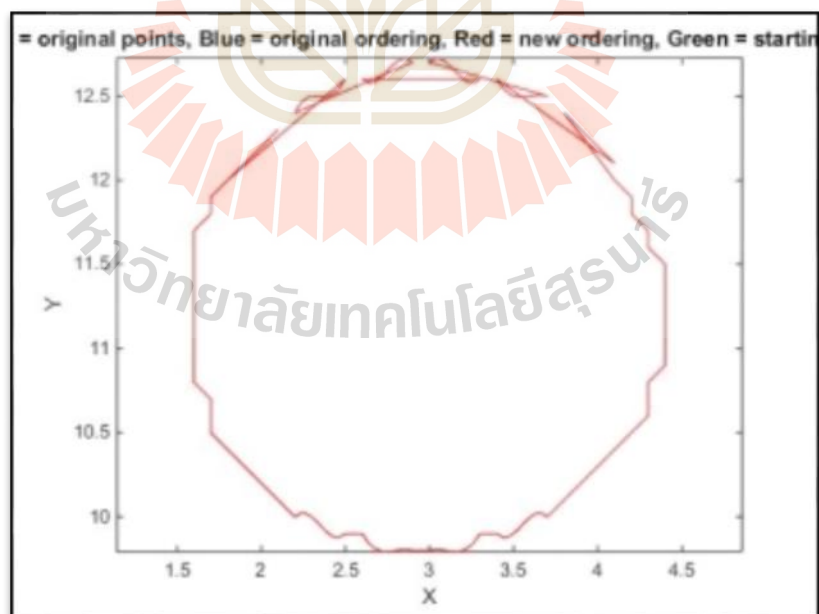
จากตัวอย่างที่ 2 : ทดสอบวาดรูปเส้นโค้งกราฟฟิซัน เมื่อแสดงผลการทำงานออกมา จะเห็นว่ารูปต้นแบบที่วาดได้นั้นก็เป็นดังเช่นตัวอย่างที่ 1 จากรูปต้นแบบ ดังรูปที่ 4.19 นั้นรูปต้นแบบเป็นเส้นราบเรียบ แต่เมื่อผ่านระบบประมวลผลภาพเส้นที่ได้ออกมาไม่ราบเรียบดังเช่นรูปต้นแบบ เมื่อนำชุดข้อมูลตำแหน่งพิกัด (x,y) ไปพล็อตกราฟได้ ดังรูปที่ 4.20 ออกมา เมื่อ

เครื่องต้นแบบหุ่นยนต์ระนาบสองมิติวาดรูปออกมา ดังรูปที่ 4.21 ก็ได้รูปที่มีความเหมือนกันกับรูปที่ผ่านการนำชุดข้อมูลไปพล็อตกราฟ

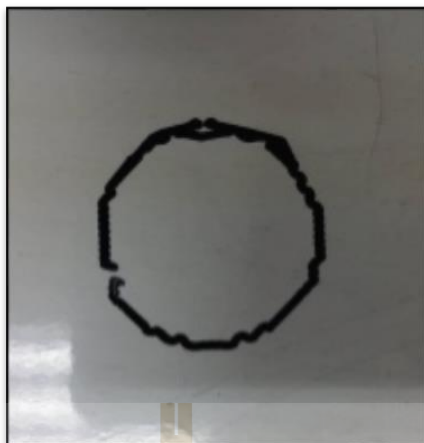
ตัวอย่างที่ 3 : ทดสอบการวาดรูปวงกลม



รูปที่ 4.22 แสดงรูปต้นแบบรูปวงกลม และภาพที่แสดงผ่านระบบประมวลภาพ



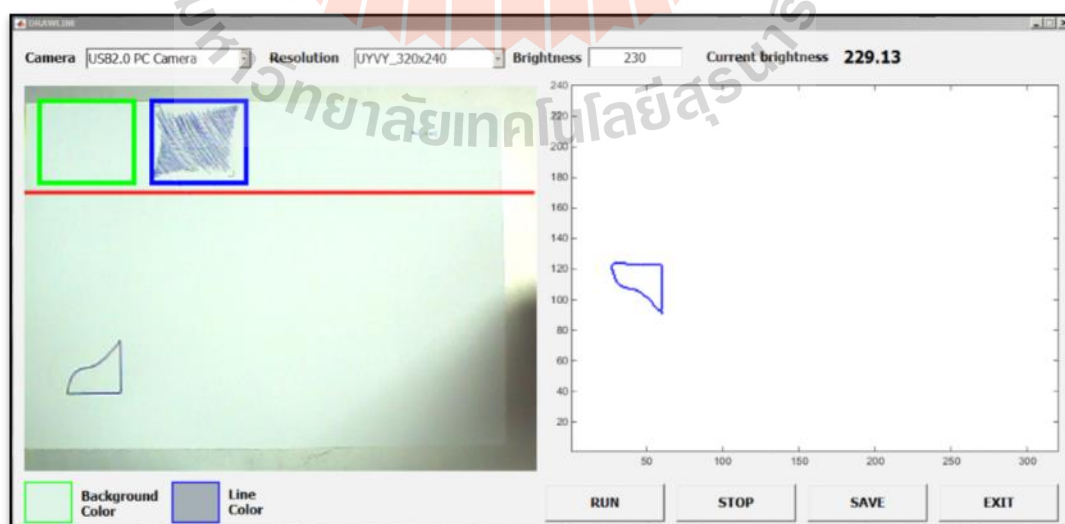
รูปที่ 4.23 แสดงรูปวงกลม จากการนำชุดข้อมูลตำแหน่งพิกัดไปพล็อตเป็นรูป



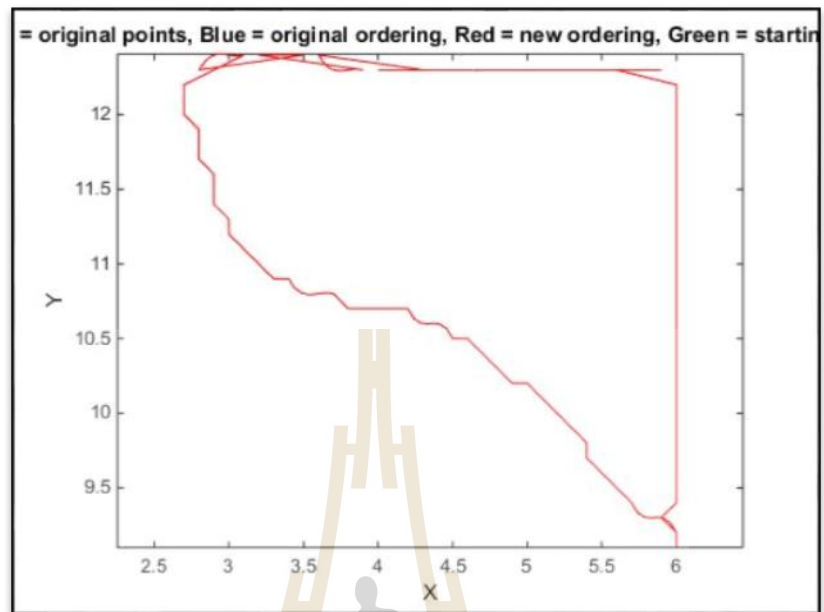
รูปที่ 4.24 แสดงรูปวงกลมที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

จากตัวอย่างที่ 3 : ทดสอบวาดรูปวงกลม เมื่อแสดงผลการทำงานออกมาจะเห็นว่ารูปต้นแบบที่วาดได้นั้นก็เป็นดังเช่นตัวอย่างที่ 1 และ 2 จากรูปต้นแบบที่แสดงดังรูปที่ 4.22 นั้นรูปต้นแบบเป็นเส้นราบเรียบ แต่เมื่อป้อนระบบประมวลผลภาพเส้นที่ได้ออกมาไม่ราบเรียบดังเช่นรูปต้นแบบ เมื่อนำชุดข้อมูลตำแหน่งพิกัด (x,y) ไปพล็อตกราฟได้ ดังรูปที่ 4.23 ออกมา เมื่อเครื่องต้นแบบหุ่นยนต์ระนาบสองมิติวาดรูปออกมา ดังรูปที่ 4.24 ก็ได้รูปที่มีความเหมือนกันกับรูปที่ผ่านการนำชุดข้อมูลไปพล็อตกราฟ

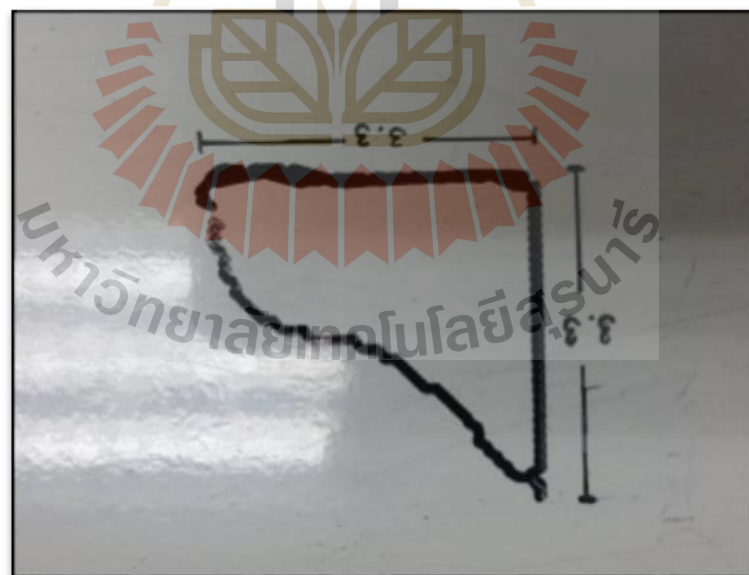
ตัวอย่างที่ 4 : ทดสอบการวาดเส้นโค้งและเส้นตรง



รูปที่ 4.25 แสดงรูปต้นแบบเส้นโค้งและเส้นตรง และภาพที่แสดงผ่านระบบประมวลผลภาพ



รูปที่ 4.26 แสดงรูปเส้นโค้งและเส้นตรงจากการนำชุดข้อมูลตำแหน่งพิกัดไปพล็อตเป็นรูป



รูปที่ 4.27 แสดงรูปเส้นโค้งและเส้นตรงที่วัดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

