



ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

เรื่อง การติดตามทำทางของมือจากข้อมูลกลศาสตร์ 3 มิติ ที่ได้จากภาพถ่าย 2 มิติ
โดย นายไกรวุฒิ หลักคำ

ได้รับอนุมัติให้นับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า

คณบดีบัณฑิตวิทยาลัย

(อาจารย์ ดร.มงคล หวังสถิตย์วงศ์)

17 พฤษภาคม 2550

คณะกรรมการสอบวิทยานิพนธ์

ประธานกรรมการ

(ผู้ช่วยศาสตราจารย์ ดร.พูลศักดิ์ โกษียาภรณ์)

กรรมการ

(ผู้ช่วยศาสตราจารย์ ไชยันต์ สุวรรณชีวะศิริ)

กรรมการ

(รองศาสตราจารย์ มนูญ พ่วงพุด)

กรรมการ

(รองศาสตราจารย์ ดร.สยาม เจริญเสียง)

การติดตามท่าทางของมือจากข้อมูลกลศาสตร์ 3 มิติ ที่ได้จากภาพถ่าย 2 มิติ

นายไกรวุฒิ หลักคำ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ปีการศึกษา 2549
ลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ชื่อ : นายไกรวุฒิ หลักคำ

ชื่อวิทยานิพนธ์ : การติดตามท่าทางของมือจากข้อมูลกลศาสตร์ 3 มิติ ที่ได้จากภาพถ่าย 2 มิติ

สาขาวิชา : วิศวกรรมไฟฟ้า

สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ที่ปรึกษาวิทยานิพนธ์ : ผู้ช่วยศาสตราจารย์ ดร.พูลศักดิ์ โกษียาภรณ์

ผู้ช่วยศาสตราจารย์ ไซยนต์ สุวรรณชีวะศิริ

ปีการศึกษา : 2549

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้เสนอระบบติดตามท่าทางของมือจากข้อมูลกลศาสตร์ 3 มิติ ที่ได้จากภาพถ่าย 2 มิติ การทำงานอาศัยหลักการพื้นฐานจากถุมือ และการจำลองแบบมาประยุกต์ร่วมกัน โดยใช้ภาพถ่ายฝ่ามือข้างขวาซึ่งสวมถุมือที่ทำสัญลักษณ์ด้วยสีบริเวณปลายนิ้ว ส่วนการออกแบบแบบจำลองมือได้วิเคราะห์จากโครงสร้างมือมนุษย์ โดยแบบจำลองมือมีจำนวนมุมหมุนอิสระ (Degree of Freedom: DOF) จำนวน 21 DOF ซึ่งแบบจำลองมือได้ถูกสร้างภายใต้โปรแกรมจำลองหุ่นยนต์ 3 มิติ โรโบซิม 2 (Robosim 2) ส่วนการควบคุมการขยับปลายนิ้วเข้าสู่ตำแหน่งที่ต้องการได้ใช้ระบบควบคุมแบบฟัซซี่ ร่วมกับเงื่อนไขหรือข้อกำหนดในการขยับมุมหมุนที่ข้อต่อของนิ้ว เขียนอัลกอริทึมในการควบคุมภายใต้โปรแกรม แมทแลป (Matlab)

จากการทดสอบได้ใช้สัญญาณภาษามืออเมริกัน (American Sign Language: ASL) ของ
 พืชชนิดต่างๆ จำนวนทั้งสิ้น 24 ท่าทาง สามารถแสดงสัญญาณภาษามืออเมริกันของพืชชนิด
 ต่างๆ ได้จำนวนทั้งสิ้น 18 ท่าทาง ได้แก่พืชชนิด A, B, C, D, E, F, G, H, I, K, L, O, P, Q, U, W, X
 และ Y โดยจำนวนรอบในการแสดงท่าทางโดยเฉลี่ยประมาณ 17 รอบ และกำหนดให้ค่าผิดพลาด
 ในการระบุตำแหน่งในแนวแกน X และ Y ของปลายนิ้ว ไม่เกินกว่า 4 พิกเซล

(วิทยานิพนธ์มีจำนวนทั้งสิ้น 99 หน้า)

คำสำคัญ : การติดตามท่าทาง, มือกล, กลศาสตร์

อาจารย์ที่ปรึกษาวิทยานิพนธ์

Name : Mr.Kraiwoot Lakkam
Thesis Title : Hand Gesture Tracking Based on 3D Hand Kinematics Data Extracted from
2D Image
Major Field : Electrical Engineering
King Kongkut's Institute of Technology North Bangkok
Thesis Advisors : Assistant Professor Dr. Poolsak Koseeyaporn
Assistant Professor Chaiyan Suwanchewasiri
Academic Year : 2006

Abstract

The thesis presents hand gesture tracking based on 3D hand kinematics data extracted from 2D image. The operation is based on glove-base methods and model-base methods. The right hand glove captured into color image contains several marks at extreme areas in different colors. A part from hand model design is analyzed based on human hand structure. Hand model possesses 21 degree of freedom (DOF), constructed in 3D Simulation Program called Robosim 2. In addition, the finger position control is designed based on fuzzy logic control utilizing joint constraints and the control algorithm is written in Matlab environment.

For experiment, action American Sign Language (ASL) with 24 postures of consonant is employed. The system is capable to demonstrate 18 postures of consonant as A, B, C, D, E, F, G, H, I, K, L, O, P, Q, U, W, X and Y. The average cycle of posture is 17 cycles and the defined error on X and Y axle is limited to 4 pixels.

(Total 99 pages)

Keywords : Tracking, Hand Model, Kinematics

Advisor

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลือ และสนับสนุนอย่างดียิ่งของผู้ช่วยศาสตราจารย์ ดร.พูลศักดิ์ โกษียาภรณ์ และผู้ช่วยศาสตราจารย์ไชยันต์ สุวรรณชีวะศิริ อาจารย์ที่ปรึกษาทั้งสองที่กรุณาให้คำแนะนำ และข้อคิดเห็นต่างๆของการวิจัยมาโดยตลอด และทุนการวิจัยบางส่วนที่ได้รับจากทุนอุดหนุนการวิจัยของบัณฑิตวิทยาลัย จึงขอขอบพระคุณบัณฑิตวิทยาลัยที่ได้ให้ทุนอุดหนุนการทำวิจัยครั้งนี้มา ณ ที่นี้ด้วย

ทำนุผู้วิจัยใคร่ขอกราบขอบพระคุณบิดา มารดา ซึ่งให้การสนับสนุนในด้านการเงิน และให้กำลังใจแก่ผู้วิจัย ขอบพระคุณครูบาอาจารย์ที่ให้ความรู้ อบรมสั่งสอน และเพื่อนๆ พี่น้องที่ให้การสนับสนุนที่เป็นกำลังใจเสมอมาจนวิทยานิพนธ์นี้สำเร็จลุล่วงได้ด้วยดี

ไกรวุฒิ หลักคำ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
กิตติกรรมประกาศ	ง
สารบัญตาราง	ช
สารบัญภาพ	ฅ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการวิจัย	2
1.3 ขอบเขตของการวิจัย	2
1.4 เครื่องมือและอุปกรณ์ที่ใช้	2
1.5 ประโยชน์ของการวิจัย	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	3
2.1 ลักษณะโครงสร้างทางกายภาพของมือมนุษย์	3
2.2 หลักกลศาสตร์	6
2.3 โปรแกรมจำลองหุ่นยนต์ในระบบ 3 มิติ	16
2.4 การประมวลผลภาพ	21
2.5 ทฤษฎีระบบโครงข่ายประสาทเทียม	27
2.6 ทฤษฎีฟัซซี่	39
บทที่ 3 การวิเคราะห์การออกแบบ	53
3.1 การวิเคราะห์โครงสร้างมือมนุษย์ และออกแบบแบบจำลองมือ	54
3.2 การออกแบบระบบประมวลผลภาพ	60
3.3 การออกแบบระบบหาตำแหน่ง 2 มิติ จากตำแหน่ง 3 มิติ	63
3.4 การวิเคราะห์การเคลื่อนไหวและออกแบบระบบควบคุม	67
บทที่ 4 ผลของการวิจัย	79
4.1 ผลการประมวลผลภาพ	79
4.2 ผลการหาตำแหน่ง 2 มิติ จากตำแหน่ง 3 มิติ	82
4.3 การทดสอบติดตามท่าทาง	83

สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	95
5.1 สรุปผลการวิจัย	95
5.2 ข้อดีของงานวิจัย	95
5.3 ข้อบกพร่องของงานวิจัย	96
5.4 ข้อเสนอแนะสำหรับการพัฒนา	96
บรรณานุกรม	97
ประวัติผู้วิจัย	99

สารบัญตาราง

ตารางที่	หน้า
3-1 พารามิเตอร์ของ D-H สำหรับนิ้วมือทั้ง 4	56
3-2 พารามิเตอร์ของ D-H ของนิ้วหัวแม่มือ	57
3-3 ขอบเขตของการตัดค่าสี RGB ของสีต่างๆของแต่ละนิ้ว	61
3-4 กฎของฟิชชีในการควบคุมการเข้าสู่ตำแหน่งจากการพับงอนิ้ว	69
3-5 กฎของฟิชชีในการควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้ว	71
3-6 กฎของฟิชชีในการควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้วเมื่อตำแหน่งของปลายนิ้วอยู่ต่ำกว่าจุดต่อ MCP	73
3-7 กฎของฟิชชีในการควบคุมการเข้าสู่ตำแหน่งในแนวแกน X	75
3-8 กฎของฟิชชีในการควบคุมการเข้าสู่ตำแหน่งในแนวแกน Y	77
4-1 ตำแหน่งสี่บริเวณปลายนิ้วของภาพที่ 4-3	82
4-2 ผลการทดสอบหาตำแหน่ง 2 มิติ จากข้อมูล 3 มิติ ด้วยระบบโครงข่ายประสาทเทียม	82
4-3 แสดงค่าผิดพลาดของตำแหน่ง 2 มิติ จากระบบประสาทกับตำแหน่งจริง	83
4-4 แสดงจำนวนรอบในการแสดงท่าทางของสัญญาณภาษามืออเมริกันในแต่ละท่าทาง	90
4-5 แสดงค่าผิดพลาดในแนวแกน X และ Y ของสัญญาณภาษามืออเมริกันในแต่ละท่าทาง	91
4-6 แสดงค่าเฉลี่ยของค่าผิดพลาดกำลังสองในการเข้าสู่ตำแหน่งของสัญญาณภาษามืออเมริกันในแต่ละท่าทาง	92

สารบัญภาพ

ภาพที่	หน้า
2-1 แสดงโครงสร้างกระดูกมือข้างขวา	4
2-2 แสดงจุดต่อระหว่างกระดูกนิ้วมือ	4
2-3 แสดงการขยับของมุมหมุนอิสระที่จุดต่อต่างๆของมือ	5
2-4 แสดงข้อต่อพื้นฐานทางจลนศาสตร์	6
2-5 แสดงแผนภาพจลนศาสตร์แบบไปข้างหน้า และจลนศาสตร์แบบผกผัน	7
2-6 แสดงตำแหน่งของจุด P	8
2-7 แสดงการหมุนรอบแกน Z	9
2-8 แสดงการหมุนของปลายแขนกล	10
2-9 แสดงพารามิเตอร์ของ Denavit-Hartenberg กรณีข้อต่อแบบหมุน	13
2-10 แสดงพารามิเตอร์ของ Denavit-Hartenberg กรณีข้อต่อแบบเลื่อน	13
2-11 แสดงรูปทรงเรขาคณิตรูปกล่อง และทรงกระบอก	17
2-12 แสดงหน้าต่าง LISP Command Line Window	18
2-13 แสดงหน้าต่างเมนู File ในโปรแกรมโรโบซิม	18
2-14 แสดงหน้าต่าง RoboEditor สำหรับเขียนโปรแกรมเพื่อสร้างหุ่นยนต์	19
2-15 แสดงหน้าต่างแสดงแบบจำลองหุ่นยนต์	19
2-16 แสดงแผนภาพมาตรฐานสามเหลี่ยมสี Commission International de l'Eclairage	22
2-17 แสดงการผสมสีแบบพื้นฐานบวก	23
2-18 แสดงการผสมสีแบบพื้นฐานลบ	23
2-19 แสดงการเก็บข้อมูลแบบดิจิทัล	24
2-20 แสดงระบบคาร์ทีเซียนของสี RGB	25
2-21 แสดงการเปรียบเทียบระหว่างข้อมูลภาพแบบระดับสีเทา กับสี RGB	26
2-22 แสดงการตัดค่าสี	27
2-23 แสดงแบบจำลองโครงข่ายประสาทของมนุษย์	28
2-24 แบบจำลองทางคณิตศาสตร์ของระบบโครงข่ายประสาทเทียม	28
2-25 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบเชิงเส้น	29
2-26 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบขั้นบันได	30
2-27 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบซิกมอยด์	30
2-28 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบไฮเพอร์บอลิก	30

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
2-29 แสดงข่ายงานแบบป้อนไปข้างหน้าชั้นเดียว	31
2-30 แสดงข่ายงานแบบป้อนไปข้างหน้าหลายชั้นโดยเชื่อมต่อแบบเต็ม	32
2-31 แสดงข่ายงานแบบป้อนไปข้างหน้าหลายชั้นโดยเชื่อมต่อบางส่วน	32
2-32 แสดงข่ายงานแบบเวียนเกิด	33
2-33 แสดงแผ่นภาพการเรียนรู้แบบมีผู้ฝึกสอน	34
2-34 แสดงแผ่นภาพการเรียนรู้แบบไม่มีผู้ฝึกสอน	34
2-35 แสดงแผ่นภาพการเรียนรู้แบบเสริมสร้าง	34
2-36 แสดงขั้นตอนวิธีการแพร่ย้อนกลับ	35
2-37 แสดงกระบวนการคำนวณไปข้างหน้า	36
2-38 แสดงกระบวนการคำนวณย้อนหลัง	37
2-39 แสดงการเปลี่ยนแปลงค่าตัวถ่วงน้ำหนักเชื่อมโยง	38
2-40 แสดงการเปรียบเทียบระหว่างบูลีนลอจิกกับพีชชีลอจิก	39
2-41 แสดงภาพของการยูเนียนของพีชชีเซต A และ B	41
2-42 แสดงภาพของการอินเตอร์เซกชันของพีชชีเซต A และ B	41
2-43 แสดงภาพของการคอมพลิเมนต์ของพีชชีเซต A	41
2-44 แสดงกราฟของฟังก์ชันสามเหลี่ยม	42
2-45 แสดงกราฟของฟังก์ชันสี่เหลี่ยมคางหมู	43
2-46 แสดงกราฟของฟังก์ชันเกาส์เซียน	43
2-47 แสดงกราฟของฟังก์ชันระฆังคว่ำ	43
2-48 แสดงกราฟของฟังก์ชันตัวเอส	44
2-49 แสดงกราฟของฟังก์ชันตัวแซด	44
2-50 แสดงตัวแปรภาษากำหนดย่านความเร็ว	45
2-51 แสดงโครงสร้างพื้นฐานของการประมวลผลระบบพีชชี	46
2-52 แสดงขั้นตอนการประมวลผลของระบบพีชชี	47
2-53 แสดงขั้นตอนที่ 1 ของการประมวลผลของระบบพีชชี	47
2-54 แสดงขั้นตอนที่ 2 ของการประมวลผลของระบบพีชชี	48
2-55 แสดงขั้นตอนที่ 3 ของการประมวลผลของระบบพีชชี	48
2-56 แสดงขั้นตอนที่ 4 ของการประมวลผลของระบบพีชชี	48

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
2-57 แสดงระบบกฎฟิซซี่	49
2-58 แสดงกราฟฟิคของวิธีการจัดองค์ประกอบแบบค่าสูงสุด-ต่ำสุด	51
2-59 แสดงกราฟฟิคของวิธีการจัดองค์ประกอบแบบค่าสูงสุดผลคูณ	52
3-1 แสดงไดอะแกรมของงานวิจัย	53
3-2 แสดงการวิเคราะห์โครงสร้างของแบบจำลองสำหรับนิ้วมือทั้ง 4	55
3-3 แสดงการวิเคราะห์โครงสร้างของแบบจำลองของนิ้วหัวแม่มือ	56
3-4 แสดงการเปรียบเทียบขนาดของภาพถ่ายมือจริงกับแบบจำลองมือ	58
3-5 แสดงแบบจำลองบริเวณฝ่ามือ	58
3-6 แสดงการเคลื่อนไหวที่จุดต่อ MCP ในลักษณะการกาง	59
3-7 แสดงการเคลื่อนไหวที่จุดต่อ MCP จุดต่อ PIP และจุดต่อ DIP ในลักษณะการพับงอ	59
3-8 แสดงการเคลื่อนไหวที่จุดต่อ CMC และจุดต่อ MCP ในลักษณะการกาง	60
3-9 แสดงการเคลื่อนไหวที่จุดต่อ CMC จุดต่อ MCP และจุดต่อ IP ในลักษณะการพับงอ	60
3-10 แสดงแบบจำลองมือ	60
3-11 แสดงภาพถ่ายมือ	61
3-12 แสดงผลการตรวจจับสีที่ปลายนิ้วมือทั้ง 5 นิ้ว	62
3-13 แสดงผลการหาตำแหน่งสีของปลายนิ้วหัวแม่มือ	62
3-14 แสดงผลการหาตำแหน่งสีที่ปลายนิ้ว	62
3-15 แสดงไดอะแกรมของการแปลงตำแหน่ง 3 มิติ เป็น 2 มิติ	63
3-16 แสดงทรงกลมขนาด 1 หน่วย บนระบบ 3 มิติ บนโปรแกรมโรโบซิม 2	63
3-17 แสดงลักษณะการเก็บข้อมูลในระนาบ ZX	64
3-18 แสดงลักษณะข้อมูล 3 มิติ ที่จัดเก็บเป็นชั้นของแกน Y ตั้งแต่ 0 ถึง 40	64
3-19 แสดงพิกัด XY ในระบบ 2 มิติ	65
3-20 แสดงโครงสร้างของระบบโครงข่ายประสาทเทียม	65
3-21 แสดงแผนภาพการฝึกฝนระบบโครงข่ายประสาทเทียม	66
3-22 แสดงการดูเข้าของระบบโครงข่ายประสาทเทียม	66
3-23 แสดงการเข้าสู่ตำแหน่งจากการพับงอนิ้ว	68
3-24 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน Y (e_y)	69
3-25 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน Y (Δe_y)	70