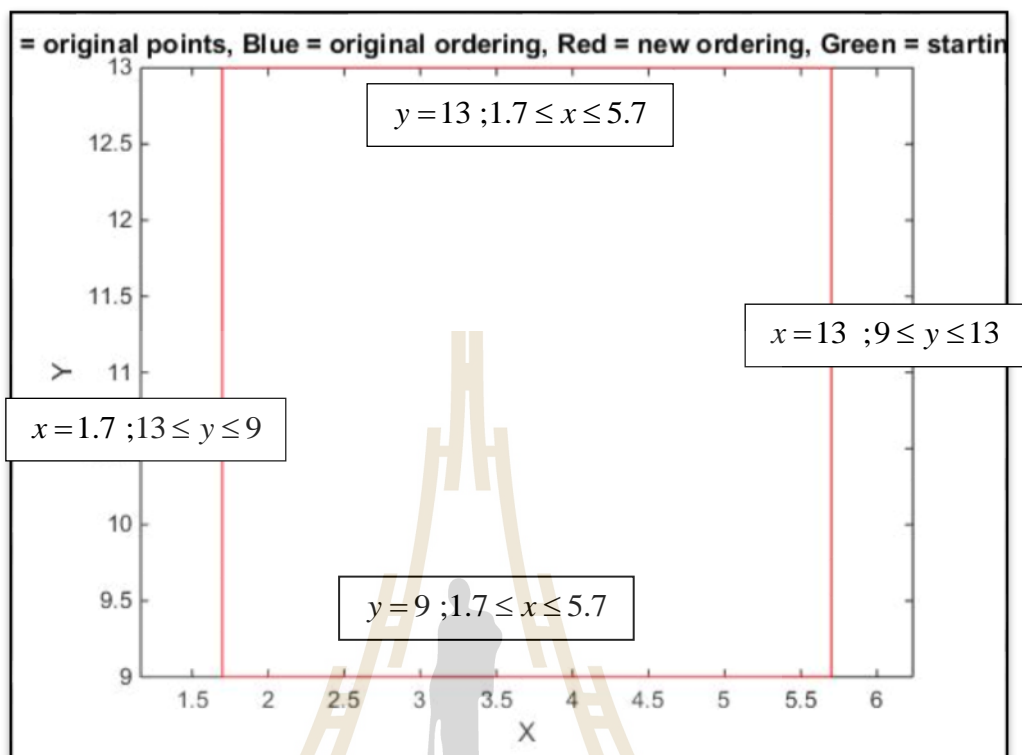
จากตัวอย่างที่ 4 : ทดสอบวาดรูปเส้น โค้งและเส้นตรงผลที่ออกมาก็เป็นเช่นเดิมดัง ตัวอย่าง 1-3 แต่ครั้งนี้ผู้วิจัยได้ทำการลองวัดขนาดเพื่อให้ผู้ที่ติดตามงานวิจัยได้เห็นว่า ขนาดของรูปใหม่ที่ได้ออกมานั้น ดังรูปที่ 4.27 มีขนาดเท่ากับรูปจริง ดังรูปที่ 4.25 และ 4.26

โดยผลจากการใช้อุปกรณ์เครื่องมืออย่างจำกัด ทำให้ประสิทธิภาพการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติและระบบประมวลผลภาพมีขีดจำกัด โดยหากต้องการให้ใหม่ที่วาดออกมาโดยเครื่องต้นแบบ มีขนาดเท่ากับรูปต้นแบบที่ออกแบบข้างต้น ก็จะต้องแลกมาด้วยการปรับความสูงของกล้องทำให้ความละเอียดและความชัดของกล้องน้อยลงไป โดยความสูงที่ทำให้ได้ขนาดที่มีค่าเท่ากับรูปต้นแบบคือตั้งกล้องที่ความสูง 50 เซนติเมตรเหนือรูปภาพ แต่ถ้าหากอยากได้ความคมชัดขึ้นก็สามารถทำได้โดยลดความสูงลงมา รูปที่เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติวาดได้จะราบเรียบขึ้น เนื่องจากกล้องจับภาพได้ชัดเจนขึ้น แต่ขนาดของรูปภาพใหม่ที่ถูกวาดด้วยเครื่องต้นแบบแขนกลระนาบสองมิติก็จะมีขนาดที่ใหญ่ขึ้นด้วย และอาจจะวาดรูปเกินขอบเขตที่จะสามารถรับได้

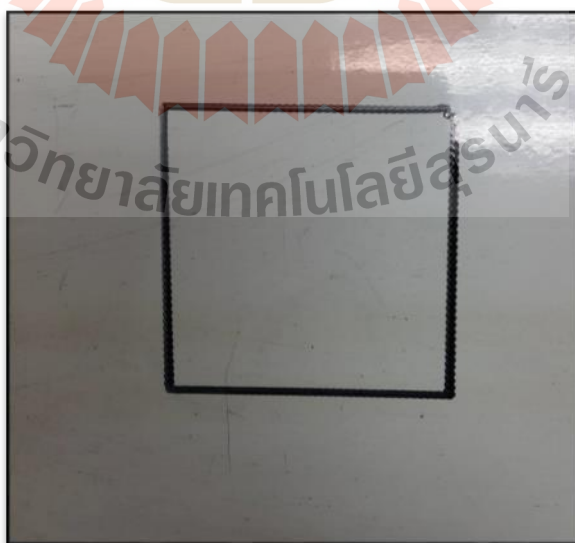
4.3.2 ผลการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติโดยไม่ผ่านระบบการประมวลผลภาพ

จากหัวข้อ 4.3.1 ข้างต้น ผู้วิจัยต้องการพิสูจน์ว่าเป็นจริงตามที่ตั้งสมมติฐานไว้หรือไม่ว่า การที่เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติจริง ๆ แล้วสามารถเขียนรูปที่มีประสิทธิภาพได้ แต่จากตัวอย่างที่ 1-4 นั้น การที่ไม่สามารถวาดรูปที่เป็นเส้นที่ราบเรียบนั้นเกิดมาจากการที่ระบบประมวลผลภาพสร้างตำแหน่งพิกัด (x,y) ที่ได้จากการถ่ายรูปด้วยกล้องที่ความละเอียดไม่สูงมาก จึงทำการทดสอบดังตัวอย่างข้างล่างดังนี้

ตัวอย่างที่ 5 : ทดลองวาดรูปสี่เหลี่ยม



รูปที่ 4.28 แสดงรูปสี่เหลี่ยมที่สร้างจากสมการ

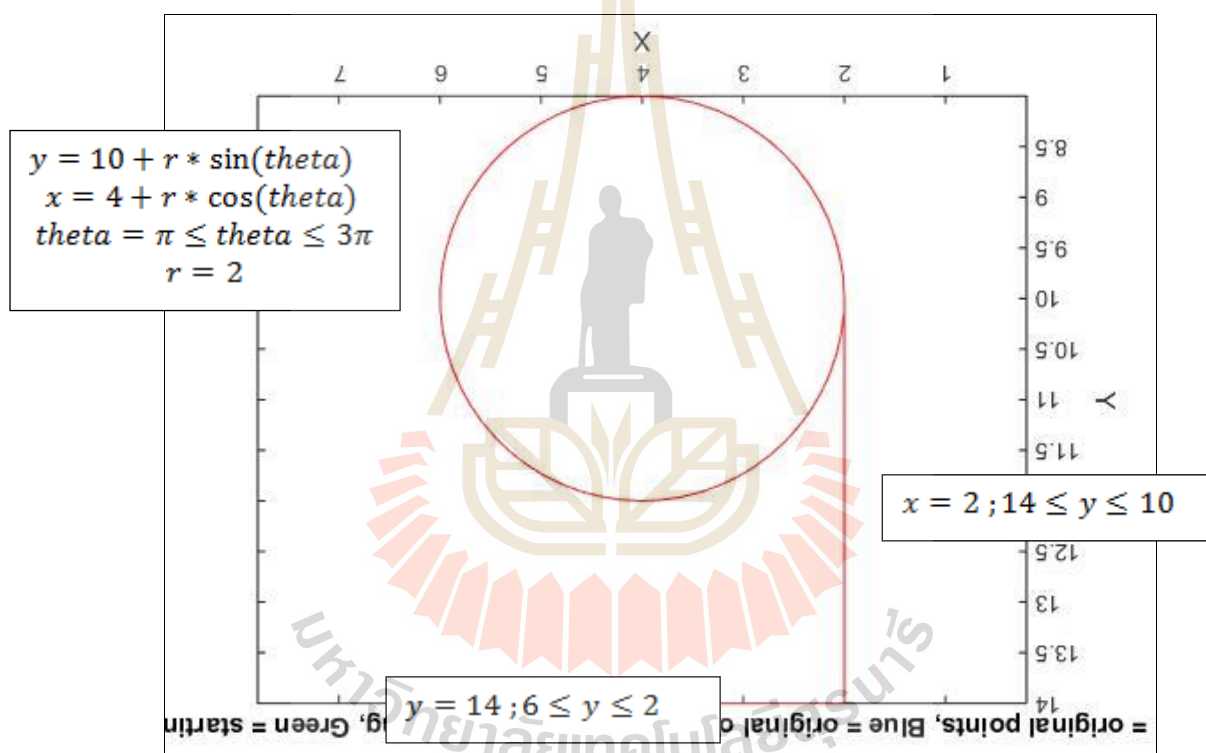


รูปที่ 4.29 แสดงรูปสี่เหลี่ยมที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

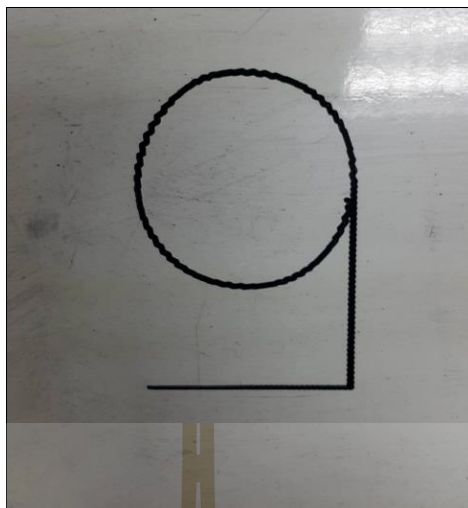
จากตัวอย่างที่ 5 ทดลองวาดรูปสี่เหลี่ยม แต่ในส่วนนี้จะแตกต่างจาก ตัวอย่างที่ 1-4 ข้างต้น เพราะในตัวอย่างที่ 5 เป็นการกำหนดจุดของรูปภาพขึ้นมาจากสมการเส้นตรงและนำตำแหน่งพิกัด (x,y) เหล่านั้นไปสั่งให้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติวาดรูปตาม

ซึ่งการทดลองนี้ก็จริงตามสมมติฐานที่ผู้วิจัยตั้งไว้ เพราะจากรูปต้นแบบที่ได้จากการสร้างสมการเส้นตรงสี่เส้นประกอบกันเป็นรูปสี่เหลี่ยม ดังรูปที่ 4.28 นั้น เมื่อนำตำแหน่งพิกัดเหล่านั้นไปป้อนให้เครื่องต้นแบบแขนกลระนาบสองมิติทำงาน ผลที่ออกมาแสดงในรูปที่ 4.29 สามารถวาดรูปออกมาเป็นเส้นที่ราบเรียบได้ตามต้องการ