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
3-26 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_2 ($\Delta\theta_2$)	70
3-27 แสดงการเข้าสู่ตำแหน่งจากการกางนิ้ว	71
3-28 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน X (e_x)	72
3-29 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน X (Δe_x)	72
3-30 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_1 ($\Delta\theta_1$)	72
3-31 แสดงทิศทางการหมุนของมุม θ_1	73
3-32 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_1 สำหรับการควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้วเมื่อตำแหน่งของปลายนิ้วอยู่ต่ำกว่าจุดต่อ MCP	74
3-33 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน X (e_x)	75
3-34 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน X (Δe_x)	75
3-35 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_1 ($\Delta\theta_1$)	75
3-36 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน Y (e_y)	78
3-37 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน Y (Δe_y)	78
3-38 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_2 ($\Delta\theta_2$)	78
4-1 แสดงสัญญาณภาษามืออเมริกันของพยัญชนะ A, B, C และ D	79
4-2 แสดงกลุ่มสี่ที่บริเวณปลายนิ้วของสัญญาณภาษามืออเมริกันของพยัญชนะ A, B, C และ D	80
4-3 แสดงตำแหน่งสี่ที่บริเวณปลายนิ้วของสัญญาณภาษามืออเมริกันของพยัญชนะ A, B, C และ D	81
4-4 แสดงตำแหน่งของทรงกลมจากหน้าต่างแสดงผล	82
4-5 แสดงการปรับมุมหมุนที่จุดต่อของแบบจำลองมือด้วยการวนซ้ำ	84
4-6 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วหัวแม่มือ	84
4-7 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วชี้	85
4-8 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วกลาง	85
4-9 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วนาง	86
4-10 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วก้อย	86
4-11 แสดงการปรับมุมหมุนที่จุดต่อของแบบจำลองมือด้วยระบบฟัซซี่	87
4-12 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วหัวแม่มือ	87

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
4-13 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิวซี	87
4-14 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิวกลาง	88
4-15 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิวนาง	88
4-16 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิวก้อย	89
4-17 แสดงการซ้อนภาพระหว่างภาพถ่าย กับภาพจากหน้าต่างแสดงผลด้วยการทดสอบ การควบคุมแบบวนซ้ำ	89
4-18 แสดงการซ้อนภาพระหว่างภาพถ่าย กับภาพจากหน้าต่างแสดงผลด้วยการทดสอบ การควบคุมด้วยระบบฟัซซี่	90
4-19 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ A	92
4-20 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ B	93
4-21 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ C	93
4-22 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ D	93
4-23 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ E	94
4-24 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ F	94

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เทคโนโลยีการใช้งานมือกลเพื่อช่วยปฏิบัติงานต่างๆแทนมนุษย์ได้เข้ามามีบทบาทในการปฏิบัติงานเป็นอย่างมาก ทั้งในงานด้านอุตสาหกรรม ด้านการแพทย์และด้านการทหาร โดยส่วนใหญ่เป็นงานที่ค่อนข้างอันตรายและต้องการความละเอียดสูง ปัจจุบันมีสถานศึกษาและองค์กรต่างๆได้ศึกษา พัฒนาและวิจัยเทคโนโลยีที่เกี่ยวข้องกับมือกลอย่างต่อเนื่อง โดยให้มือกลมีพื้นฐานทางกายภาพใกล้เคียงมือมนุษย์ (Human Hand Model Based on Anatomy) ทั้งในส่วนการออกแบบโครงสร้าง โปรแกรมในการประมวลผลและวิธีการควบคุมมือกล ซึ่งการควบคุมมือกลนั้นจำเป็นต้องติดต่อสื่อสารระหว่าง มือกลกับผู้ใช้งาน โดยผ่านคอมพิวเตอร์ซึ่งแต่เดิมการควบคุมจะใช้อุปกรณ์ต่างๆ เข้ามาเป็นสื่อกลางในการควบคุมการทำงาน ตัวอย่างเช่น แป้นพิมพ์ เมาส์และจอยสติ๊ก เป็นต้น แต่อุปกรณ์เหล่านี้มีผลกระทบต่อความไว สมรรถนะและความเป็นธรรมชาติในการควบคุมของมือกล

จากเหตุผลดังกล่าวจึงมีการพัฒนาการควบคุมมือกลด้วยวิธีใหม่ โดยผู้ใช้งานสามารถควบคุมได้อย่างเป็นอิสระและเป็นธรรมชาติ จากการวิจัยของ Sturman และ Zeltzer (1994) ได้อาศัยวิธีการพื้นฐานจากถุงมือ (Glove Based Method) ซึ่งสามารถรับข้อมูลจากจุดต่างๆของมือได้โดยตรง รวดเร็วและน่าเชื่อถือ จะเห็นได้ว่าการควบคุมโดยใช้วิธีการพื้นฐานจากถุงมือ นั้นมีลักษณะใกล้เคียงมือมนุษย์ แต่ยังคงขาดความสะดวกในการใช้งาน เนื่องจากต้องมีการสวมอุปกรณ์ต่างๆ เช่น อุปกรณ์ตรวจจับและสายต่อพ่วงรับส่งสัญญาณ นอกจากนั้นราคาของอุปกรณ์ตรวจจับยังมีราคาสูง จากงานวิจัยของ Kuch และ Huang (1995) ได้เสนอแนวทางในการสร้างแบบจำลองมือกลเพื่อวิเคราะห์และควบคุมโดยใช้หลักการพื้นฐานจากการมองเห็น (Vision Based Method) แบ่งได้ 2 ประเภท ตามรายงานของ Haiying และ Matthew (2005) วิธีแรกใช้หลักการพื้นฐานจากภาพที่ปรากฏ (Appearance Based Method) ข้อมูลที่ได้จากตำแหน่งสำคัญต่างๆจากภาพเงาหรือส่วนโค้ง แสดงรูปร่างของมือ (Contour Based Method) จะนำมาใช้สร้างแบบจำลองในการควบคุมมือกล แต่มีข้อจำกัดหากภาพเงาที่ได้มีการบังของนิ้วอาจเกิดความผิดพลาดของข้อมูลได้ วิธีที่สองใช้หลักการพื้นฐานจากการจำลองแบบ (Model Based Method) ข้อมูลที่ได้จากวิธีนี้จะเป็นข้อมูลบนระบบ 3 มิติ โดยจะจับส่วนสำคัญต่างๆ ของภาพจริง 2 มิติ มาคำนวณหาตำแหน่งของมือบนระบบ 3 มิติ และมุมของข้อต่อนิ้ว เพื่อใช้สร้างแบบจำลองในการควบคุมมือกลต่อไป

จากหลักการดังกล่าว ผู้วิจัยมีแนวคิดให้แบบจำลองแสดงท่าทางตามภาพจริง เพื่อควบคุมมือกลโดยอาศัยวิธีการพื้นฐานจากถ่วงมือ และจากการจำลองแบบมาประยุกต์ร่วมกัน ซึ่งจับส่วนสำคัญต่าง ๆ บนภาพจริง 2 มิติ ด้วยการทำสัญลักษณ์ที่ตำแหน่งต่าง ๆ บนนิ้ว และหาตำแหน่งของนิ้วจากภาพจริง 2 มิติ เพื่อเปรียบเทียบกับตำแหน่งจากภาพ 2 มิติ ซึ่งได้จากการถ่ายโอนตำแหน่งของแบบจำลองมือกล 3 มิติ ให้มีความผิดพลาดต่ำที่สุด เพื่อใช้แบบจำลองมือกล 3 มิติ แสดงท่าทางตามภาพมือจริง ในส่วนการทดสอบการทำงานของแบบจำลองของมือกลใช้โปรแกรมจำลองหุ่นยนต์แบบ 3 มิติ ที่มีชื่อว่า โปรแกรมโรโบซิม 2 (Robosim 2) และใช้ทำงานร่วมกันกับโปรแกรมสำเร็จรูปแมทแลป (MATLAB)

1.2 วัตถุประสงค์ของการวิจัย

- 1.2.1 เพื่อออกแบบและสร้างแบบจำลองมือ 3 มิติ
- 1.2.2 เพื่อสร้างระบบติดตามท่าทางโดยอาศัยข้อมูลจากภาพจริง 2 มิติ

1.3 ขอบเขตของการวิจัย

- 1.3.1 ออกแบบระบบการติดตามตำแหน่งของนิ้วมือ จากภาพจริง 2 มิติ
- 1.3.2 ออกแบบและสร้างแบบจำลองมือกล 3 มิติ ด้วยโปรแกรมโรโบซิม 2
- 1.3.3 ภาพถ่ายฝ่ามือข้างขวาและภาพถ่ายจากด้านตรงจะถูกนำมาพิจารณา
- 1.3.4 ภาพเคลื่อนไหวของนิ้วมือและมือจะไม่นำมาพิจารณา

1.4 เครื่องมือและอุปกรณ์ที่ใช้

- 1.4.1 ไมโครคอมพิวเตอร์ ที่มีหน่วยประมวลผลกลางความเร็วไม่ต่ำกว่า 1 GHz
- 1.4.2 โปรแกรมสำเร็จรูป โรโบซิม 2.0
- 1.4.3 โปรแกรมสำเร็จรูป แมทแลป 6.5
- 1.4.4 กล้องดิจิทัล

1.5 ประโยชน์ของการวิจัย

- 1.5.1 ได้แนวทางในการพัฒนาการออกแบบและสร้างแบบจำลองมือกล 3 มิติ
- 1.5.2 ได้แนวทางในการพัฒนาการควบคุมมือกลด้วยการติดตามท่าทางของมือ
- 1.5.3 เสริมสร้างศักยภาพทางด้านการออกแบบและพัฒนาเทคโนโลยีในประเทศ

บทที่ 2

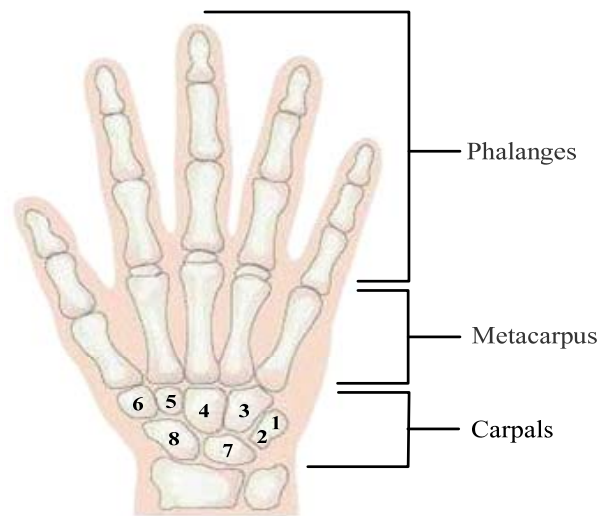
ทฤษฎีที่เกี่ยวข้อง

วิทยานิพนธ์เรื่อง การติดตามท่าทางของมือจากข้อมูลกลศาสตร์ 3 มิติที่ได้จากภาพถ่าย 2 มิติ ผู้วิจัยได้ศึกษาลักษณะทางกายภาพของมือมนุษย์ เพื่อออกแบบและสร้างแบบจำลองมือด้วยโปรแกรมจำลองหุ่นยนต์ในระบบ 3 มิติ โดยควบคุมให้แบบจำลองมือแสดงท่าทางตามภาพถ่าย ผู้วิจัยได้ศึกษาทฤษฎีที่เกี่ยวข้อง ดังหัวข้อต่อไปนี้

- 2.1 ลักษณะโครงสร้างทางกายภาพของมือมนุษย์
- 2.2 หลักจลนศาสตร์
- 2.3 โปรแกรมจำลองหุ่นยนต์ในระบบ 3 มิติ
- 2.4 การประมวลผลภาพ
- 2.5 ทฤษฎีระบบโครงข่ายประสาทเทียม
- 2.6 ทฤษฎีฟัซซี่

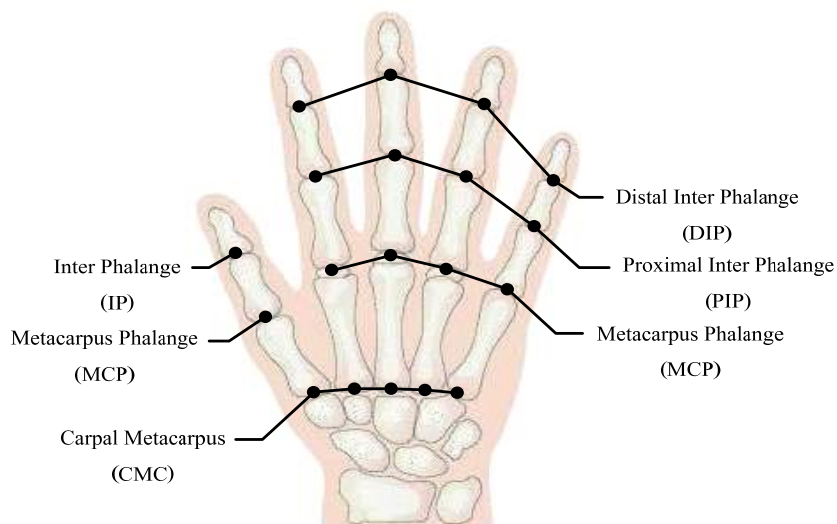
2.1 ลักษณะโครงสร้างทางกายภาพของมือมนุษย์ (Jingzhou Y., 2004)

แขนมนุษย์ประกอบด้วยฝ่ามือที่ติดกับปลายแขน โดยข้อต่อระหว่างฝ่ามือกับปลายแขนเรียกว่าข้อมือ ที่ปลายของฝ่ามือประกอบด้วยนิ้วมือจำนวน 5 นิ้ว ด้วยกันคือ นิ้วหัวแม่มือ นิ้วชี้ นิ้วกลาง นิ้วนาง และ นิ้วก้อย นิ้วมือสามารถพับงอไปด้านหลังเหนือฝ่ามือสำหรับการหยิบจับวัตถุสิ่งของต่างๆได้ มือมนุษย์นั้นมีกระดูกจำนวนทั้งหมด 27 ชิ้น ซึ่งกระดูกมือมนุษย์ทั้งหมดได้ถูกแบ่งออกเป็น 3 ส่วนด้วยกัน ส่วนแรกอยู่ในบริเวณข้อมือ (Carpals) ซึ่งเป็นกระดูกชิ้นเล็กจำนวนทั้งสิ้น 8 ชิ้น ส่วนที่สองเป็นกระดูกบริเวณฝ่ามือ (Metacarpus) มีจำนวนทั้งสิ้น 5 ชิ้น ด้วยกัน ส่วนที่สามมีจำนวนทั้งสิ้น 14 ชิ้น เป็นกระดูกนิ้ว (Phalanges) โดยกระดูกนิ้วมือนั้นยังแบ่งออกเป็นส่วนของนิ้วหัวแม่มือจำนวน 2 ชิ้น และนิ้วทั้ง 4 นิ้ว นิ้วละ 3 ชิ้นด้วยกัน แสดงดังภาพที่ 2-1



ภาพที่ 2-1 แสดงโครงสร้างกระดูกมือข้างขวา

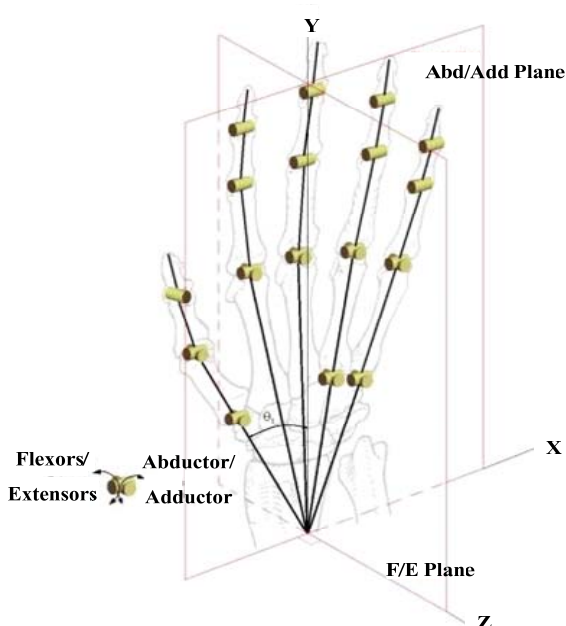
ข้อต่อที่เชื่อมต่อระหว่างกระดูกนิ้วมือไว้ด้วยกันสามารถแบ่งได้ดังนี้ จุดต่อระหว่างกระดูกข้อมือกับกระดูกฝ่ามือ (Carpal Metacarpus: CMC) จะอยู่บริเวณข้อมือของทุกนิ้ว จุดต่อระหว่างกระดูกฝ่ามือกับกระดูกนิ้ว (Metacarpus Phalange: MCP) จะอยู่บริเวณฐานของนิ้วมือทุกนิ้ว เฉพาะนิ้วหัวแม่มือจะมีจุดต่อระหว่างกระดูกนิ้วกับกระดูกนิ้วส่วนปลายของนิ้วหัวแม่มือ (Inter Phalange: IP) จุดต่อระหว่างกระดูกนิ้วของนิ้วมือทั้ง 4 นิ้ว (Proximal Inter Phalange: PIP) โดยไม่ใช่จุดต่อสุดท้ายจุดต่อระหว่างกระดูกนิ้วกับกระดูกนิ้วส่วนปลาย (Distal Inter Phalange: DIP) โดยจุดต่อแบบ PIP และ DIP จะมีเฉพาะนิ้วมือทั้ง 4 เท่านั้น แสดงดังภาพที่ 2-2



ภาพที่ 2-2 แสดงจุดต่อระหว่างกระดูกนิ้วมือ

การเคลื่อนไหวของนิ้วมือนมนุษย์เกิดจากทำงานร่วมกันระหว่างกล้ามเนื้อและเส้นเอ็น ทำให้เกิดการพับงอของนิ้วมือ (Flexors/Extensors) กับการกางของนิ้วมือ (Abductor/Adductor) โดยการพับงอของนิ้วมือ เกิดจากการดึงของกล้ามเนื้อข้างใต้ข้อต่อ (Flexors) และการยืดตัวของกล้ามเนื้อ (Extensors) ส่วนการกางของนิ้วมือเกิดจากการดึงของกล้ามเนื้อให้นิ้วกางออก (Abductor) และการคลายของกล้ามเนื้อ (Adductor) ด้วยเหตุนี้มือมนุษย์จึงสามารถใช้งานหยิบจับสิ่งของต่างๆได้

การเคลื่อนไหวของนิ้วมือที่บริเวณจุดต่อระหว่างกระดูกส่วนต่างๆ มีการเคลื่อนไหวที่แตกต่างกันดังนี้ บริเวณจุดต่อระหว่างกระดูกข้อมือกับกระดูกฝ่ามือ (CMC) ของนิ้วชี้และนิ้วกลางจะไม่สามารถขยับได้ แต่ในส่วนของนิ้วหัวแม่มือ นิ้วนางและนิ้วก้อย สามารถขยับได้โดยมีมุมหมุนอิสระ (Degree of Freedom: DOFs) อยู่ 2 มุม ลักษณะการขยับจะเป็นแบบการพับงอ (Flexors/Extensors) และการกางของนิ้วมือ (Abductor/Adductor) บริเวณจุดต่อระหว่างกระดูกฝ่ามือกับกระดูกนิ้ว (MCP) ของทุกนิ้วสามารถขยับได้โดยมีมุมหมุนอิสระ (DOFs) อยู่ 2 มุม ลักษณะการขยับจะเป็นแบบการพับงอ (Flexors/Extensors) และการกางของนิ้วมือ (Abductor/Adductor) บริเวณจุดต่อระหว่างกระดูกนิ้วกับกระดูกนิ้วส่วนปลายของนิ้วหัวแม่มือ (IP) สามารถขยับได้โดยมีมุมหมุนอิสระ (DOFs) อยู่ 1 มุม มีลักษณะการขยับจะเป็นแบบการกางของนิ้วมือ (Abductor/Adductor) ส่วนบริเวณจุดต่อระหว่างกระดูกนิ้ว (PIP) และบริเวณจุดต่อระหว่างกระดูกนิ้วกับกระดูกนิ้วส่วนปลาย (DIP) ของนิ้วมือทั้ง 4 นิ้ว จะมีมุมหมุนอิสระ (DOFs) เท่ากันคือมี 1 มุม โดยลักษณะการขยับเป็นแบบการพับงอ (Flexors/Extensors) แสดงมุมหมุนอิสระของข้อต่อต่างๆของมือดังภาพที่ 2-3



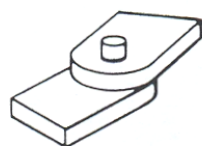
ภาพที่ 2-3 แสดงการขยับของมุมหมุนอิสระที่จุดต่อต่างๆของมือ

จากภาพที่ 2-3 พบว่าจำนวนมุมหมุนอิสระทั้งหมดของมือมนุษย์มีจำนวนทั้งสิ้น 25 มุม ส่วนการขยับนิ้วมือแบบการพับงอ (Flexors/Extensors) สามารถเขียนเป็นระนาบของการพับงอของนิ้ว (F/E Plane) ซึ่งอยู่บนระนาบเดียวกับระนาบของแกน Y-Z และการขยับนิ้วมือแบบการกางนิ้ว (Abductor/Adductor) สามารถเขียนเป็นระนาบของการกางนิ้ว (Abd/Add Plane) ซึ่งอยู่บนระนาบเดียวกับระนาบของแกน X-Y ดังภาพที่ 2-3

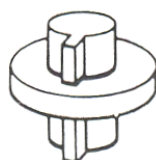
2.2 หลักจลนศาสตร์ (สถาพร, 2548)

จลนศาสตร์ (Kinematics) คือ การศึกษาทางด้านเรขาคณิตโดยเฉพาะการเคลื่อนที่ของแขนกลในหุ่นยนต์โดยไม่คิดแรง รูปร่าง ขนาด และน้ำหนัก การศึกษาจะคำนวณหาตำแหน่ง ความเร็ว และความเร่งของการเคลื่อนที่ของแขนกลผ่านระบบก้านโยง (Linkages) หรือกลไก หากพิจารณาหุ่นยนต์หรือแขนกลในทางจลนศาสตร์แขนกลเกิดจากต่อกันของก้านโยงตั้งแต่ 2 ก้านขึ้นไปซึ่งเป็นชิ้นส่วนคู่ทางจลนศาสตร์โดยชิ้นส่วนที่ต่อกันจะเกิดที่บริเวณข้อต่อ (Joints) ซึ่งการต่อที่บริเวณข้อต่อมี 2 แบบ คือ

ก้านโยงที่มีแนวสัมผัสเป็นแบบพื้นที่ผิวสัมผัส เรียกข้อต่อแบบนี้ว่า Lower Pairs ข้อต่อแบบนี้มี 2 แบบด้วยกันคือ ข้อต่อแบบหมุน (Revolute Joint) และข้อต่อแบบเลื่อน (Prismatic Joint) แสดงดังภาพที่ 2-4



ก) ข้อต่อแบบหมุน



ข) ข้อต่อแบบเลื่อน

ภาพที่ 2-4 แสดงข้อต่อพื้นฐานทางจลนศาสตร์

ก้านโยงที่มีการสัมผัสเป็นแบบจุด เรียกแบบนี้ว่า แบบ Higher Pairs กลไกที่พบเห็นมาได้แก่ ชุดกลไกลูกเบี้ยว และชุดเกียร์ เป็นต้น

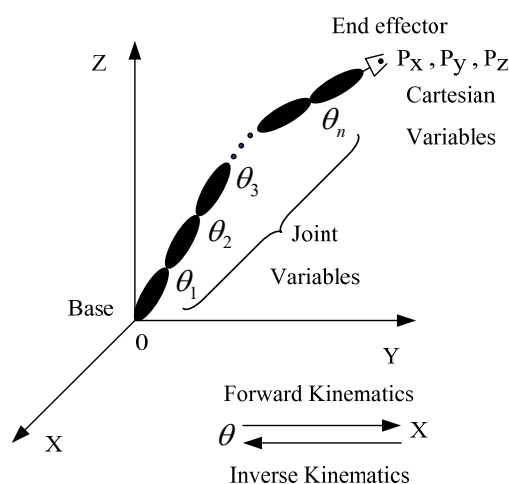
ปัญหาทางด้านจลนศาสตร์ของหุ่นยนต์มี 2 แบบ คือ จลนศาสตร์แบบไปข้างหน้า (Forward Kinematics) และจลนศาสตร์แบบผกผัน (Inverse Kinematics)

2.2.1 จลนศาสตร์แบบไปข้างหน้า (Forward Kinematics)

การคำนวณหาตำแหน่งที่ปลายของแขนกล (End Effectors) โดยคำนวณจากสมการการเคลื่อนที่ของหุ่นยนต์คำตอบที่ออกมาอยู่ในรูปของจุด (X,Y,Z) หรือระบบคาร์ทีเซียน โดยต้องทราบตัวแปรที่ข้อต่อของแต่ละข้อต่อ (Joint Variables) กรณีข้อต่อแบบหมุนตัวแปรที่ต้องทราบค่าคือ มุม θ ที่ทราบค่าในแต่ละข้อต่อนำไปคำนวณหาค่าพิกัดที่ปลายของแขนกล

2.2.2 จลนศาสตร์แบบผกผัน (Inverse Kinematics)

การคำนวณค่าตัวแปรของแต่ละข้อต่อโดยกำหนดตำแหน่งที่ปลายของแขนกลวิธีการคำนวณจลนศาสตร์แบบผกผันนี้ในบางครั้งมีได้หลายคำตอบหรือไม่สามารถหาคำตอบได้ การคำนวณนี้ค่อนข้างยุ่งยากกว่าการคำนวณจลนศาสตร์แบบไปข้างหน้า ซึ่งคำตอบของวิธีการคำนวณจลนศาสตร์ผกผันนี้อาจเป็นได้ 2 รูปแบบคือ แบบปิด (Closed Form) แบบคำนวณซ้ำเชิงตัวเลข (Numerical Iterative Form) โดยวิธีการแบบปิด (Closed Form) นั้นสามารถหาคำตอบให้อยู่ในรูปของฟังก์ชันซึ่งง่ายต่อการคำนวณหาค่าเพราะสามารถแทนค่าในตัวแปรของฟังก์ชัน ส่วนวิธีการแบบคำนวณซ้ำเชิงตัวเลข (Numerical Iterative Form) นั้นใช้วิธีสมมุติค่าตัวเลขเริ่มต้นแล้วทำซ้ำไปจนกว่าค่านั้นไม่มีการเปลี่ยนแปลงหรือเรียกว่า ลู่เข้า (Converge) ซึ่งเป็นการยากต่อการคำนวณเพราะในบางครั้งอาจหาคำตอบไม่ได้



ภาพที่ 2-5 แสดงแผนภาพจลนศาสตร์แบบไปข้างหน้า และจลนศาสตร์แบบผกผัน

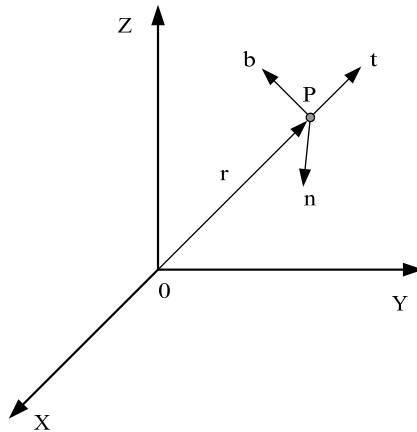
2.2.3 ตำแหน่ง (Position)

การบอกตำแหน่งโดยทั่วไปมักบอกโดยอ้างอิงกับสิ่งที่ยึดติดอยู่ซึ่งก็คือแกนอ้างอิง (Reference Frame) และระยะทางโดยประมาณ การสั่งงานของหุ่นยนต์หรือแขนกลให้เคลื่อนที่ไปยังตำแหน่งที่ต้องการเป็นการยาก เพราะหุ่นยนต์ไม่สามารถรับรู้สภาพแวดล้อมได้ดีเท่ากับมนุษย์ แต่ถ้าสามารถบอกตำแหน่งก็จะสามารถทำให้หุ่นยนต์หรือแขนกลเคลื่อนที่ไปยังเป้าหมายได้ การเข้าหาตำแหน่งจะต้องมีการกำหนดจุดพิกัด (Coordinates) เพื่อให้แขนกลรู้ถึงตำแหน่งของวัตถุ โดยต้องมีการกำหนดแกนอ้างอิงด้วยในที่นี้ใช้การบอกพิกัดแบบคาร์ทีเซียน ทิศทางของแกน Z มีนิยามจากกฎมือขวา

กรณีหนังสือวางอยู่บนโต๊ะและแฉกสี่จุดหนึ่งบนหนังสือเพื่อใช้เป็นจุดอ้างอิงแล้วเรียกจุดนั้นว่าจุด P เราสามารถบอกได้ว่าหนังสือเล่มนี้อยู่ที่ตำแหน่งสัมพันธ์กับแกนอ้างอิง ซึ่งอาจจะวาง

จุดศูนย์กลางของแกนอ้างอิงไว้ที่มุมหนึ่งของโต๊ะเราสามารถแสดงพิกัดของจุด P ได้ว่า $P = (x,y,z)$ หรือเขียนเป็นเวกเตอร์คอลัมน์ (Column Vector) มีขนาด 3×1 ได้ดังสมการที่ 2-1 โดยจุด P แสดงถึงจุดบนวัตถุ ภาพที่ 2-6 และ r คือ เวกเตอร์ตำแหน่ง (Position Vector) จากจุด O ถึง P

$$r = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2-1)$$



ภาพที่ 2-6 แสดงตำแหน่งของจุด P

2.2.4 เมตริกซ์การหมุน (Rotation Matrix)

เมตริกซ์การหมุนใช้สัญลักษณ์ R เป็นตัวแปรที่ใช้ระบุว่าวัตถุนั้นมีการหมุนเปลี่ยนไปจากแกนอ้างอิงเท่าใดแสดงได้ด้วยเวกเตอร์หนึ่งหน่วย 3 ทิศทาง คือ เวกเตอร์ n , t , b ดังแสดงบนจุด P ในภาพที่ 2-6 โดยที่ n , t , b เป็นเวกเตอร์หนึ่งหน่วย ดังสมการที่ 2-2

$$R = \begin{bmatrix} n_x & t_x & b_x \\ n_y & t_y & b_y \\ n_z & t_z & b_z \end{bmatrix} \quad (2-2)$$

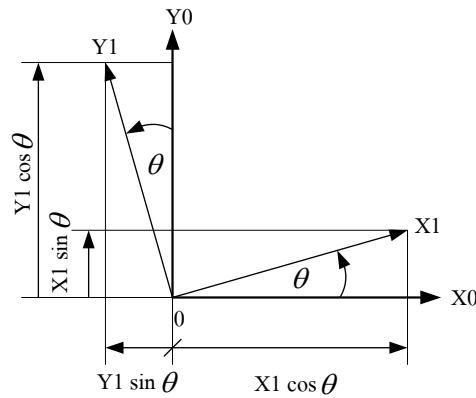
เมตริกซ์การหมุน R แสดงถึงการหมุนของวัตถุสัมพันธ์กับแกนอ้างอิงที่ติดอยู่กับที่ สังเกตว่าในเวกเตอร์คอลัมน์ของเมตริกซ์การหมุน R ประกอบด้วย เวกเตอร์ n , t , b ซึ่งตั้งฉากซึ่งกันและกัน (Perpendicular) ในกรณีเมตริกซ์การหมุน R มีคุณสมบัติออร์โธนอร์มัล (Orthornormal) ซึ่งมีคุณสมบัติที่เวกเตอร์ทั้ง 3 ตั้งฉากกันและมีขนาดหนึ่งหน่วย สามารถเขียนได้สมการที่ 2-3

$$n \cdot t = t \cdot b = b \cdot n = 0 \quad (2-3)$$

จากคุณสมบัติออร์โธนอร์มัลของเวกเตอร์ทั้ง 3 สามารถพิสูจน์การหาอินเวอร์สของเมตริกซ์การหมุนดังสมการที่ 2-4

$$R^T R = \begin{bmatrix} n^T n & n^T t & n^T b \\ t^T n & t^T t & t^T b \\ b^T n & b^T t & b^T b \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I \quad (2-4)$$

สรุปได้ว่าการทรานสโฟรีสของเมตริกซ์การหมุนนั้นคือการอินเวอร์สการคูณของเมตริกซ์การหมุนนั่นเองหากหาเมตริกซ์การหมุนรอบแกน Z สังเกตจากภาพที่ 2-7 แกนอ้างอิงที่ 1 หมุนไปจากแกนอ้างอิงที่ 0 รอบแกน Z เป็นมุม θ



ภาพที่ 2-7 แสดงการหมุนรอบแกน Z

สามารถแสดงเวกเตอร์ในแกนอ้างอิงที่ 1 นำเสนอในแกนอ้างอิงที่ 0 ได้ดังสมการที่ 2-5

$$\begin{aligned} x_0 &= x_1 \cos \theta - y_1 \sin \theta \\ y_0 &= x_1 \sin \theta + y_1 \cos \theta \\ z_0 &= z_1 \end{aligned} \quad (2-5)$$

จากสมการที่ 2-5 เขียนอยู่ในรูปของเวกเตอร์ได้ $X_0 = R_z(\theta)X_1$ สามารถแยกเมตริกซ์การหมุนในแกน Z ได้ดังสมการที่ 2-6

$$R_z(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2-6)$$

ทำนองเดียวกันสามารถหาเมตริกซ์การหมุนรอบแกน X และ Y ได้เช่นเดียวกันดังสมการที่ 2-7 และ 2-8

$$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \quad (2-7)$$

$$R_y(\beta) = \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix} \quad (2-8)$$