ตัวอย่างที่ 6 : ทดลองวาดรูปเลขเก้า



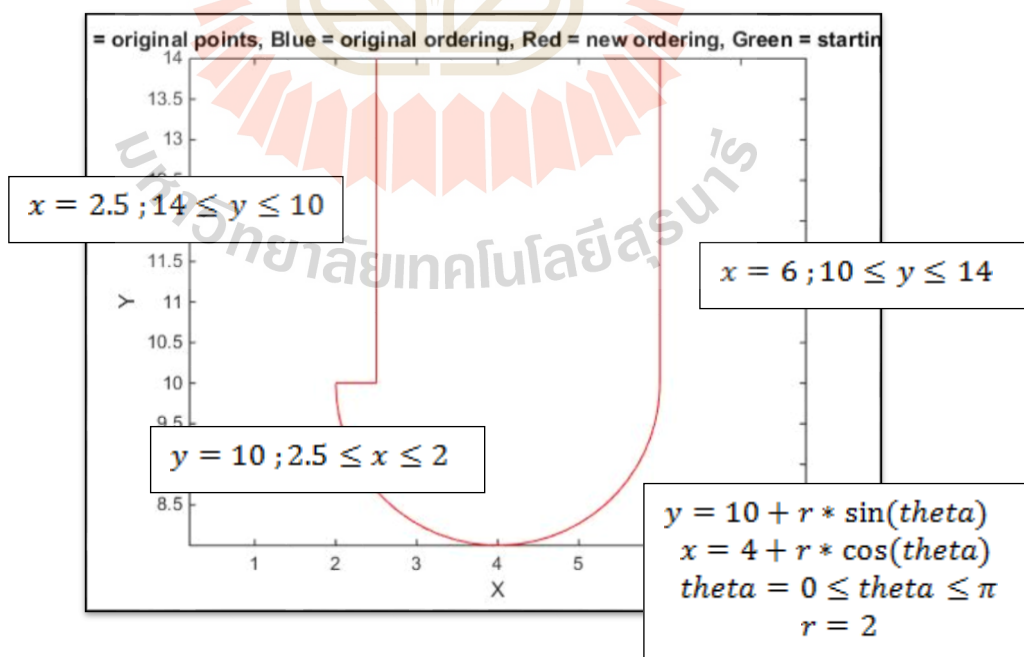
รูปที่ 4.30 แสดงรูปเลขเก้าที่สร้างจากสมการ



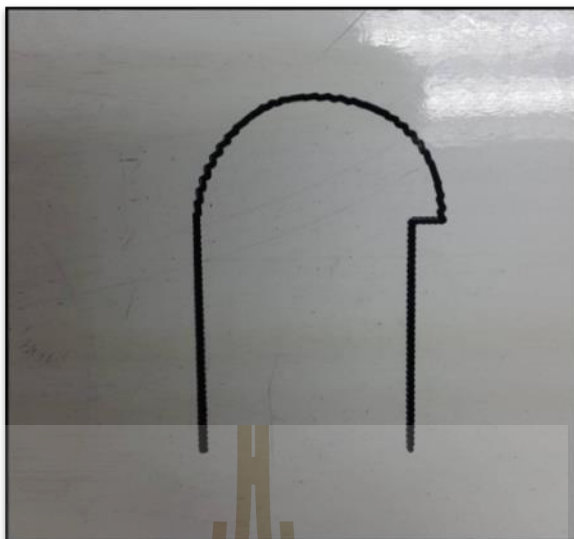
รูปที่ 4.31 แสดงรูปเลขเก้าที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

จากตัวอย่างที่ 6 ทดลองวาดรูปเลขเก้า ซึ่งในรูปนี้จะประกอบไปด้วยส่วนของเส้นโค้ง และเส้นตรง ซึ่งจากรูปต้นแบบ ดังรูปที่ 4.30 สามารถทำให้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติวาดรูปออกมา ดังรูปที่ 4.31 ได้เป็นอย่างดี

ตัวอย่างที่ 7 : ทดลองวาดรูป ก.ไก่



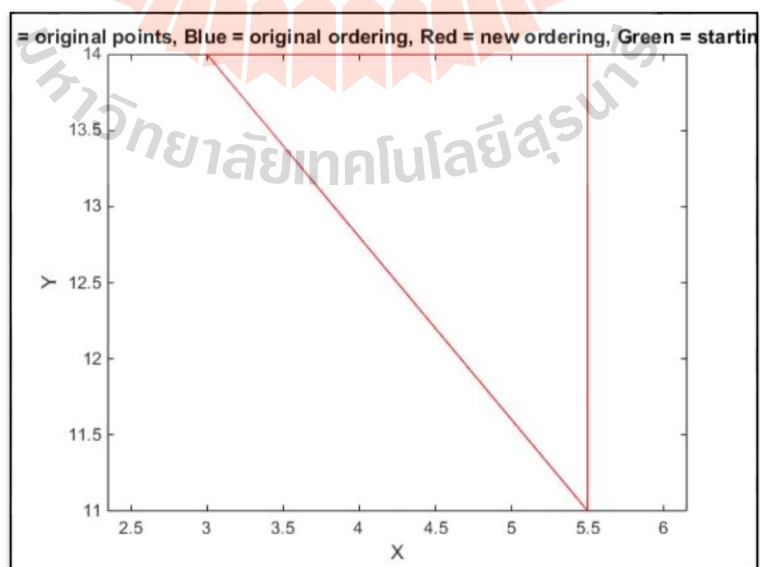
รูปที่ 4.32 แสดงรูป ก.ไก่ ที่สร้างจากสมการ



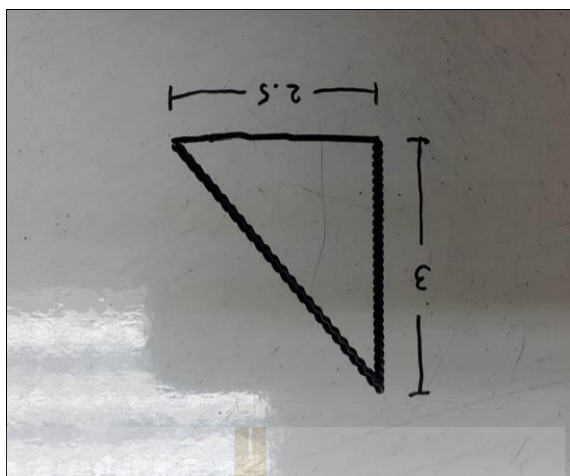
รูปที่ 4.33 แสดงรูป ก. ไม้ ที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

จากตัวอย่างที่ 7 ทดลองวาดรูป ก. ไม้ ซึ่งในรูปนี้จะประกอบไปด้วยส่วนของเส้นโค้งและเส้นตรง เช่นตัวอย่างที่ 6 ก็ยังมีผลลัพธ์เช่นเดิม ดังรูปที่ 4.32 และ 4.33 โดยรูปที่วาดออกมาด้วยเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ เปรียบเทียบกับรูปจากภาพต้นแบบมีความเหมือนมาก เป็นที่น่าพอใจ

ตัวอย่างที่ 8 : ทดลองวาดรูปสามเหลี่ยม



รูปที่ 4.34 แสดงรูปสามเหลี่ยมที่สร้างจากสมการ



รูปที่ 4.35 แสดงรูปสามเหลี่ยมที่วาดผ่านเครื่องต้นแบบหุ่นยนต์แกนกระนาบสองมิติ

จากตัวอย่างที่ 8 ทดลองวาดรูปสามเหลี่ยม ผลลัพธ์ก็เป็นอย่างที่ 5-7 ที่ผ่าน มา เป็นที่น่าพึงพอใจมาก โดยในตัวอย่างที่ 8 นี้ ผู้วิจัยได้ทำการลองวัดขนาดของรูปสามเหลี่ยมที่ถู กวาดออกมาด้วยเครื่องต้นแบบหุ่นยนต์แกนกระนาบสองมิติ ปรากฏว่ามีขนาด ความยาวด้านฐาน 2.5 เซนติเมตร และความยาวด้านสูง 3 เซนติเมตร ซึ่งขนาดที่วาดออกมานั้นตรงกับรูปต้นแบบที่ สร้างขึ้น จึงสามารถสรุปได้ว่าเครื่องต้นแบบหุ่นยนต์แกนกระนาบสองมิติ มีความสามารถในการ วาดรูปที่มีความราบเรียบออกมาได้

4.4 ผลการทดสอบเวลาการทำงานของงานวิจัย

ผลการทดสอบด้านเวลาจะแบ่งออกเป็นสามช่วงเวลา อันได้แก่

4.4.1 ช่วงแรกคือช่วงของการออกแบบ จะเป็นช่วงที่เริ่มตั้งแต่เปิดโปรแกรมประมวลผล ภาพ ออกแบบรูปต้นแบบ จนกระทั่งหยุดโปรแกรมประมวลผลภาพ

4.4.2 ช่วงต่อมาจะเป็นช่วงประมวลผลภาพและส่งข้อมูล จะเริ่มนับเวลาต่อจากการปิด โปรแกรมประมวลผลภาพ ระบบประมวลผลภาพก็จะส่งข้อมูลต่อไปสั่งงานให้มอเตอร์ทำงาน

4.4.3 ช่วงสุดท้ายช่วงทำงานของเครื่องต้นแบบหุ่นยนต์แกนกระนาบสองมิติ

ผลในการทดสอบเวลาการทำงานของงานวิจัยนี้จะแสดงโดยหาค่าเฉลี่ยของเวลาของการ ทำงานออกมาแสดงให้เห็น ดังแสดงในตารางที่ 4.4

ตารางที่ 4.4 แสดงเวลาทำงานเฉลี่ยของงานวิจัย

ตัวอย่างที่	จำนวนพิกัด ตำแหน่งของ ชุดข้อมูล	ผลการทดสอบเวลาในการทำงาน (วินาที)			
		เวลาช่วง ที่ 1	เวลาช่วงที่ 2	เวลาช่วงที่ 2 / จำนวน พิกัดตำแหน่ง	เวลาช่วงที่ 3
1	268	26.125	593.622	2.215	97.258
2	143	14.584	312.169	2.183	65.751
3	128	13.785	271.232	2.119	62.000
4	167	18.016	353.539	2.117	71.750
5	167	25.874	353.706	2.118	71.750
6	144	15.327	304.870	2.117	66.006
7	119	13.750	257.970	2.168	58.584
8	81	10.870	182.574	2.254	50.250
เฉลี่ย		17.291	328.710	2.161	67.919

จากผลการทดสอบเรื่องเวลาการทำงานของงานวิจัยนี้ได้ผลว่า

ช่วงที่ 1 : ช่วงของการออกแบบ จะเป็นช่วงที่เริ่มตั้งแต่เปิดโปรแกรมประมวลผลภาพ ออกแบบรูปต้นแบบ จนกระทั่งหยุดโปรแกรมประมวลผลภาพ จะใช้เวลาเฉลี่ยโดยประมาณ 18 วินาที ขึ้นอยู่กับความเร็วในการออกแบบ และความเร็วของตัวคอมพิวเตอร์ที่ใช้ในการเปิดและทำงานโปรแกรมประมวลผลภาพเป็นหลัก

ช่วงที่ 2 : ช่วงประมวลผลภาพและส่งข้อมูล จะเริ่มนับเวลาต่อจากการปิดโปรแกรมประมวลผลภาพ ระบบประมวลผลภาพก็จะส่งข้อมูลต่อไปสั่งงานให้มอเตอร์ทำงาน ซึ่งช่วงนี้จะ เป็นช่วงที่ใช้เวลาในการทำงานมากที่สุด เพราะการส่งข้อมูลของรูปต้นแบบแต่ละรูปใช้เวลานาน เนื่องจากข้อมูลมีปริมาณที่มาก ผู้วิจัยได้ทดลองทำการลดเวลาในการส่งข้อมูลลงแล้ว แต่ผลคือ ข้อมูลที่ถูกส่งออกไปนั้น ถูกส่งด้วยความเร็วมากเกินไปทำให้ไมโครคอนโทรลเลอร์ที่รับค่าของชุด ข้อมูลเหล่านั้นไม่สามารถอ่านค่าได้ทัน โดยเวลาเฉลี่ยของการทำงานในช่วงนี้จะคิดเป็นเวลาต่อ จำนวนพิกัดตำแหน่ง (x,y) เนื่องจากรูปต้นแบบแต่ละรูปนั้นมีชุดข้อมูลที่ไม่เท่ากัน โดยมีเวลาเฉลี่ย ต่อที่ 2.161 วินาทีต่อการส่งชุดข้อมูล 1 ชุด

ช่วงที่ 3 : ช่วงสุดท้ายช่วงทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ ช่วงนี้จะ ใช้เวลาในการทำงานที่ไม่นานมาก เนื่องจากชุดข้อมูลได้ถูกส่งมาและจดจำอยู่ในส่วนจดจำ เรียบร้อยแล้ว จึงใช้เวลาทำงานอย่างต่อเนื่องในเวลาสั้น ๆ เฉลี่ยประมาณ 68 วินาที

4.5 ผลการทดสอบความถูกต้องของงานวิจัย

ในส่วนของการทดสอบความถูกต้องนั้น ผู้วิจัยได้แบ่งการทดสอบออกเป็นสองส่วน ได้แก่ ส่วนแรกคืองานวิจัยทั้งระบบ ซึ่งประกอบด้วย ระบบการประมวลผลภาพ และการทำงานของ เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ และส่วนถัดมาคือ การทดสอบเฉพาะการทำงานของ เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติว่ามีความถูกต้องคิดเป็นเปอร์เซ็นต์ความผิดพลาด เท่าใด โดยคำนวณได้จากสูตร ดังสมการที่ 4.9

$$\% Error = \left| \frac{Experimental - Real}{Real} \right| \times 100\% \quad (4.9)$$

4.5.1 การทดสอบความถูกต้องทั้งระบบ

จากตัวอย่างที่ 4 : ทดสอบการวาดเส้นโค้งและเส้นตรง จากรูปที่ 4.27 ที่แสดงดังที่ ผ่านมาจะเห็นว่า จากรูปจริงมีความสูง คือ วัดได้จาก 9 เซนติเมตร ถึง 12.4 เซนติเมตร เท่ากับ 3.4 เซนติเมตร และมีความกว้าง คือ วัดได้จาก 2.7 เซนติเมตร ถึง 6 เซนติเมตร เท่ากับ 3.3 เซนติเมตร ซึ่งเมื่อโปรแกรมประมวลผลภาพทำการประมวลผลและสั่งให้เครื่องต้นแบบหุ่นยนต์แขนกล ระนาบสองมิติวาดรูปออกมา ได้ค่าออกมาดังนี้ คือ มีความสูง 3.3 เซนติเมตร และความกว้าง 3.3 เซนติเมตร

เมื่อนำค่าที่ได้จากการทดลองไปหาเปอร์เซ็นต์ความผิดพลาดโดยแทนค่าในสมการที่ 4.9 จะได้ดังนี้

- ความผิดพลาดของความสูง

$$\% Error = \left| \frac{Experimental - Real}{Real} \right| \times 100\%$$

$$\% Error = \left| \frac{3.3 - 3.4}{3.4} \right| \times 100\%$$

$$\% Error = 2.94\%$$

- ความผิดพลาดของความกว้าง

$$\% Error = \left| \frac{Experimental - Real}{Real} \right| \times 100\%$$

$$\% Error = \left| \frac{3.3 - 3.3}{3.3} \right| \times 100\%$$

$$\% Error = 0\%$$

จากตัวอย่างที่ 4 ทำให้เห็นว่าเมื่อทั้งระบบทำงานร่วมกันทั้งระบบการประมวลผลภาพ และการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ จะมีความผิดพลาดเกิดขึ้นในงานวิจัย ซึ่งจากการเก็บข้อมูลได้ผลว่า งานวิจัยนี้ทั้งระบบจะมีความผิดพลาดเกิดขึ้นแต่จะไม่เกิน 3%

4.5.2 การทดสอบความถูกต้องเฉพาะการทำงานของเครื่องต้นแบบ ฯ

ในส่วนของการทดสอบเฉพาะการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิตินั้น ผู้วิจัยได้ทำการสร้างสมการเพื่อสร้างเป็นรูปทรงแบบเรขาคณิตขึ้นมา โดยตัวอย่างที่จะนำมาแสดงในส่วนนี้ คือ ตัวอย่างที่ 8 : ทดลองวาดรูปสามเหลี่ยม ดังรูปที่ 4.35 แสดงดังที่ผ่านมาจะเห็นว่า จากรูปจริงมีความสูง คือ วัดได้จาก 11 เซนติเมตร ถึง 14 เซนติเมตร เท่ากับ 3 เซนติเมตร และมีความกว้าง คือ วัดได้จาก 3 เซนติเมตร ถึง 5.5 เซนติเมตร เท่ากับ 2.5 เซนติเมตร ซึ่งเมื่อโปรแกรมประมวลผลภาพทำการประมวลผลและสั่งให้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติวาดรูปออกมา ได้ค่าออกมาดังนี้ คือ มีความสูง 3 เซนติเมตร และความกว้าง 2.5 เซนติเมตร

เมื่อนำค่าที่ได้จากการทดลองไปหาเปอร์เซ็นต์ความผิดพลาดโดยแทนค่าในสมการที่ 4.9 จะได้ดังนี้

- ความผิดพลาดของความสูง

$$\% Error = \left| \frac{Experimental - Real}{Real} \right| \times 100\%$$

$$\% Error = \left| \frac{3 - 3}{3} \right| \times 100\%$$

$$\% Error = 0\%$$

- ความผิดพลาดของความกว้าง

$$\% Error = \left| \frac{Experimental - Real}{Real} \right| \times 100\%$$

$$\% Error = \left| \frac{2.5 - 2.5}{2.5} \right| \times 100\%$$

$$\% Error = 0\%$$

จากตัวอย่างที่ 8 ทำให้เห็นว่าเมื่อทำการตัดระบบการประมวลผลภาพออกไปแล้วนั้น เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ สามารถทำงานได้อย่างมีประสิทธิภาพมากขึ้น ซึ่งจากการเก็บข้อมูล ได้พบว่า ความผิดพลาดที่เกิดขึ้นในเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติมีความผิดพลาดที่น้อยมากหรือแทบจะไม่มีผิดพลาดเกิดกับเครื่องต้นแบบ คิดเป็นความผิดพลาด 0%

4.6 สรุปผลการทดสอบ

จากการทดสอบการทำงานของงานวิจัยนี้ ตั้งแต่การทำงานของระบบประมวลผลภาพ, การทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ, ระบบการทำงานร่วมกันของระบบประมวลผลภาพและเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ และการทดสอบเวลาในการทำงานของงานวิจัยได้ผลออกมาว่า ระบบประมวลผลภาพสามารถจับภาพใด ๆ ก็ตามที่ถูกกำหนดด้วยสีของปากกาที่ตรงกับภาพต้นแบบได้ และสร้างรูปเสมือนขึ้นมาจากหลักการแปลงรูปต้นแบบเป็นพิกเซล และยังสามารถส่งข้อมูลไปให้เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติได้

ในการทำงานของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติสามารถทำงานได้อย่างมีประสิทธิภาพ โดยขีดความสามารถของหุ่นยนต์สามารถวาดรูปใด ๆ ก็ตามที่อยู่ในพื้นที่ทำงานได้อย่างดีเยี่ยม จากที่เห็นว่าแม้จะทำการนำตำแหน่งพิกัดที่เกิดจากการพล็อตสมการมาทำการสั่งให้เครื่องต้นแบบหุ่นยนต์ระนาบสองมิติวาดรูปนั้น บางส่วนยังไม่ราบเรียบ เป็นเพราะนี่คือขีดความสามารถสูงสุดที่เครื่องต้นแบบสามารถทำได้แล้วด้วยอุปกรณ์ที่มีอยู่อย่างจำกัด

การทดสอบเวลาในการทำงานวิจัย ได้ผลคือ ช่วงแรกช่วงการออกแบบนั้นจะใช้เวลามากหรือน้อยขึ้นอยู่กับความเร็วในการออกแบบ และความเร็วของตัวคอมพิวเตอร์ที่ใช้ในการเปิดและทำงาน โปรแกรมประมวลผลภาพเป็นหลัก มีเวลาเฉลี่ยประมาณ 18 วินาที ช่วงต่อมาช่วงประมวลผลภาพและการส่งข้อมูลเป็นช่วงที่ใช้เวลานานที่สุด โดยขึ้นอยู่กับว่ารูปต้นแบบจะมีจำนวนชุดข้อมูลตำแหน่งพิกัด (x,y) ที่ชุดข้อมูล ยิ่งภาพซับซ้อนก็จะใช้เวลานานมากขึ้น โดยใช้เวลาเฉลี่ยประมาณ 2.161 วินาทีต่อการส่งข้อมูลหนึ่งชุด สุดท้ายช่วงที่เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติทำงานใช้เวลาในการทำงานที่ไม่นานมาก เนื่องจากชุดข้อมูลได้ถูกส่งมาและจดจำอยู่ในส่วนจดจำเรียบร้อยแล้ว จึงใช้เวลาทำงานอย่างต่อเนื่องในเวลาสั้น ๆ เฉลี่ยที่ประมาณ 68 วินาที

สุดท้ายการทดสอบความถูกต้องในการทำงานวิจัยระบบการทำงานร่วมกันของระบบประมวลผลภาพและเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ ในส่วนนี้นั้นด้วยอุปสรรคของอุปกรณ์และเครื่องมือทดลองที่มีอยู่อย่างจำกัด ทำให้รูปภาพใหม่ที่ถูกวาดออกมาจากเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติยังไม่มีควมราบเรียบเหมือนรูปต้นแบบคิดเป็นความผิดพลาดประมาณไม่เกิน 3% ซึ่งผู้วิจัยได้หาต้นเหตุของปัญหาดังกล่าวออกมา สรุปได้ว่า เกิดจากระบบประมวลผลภาพ เพราะเมื่อลองทำการตัดในส่วนของระบบประมวลผลภาพออกไปนั้นเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ สามารถทำงานได้อย่างเต็มความสามารถ คิดเป็นความผิดพลาดที่น้อยมาก เท่ากับ 0% หรือแทบจะไม่มีควมผิดพลาดเกิดขึ้นในเครื่องต้นแบบหุ่นยนต์ระนาบสองมิติ