หากนำเมตริกซ์การหมุนมาเรียงกันเมื่อมีการหมุนรอบหลายแกน โดยมีการหมุนครั้งแรกเป็น R_0 จากนั้นหมุนรอบแกน X ด้วยมุม α จะได้การหมุนใหม่ ดังสมการที่ 2-9

$$R' = R_x(\alpha)R_0 \quad (2-9)$$

จากนั้นทำการหมุนรอบแกน Y เป็นมุม β สามารถเขียน ดังสมการที่ 2-10

$$R'' = R_y(\beta)R_x(\alpha)R_0 \quad (2-10)$$

และการหมุนลำดับที่ 3 รอบแกน Z แสดงดังสมการที่ 2-11

$$R''' = R_z(\theta)R_y(\beta)R_x(\alpha)R_0 \quad (2-11)$$

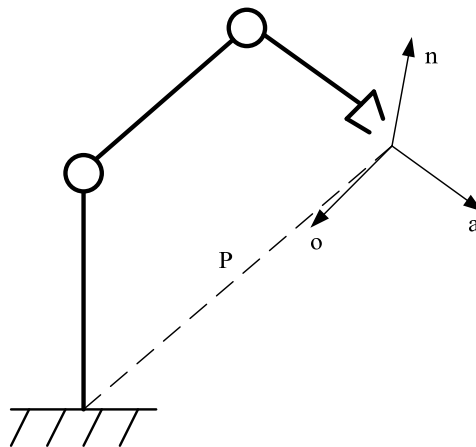
สามารถจัดรูปสมการที่ 2-11 ได้ใหม่ดังสมการที่ 2-12

$$R_{zyx} = R_z(\theta)R_y(\beta)R_x(\alpha) \quad (2-12)$$

สมการที่ 2-12 อธิบายการหมุน 3 แบบ คือ โรลล์ (Roll) พิตช์ (Pitch) และยอว์ (Yaw) โดย โรลล์ คือ การหมุนรอบแกน Z ส่วน พิตช์ คือ การหมุนรอบแกน Y และ ยอว์ คือ การหมุนรอบแกน X

2.2.5 สมการจลนศาสตร์

การบอกตำแหน่ง และการหมุนของปลายแขนกล (End Effectors) สามารถแสดงอยู่ในรูปของสมการบอกตำแหน่งด้วยเวกเตอร์ P และการหมุนด้วย (n, o, a) ดังภาพที่ 2-8



ภาพที่ 2-8 แสดงการหมุนของปลายแขนกล

ส่วนประกอบของเวกเตอร์ทั้ง 3 กำหนดขึ้นในระบบพิกัดตามแกนคาร์ทีเซียนที่ขีดติดอยู่กับฐานของหุ่นยนต์โดยที่เวกเตอร์ a คือ เวกเตอร์ขนาดหนึ่งหน่วยมีทิศทางเคลื่อนที่พุ่งเข้าหาวัตถุ ส่วนเวกเตอร์ o มีทิศทางตามนิ้วจับชิ้นงานทั้ง 2 นิ้ว ในที่นี้คือมีทิศทางตามจุดปลายทั้ง 2 ของนิ้ว และเวกเตอร์ n คือ เวกเตอร์ที่ตั้งฉาก โดยเวกเตอร์ทั้งหมดเป็นไปตามกฎมือขวาคือ $n=o \times a$ การบอกตำแหน่งสามารถบอกโดยใช้ระบบพิกัดอื่นได้อีก เช่นระบบพิกัดทรงกระบอก (Cylindrical Coordinates) และระบบพิกัดทรงกลม (Spherical Coordinates)

ส่วนการหมุนสามารถใช้วิธีการแบบอื่นได้ เช่น การหมุนแบบมุมออยเลอร์ (Euler Angles) และการหมุนแบบ RPY (Roll-Pitch-Yaw)

2.2.6 มุมออยเลอร์ (Koivo Antti J., 1989)

การหมุนแบบมุมออยเลอร์สามารถอธิบายได้ด้วยการหมุนรอบแกน 3 แกนโดยเริ่มจากแกน Z ด้วยมุม ϕ และหมุนรอบแกน Y ด้วยมุม θ และหมุนรอบแกน Z ด้วยมุม ψ หรือเรียกการหมุนนี้ว่า 2-3-2 ดังสมการที่ 2-13

$$Euler(\phi, \theta, \psi) = R_z(\phi)R_y(\theta)R_z(\psi) \quad (2-13)$$

$$Euler(\phi, \theta, \psi) = \begin{bmatrix} C\phi & -S\phi & 0 \\ S\phi & C\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & -S\theta \\ 0 & S\theta & C\theta \end{bmatrix} \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$Euler(\phi, \theta, \psi) = \begin{bmatrix} C\phi C\theta C\psi - S\phi S\psi & -C\phi C\theta S\psi - S\phi C\psi & C\phi S\theta \\ S\phi C\theta C\psi + C\phi S\psi & -S\phi C\theta S\psi + C\phi C\psi & S\phi S\theta \\ -S\theta C\psi & S\phi S\psi & C\theta \end{bmatrix}$$

ดังนั้นการหมุนของเวกเตอร์ n, o, a ณ จุดปลายของแขนกลเขียนอยู่ในรูปของ Euler Coordinates ϕ, θ, ψ ได้ดังสมการที่ 2-14

$$\begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} = \begin{bmatrix} C\phi C\theta C\psi - S\phi S\psi & -C\phi C\theta S\psi - S\phi C\psi & C\phi S\theta \\ S\phi C\theta C\psi + C\phi S\psi & -S\phi C\theta S\psi + C\phi C\psi & S\phi S\theta \\ -S\theta C\psi & S\phi S\psi & C\theta \end{bmatrix} \quad (2-14)$$

จากสมการที่ 2-14 กำหนดให้ $C\phi = \cos \phi, S\phi = \sin \phi, C\theta = \cos \theta, S\theta = \sin \theta, C\psi = \cos \psi$ และ $S\psi = \sin \psi$

2.2.7 การหมุนแบบ RPY (Roll, Pitch, Yaw)

การหมุนแบบ RPY การหมุนนี้เป็นการหมุน 3 แบบ คือ หมุนรอบแกน a, o, n ตามลำดับ โดยมีข้อจำกัดว่าแกนปัจจุบันจะขนานไปกับแกนอ้างอิง ดังนั้นการหมุนจึงเหมือนกับการหมุนแกน

อ้างอิงก่อนการหมุน RPY ถ้ากรณีแกนปัจจุบันไม่ขนานกับแกนอ้างอิง การหมุนขั้นสุดท้ายจะเป็นการรวมกันกับการหมุนก่อนหน้านี้และ Post Multiply ด้วย RPY ลำดับของการหมุนประกอบไปด้วย

โรลล์ เป็นการหมุนรอบแกน Z ที่เคลื่อนที่ หรือแกน a

พิทช์ เป็นการหมุนรอบแกน Y ที่เคลื่อนที่ หรือแกน o

ยอว์ เป็นการหมุนรอบแกน X ที่เคลื่อนที่ หรือแกน n

เมตริกซ์การหมุนแบบโรลล์ พิตช์ ยอว์ สามารถพิจารณาจากสมการที่ 2-15

$$RPY(\gamma, \beta, \alpha) = R_z(\gamma)R_y(\beta)R_x(\alpha) \quad (2-15)$$

$$RPY(\gamma, \beta, \alpha) = \begin{bmatrix} C\gamma & -S\gamma & 0 \\ S\gamma & C\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\beta & 0 & S\beta \\ 0 & 1 & 0 \\ -S\beta & 0 & C\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix}$$

$$RPY(\gamma, \beta, \alpha) = \begin{bmatrix} C\beta C\gamma & S\alpha S\beta C\gamma \cdot C\alpha S\gamma & C\alpha S\beta C\gamma + S\alpha S\gamma \\ C\beta S\gamma & S\alpha S\beta S\gamma + C\alpha C\gamma & C\alpha S\beta S\gamma \cdot S\alpha C\gamma \\ -S\beta & S\alpha C\beta & C\alpha C\beta \end{bmatrix}$$

คำตอบของการหมุนของเวกเตอร์ a, o, n สามารถหาได้ดังสมการที่ 2-16

$$\begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} = \begin{bmatrix} C\beta C\gamma & S\alpha S\beta C\gamma \cdot C\alpha S\gamma & C\alpha S\beta C\gamma + S\alpha S\gamma \\ C\beta S\gamma & S\alpha S\beta S\gamma + C\alpha C\gamma & C\alpha S\beta S\gamma \cdot S\alpha C\gamma \\ -S\beta & S\alpha C\beta & C\alpha C\beta \end{bmatrix} \quad (2-16)$$

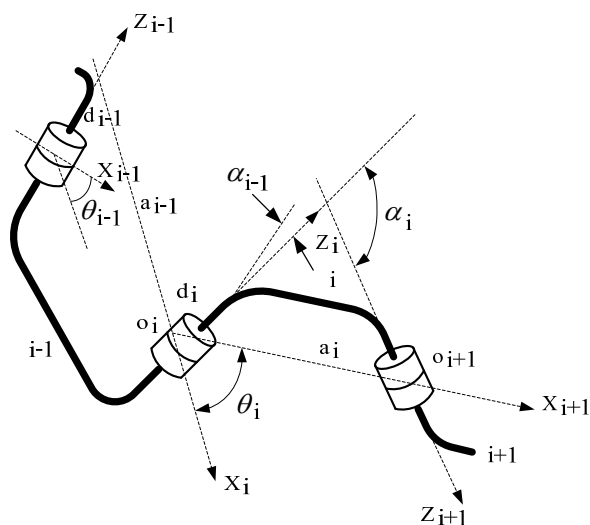
จากสมการที่ 2-16 กำหนดให้ $C\gamma = \cos \gamma, S\gamma = \sin \gamma, C\beta = \cos \beta, S\beta = \sin \beta,$

$C\alpha = \cos \alpha$ และ $S\alpha = \sin \alpha$

2.2.8 พารามิเตอร์ของ Denavit-Hartenberg

ในการสร้างแขนกลทั่วไปนั้นมักประกอบไปด้วยก้านโยงที่ต่อกันแบบอนุกรมด้วยข้อต่อ จากนั้นจะกำหนดแกนอ้างอิงที่ติดกับวัตถุในแต่ละก้านโยง เพื่อคำนวณการเปลี่ยนแกนอ้างอิงแต่ละก้านโยงไปยังปลายสุดท้ายหรือจุดปลายของแขนกล โดยทั่วไปมักเป็นมือที่ใช้หยิบจับชิ้นงาน จากการคำนวณนี้สามารถบอกได้ว่าตำแหน่งของมือจับชิ้นงานนั้นอยู่ตำแหน่งใดในระนาบ 3 มิติ

ในการคำนวณสามารถหาพิสดารต่างๆ ลำดับต่อไป คือ ทำอย่างไรเพื่อใช้ตัวแปรที่กล่าวข้างต้นนั้นมาใช้กับหุ่นยนต์วิธีที่เป็นที่ยอมรับวิธีหนึ่งได้แก่ พารามิเตอร์ของ Denavit-Hartenberg หรือใช้ตัวย่อ D-H Parameters จากภาพที่ 2-9 บอกถึงพารามิเตอร์ต่างๆใน D-H Parameters ในกรณีของข้อต่อแบบหมุน (R: Revolute Joint) ตัวแปรที่เปลี่ยนแปลง คือ มุม θ โดยระยะ d คงที่



ภาพที่ 2-9 แสดงพารามิเตอร์ของ Denavit-Hartenberg กรณีสอดคล้องแบบหมุน

นิยามของตัวแปรดังนี้

a_i เป็นระยะจากแกน Z_{i-1} ถึงแกน Z_i วัดตามแนวแกน X_i

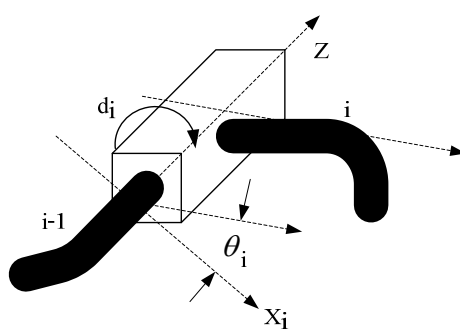
d_i เป็นระยะจากแกน X_{i-1} ถึงแกน X_i วัดตามแนวแกน Z_{i-1}

α_i เป็นมุมบิดระหว่าง Z_{i-1} กับ Z_i รอบแกน X_i

θ_i มุมระหว่างแกน X_{i-1} ถึงแกน X_i หมุนรอบแกน Z_{i-1}

ส่วนกรณีข้อต่อแบบเลื่อน (P: Prismatic Joint) ตัวแปรที่เปลี่ยนค่า คือ ระยะ d ดังภาพที่ 2-10

โดยมุม θ มีค่าคงที่



ภาพที่ 2-10 แสดงพารามิเตอร์ของ Denavit-Hartenberg กรณีสอดคล้องแบบเลื่อน

วิธีการกำหนดแกนมีดังนี้

1. เริ่มจากที่ฐานกำหนดแกนอ้างอิงเป็น o ร่วมกับแกนอ้างอิงหลัก

2. กำหนดแกนในแต่ละก้านโยงตามกฎมือขวาโดยให้แกนหมุนเป็นแกนเริ่มต้นที่ o เช่น Z_0

ในส่วนข้อต่อแบบหมุนหรือข้อต่อแบบเลื่อนทิศทางของแกน Z ไปในทิศทางออกจากก้านโยง

3. กำหนดแกนของก้านโยง ถัดไปให้ดูตามแกนการหมุนโดยให้แกนนั้นเป็นแกน Z_I
4. ทิศทางของแกน X_I หาได้โดยได้มาจากผลลัพท์ของ Cross Product $Z_0 \times Z_I$ แต่ละแกนจะถูกกำหนดที่ส่วนปลายของข้อต่อของก้านโยง
5. กรณีมีระยะออฟเซตจะมีระยะ d หรือถ้าไม่มี $d = 0$
6. แกน Z อยู่ร่วมในแนวเดียวกับแกนการหมุนของข้อต่อ

2.2.9 เมตริกซ์การแปลง (Transformation Matrix)

ในวิชาหุ่นยนต์จุดสนใจจุดหนึ่งโดยทั่วไปคือ ตำแหน่งของวัตถุในระนาบ 3 มิติ จากเมตริกซ์การหมุน ซึ่งบอกถึงการหมุนของวัตถุใน 3 มิติ และเพิ่มการบอกตำแหน่งพิกัดเข้าไปอีก 1 หลัก จะได้เมตริกซ์ใหม่นี้เรียกว่า เมตริกซ์การแปลง (Transformation Matrix) โดยมีขนาด 4×4 มีลักษณะดังสมการที่ 2-17

$$T_{NA} = \begin{bmatrix} \text{Rotation}_{(3 \times 3)} & \text{Translation}_{(3 \times 1)} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & P \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-17)$$

โดย N คือ แกนอ้างอิงเริ่มต้น และ A คือ แกนอ้างอิงสุดท้ายเช่น วัตถุมีการหมุนรอบแกน X_0 เป็นมุม θ และเลื่อนไปยังตำแหน่ง $(1,1,0)$ ที่ตำแหน่งของแกนอ้างอิงที่ 1 สามารถเขียนได้ดังสมการที่ 2-18

$$T_{01} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & \cos \theta & -\sin \theta & 1 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-18)$$

เมตริกซ์การแปลงนั้นมีคุณสมบัติเช่นเดียวกับเมตริกซ์หมุนคือ สามารถนำมาต่อกันสำหรับกรณีที่มีการเคลื่อนย้ายวัตถุในหลายๆ แกนอ้างอิงดังสมการที่ 2-19

$$T_{03} = T_{01} T_{12} T_{23} \quad (2-19)$$

จากสมการที่ 2-19 แสดงถึง T_{03} สามารถอ้างอิงจากแกนอ้างอิงที่ 1 ไปยังแกนอ้างอิงที่ 2 และแกนอ้างอิงที่ 3 อย่างไรก็ตามการหาอินเวอร์สของ T นั้นไม่เหมือนกับเมตริกซ์การหมุน เพราะเมตริกซ์การแปลงไม่มีคุณสมบัติออร์ทอโนมัลการหาอินเวอร์สเมตริกซ์ T หาได้จากสมการที่ 2-20

$$T^{-1} = \begin{bmatrix} R^T & (-R^T P) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-20)$$

จากเมตริกซ์การแปลงเราสามารถเขียนอยู่ในรูปเมตริกซ์การเคลื่อนย้าย (Translation Matrix) เพียงอย่างเดียวดังสมการที่ 2-21 (Schilling R., 1990)

$$Trans(p_x, p_y, p_z) = \begin{bmatrix} 1 & 0 & 0 & p_x \\ 0 & 1 & 0 & p_y \\ 0 & 0 & 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-21)$$

ในทำนองเดียวกันยังสามารถเขียนเป็นเมตริกซ์การหมุนในแนวแกน X, Y และ Z ดังสมการที่ 2-22 ถึงสมการที่ 2-24 (Schilling R., 1990)

$$Rot_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta_x & -\sin \theta_x & 0 \\ 0 & \sin \theta_x & \cos \theta_x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-22)$$

$$Rot_y(\theta_y) = \begin{bmatrix} \cos \theta_y & 0 & \sin \theta_y & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_y & 0 & \cos \theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-23)$$

$$Rot_z(\theta_z) = \begin{bmatrix} \cos \theta_z & -\sin \theta_z & 0 & 0 \\ \sin \theta_z & \cos \theta_z & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-24)$$

หากนำพารามิเตอร์ของ Denavit-Hartenberg มาเขียนเป็นเมตริกซ์การแปลงสามารถเขียนในรูปของการเรียงต่อกันของเมตริกซ์การเคลื่อนย้ายและเมตริกซ์การหมุนดังสมการที่ 2-25

$$T_{NA} = Trans(0, 0, d_i) Rot_z(\theta_i) Rot_x(\alpha_i) Trans(a_i, 0, 0) \quad (2-25)$$

$$T_{NA} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta_i & -S\theta_i & 0 & 0 \\ S\theta_i & C\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha_i & -S\alpha_i & 0 \\ 0 & S\alpha_i & C\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{NA} = \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-26)$$