บทที่ 5

สรุปงานวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

งานวิจัยนี้เป็นการติดตามชิ้นงานของแขนกลระนาบสองมิติด้วยวิธีการมองเห็นของเครื่องจักร เพื่อออกแบบอัลกอริทึมของระบบการมองเห็นเครื่องจักร จัดสร้างเครื่องต้นแบบระดับห้องปฏิบัติการ และออกแบบระบบการทำงานร่วมกันระหว่างระบบการมองเห็นเครื่องจักรและเครื่องต้นแบบห้องปฏิบัติการ โดยผลการวิจัยแต่ละระบบสามารถทำงานได้ตามความวัตถุประสงค์ที่วางไว้ และยังสามารถทำงานร่วมกันได้อย่างเป็นที่น่าพึงพอใจมาก และจากผลการทดสอบความผิดพลาดปรากฏว่าในงานวิจัยนี้มีเปอร์เซ็นต์ความผิดพลาดประมาณไม่เกิน 3% ซึ่งผู้วิจัยได้หาต้นเหตุของปัญหาดังกล่าวออกมา สรุปได้ว่า เกิดจากระบบประมวลผลภาพ เพราะเมื่อลองทำการตัดในส่วนจากระบบประมวลผลภาพออกไปนั้น เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติสามารถทำงานได้อย่างเต็มความสามารถ คิดเป็นความผิดพลาดที่น้อยมาก เท่ากับ 0% หรือแทบจะไม่มีผิดพลาดเกิดขึ้นในเครื่องต้นแบบหุ่นยนต์ระนาบสองมิติ และระบบประมวลผลภาพในงานวิจัยนี้สามารถนำไปพัฒนาต่อยอดในอนาคตสามารถร่วมกับเครื่องจักรอัตโนมัติอื่น ๆ ได้ต่อไป

5.2 ข้อเสนอแนะ

5.2.1 ความละเอียดของกล้องที่นำมาใช้ ส่งผลต่อความแม่นยำในการตรวจสอบ หากนำกล้องที่มีความละเอียดสูงมาใช้งานความถูกต้องในการประมวลผลก็จะมากขึ้นด้วย และในงานที่ต้องการความแม่นยำสูงควรเลือกใช้กล้องที่มีความละเอียดสูง แต่ราคาก็จะสูงขึ้นตามไปด้วย

5.2.2 การจัดวางตำแหน่งของกล้องเพื่อการตรวจสอบ ก็จะมีผลต่อการตรวจสอบและการบอกตำแหน่งในการประกอบด้วย จึงต้องจัดวางตำแหน่งให้เหมาะสม ควบคุมพื้นที่ในการตรวจค้นหา หากจัดวางกล้องในตำแหน่งที่สูงเกินไปภาพที่ออกมาจะได้ภาพที่เล็กและไม่ชัดเจน แต่หากจัดวางกล้องใกล้ภาพต้นแบบมากเกินไปจะได้ภาพที่ชัดเจนดี แต่ขนาดที่ออกมาจะใหญ่เกินจริง

5.2.3 สเตปเปอร์มอเตอร์เป็นส่วนที่มีความสำคัญมากเพราะเป็นต้นกำลังที่ใช้ในการขับเคลื่อนเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ ดังนั้นการเลือกใช้มอเตอร์ที่มีความละเอียดมากขึ้นก็จะทำให้ได้งานที่มีความราบเรียบมากยิ่งขึ้น แต่ราคาก็จะสูงขึ้นตามไปด้วย

5.2.4 ไมโครคอนโทรลเลอร์รุ่นที่ใช้ในงานวิจัยมีขนาดเล็ก ซึ่งสามารถเพิ่มประสิทธิภาพในงานวิจัยได้ โดยเลือกใช้ไมโครคอนโทรลเลอร์รุ่นที่มีขนาดใหญ่กว่า ซึ่งจะทำให้สามารถวาดรูปที่ซับซ้อนมากขึ้นได้

5.2.5 งานวิจัยนี้ต้องการออกแบบในขอบเขตพื้นที่ที่กำหนดไว้เท่านั้น เพราะว่าสเปคเปออร์มอเตอร์ตัวหลักได้มีการกำหนดมาให้เคลื่อนที่ในขอบเขต 0 องศา ถึง 90 องศา

5.2.6 รูปภาพต้นแบบที่ทำการออกแบบนั้นจะต้องเป็นรูปภาพต่อเนื่องไม่สามารถวาดรูปหลาย ๆ รูปในรูปภาพต้นแบบเดียวกันได้

5.2.7 ภายในรูปปิดใด ๆ จะไม่สามารถแสดงผลออกมาได้ ดังนั้นงานวิจัยนี้จึงเน้นไปที่การวาดรูปกรอบ หรือขอบภายนอกเท่านั้น



รายการอ้างอิง

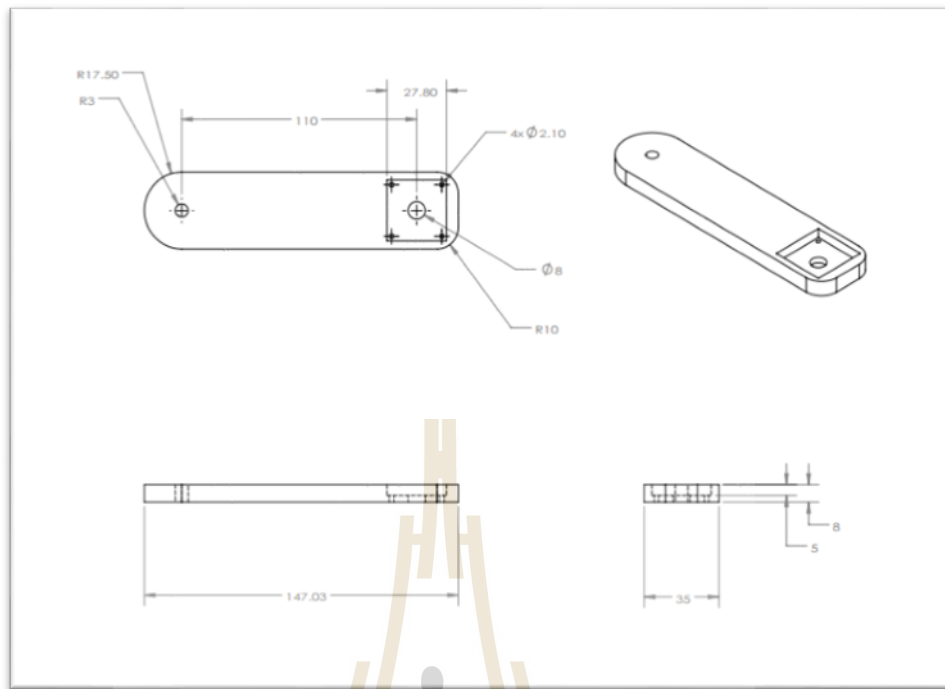
- Barhoumi EM., and Ben salah B. (2011). New Positioning Control of Stepper Motor using BP Neural Networks. **Journal of Emerging Trends in Computing and Information Sciences**, Volume 2, No. 6, JUNE 2011, ISSN 2079-8407
- Bhavana C. Bendale and Prof. Anil R. Karwankar. (2015). Moving Object Tracking in Video Using MATLAB. **International Journal of Electronics, Communication & Soft Computing Science and Engineering**, ISSN: 2277-9477, Volume 2, Issue 1
- Boyle, W.S., and Smith, G.E. (1970). **Charge Coupled Semiconductor Devices**. Manuscript received January 29,1970
- Frans W. Cornelissen and Enno M. Peters. (2002). The Eyelink Toolbox: Eye tracking with MATLAB and the Psychophysics Toolbox, **Behavior Research Methods, Instruments, & Computers**, 2002, 34(4), 613-617.
- Ivan Virgala, Michal Kelemen, Alexander Gmitterko and Tomas Liptak. (2015). Control of Stepper Motor by Microcontroller. **Journal of Automation and Control**, 2015, Vol. 3, No. 3, 131-134.
- Mahmoud Gouasmi, Mohammed Ouali, Brahim Fernini and M'hamed Meghatria. (2012). Kinematic Modelling and Simulation of a 2-R Robot Using Solidworks and Verification by MATLAB/Simulink. **International Journal of Advanced Robotic Systems**, Int J Adv Robotic Sy, 2012, Vol. 9, 245:2012
- Muzamil Bhat. (2014). Digital Image Processing. **International Journal of Scientific & Technology Research**, Volume 3, Issue 1, January 2014, ISSN: 2277-8616
- Pradyumna Deshpande and Prashasti Kanikar. (2012). Pixel Based Digital Image Forgery Detection Techniques. **International Journal of Engineering Research and Application (IJERA)** ISSN: 2248-9622, Vol. 2, Issue 3, May-Jun 2012, pp. 539-543.

- Pratiksha N.Balai and Jimit A.Talati. (2013). Position Control of Hybrid Stepper Motor Using PIC16F877A Microcontroller. **International Journal of Science and Research (IJSR)**, ISSN (Online): 2319-7064, Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438
- Rajkumar Jain and Pushparaj Mani Pathak. (2008). Trajectory planning of 2 DOF planar space robot without attitude controller. **World Journal of Modelling and Simulation**, ISSN 1746-7233, England, UK, Vol. 4 (2008), No. 3, pp. 196-204.
- R. Hari Sudhan, M.Ganesh Kumar, A.Udhaya Prakash, S.Anu Roopa Devi and P.Sathiya. (2015). ARDUINO ATMEGA-328 MICROCONTROLLER. **INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING**, Vol. 3, Issue 4, April 2015
- S.B. Dworkin and T.J.Nye. (2006). Image Processing for machine vision measurement of hot formed parts. **Journal of Materials Processing Technology** 174 (2006): 1-6.
- Shafeeque Ahmed, Fini Fathima, S. Prabhakar Karthikeyan, Sarat Kumar Sahoo, Shriram. S.Rangarajan, and D.P.Kothari. (2015). **IEEE**, 978-1-4799-8641-5/15/\$31.00, @2015 IEEE
- Shrddha Baldha, Jimit Shukla and Kuldip Tarpara. (2015). Design and Simulation of Two-Phase Hybrid Stepper Motor with Current Tracking. **National Conference on emerging Trends in Computer, Electrical & Electronics (ETCEE-2015)**, **International Journal of Advance Engineering and Research Development (IJAERD)**, e-ISSN: 2348-4470, print-ISSN: 2348-6406, Impact Factor: 3.134
- Ticiano Monte Lucioda Silva, Mateus Bonamigo Zupiroli, Rafael Pereira and Antonio C.P. Brasil Junior. (2007). A PROPOSED AIRFOIL SHAPE PARAMETRIZATION FOR AERODYNAMIC OPTIMIZATION. **19th International Congress of Mechanical Engineering** (pp 1-10). Brazil: ABCM.