จากสมการที่ 2-26 กำหนดให้ $C\theta_i = \cos \theta_i$, $S\theta_i = \sin \theta_i$, $C\alpha_i = \cos \alpha_i$ และ $S\alpha_i = \sin \alpha_i$ จากเมตริกซ์การแปลงซึ่งได้จาก D-H parameters ดังสมการที่ 2-26 เมื่อนำมาเทียบกับสมการที่ 2-17 พบว่าหากแบ่งเมตริกซ์ในสมการที่ 2-26 ออกเป็นส่วนแล้วพบว่าเมตริกซ์ย่อยที่มุมบนด้านซ้ายที่มีขนาด 3×3 เป็นเมตริกซ์การหมุนของก้านโยงที่ i ส่วนเมตริกซ์ย่อยที่มุมบนด้านขวาที่มีขนาด 3×1 เป็นเมตริกซ์ตำแหน่งของก้านโยง

2.3 โปรแกรมจำลองหุ่นยนต์ในระบบ 3 มิติ (Koseeyaporn P., 2002)

โปรแกรมโรโบซิม 2 (Robosim 2) เป็นโปรแกรมจำลองการทำงานหุ่นยนต์ในระบบ 3 มิติ โปรแกรมแบบจำลองนี้ใช้สำหรับการสร้างแบบจำลองหุ่นยนต์ชนิดต่างๆรวมไปถึงสภาพแวดล้อมหรือวัตถุที่ต้องการสร้างขึ้นและสามารถแสดงการเคลื่อนที่ หรือการทำงานของแบบจำลองหุ่นยนต์ในหน้าต่างการทำงาน (RoboSvr Window) การสร้างแบบจำลองมีวัตถุประสงค์เพื่อออกแบบหุ่นยนต์ตลอดจนให้เห็นขั้นตอนหรือขบวนการควบคุมการทำงาน ซึ่งผลจากการจำลองคือ ความปลอดภัย การคาดการณ์ล่วงหน้า เทคนิคการควบคุม ค่าใช้จ่ายที่ลดลง และสามารถทำซ้ำได้หลาย ๆ ครั้ง

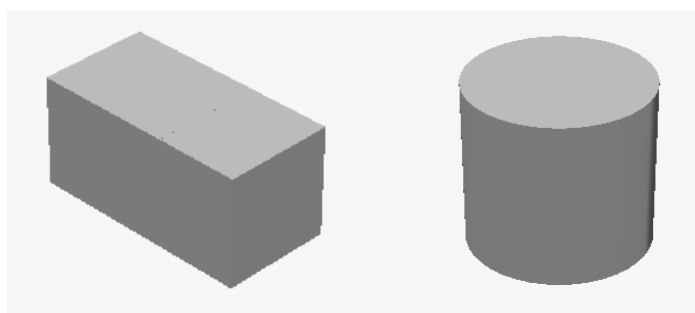
สำหรับโปรแกรมจำลองการทำงานของหุ่นยนต์นี้ใช้คอมพิวเตอร์ เพื่อช่วยประหยัดค่าใช้จ่ายในด้านการออกแบบการสร้างหุ่นยนต์ ทั้งยังสามารถควบคุมและแสดงการทำงานได้ เนื่องจากหุ่นยนต์ที่พบในงานจริงมีหลายรูปแบบและเทคนิคขบวนการควบคุมแตกต่างกัน ดังนั้นแบบจำลองจะช่วยเพิ่มทักษะรวมถึงลดความซับซ้อนอีกด้วย เพราะแบบจำลองสามารถเปลี่ยนแปลงรูปแบบหรือขั้นตอนการทำงานได้อย่างรวดเร็ว มีการเพิ่มหรือลดองค์ประกอบได้ง่าย

โปรแกรมโรโบซิมถูกพัฒนาตั้งแต่ปี 1988 ระหว่างองค์การนาซา กับมหาวิทยาลัยแวนเดอร์บิลท์ ประเทศสหรัฐอเมริกาโดย Fernandez และพัฒนาเรื่อยมาจนถึงปี 1993 สามารถทำงานบนวินโดวส์เวอร์ชันที่แสดงผลเป็นกราฟฟิกโดย Springfield และในปี 2000 ได้ถูกพัฒนาให้มีความสามารถทำงานบนปฏิบัติการวินโดวส์ แสดงผลการทำงานของกราฟฟิกชิ้นงานจากการสร้างรายการคำสั่ง โดย P. Koseeyaporn การใช้งานของโปรแกรมมีข้อแนะนำของคอมพิวเตอร์สำหรับการทำงาน คือเป็นคอมพิวเตอร์ที่มีความเร็วในการประมวลผลไม่น้อยกว่า 486 หรือเพนเทียมโปรเซสเซอร์ 100 MHz หน่วยความจำแบบแรมต้องไม่ต่ำกว่า 16 Mbytes และระบบปฏิบัติการไม่น้อยกว่าเวอร์ชันวินโดวส์ 95

คุณสมบัติของแบบจำลองหุ่นยนต์ในโปรแกรมโรโบซิม 2 มีแบบจำลองหุ่นยนต์มากกว่าหนึ่งแบบแต่ละแบบมีหน้าที่ต่างกัน รวมไปถึงแบบจำลองทางกลไก แบบจำลองทางเรขาคณิต และแบบจำลองอุปกรณ์ตรวจจับ ผลจากแบบจำลองการเคลื่อนที่ได้มาจาก แบบจำลองที่กล่าวมาข้างต้น

2.3.1 แบบจำลองเรขาคณิต

อธิบายในส่วนของชิ้นส่วนโครงสร้างทางด้านฟิสิกส์ ที่นำมาสร้างเป็นหุ่นยนต์ โดยแบบจำลองที่นำมาใช้จะเป็นรูปทรงเรขาคณิตพื้นฐานเช่น ทรงสี่เหลี่ยมผืนผ้า ทรงกระบอก รูปสี่เหลี่ยม ทรงแกรวย ทรงกลม ทรงครึ่งวงกลม หรือส่วนโค้งของวงกลม โดยแบบจำลองที่สร้างขึ้นประกอบด้วยจำนวนเส้นเล็กๆที่นำมาต่อกันจนเป็นขอบของวัตถุออกมาเป็นพื้นที่โค้งและผิวของทรงเรขาคณิตที่ต้องการ ดังภาพที่ 2-11



ภาพที่ 2-11 แสดงรูปทรงเรขาคณิตรูปกล่อง และทรงกระบอก

2.3.2 แบบจำลองอุปกรณ์ตรวจจับ

ส่วนนี้มีความสำคัญเมื่อใช้ในงานจริงตัวอย่างอุปกรณ์ตรวจจับเช่น จับภาพ ตรวจจับโลหะ ลิมิทเซนเซอร์ เซนเซอร์แบบสัมผัสมีความจำเป็นในการจำลองการทำงานพร้อมกับการจำลองการเคลื่อนที่ของหุ่นยนต์ เพื่อนำผลที่ได้ไปใช้งานจริงโดยโปรแกรมโรโบซิม มีส่วนของอุปกรณ์เหล่านี้

2.3.3 การจำลองสภาพแวดล้อม

การจำลองสภาพแวดล้อมในพื้นที่ใช้งานมีความสำคัญต่อการควบคุมเพราะจะแสดงให้เห็นถึงสภาพแวดล้อมที่หุ่นยนต์สามารถทำงานได้ โดยที่สามารถเปลี่ยนแปลงสภาพแวดล้อมให้มีความแตกต่างกันได้ยกตัวอย่างสภาพแวดล้อมได้นำ อวกาศ

2.3.4 ภาษาที่ใช้ในโปรแกรมแบบจำลองหุ่นยนต์

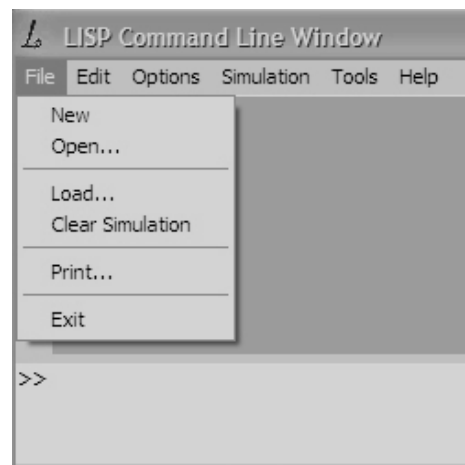
ประกอบด้วยองค์ประกอบพื้นฐานของการสร้างวัตถุ การควบคุมการเคลื่อนที่ การติดต่อกับพอร์ตอินพุตและเอาต์พุต สามารถเขียนโปรแกรมและแสดงการทำงานได้ทันทีทั้งหุ่นยนต์ พื้นที่ที่ทำงาน และสภาพแวดล้อมที่สร้างขึ้น บ้างครั้งสามารถเขียนเป็นโปรแกรมแบบอินเตอร์แอคทีฟโดยผ่านพอร์ตอินพุตได้

ระบบแบบจำลองหุ่นยนต์โรโบซิมจะแสดงผลเป็นกราฟฟิคบนระบบ 3 มิติ ที่ประกอบด้วย ส่วนต่างๆดังกล่าวนมา ภาพที่ 2-12 แสดงหน้าต่าง LISP Command Line Window



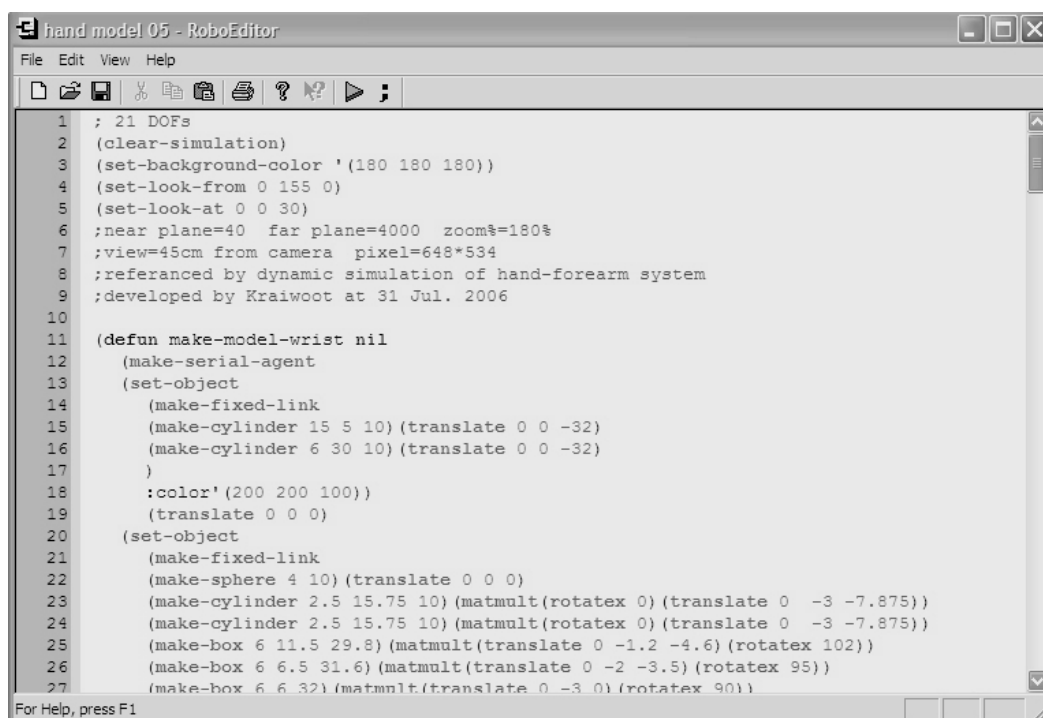
ภาพที่ 2-12 แสดงหน้าต่าง LISP Command Line Window

ในส่วนส่วนของหน้าต่างใช้งานที่มีส่วนของเมนูอยู่ด้านบนประกอบไปด้วย เครื่องพิมพ์ หน้าต่างแก้ไข ตัวเลือกอื่น แบบจำลองตัวอย่าง และคู่มือการใช้คำสั่ง แสดงดังภาพที่ 2-13



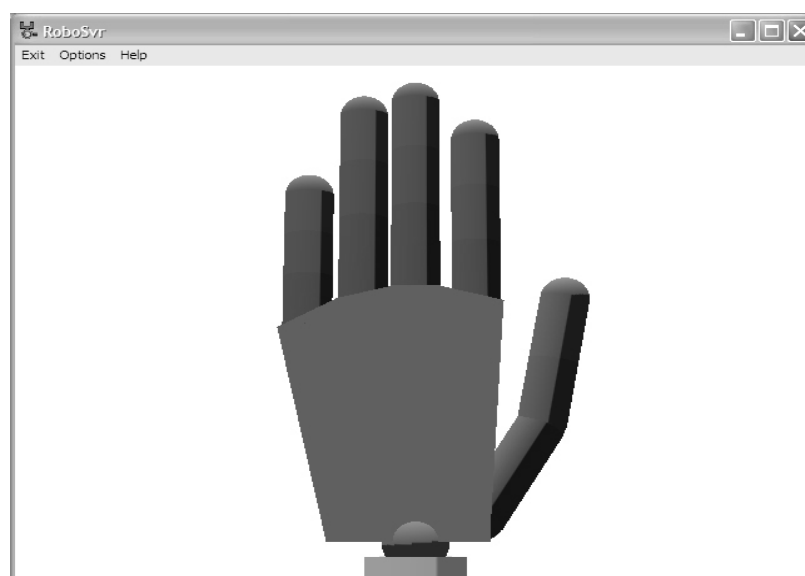
ภาพที่ 2-13 แสดงหน้าต่างเมนู File ในโปรแกรมโรโบซิม

การสร้างหุ่นยนต์สามารถเขียนโปรแกรมเพื่อสร้างแบบจำลองหุ่นยนต์ได้บนหน้าต่าง RoboEditor ซึ่งจะเก็บเป็นไฟล์ *.lsp แสดงดังภาพที่ 2-14



ภาพที่ 2-14 แสดงหน้าต่าง RoboEditor สำหรับเขียนโปรแกรมเพื่อสร้างหุ่นยนต์

เมื่อรันโปรแกรมโรโบซิมโปรแกรมจะสร้างแบบจำลองหุ่นยนต์เขียนไว้บนหน้าต่างแสดงแบบจำลองหุ่นยนต์ (RoboSvr Window) แสดงดังภาพที่ 2-15



ภาพที่ 2-15 แสดงหน้าต่างแสดงแบบจำลองหุ่นยนต์

ภาษาที่ใช้เขียนโปรแกรมจะเรียกว่า ภาษาลิปส์ (LISP Language) รูปแบบของภาษาลิปส์เป็นแบบภาษาที่เข้าใจง่ายที่เป็นสัญลักษณ์ และเป็นรายการของคำสั่งเช่น จำนวน ข้อความ คำเฉพาะ

และ nil การพิมพ์ nil ในโปรแกรมเทียบได้กับการพิมพ์ () ยกตัวอย่างของรายการ (1 2 3) สามารถเขียนได้เป็น (1 2 3 nil) สัญลักษณ์ที่เป็นตัวแปร เป็นการกำหนดค่าของตัวแปร ยกตัวอย่างถ้าให้ค่าของ 2 กำหนดเท่ากับสัญลักษณ์ a ดังนั้นผลลัพธ์ของ a มีค่าเท่ากับ 2 สามารถแบ่งเป็น 3 กลุ่มคือ กลุ่มพิเศษ กลุ่มมาโคร และกลุ่มฟังก์ชัน

กลุ่มพิเศษ จะไม่ถูกประเมินค่า กลุ่มมาโครคล้ายกับกลุ่มพิเศษ เนื่องจากได้มาจากกลุ่มพิเศษคือไม่ถูกประเมินค่า กลุ่มฟังก์ชันจะมีการประเมินค่าจากส่วนของฟังก์ชัน และผลของการประเมินค่าเป็นกลุ่มของฟังก์ชันด้วย ความแตกต่างของกลุ่มพิเศษและกลุ่มฟังก์ชันจะเห็นได้อย่างชัดเจนในตัวอย่างของตัวแปร ยกตัวอย่าง เครื่องหมาย => จะแสดงผลลัพธ์ที่พิมพ์เข้าไปในระบบเครื่องหมาย (;) จะหมายถึงแสดงการสิ้นสุดคำสั่ง และแสดงเป็นส่วนของการอธิบาย ไม่ถูกนำมาใช้ในโปรแกรม เครื่องหมาย (') ในกลุ่มพิเศษยกตัวอย่าง 'a จะหมายถึงการอ้างอิง (Quote a) ตัวอย่างของการใช้สัญลักษณ์ในกลุ่มต่าง ๆ

```
1 => 1
```

```
"hello world" => "hello world"
```

```
nil => nil
```

```
() => nil
```

```
(setq a 1) => 1
```

```
a => 1
```

```
'a => a
```

(Setq) จะเป็นกลุ่มพิเศษ ถูกใช้สำหรับการสร้างตัวแปร ส่วนแรกของ setq จะไม่ถูกประเมินค่า แต่ส่วนที่สองจะถูกประเมินค่าผลจากการประเมินค่าในส่วนที่สอง จะเก็บไว้ในส่วนแรก ตัวอย่างเช่น

```
(setq a 1 b 2) => 2
```

```
(setq a 2 b (+ a 2)) => 4
```

```
b => 4
```

กลุ่มฟังก์ชัน เป็นค่ากลุ่มฟังก์ชันที่ทุกค่าจะถูกประเมินค่า การสร้างฟังก์ชันในลิปส์ และการใช้สามารถที่จะกำหนดเป็นฟังก์ชันขึ้นได้ใหม่โดยรูปแบบเป็น defun ตัวอย่างการใช้เครื่องหมาย +, -, * และ /

```
(+ 1 2) => 3
```

```
(+ 2) => 2
```

```
(- 5 2 1) = 2
```

```
(* 2 (+ 3 4)) => 14
```

```
(setq a (* 3 5)) => 15
```

```
(/ 1 2) => 0
```

```
(/ 1. 2) => 0
```

```
(/ 1.0 2) => 0.5
```

ในสามตัวอย่างสุดท้าย จะแสดงให้เห็นถึงความแตกต่างของจำนวนเต็มและจำนวนทศนิยม ผลของการหารที่เป็นจำนวนเต็ม และทศนิยม

การสร้างฟังก์ชัน (Defined Functions) ผู้ใช้สามารถสร้างฟังก์ชันใหม่โดยใช้รูปแบบของกลุ่มพิเศษ คือ defun ตัวอย่างเช่น

(defun cube(x)(* x x x)) จากคำสั่งฟังก์ชันที่สร้างชื่อ cube จากฟังก์ชันนำมาใช้ (cube 3) มีค่าเท่ากับ 27

```
>> (defun cube (x)(* x x x))
```

```
CUBE
```

```
>> (cube 3)
```

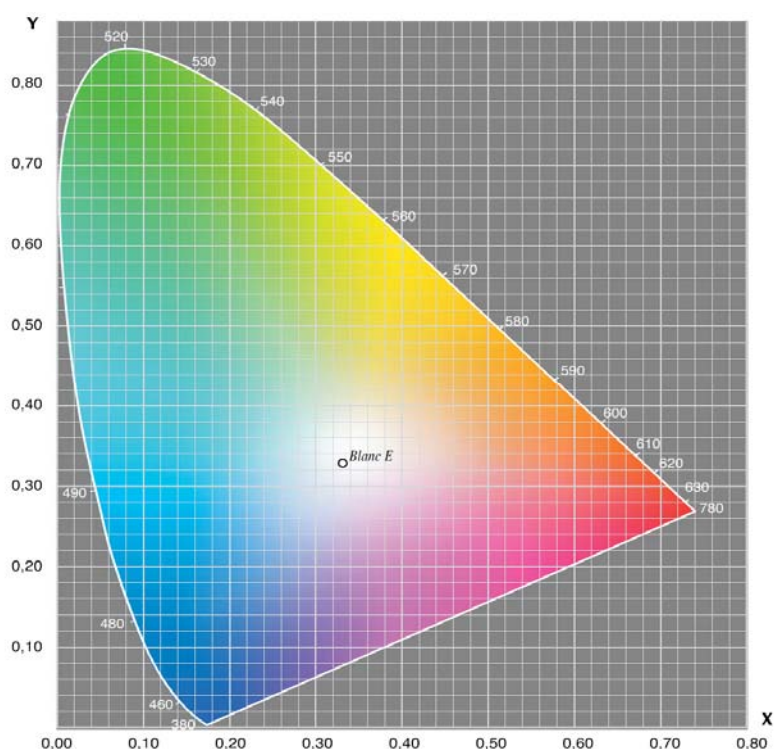
```
27
```

2.4 การประมวลผลภาพ (สละไกร, 2548)

การประมวลผลภาพ (Image Processing) เป็นการประยุกต์ใช้งานการประมวลผลสัญญาณแบบ 1 มิติ บนสัญญาณ 2 มิติ เช่น ภาพนิ่ง ภาพถ่ายหรือภาพวิดีโอ และยังรวมถึงสัญญาณ 2 มิติ อื่นๆที่ไม่ใช่ภาพด้วย การประมวลผลภาพนั้นจะอยู่ในรูปของการประมวลผลสัญญาณอะนาล็อก (Analog) โดยใช้อุปกรณ์ปรับแต่งแสง (Optics) ยกตัวอย่างเช่น ฮอโลกราฟี (Holography) เนื่องจากอุปกรณ์คอมพิวเตอร์มีราคาถูก และความเร็วในการประมวลผลสูงขึ้น การประมวลผลภาพดิจิทัล (Digital Image Processing) จึงได้รับความนิยมมาก เนื่องจากมีความแม่นยำ

2.4.1 ทฤษฎีสี

สี คือลักษณะความเข้มของแสงที่ปรากฏแก่สายตาให้เห็นเป็นสีผ่านกระบวนการรับรู้ด้วยการมองเห็น โดยที่ตาได้วิเคราะห์ข้อมูลพลังงานแสง ผ่านประสาทสัมผัสการมองเห็น ผ่านศูนย์สลับเปลี่ยนในสมองไปสู่ศูนย์การมองเห็นวิเคราะห์แยกแยะให้รับรู้ วิธีการตรวจวัดสีจะต้องอ้างอิงกับ Commission International de l'Eclairage หรือ CIE โดยถือว่าการตรวจวัดมาตรฐานสามเหลี่ยมสี CIE



ภาพที่ 2-16 แสดงแผนภาพมาตรฐานสามเหลี่ยมสี Commission International de l'Eclairage

สามเหลี่ยมสี CIE ดังภาพที่ 2-16 แสดงถึงรูปสามเหลี่ยมเอ็กมา นำเสนอไว้ในปี 1931 โดยการวิเคราะห์สีจากแสงสเปกตรัม สัมพันธ์กับความยาวคลื่นแสงแสดงถึงสีขาวท่ามกลางแสงสเปกตรัมรอบรูปเอ็กมาโค้งรูปเอ็กมาแสดงความยาวคลื่นจาก 380 – 780 นาโนเมตรสามเหลี่ยมสี CIE สร้างตามระบบความสัมพัทธ์พิกัด XY คาร์ทีเซียน (Cartesian Coordination System) ในทางคณิตศาสตร์จากมุมตรงข้าม 3 มุมของรูปเอ็กมาคือ สีน้ำเงินม่วงเข้มประมาณ 380 นาโนเมตร สีเขียวประมาณ 520 นาโนเมตร และสีแดงประมาณ 780 นาโนเมตร คือสีจากแสงที่นำมาผสมกันและก่อให้เกิดสีต่างๆขึ้น แสงสีแดงมีความยาวคลื่นสูงสุด แต่มีความถี่คลื่นต่ำสุดจะหักเหน้อยที่สุด

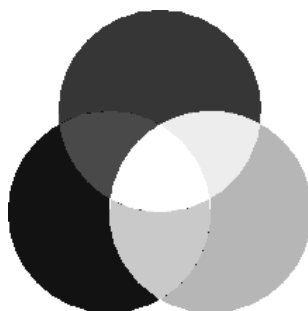
โครงสร้างของสามเหลี่ยมสี CIE เกิดจากการทดลองค้นคว้าทางวิทยาศาสตร์ ในระบบการพิมพ์อุตสาหกรรม การถ่ายภาพ ภาพยนตร์ โทรทัศน์ ได้ใช้โครงสร้างนี้เป็นหลัก ในระบบการพิมพ์ได้ใช้สีจากด้าน 3 ด้านของรูปเอ็กมาคือ สีเหลือง สีฟ้า สีม่วงแดง และสีดำเป็นหลัก ในการถ่ายภาพยนตร์ โทรทัศน์ จอคอมพิวเตอร์ ใช้สีจากมุมทั้งสาม คือ แดง เขียว น้ำเงิน เป็นต้น นักวิทยาศาสตร์ได้กำหนดแม่สีของแสงไว้ 3 สี คือ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) แสงทั้งสามสี เมื่อนำมาฉายรวมกัน จะทำให้เกิดสีต่างๆขึ้นมาคือ

แสงสีแดง + แสงสีเขียว = แสงสีเหลือง (Yellow)

แสงสีแดง + แสงสีน้ำเงิน = แสงสีแดงมagenta (Magenta)

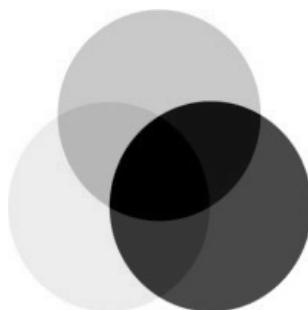
แสงสีน้ำเงิน + แสงสีเขียว = แสงสีฟ้าไซแอน (Cyan)

หากแสงทั้งสามสีมารวมกันจะได้แสงสีขาว หรือไม่มีสี ซึ่งสามารถสังเกตแม่สีของแสงได้จากโทรทัศน์หรือจอคอมพิวเตอร์สี โดยใช้แว่นขยายส่องดูหน้าจอจะเห็นเป็นแพทเทิร์นสีแสงสว่าง 3 สี คือ แดง เขียว และน้ำเงิน แสงที่เป็นแม่สี คือ สีแดง สีน้ำเงิน และสีเขียว จะเรียกว่าสีพื้นฐานบวก (Additive Primary Colors) เกิดจากการหักเหของแสงสีขาว แสดงดังภาพที่ 2-17



ภาพที่ 2-17 แสดงการผสมสีแบบพื้นฐานบวก

สีใหม่ที่เกิดจากการผสมกันของแม่สีของแสงทั้งสามจะเรียกว่า สีพื้นฐานลบ (Subtractive Primary Colors) คือ สีฟ้าไซแอน (Cyan) สีแดงมาเจนต้า (Magenta) และสีเหลือง (Yellow) แสดงดังภาพที่ 2-18



ภาพที่ 2-18 แสดงการผสมสีแบบพื้นฐานลบ

2.4.2 ระบบการมองเห็นภาพ

ระบบการมองเห็นภาพ หมายถึงทุกสิ่งที่เป็นต่อการได้มาซึ่งรหัสทางดิจิทัลที่ใช้แทนภาพนั้นๆ การปรับปรุงเปลี่ยนแปลงข้อมูลและรวมทั้งการนำเสนอภาพที่ได้หลังการปรับปรุง ซึ่งระบบการมองเห็นภาพนี้อาจมีความยุ่งยากซับซ้อนต่างกันตามลักษณะการใช้งานสามารถแบ่งเป็นขั้นตอนกว้างๆ ได้ 3 ขั้นตอนดังนี้

2.4.2.1 การได้มาซึ่งภาพ (Image Acquisition)

การแปลงภาพที่เห็นในลักษณะทางกายภาพให้เป็น เซตของข้อมูลทางดิจิทัล ซึ่งอาจจะรับภาพนั้นจากกล้อง สแกนเนอร์ กล้องวิดีโอที่เป็นอุปกรณ์ดิจิทัล ซึ่งภาพแต่ละภาพที่ได้รับจาก

อุปกรณ์เหล่านี้จะมีรายละเอียดและความคมชัดสูงขึ้นอยู่กับอัตราการสุ่มสัญญาณ (Sampling Rate) ของอุปกรณ์รับภาพ

2.4.2.2 กระบวนการประมวลผลภาพ (Image Processing)

มีหน้าที่หลักในการสร้างภาพขึ้นมาโดยการแยกแยะส่วนที่เราต้องการ หรือสนใจออกจากพื้นหลัง หรือสิ่งรบกวนต่างๆ เช่น การหาขอบภาพ การกรองภาพ การเปลี่ยนแปลงค่าสีของภาพ เป็นต้น การประมวลผลภาพจะเป็นการประมวลผลเชิงตัวเลข ซึ่งขั้นตอนดังกล่าวนี้จะถูกนำมาประมวลผลด้วยเครื่องคอมพิวเตอร์ โดยภาพที่นำมาประมวลผลจะถูกแทนด้วยตัวเลขในรูปของเมตริกซ์ แต่ภาพที่ได้ส่วนมากจะอยู่ในรูปของฟังก์ชันต่อเนื่อง $f(x,y)$ ในระนาบ 2 มิติ โดยจะเป็นสัดส่วนกับความสว่าง หรือความเข้มของภาพที่ตำแหน่ง (x,y) ซึ่งเรียกว่า ระดับสีเทา (Grey Level)

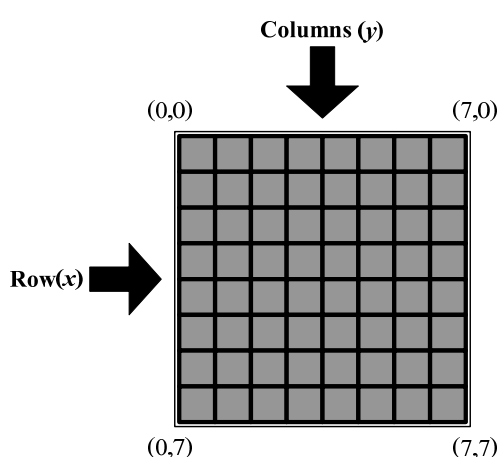
2.4.2.3 ผลที่ได้และการแสดงผล (Output and Display)

รูปแบบของการแสดงผลที่ได้จะขึ้นอยู่กับการนำข้อมูลที่ได้ไปใช้กับงานประเภทใด ซึ่งอาจนำไปแสดงผลที่เครื่องพิมพ์ หรือผ่านหน้าจอเป็นต้น

การประมวลผลภาพสามารถทำได้ 2 ลักษณะคือ กระบวนการปรับปรุงคุณภาพให้ดีขึ้น เพื่อนำไปแสดงผลต่อไป กับกระบวนการนำข้อมูลรูปภาพเพื่อไปใช้ต่อในการเก็บ การส่งข้อมูล และการวิเคราะห์ต่างๆ

2.4.3 การแทนภาพด้วยข้อมูลแบบดิจิทัล

ข้อมูลภาพแบบดิจิทัล (Digital Image) เป็นภาพที่ถูกแปลงมาจากภาพอะนาล็อกให้อยู่ในรูปของตัวเลข โดยการนำภาพอะนาล็อกมาแยกเป็นพื้นที่สี่เหลี่ยมเล็กๆ ที่เรียกว่า พิกเซล (Pixel) ดังภาพที่ 2-19



ภาพที่ 2-19 แสดงการเก็บข้อมูลแบบดิจิทัล

ในแต่ละพิกเซลจะถูกระบุตำแหน่งด้วยพิกัด (x,y) และค่าระดับสีเทาของพิกเซล โดยสามารถแปลงรูปภาพเป็นข้อมูลแบบดิจิทัลได้ด้วยขั้นตอนดังนี้

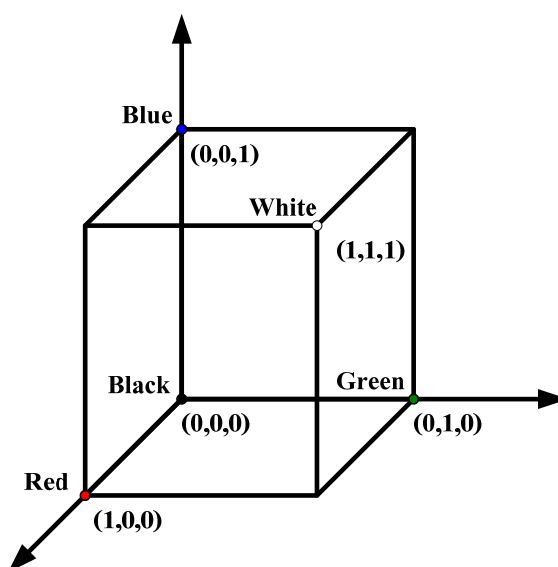
เมื่อนำสัญญาณอะนาล็อกที่ต้องการประมวลผลมาผ่านส่วนที่เรียกว่า ดิจิไทเซอร์ (Digitizer) ซึ่งจะมีหน้าที่ในการเปลี่ยนสัญญาณอะนาล็อกให้เป็นสัญญาณดิจิทัล จากนั้นทำการควอนไทซ์ (Quantizing) เพื่อที่จะประมวลสัญญาณด้วยคอมพิวเตอร์ ฟังก์ชันของภาพ $f(x,y)$ จะถูกทำให้เป็นสัญญาณที่ไม่ต่อเนื่องทั้งระนาบของภาพ ซึ่งเราเรียกว่า การสุ่มภาพ (Image Sampling) ของฟังก์ชันที่เรียกว่า การควอนไทซ์เซชันระดับสีเทา (Grey Level Quantization) จะได้ข้อมูลเป็นแบบดิจิทัล

2.4.4 การประมวลผลแบบสเปเชียล

การประมวลผลแบบสเปเชียล จะแบ่งรูปออกเป็นฟังก์ชันสองตัวแปร $f(x,y)$ ซึ่ง x และ y เป็นโคออร์ดิเนต (Coordinate) ของภาพนั้นๆ และขนาดของ f . ทุกๆ โคออร์ดิเนต (x,y) คือความเข้มของสี (Intensity) ของรูปภาพ ณ.จุดนั้นๆ ทุกๆ x,y และ f มีขนาดที่จำกัดและเป็นฟังก์ชันที่ไม่ต่อเนื่อง (Discrete Function) เราจะเรียกรูปภาพนั้นว่า รูปภาพดิจิทัล ขอบเขตของการประมวลผลทางดิจิทัลคือ การประมวลผลโดยใช้ดิจิทัลคอมพิวเตอร์

2.4.5 ทฤษฎีสี RGB

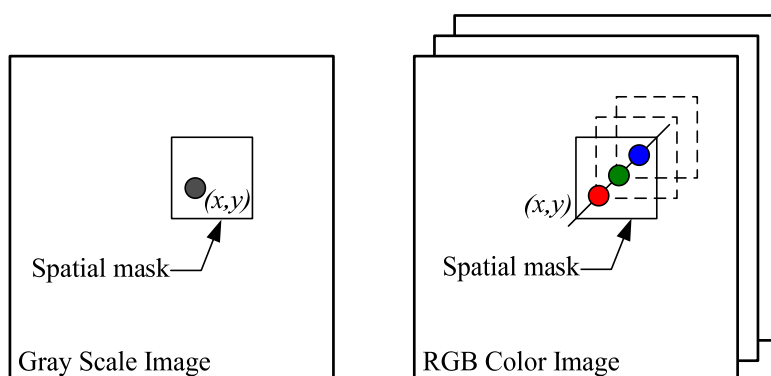
สี RGB เป็นสีที่อยู่บนระบบคาร์ทีเซียน (Cartesian Coordinate System) มีแม่สีหลักอยู่ทั้งหมด 3 สีคือ สีแดง สีเขียว และสีน้ำเงิน