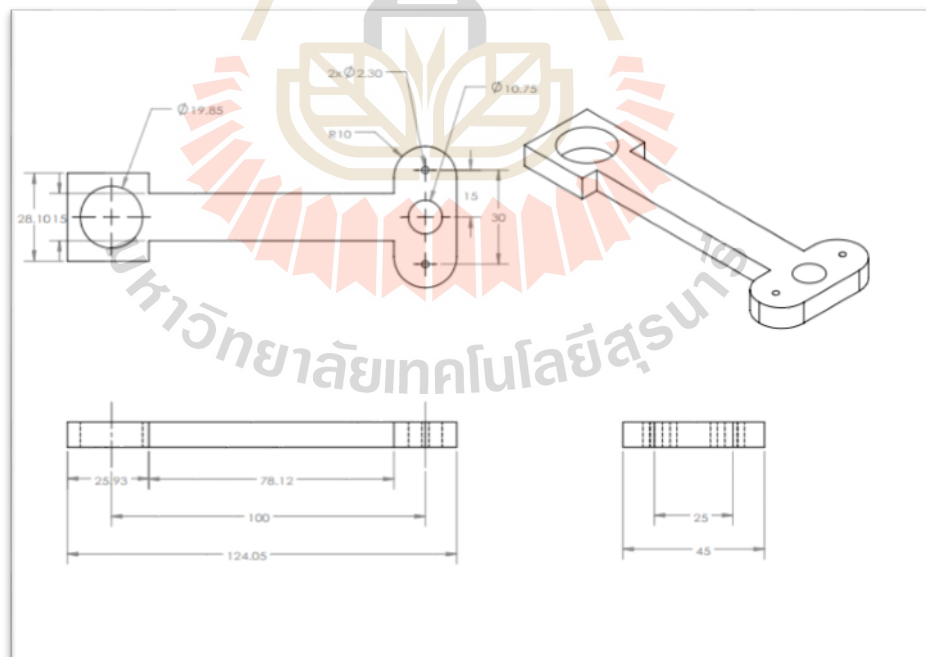
ภาคผนวก ก

Drawing เครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ

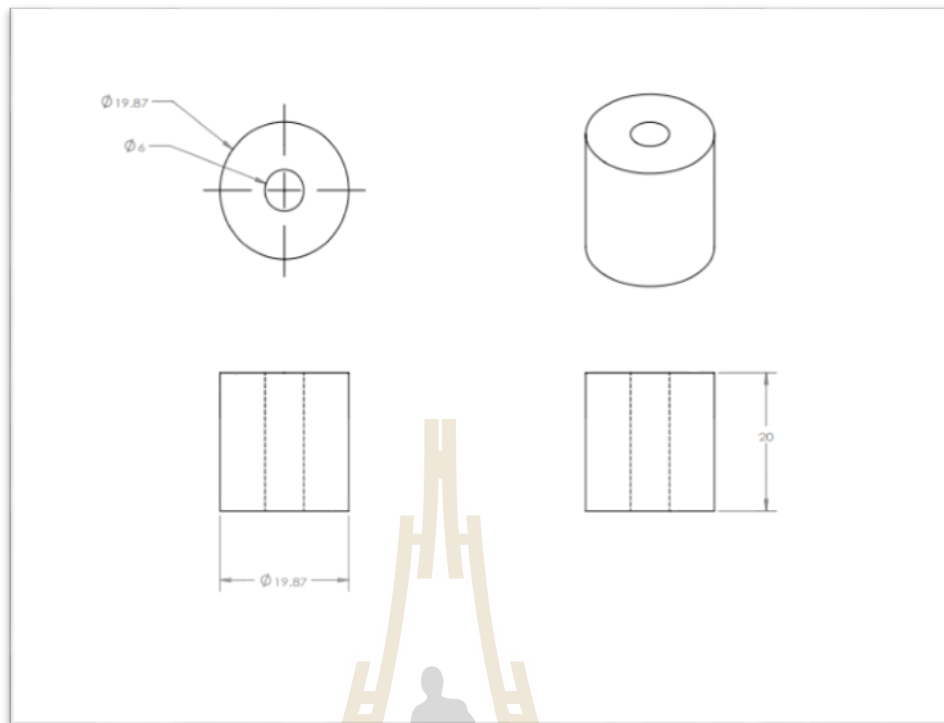
มหาวิทยาลัยเทคโนโลยีสุรนารี



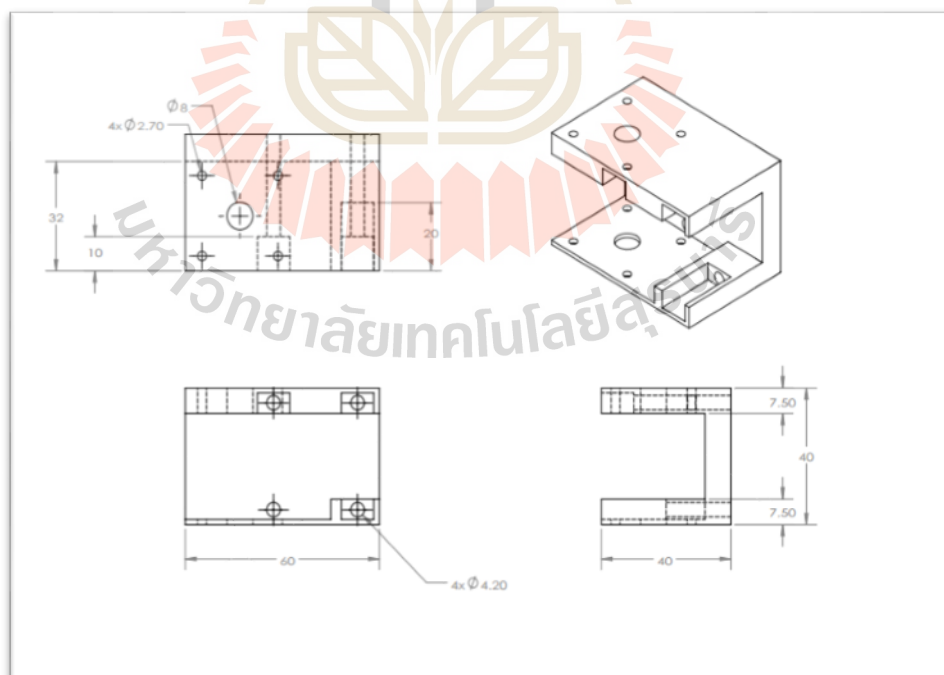
รูปที่ ก.1 แขนลำดับที่หนึ่งของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



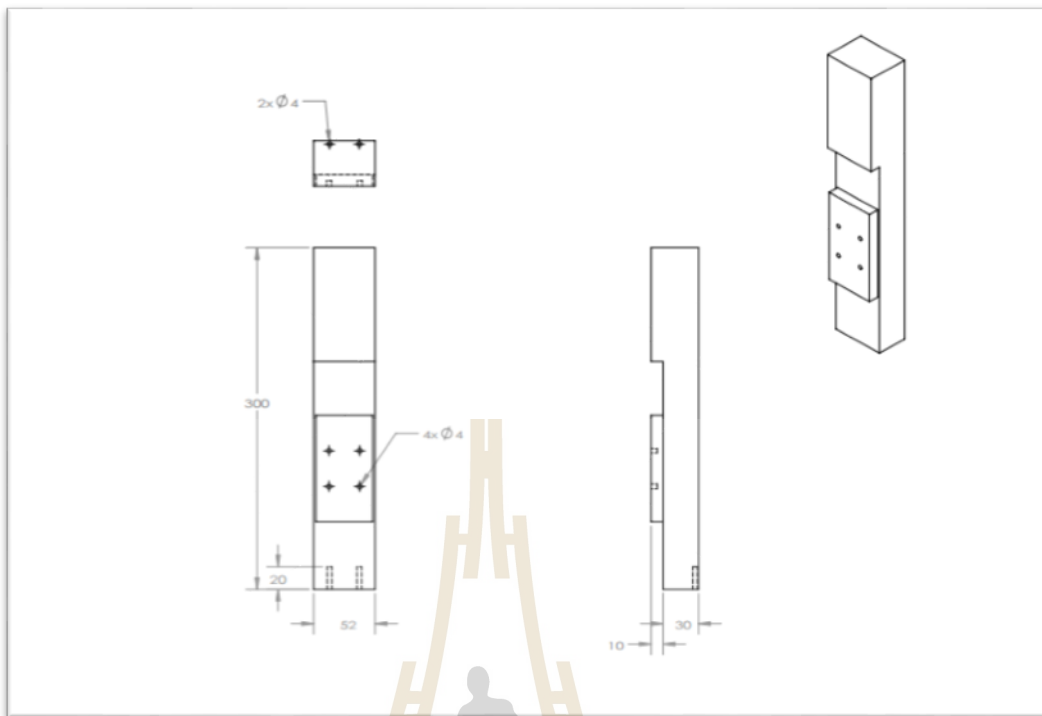
รูปที่ ก.2 แขนลำดับที่สองของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



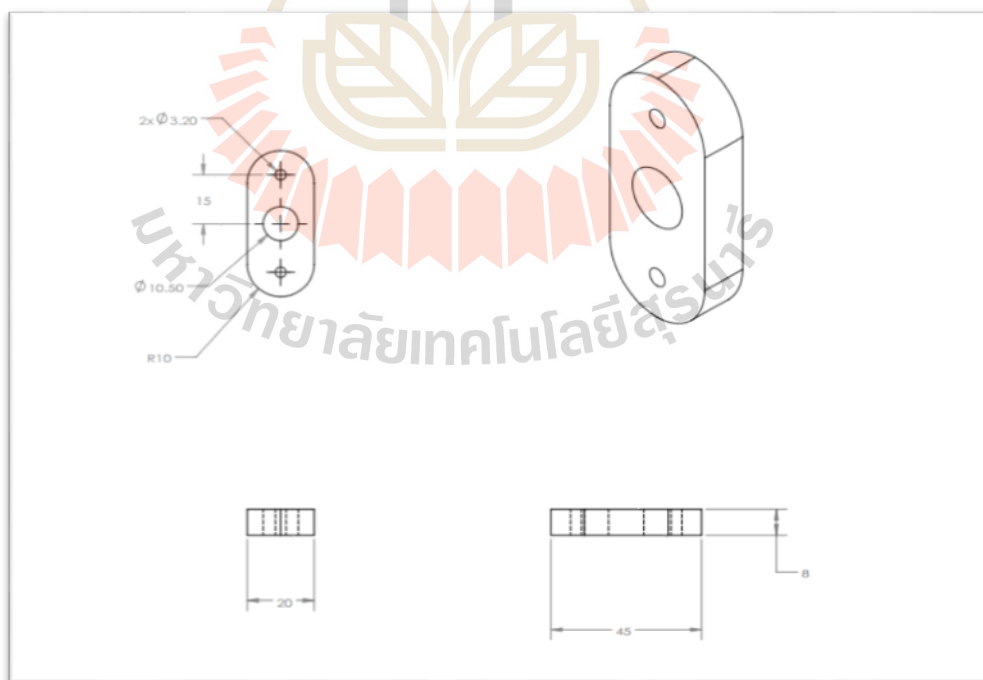
รูปที่ ก.3 กัปปิ้งของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