ภาพที่ 2-20 แสดงระบบคาร์ทีเซียนของสี RGB

จากภาพที่ 2-20 แสดงระบบคาร์ทีเซียนของสี RGB ซึ่งประกอบด้วยรูประดับสีเทา (Grey Scale) ทั้งหมด 3 ชั้น เข้าด้วยกันคือ ชั้นของสีแดง สีเขียว และสีน้ำเงิน เข้ามารวมกันเป็นชั้นเดียว และสามารถแสดงได้ทั้ง สีแดง สีเขียว และสีน้ำเงินได้

$$f(x,y) \text{ ในโหมด RGB ณ.จุดใดๆ} = \{R,G,B\}$$



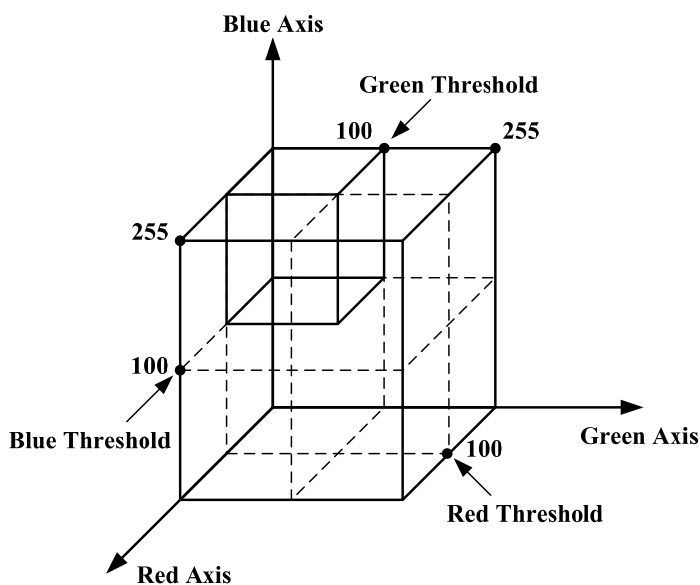
ภาพที่ 2-21 แสดงการเปรียบเทียบระหว่างข้อมูลภาพแบบระดับสีเทากับสี RGB

จำนวนของสีที่สามารถแสดงได้จะแปรตามจำนวนบิตที่ใช้แสดงค่าความเข้มของแต่ละสีซึ่งเรียกว่า Pixel Depth รูปภาพที่มีจำนวน Pixel Depth 8 บิต จะหมายถึงรูปที่มีจำนวนบิตของแต่ละสีเท่ากับ 8 บิต รวมเข้าด้วยกันสามสี ตามภาพที่ 2-21 หากเป็นภาพขนาด 24 บิต สามารถแสดงจำนวนสีได้เท่ากับ $(2^8)^3 = 16,777,246$ สี จะมีค่าของความเข้มของแต่ละสีของทั้งสามสีเป็น 256 สี แสดงค่าได้ระหว่าง 0-255

หากต้องการค่าของข้อมูลที่ละเอียดมากขึ้น จำเป็นต้องเพิ่มจำนวนบิตการแสดงผลของแต่ละพิกเซล เช่น หากแบ่งความเข้มของการส่องสว่างไว้ 4 ระดับ ต้องใช้เลขฐานสองจำนวน 2 บิต และจำนวน 4 บิต สำหรับ 16 ระดับ, จำนวน 8 บิต สำหรับ 256 ระดับ ที่ใช้ในระดับสีเทานี้มักเป็นเลขยกกำลังของ 2 ค่าที่ต่ำที่สุดคือ 0 กำหนดให้เป็นสีดำและ 1 หรือตัวเลขที่น้อยกว่าค่าสูงสุดของระดับสีเทาอยู่ 1 (เช่น 15 สำหรับระดับเทา 16 ระดับ) แทนสีขาว ค่าที่กำหนดไว้ในแต่ละพิกเซลนี้ นิยมใช้เป็นจำนวนเต็ม ในยุคแรกๆ ของระบบการมองเห็นภาพจะใช้ระบบเลขฐานสอง แต่ในปัจจุบันเทคโนโลยีไมโครโปรเซสเซอร์เข้ามามีบทบาทมากขึ้น การแบ่งระดับเป็น 16, 64 หรือ 256 เป็นเรื่องธรรมดา แต่ทั้งนี้ในการมองเห็นของมนุษย์จะแยกแยะความแตกต่างได้เพียง 10-15 ระดับเท่านั้น การแบ่งโดยละเอียดเป็น 64 หรือ 256 ระดับอาจจะนำไปประยุกต์ใช้กับงานประมวลผลภาพหรือกระบวนการอุตสาหกรรมอื่นๆ ซึ่งจะเห็นว่าจำนวนระดับสีเทาเป็นตัวจำกัดความละเอียดของภาพ โดยทั่วไปแล้วยิ่งแบ่งระดับสีเทาเป็นหลายระดับจะเป็นการเพิ่มคุณภาพด้วย และการเพิ่มจำนวนพิกเซลจาก 35×35 เป็น 250×250 ก็เป็นการเพิ่มความละเอียด (Resolution) และรายละเอียด (Detail) ของภาพเช่นกัน จะเห็นว่าความแตกต่างกับการขยายภาพ (Zoom) โดยการขยายภาพนี้เป็น การขยายแต่ละพิกเซลให้ใหญ่ขึ้น ไม่ได้เป็นการเพิ่มจำนวนพิกเซล

2.4.6 การตัดค่าสี (Color Threshold)

การตัดค่าสีเป็นกระบวนการหนึ่งในวิธีการประมวลผลภาพการตัดค่าสีแต่ละค่าในแกน RGB จะแบ่งโดยค่าสีที่ถูกกำหนดไว้ในกรอบรูปทรงสี่เหลี่ยม ดังภาพที่ 2-22

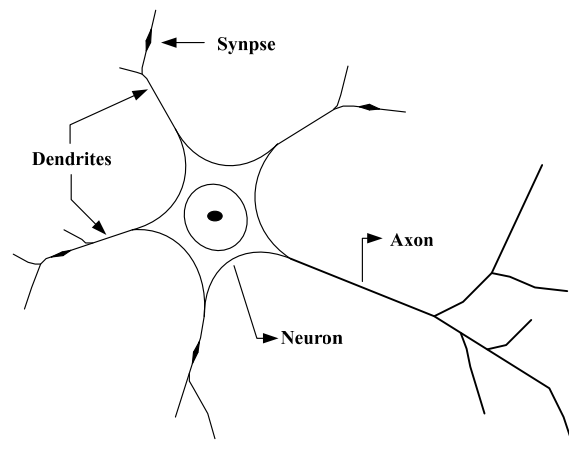


ภาพที่ 2-22 แสดงการตัดค่าสี

จากภาพที่ 2-22 เป็นการตัดเฉพาะค่าสีที่ต้องการ โดยจะเลือกเฉพาะสีที่อยู่ในบริเวณทรงสี่เหลี่ยมของแกน RGB สังเกตได้ว่าค่าสีในแกน R และ G ที่ต้องการจะอยู่ในช่วง 0 ถึง 100 ส่วนค่าสีในแกน B ที่ต้องการจะอยู่ในช่วง 100 ถึง 255 ส่วนค่าอื่นที่อยู่นอกกรอบจะถูกตัดออกไป

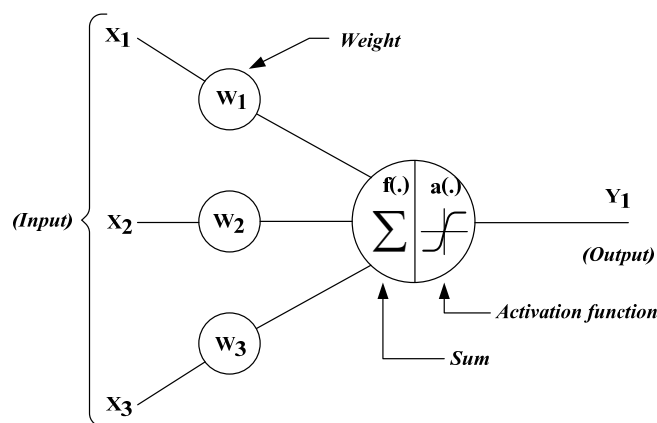
2.5 ทฤษฎีระบบโครงข่ายประสาทเทียม (Chin-Teng Lin, 1996)

ระบบโครงข่ายประสาทเทียม (Artificial Neural Network) เป็นการจำลองการทำงานโครงข่ายประสาทของมนุษย์ในรูปแบบของแบบจำลองทางคณิตศาสตร์ (Mathematical Model) โดยโครงข่ายประสาทของมนุษย์ประกอบด้วยองค์ประกอบ 3 ส่วน คือ เซลประสาท (Neuron or Soma) เส้นประสาท (Axon) และใยประสาท (Dendrites) โดยเส้นประสาทจะเสมือนท่อนำสัญญาณจากเซลล์ประสาทออกไปส่วนใยประสาทจะเชื่อมต่อกับเซลล์ประสาทอื่นๆที่ปลายของใยประสาทจะมีปมเล็กๆเรียกว่า ไสแวนป์ส (Synapse) เมื่อใยประสาทได้รับสัญญาณจากเซลล์ประสาทข้างเคียงส่งผ่านมายังไซแนปส์ไปยังเซลล์ประสาท เพื่อประมวลผลแล้วส่งสัญญาณผ่านเส้นประสาทออกไป แสดงแบบจำลองระบบโครงข่ายประสาทของมนุษย์ดังภาพที่ 2-23



ภาพที่ 2-23 แสดงแบบระบบจำลองโครงข่ายประสาทของมนุษย์

ในปี ค.ศ. 1943 McCulloch และ Pitts ได้นำเสนอบทความวิชาการ “Boolean brain” ซึ่งเป็นจุดกำเนิดของการจัดรูปแบบแบบจำลองทางคณิตศาสตร์ของระบบโครงข่ายประสาทเทียม ระบบโครงข่ายประสาทเทียมของ McCulloch และ Pitts ได้ถูกเรียกว่า ระบบโครงข่ายประสาทเทียมแบบ M-P กระบวนการทำงานของระบบโครงข่ายประสาทเทียมจะคำนวณผลรวมของน้ำหนักเชื่อมโยง (Connection Weights) ทางด้านอินพุตที่ถูกกระตุ้นเพื่อการตัดสินใจและส่งผลไปยังเอาต์พุต โดยน้ำหนักจะแทนที่ไซแนปส์ (Synapse) ที่ต่อกับเซลล์ประสาทด้านอินพุตจะแทนที่ใยประสาท (Dendrites) และเส้นประสาท (Axon) จะเป็นเอาต์พุตของระบบโครงข่ายประสาทเทียม แสดงดังภาพที่ 2-24



ภาพที่ 2-24 แบบจำลองทางคณิตศาสตร์ของระบบโครงข่ายประสาทเทียม

2.5.1 แบบจำลองของระบบโครงข่ายประสาทเทียม (ณชล, 2545)

แบบจำลองของระบบโครงข่ายประสาทเทียมที่ใช้ทั่วไปจะมีโครงสร้างพื้นฐานที่เหมือนกัน โดยมีส่วนประกอบดังต่อไปนี้

2.5.1.1 ค่าน้ำหนักเชื่อมโยง (Connection Weights) เป็นสิ่งที่ได้หลังจากการเรียนรู้ของระบบโครงข่ายประสาทเทียมเสร็จหรือบางครั้งเรียกว่า ค่าความรู้ (Knowledge) ค่าน้ำหนักเชื่อมโยงจะเปลี่ยนแปลงจากสถานะเริ่มต้นไปยังสถานะสุดท้าย เมื่อนำค่าน้ำหนักเชื่อมโยงมาประมวลผลร่วมกับข้อมูลเข้า (Input) ของระบบโครงข่ายประสาทเทียม โดยผ่านกระบวนการทางคณิตศาสตร์ที่ได้เรียกว่า ฟังก์ชันผลรวม (Summation Functions) ซึ่งเป็นผลรวมของข้อมูลเข้าสู่ระบบ (X_i) และค่าน้ำหนักเชื่อมโยง (W_i) ของข้อมูลนั้นจากข้อมูลเข้าที่ i เท่ากับ 1 จนถึงข้อมูลเข้าตัวที่ m ดังสมการที่ 2-27

$$f = \sum_{i=1}^m w_i x_i \quad (2-27)$$

2.5.1.2 ฟังก์ชันการกระตุ้น (Activation Function) เป็นการคำนวณหาขีดระดับของการส่งข้อมูลออก (y) ฟังก์ชันการกระตุ้นจะประมวลผลจากค่าของฟังก์ชันผลรวม (f) โดยสามารถแปลงเป็นสมการทางคณิตศาสตร์ ดังสมการที่ 2-28

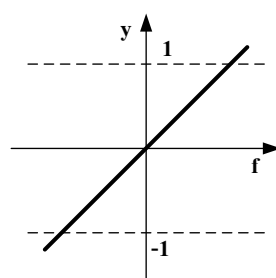
$$y_i = a(f) \quad (2-28)$$

ฟังก์ชันการกระตุ้นหรือฟังก์ชันการแปลงค่า (Transfer Function) ส่วนใหญ่นิยมใช้ได้แก่ ฟังก์ชันเชิงเส้น (Linear Function) ฟังก์ชันขั้นบันได (Unit Step Function) ฟังก์ชันซิกมอยด์ (Sigmoid Function) และฟังก์ชันไฮเพอร์บอลิก (Hyperbolic Function) โดยฟังก์ชันแต่ละชนิดสามารถอธิบายได้ดังนี้

ฟังก์ชันเชิงเส้น (Linear Function) เขียนเป็นสมการ ดังสมการที่ 2-29

$$y_i = a(f) \quad (2-29)$$

ให้ผลลัพธ์ของข้อมูลออกทุกๆค่า f ดังภาพที่ 2-25

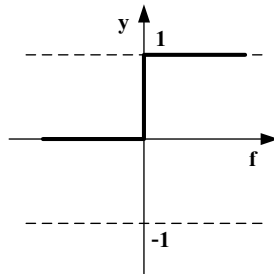


ภาพที่ 2-25 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบเชิงเส้น

ฟังก์ชันขั้นบันได (Unit Step Function) เขียนเป็นสมการ ดังสมการที่ 2-30

$$y_i = \begin{cases} 1 & \text{ถ้า } f \geq 0 \\ 0 & \text{ถ้า } f < 0 \end{cases} \quad (2-30)$$

ให้ผลลัพธ์ของข้อมูลระหว่าง 0 และ 1 ดังภาพที่ 2-26

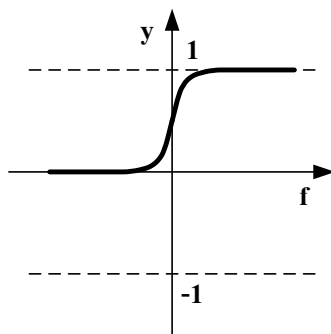


ภาพที่ 2-26 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบขั้นบันได

ฟังก์ชันซิกมอยด์ (Sigmoid Function) เขียนเป็นสมการ ดังสมการที่ 2-31

$$y_i = \frac{1}{1 + e^{-f}} \quad (2-31)$$

ให้ผลลัพธ์ของข้อมูลระหว่าง 0 และ 1 ดังภาพที่ 2-27

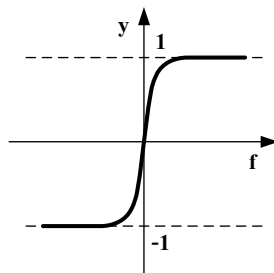


ภาพที่ 2-27 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบซิกมอยด์

ฟังก์ชันไฮเพอร์บอลิก (Hyperbolic Function) เขียนเป็นสมการที่ 2-32

$$y_i = \frac{2}{1 + e^{-f}} - 1 \quad (2-32)$$

ให้ผลลัพธ์ของข้อมูลระหว่าง 0 และ 1 ดังภาพที่ 2-28



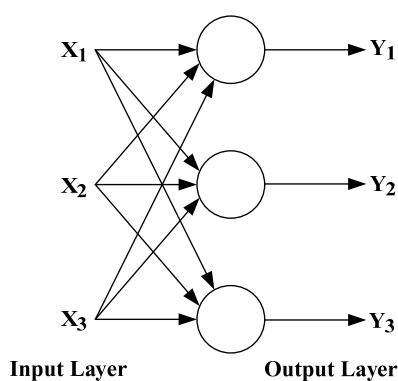
ภาพที่ 2-28 แสดงผลลัพธ์ของฟังก์ชันการกระตุ้นแบบไฮเพอร์บอลิก

2.5.2 สถาปัตยกรรมของระบบโครงข่ายประสาทเทียม (ณชล, 2545)

ระบบโครงข่ายประสาทเทียมถูกสร้างขึ้นโดยการนำระบบประสาทมาประกอบรวมกันสามารถจำแนกประเภทได้เป็น 3 ประเภท คือ โครงข่ายงานแบบป้อนไปข้างหน้าชั้นเดียว (Single Layer Feed Forward) โครงข่ายงานแบบป้อนไปข้างหน้าหลายชั้น (Multilayer Feed Forward) และโครงข่ายงานแบบเวียนเกิด (Recurrent Network)

2.5.2.1 ข่ายงานแบบป้อนไปข้างหน้าชั้นเดียว (Single Layer Feed Forward)

ข่ายงานแบบป้อนไปข้างหน้าชั้นเดียวมีโครงสร้างเป็นชั้นของระบบโครงข่ายประสาทเทียมเรียงตัวเป็นแถวเดียวกัน ดังภาพที่ 2-29