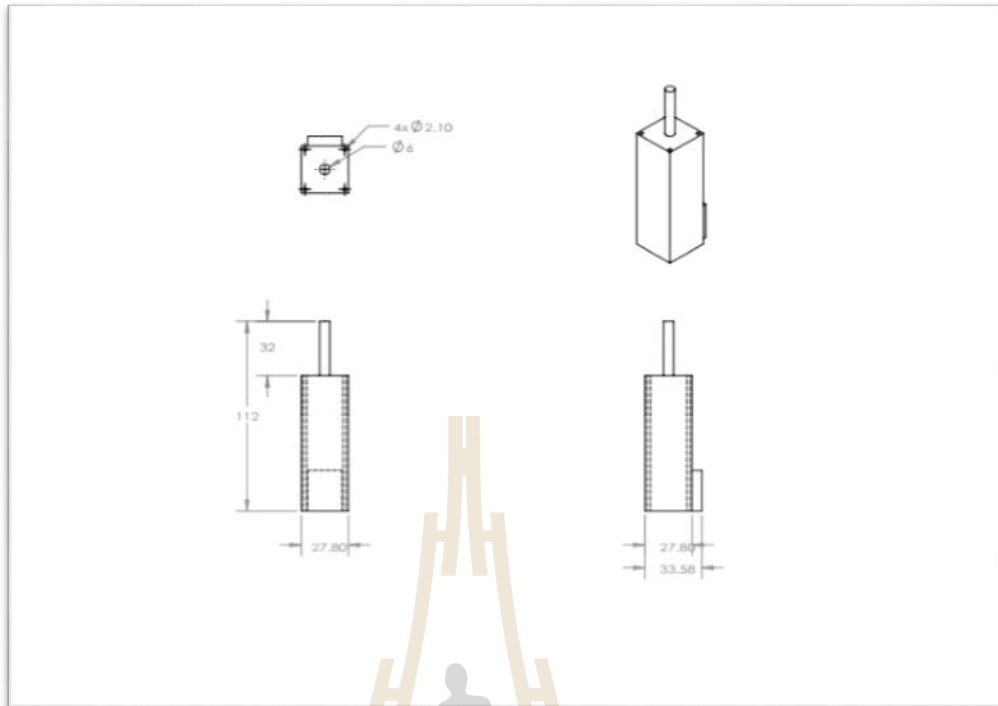
รูปที่ ก.4 ฐานรองสำหรับยึดกับ Linear Robot ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



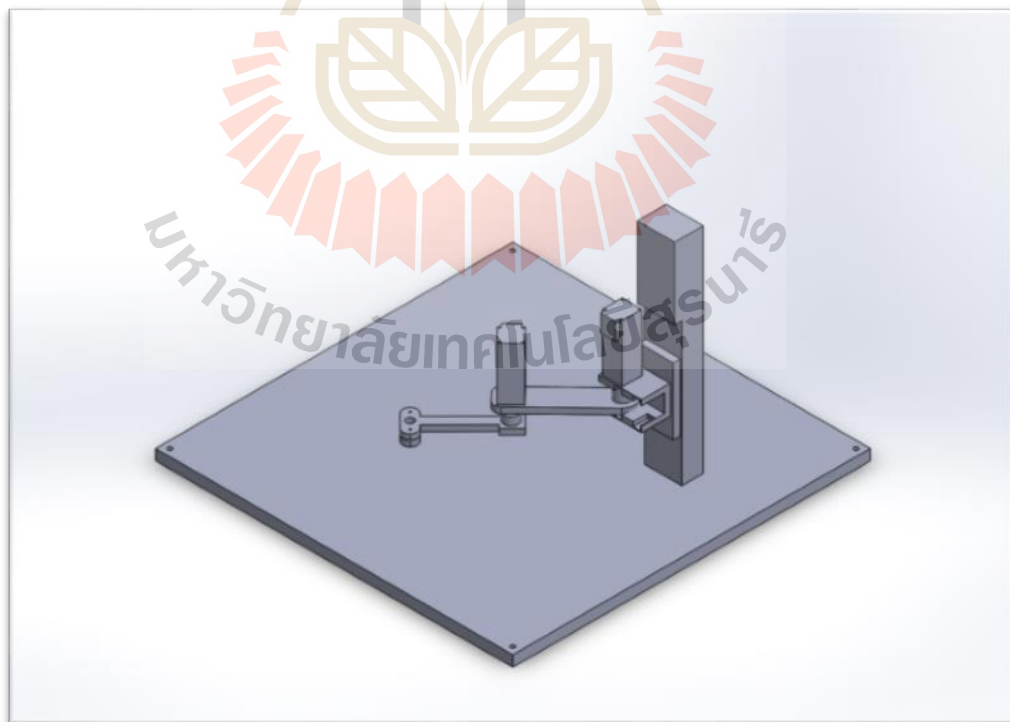
รูปที่ ก.5 Linear Robot ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



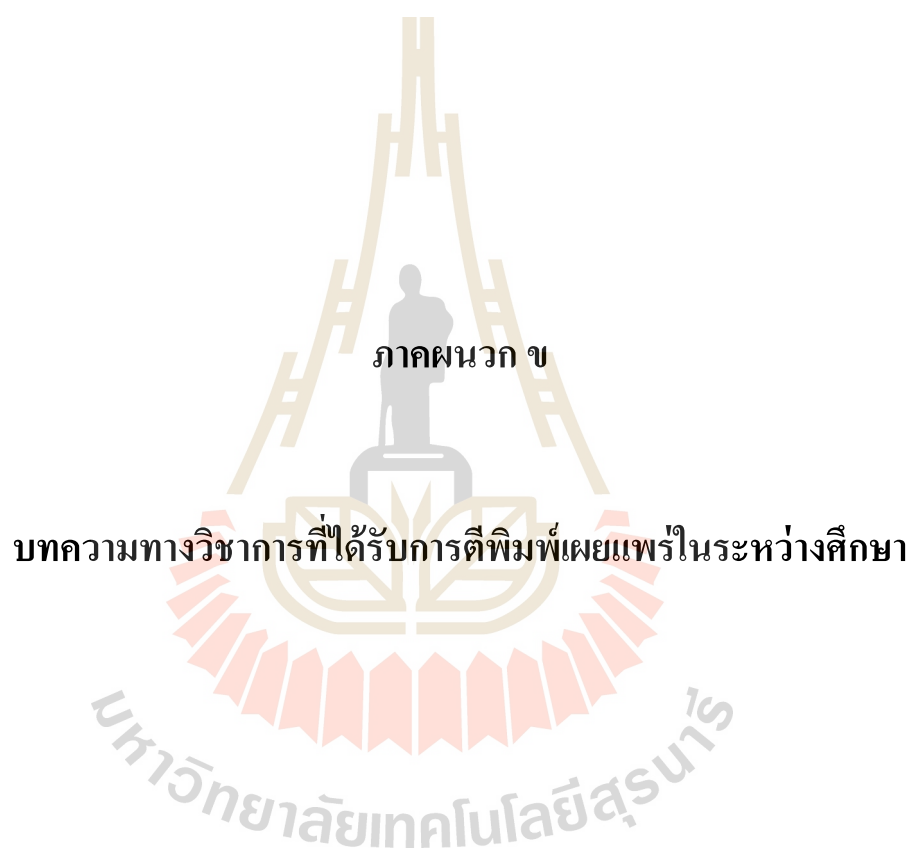
รูปที่ ก.6 ฐานจับปากกาของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



รูปที่ ก.7 สเตปเปอร์มอเตอร์ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



รูปที่ ก.8 Assembly ของเครื่องต้นแบบหุ่นยนต์แขนกลระนาบสองมิติ



ภาคผนวก ข

บทความทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายชื่อบทความที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

Sinmaroeng, P., Uangpairoj, P., and Chamniprasart, K. (2016). **Robotic Curve Tracking And Reproduction Using Machine Vision**. South East Asian Technical University Consortium (SEATUC 2016). 22 - 24 February. SIT. Japan.



ROBOTIC CURVE TRACKING AND REPRODUCTION USING MACHINE VISION

Phuwarin Sinmaroeng, Pichitra Uangpairoj, Kontorn Chamniprasart*

School of Mechanical Engineering, Institute of Engineering,
Suranaree University of Technology,

*e-mail: kontorn@sut.ac.th, telephone number :0-4422-4286, fax number :0-4422-4220

ABSTRACT

At the present, the role of the industrial technology is growing rapidly and is continuously developed. The manufacturing process consists of several sub-processes such as design, assembly and inspection. The design process is one of the important processes. This is currently designed to apply to robot and automation systems in the industrial manufacturing. Curve tracking and reproduction using machine vision and the 2-R planar robot have been broadly used for many applications in industry. For example, in the design industry, design engineers usually use free hand drawing that may contains with unusual curves or irregular shapes, the curves are drawn from imagination. This article focuses on the machine vision that is combined with a 2-R planar robot. First, the image is captured by webcam camera. Then the coordinates (x,y) are located using MATLAB software. The curve tracking can be conducted by connecting those coordinate with 1-D spline interpolation method. Because this method is simple and easy to perform curve fitting which provides the same as the original curve. After that, the 2-R planar robot is driven to move by stepping motor that are controlled via Arduino microcontroller (Arduino Mega 2560). Reverse kinematics is performed to enhance robotic reproduction, the robot can reproduce pictures as same as the original picture. The result is impressive. Now we can find the coordinates (x,y) by transforming the analog signal into pixel. The curve then can be draw using Spline function in MATLAB software. The result found that this program is able to reproduce irregular curve very well. In the future work, the robot will be driven to draw the curve pattern.

KEYWORDS

Curve tracking and reproduction, 2-R planar robot and 1-D spline interpolation method

1. INTRODUCTION

Currently, Curve tracking and reproduction are another typical application of robots with vision systems, having a broad range of industrial applications in different fields such as metal cutting, welding, cloth cutting, band automatic inputting of prints. In those kinds of applications, the robot with a vision system tracks the curves in the prints and records their coordinates. The usage of an automatic input of prints is realized. The design process is one of important processes. The designing is designed to apply for robot, automation system in industry. It is considered as one of the most challenges task for design engineers. However, the main problem is most of design engineers often design patterns by free hand drawing which is difficult to use such patterns with robots. Sometimes the patterns are too complicated for robot to understand and work effectively. The drawing is not geometry, it often consists of irregular curve as ideal designs. Consequently, the design engineer cannot apply the CAD program to all sketching. The curve is tracked using cameras that are combined with MATLAB software to increase the efficiency of the design. This system is convenient and popular in the industry. To increase stability of robotic operation in an uncertain environment, a recent trend is to use vision information in the dynamic feedback loop. There are two working styles. The first style uses cameras mounting at the end of the robot forearm, which is referred to as an "eye-in-hand" structure and the second one uses static cameras. Due to different situations, the first working styles is good for tracking with a high accuracy and the second one is fast tracking, simply, easy to control and low cost. (Jing Lin, et al., 1999 and Cao, et al., 1990). Therefore, this research aims to use machine vision to generate a part of end effector of 2-R planar robot for tracking and reproducing the complex curves. This work is divided into two parts: the first part is programming. And the second part is prototyping the robot for reproduction the curve. However this paper present only the programming part.

2. EXPERIMENT

2.1 Experimental Apparatus

2.1.1 MATLAB Software

MATLAB is called Matrix Laboratory is a multi-paradigm numerical computing environment and four-generation programming language developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with program written in other languages, including C, C++, Java, Fortran and Python

In this research, MATLAB is used to process images captured by the webcam camera. MATLAB processes the image by transforming from analog signal to pixel. Then the coordinates (x,y) are located using image processing function in MATLAB software. The coordinates are used by the robot to reproduce picture that is similar to the original picture.

2.1.2 Webcam Camera

A webcam camera is a video camera that feeds or streams its image in real time to or through a computer to computer network. When the video is captured by the computer, the video stream may be saved, viewed or sent to other networks via systems such as the internet, and email as an attachment. When it is sent to a remote location, the video stream may be saved, viewed or sent there. Unlike an IP camera (which connects using Ethernet or Wi-Fi), a webcam is generally connected by a USB cable, or similar cable, or built into computer hardware, such as laptops.

Webcam camera will capture the image that drawn by design engineer. The image is sent to process by image processing function in MATLAB software. And this research installs static cameras method to increase efficiency of image processing, and reduce tracking time.

2.1.3 Stepper motor

A stepper motor is an electromechanical device which converts electrical pulses into discrete mechanical movements. The shaft or spindle of a stepper motor rotates in discrete step increments when electrical command pulses are applied to it in the proper sequence. The motors rotation has several direct relationships to these applied input pulses. The sequence of the applied pulses is directly related to the direction of motor shafts rotation. The speed of the motor shafts rotation is directly related to the frequency of the input pulses and the length of rotation is directly related to the number of input pulses applied.

The advantages of stepper motor is simple and convenient for control circuit or microprocessor. In addition, the stepper motor can run with higher accuracy and higher torque compared with other motors.

2.1.4 Microcontroller

Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that uses a separate microprocessor,

memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes. Mixed signal microcontrollers are common, integrating analog components needed to control non-digital electronic systems.

In this research, Arduino Mega 2560 is used as a mechanism to control the stepper motor. The advantage of this board is low cost, easy to use and there are plenty of input/output channel.

2.1.5 2-R Planar Robot

2-R Planar robot is a 2 degree-of-freedom (DOF). It has advantages in term of easy to calculate and low cost. The kinematic of 2-R Planar Robot will be described in the next section.

2.2 Technique

2.2.1 Kinematics of 2-R Planar Robot

Forward kinematics of the robot : The two degree of freedom planar redundant manipulator is implemented as shown in Figure 1. The joints axes are assigned based on the Denavit-Hartenberg representation (Table 1). The Manipulator moves on a vertical plane, the gravity force is also considered in the numerical simulation.

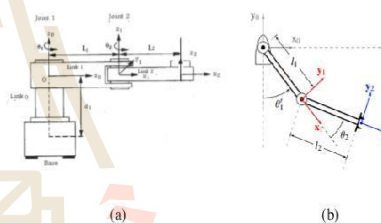


Figure 1, The two-joint 2-R robot

Table 1. D-H Parameter of the 2-R robot

Link	a_i	α_i	d_i	θ_i
1	l_1	0	0	θ_1
2	l_2	0	0	θ_2

The (4x4) rigid transformation matrix of the first link is expressed as follows

$$T_1^0 = \begin{bmatrix} c_1 & -s_1 & 0 & l_1 c_1 \\ s_1 & c_1 & 0 & l_1 s_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

While the matrix of the second link is given by

$$T_2^1 = \begin{bmatrix} c_2 & -s_2 & 0 & l_2 c_2 \\ s_2 & c_2 & 0 & l_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Then we can get the over all manipulator transformation matrix as

$$T_2^0 = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1 c_1 + l_2 c_{12} \\ s_{12} & c_{12} & 0 & l_1 s_1 + l_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

(Mahmoud Gouasmi, et al., 2012).

2.2.2 The Digital Image

Raster graphic is one type of digital image. Raster graphic consists of with irregular shape also called Bit-mapped which convert from analog devices, such as a scanner or a digital camera.

Figure 3. shows the magnification image of the cloud. The Figure 4 is magnified until the small square is found as shown in Figure 4. Each square called “pixels” (picture element).

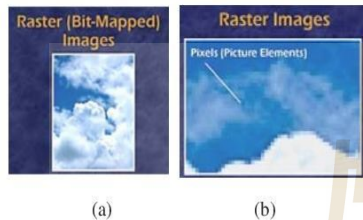


Figure 3. The digital image or Raster graphic

Each pixel has its own numbered address with different colours as shown in Figure 4.

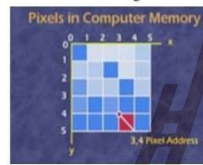


Figure 4. Pixels in Computer Memory

2.2.3 1-D Cubic Spline Interpolation

The cubic spline is used for a piecewise cubic interpolating function. The term “spline” refers to an instrument used in drafting. It is a thin, flexible wooden or plastic tool that is passed through given data points and defines a smooth curve in between. The physical spline minimizes potential energy subjected to the interpolation constraints. The corresponding mathematical spline must have a continuous second derivative and satisfy the same interpolation constraints. The breakpoints of a spline are also referred to as its knots. (x. C. De Boor., 1978).

The world of splines extends far beyond the basic one-dimensional, cubic, interpolatory spline we are describing here. There are multidimensional, high-order, variable knots for spline approximation. The spline interpolation can be performed using the Spline Toolbox for MATLAB

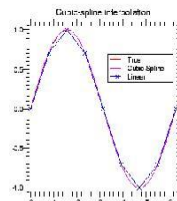


Figure 5. Compare linear and cubic Spline

3. STEP OF ANALYSIS

Procedure of this research has 6 steps Table 2. Procedure of this research



3.2.1 Step 1 : Design engineer draw the picture, the curve or irregular curve to the paper by pen.

3.2.2 Step 2 : The picture is captured by webcam camera.



Figure 6. Prototyping picture

3.2.3 Step 3 : The coordinates (x,y) are found using MATLAB software.

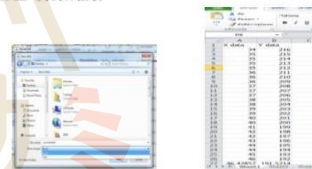


Figure 7. Coordinate (x,y) of picture

3.2.4 Step 4 : The curve tracking can be conducted by connecting those coordinate with 1-D spline interpolation method.

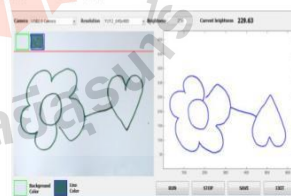


Figure 8. Plot Spline by MATLAB Software

3.2.5 Step 5 : From equation (3), over all manipulator transformation matrix is.

$$T_2^0 = \begin{bmatrix} c_{12} & -s_{12} & 0 & l_1 c_1 + l_2 c_{12} \\ s_{12} & c_{12} & 0 & l_1 s_1 + l_2 s_{12} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

this research can determine the workspace of robot that shown in Figure 9. The red line is link1 (l_1) with the length of 15 cm., it can move $0^\circ - 180^\circ$. And the blue line is link2 (l_2) with the length of 10 cm., it can move $0^\circ - 360^\circ$.

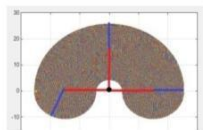


Figure 9. Workspace of robot

The 2-R planar robot is driven to move by stepping motor that are controlled via Arduino microcontroller (Arduino Mega 2560).

3.2.6 Step 6 : Reverse kinematics is performed to enhance robotic reproduction, the robot can reproduce pictures as same as the original picture.

4. CONCLUSION

This article describes irregular curve tracking using Spline function in MATLAB software. The result found that this program is able to reproduce irregular curve very well. The result indicates that integrated system is static camera, find coordinate by image processing and curve fitting by spline interpolation can be possibly used for reproduce irregular curve efficiently.

In second part (step 5 and 6) is the prototyping of the 2R Planar robot for reproducing the curve, The author and team will conduct it as the future work. Which we expected reverse kinematics is performed to enhance robotic reproduction, the robot can reproduce pictures as same as the original picture.

5. REFERENCES

1. Jing Lin, Hui-Tang Chen, Ping Jiang, Yue-Juan Wang, Curve Tracking and Reproduction by a Robot with a Vision System, Journal of Robotic Systems, 1999. 16(10): p. 547-556
2. B.L. Cao, X. Wang, and H.T. Chen, A real-time visual servoing control scheme for robot manipulator. Proc Int Conf Automat Robotics Computer Vision, 1990. pp. 936-939.
3. Mahmoud Gouasmi, Mohammed Ouali, Brahim Fernini, M'hamed Meghatria, Kinematic Modelling and Simulation of a 2-R Robot Using Solidworks and Verification by MATLAB/Simulink, International Journal of Advanced Robotic Systems, 2012. Vol. 9: p. 245
4. x. C. De Boor, A practical Guide to Splines, Springer-Verlag, New York, 1978

6. NOMENCLATURE

- C_i : $\cos(\theta_i)$
 S_i : $\sin(\theta_i)$
 C_{ij} : $\cos(\theta_i + \theta_j)$
 l : Length of each link [cm]



Phuwarin Sinmaroeng

received the B.E. (2013) degree in Aeronautical Engineering from Suranaree University of Technology.



Pichitra Uangpairoj

received the B.S. (2007) degree in Food Technology, M.E. (2010) degree in Mechanical Engineering from Suranaree University of Technology, and Ph.D. (2013) degree in Functional Control Systems from Shibaura Institute of Technology.



Kontorn Chammiprasart

Received the B.S. (1980) degree in Mechanical Engineering from (First Class Honer) from Royal Thai Air Force Academy, Thailand, M.Sc. (1987) degree in Mechanical Engineering from University of Pittsburgh, USA, and Ph.D. (1992) degree in Mechanical Engineering from University of Pittsburgh, USA

ประวัติผู้เขียน

นายภูวรินทร์ สินมะเริง เกิดเมื่อวันที่ 29 กรกฎาคม พ.ศ. 2535 มีภูมิลำเนาอยู่ที่อำเภอเมือง จังหวัดนครราชสีมา สำเร็จการศึกษาชั้นประถมศึกษาที่โรงเรียนมารีย์วิทยา จังหวัดนครราชสีมา สำเร็จการศึกษาชั้นมัธยมศึกษาที่โรงเรียนมารีย์วิทยา จังหวัดนครราชสีมา และสำเร็จการศึกษาระดับปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมอากาศยาน จากมหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา เมื่อภาคการศึกษาที่ 2 ปีการศึกษา 2556 ภาคการศึกษาที่ 3 ปีการศึกษาเดียวกัน เข้าศึกษาต่อในระดับปริญญาโท วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมเครื่องกล สำนักวิชา วิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา

ขณะที่ศึกษาต่อในระดับมหาบัณฑิตนั้น ได้มีประสบการณ์โดยเป็นผู้สอนในรายวิชาดังนี้

1. ปฏิบัติการวิศวกรรมเครื่องกล 1
2. ปฏิบัติการวิศวกรรมเครื่องกล 2
3. ปฏิบัติการวิศวกรรมการผลิต 1
4. ปฏิบัติการวิศวกรรมการผลิต 2

ในระหว่างการศึกษาระดับปริญญาโทได้นำเสนอผลงานทางวิชาการ ตามที่ได้นำเสนอแสดง
ในภาคผนวก ข

มหาวิทยาลัยเทคโนโลยีสุรนารี