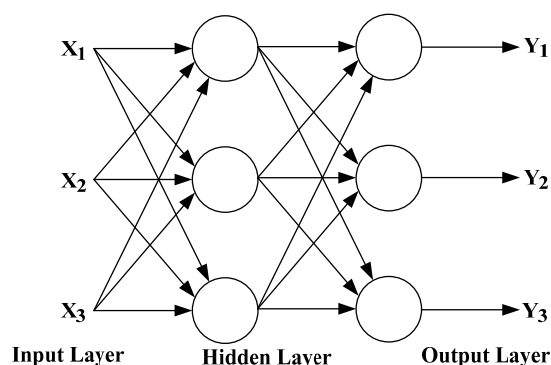


ภาพที่ 2-29 แสดงข่ายงานแบบป้อนไปข้างหน้าชั้นเดียว

จากภาพที่ 2-29 แถวของข้อมูลเข้าระบบเรียกว่า ชั้นข้อมูลเข้า (Input Layer) ในขณะที่แถวของข้อมูลออกจากระบบเรียกว่า ชั้นข้อมูลออก (Output Layer) ข้อมูลที่ออกจากระบบจะรับข้อมูลเข้าเดียวกันจากชั้นข้อมูลเข้า หลังจากการคำนวณแบบป้อนไปข้างหน้าสิ้นสุดลงชั้นข้อมูลออกจะเป็นข้อมูลออกของระบบโครงข่ายประสาทเทียม

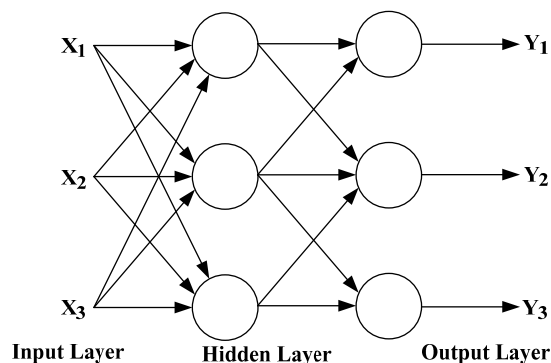
2.5.2.2 ข่ายงานแบบป้อนไปข้างหน้าหลายชั้น (Multilayer Feed Forward)

ข่ายงานแบบป้อนไปข้างหน้าหลายชั้นถูกสร้างโดยการต่อโครงข่ายงานแบบป้อนไปข้างหน้าชั้นเดียวเข้าด้วยกัน ข้อมูลออกจากชั้นแรกของระบบจะถูกส่งไปเป็นข้อมูลเข้าของชั้นต่อไป ข้อมูลออกของระบบในชั้นก่อนจะถูกเรียกว่า ชั้นซ่อน (Hidden Layer) ภาพข่ายงานแบบป้อนไปข้างหน้าหลายชั้น แสดงดังภาพที่ 2-30



ภาพที่ 2-30 แสดงข่ายงานแบบป้อนไปข้างหน้าหลายชั้นโดยเชื่อมต่อแบบเต็ม

ระบบโครงข่ายประสาทเทียมในชั้นซ่อนจะรับข้อมูลจากชั้นข้อมูลเข้า ข้อมูลออกของชั้นซ่อนจะถูกใช้เป็นข้อมูลเข้าของระบบในชั้นข้อมูลออกจากภาพที่ 2-30 จะเรียกการเชื่อมต่อแบบนี้ว่า การเชื่อมต่อแบบเต็ม (Fully Connected) เนื่องจากทุกปมของระบบโครงข่ายประสาทเทียมในแต่ชั้นจะต่อกับทุกปมของระบบโครงข่ายประสาทเทียมในชั้นถัดไป

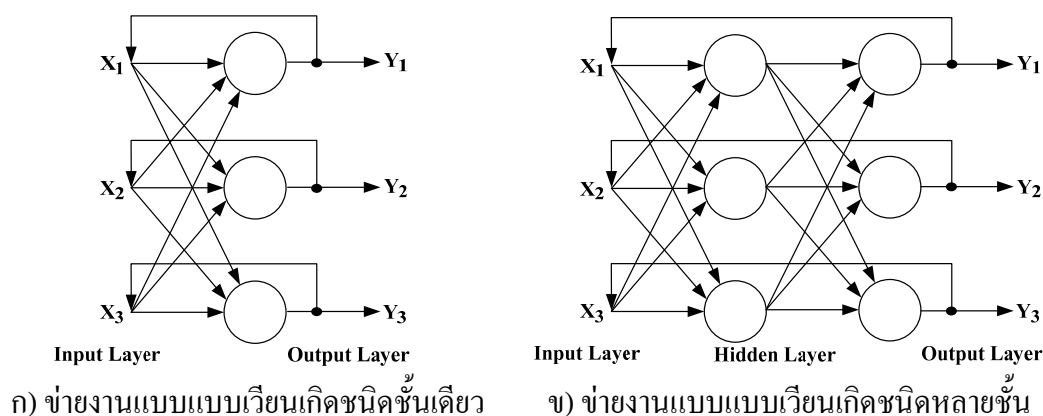


ภาพที่ 2-31 แสดงข่ายงานแบบป้อนไปข้างหน้าหลายชั้นโดยเชื่อมต่อบางส่วน

ในทางตรงกันข้ามทุกปมของระบบโครงข่ายประสาทเทียมในแต่ชั้นจะต่อกับปมของระบบโครงข่ายประสาทเทียมในชั้นถัดไปเพียงบางส่วน จะเรียกการเชื่อมต่อแบบนี้ว่า การเชื่อมต่อบางส่วน (Not Fully Connected) แสดงดังภาพที่ 2-31

2.5.2.3 ข่ายงานแบบเวียนเกิด (Recurrent Network)

ข่ายงานแบบเวียนเกิดแตกต่างจากข่ายงานแบบป้อนไปข้างหน้าคือ มีวงป้อนกลับ (Feedback Loop) อย่างน้อยหนึ่งวงอยู่ในข่ายงาน ดังภาพที่ 2-32



ภาพที่ 2-32 แสดงข่ายงานแบบเวียนเกิด

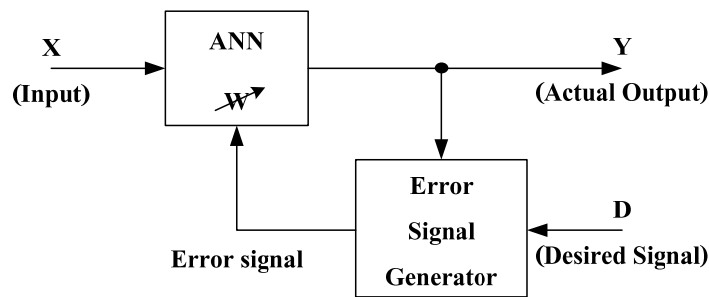
ข่ายงานแบบเวียนเกิดนั้นมีทั้งแบบชั้นเดียว และหลายชั้นแสดงไว้ในภาพที่ 2-32 โดยข่ายงานแบบเวียนเกิดชนิดหลายชั้น จะมีชั้นซ่อน เช่นเดียวกับข่ายงานระบบแบบป้อนไปข้างหน้าหลายชั้น

2.5.3 กระบวนการเรียนรู้ (Learning Process)

กระบวนการเรียนรู้เป็นสิ่งสำคัญสำหรับระบบโครงข่ายประสาทเทียม เนื่องจากระบบโครงข่ายประสาทเทียม จะเก็บข้อมูลโดยอาศัยกระบวนการเรียนรู้หรือการฝึกสอน (Training) สามารถแบ่งได้ 2 ประเภทคือ กฎการเรียนรู้ (Learning Rule) และรูปแบบการเรียนรู้ (Learning Paradigms) ในวิทยานิพนธ์นี้จะอธิบายในส่วนของรูปแบบการเรียนรู้ ซึ่งสามารถจำแนกออกได้เป็น 3 ประเภท คือ การเรียนรู้แบบมีผู้ฝึกสอน (Supervised Learning) การเรียนรู้แบบไม่มีผู้ฝึกสอน (Unsupervised Learning) และการเรียนรู้แบบเสริมสร้าง (Reinforcement Learning)

2.5.3.1 การเรียนรู้แบบมีผู้ฝึกสอน (Supervised Learning)

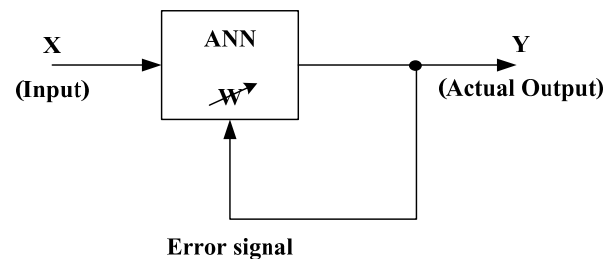
ระบบโครงข่ายประสาทเทียมจะสามารถเก็บรวบรวมความรู้ได้โดยการใช้ข้อมูลจากภายนอกในการฝึกสอน (External Training Signal) หรือค่าที่ต้องการ (Desired Signal) เป็นการเรียนรู้แบบมีเป้าหมาย การเรียนรู้แบบนี้จะไม่สามารถพัฒนาการเรียนรู้จากข้อมูลรูปแบบใหม่ได้ (Non Incremental Learning) กล่าวคือ แม้ข่ายงานจะมีความสามารถในการจดจำจากสิ่งที่เคยเรียนรู้ เพื่อใช้ประมวลผลข้อมูลที่มีลักษณะคล้ายกัน ซึ่งเป็นคุณสมบัติของข่ายงานที่มีการเรียนรู้แบบมีผู้ฝึกสอน แต่เมื่อนำไปใช้แก้ปัญหารูปแบบใหม่ซึ่งต่างจากที่เรียนรู้ จำเป็นต้องทำการฝึกสอนให้เรียนรู้รูปแบบของข้อมูลนั้นใหม่อีกครั้ง แผ่นภาพการเรียนรู้แบบมีผู้ฝึกสอน ดังภาพที่ 2-33



ภาพที่ 2-33 แสดงแผนภาพการเรียนรู้แบบมีผู้ฝึกสอน

2.5.3.2 การเรียนรู้แบบไม่มีผู้ฝึกสอน (Unsupervised Learning)

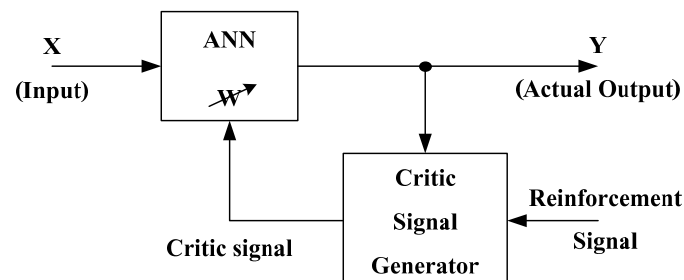
การเรียนรู้แบบไม่มีผู้ฝึกสอนนั้นจะไม่มีผู้ฝึกสอนซึ่งทำหน้าที่ป้อนสัญญาณฝึกสอนให้กับระบบโครงข่ายประสาทเทียม แต่ระบบโครงข่ายประสาทเทียมจะมีการปรับค่าน้ำหนักเชื่อมโยงเองเพื่อเข้าหาเป้าหมายแผนภาพการเรียนรู้แบบไม่มีผู้ฝึกสอน แสดงดังภาพที่ 2-34



ภาพที่ 2-34 แสดงแผนภาพการเรียนรู้แบบไม่มีผู้ฝึกสอน

2.5.3.3 การเรียนรู้แบบเสริมสร้าง (Reinforcement Learning)

เป็นการเรียนรู้ที่ให้คำตอบว่าผิดหรือถูกแต่ไม่ได้บอกว่าคำตอบที่ถูกคืออะไร โดยคำตอบที่ได้จะนำมาวิเคราะห์ความถูกต้อง จากนั้นจะนำสัญญาณไปปรับค่าน้ำหนักเชื่อมโยงดังภาพที่ 2-35

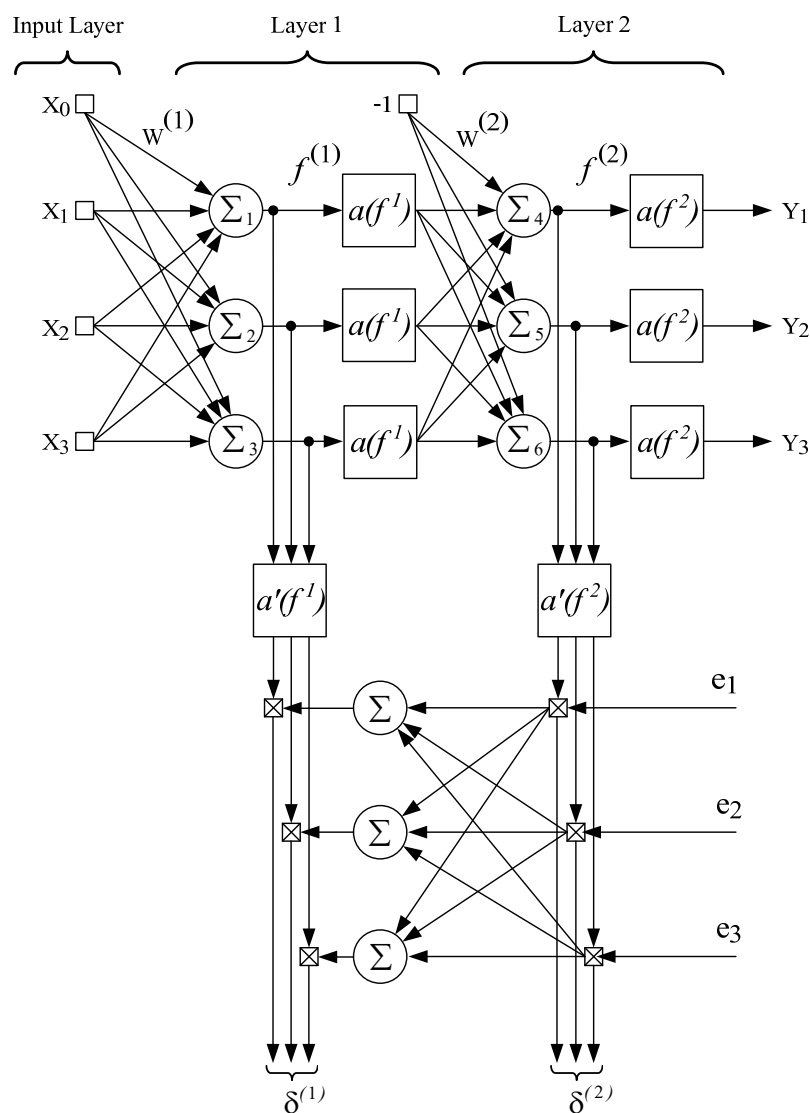


ภาพที่ 2-35 แสดงแผนภาพการเรียนรู้แบบเสริมสร้าง

2.5.4 วิธีการแพร่ย้อนกลับ (Back Propagation Algorithm)

วิธีการแพร่ย้อนกลับเป็นส่วนหนึ่งที่สำคัญของกระบวนการเรียนรู้แบบมีผู้ฝึกสอน เพื่อช่วยในการปรับค่าน้ำหนักเชื่อมโยงของระบบโครงข่ายประสาทเทียม ซึ่งถูกประยุกต์ใช้กับข่ายงาน

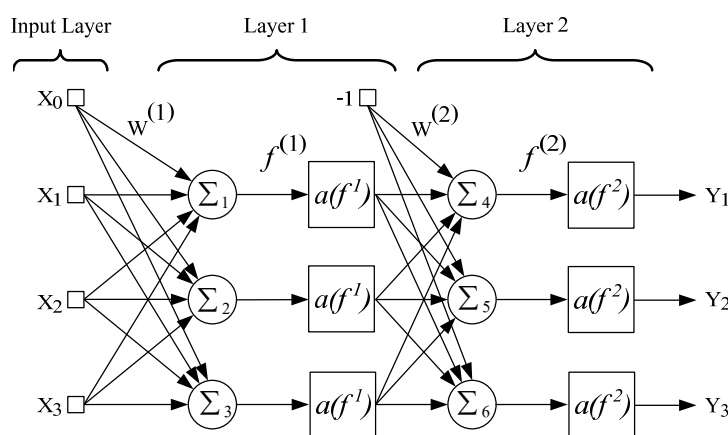
แบบป้อนไปข้างหน้าหลายชั้น พื้นฐานสำหรับการปรับค่าน้ำหนักเชื่อมโยงได้อาศัยวิธีการลดความชัน (Gradient Descent) วิธีการแพร่ย้อนกลับ แบ่งได้เป็น 2 ส่วน คือ การคำนวณไปข้างหน้า (Forward Computation) และการคำนวณย้อนหลัง (Backward Computation) ขั้นตอนวิธีการแพร่ย้อนกลับ แสดงดังภาพที่ 2-36



ภาพที่ 2-36 แสดงขั้นตอนวิธีการแพร่ย้อนกลับ

2.5.4.1 การคำนวณไปข้างหน้า (Forward Computation)

ขั้นตอนนี้จะเริ่มขึ้นเมื่อระบบโครงข่ายประสาทเทียมได้รับข้อมูลเข้าและคำนวณผลรวมของข้อมูลเข้าแล้วส่งผลลัพธ์ที่ได้เข้ามาประมวลผลโดยฟังก์ชันการกระตุ้น จากนั้นจะส่งผลการคำนวณต่อไปข้างหน้า ดังภาพที่ 2-37



ภาพที่ 2-37 แสดงกระบวนการคำนวณไปข้างหน้า

จากภาพที่ 2-37 การคำนวณผลรวมของระบบโครงข่ายประสาทเทียมในชั้นแรกสามารถหาได้จากสมการที่ 2-33

$$f_q = \sum_{j=0}^m w_{qj} x_j \quad (2-33)$$

โดย f_q คือ ฟังก์ชันของผลรวมในระบบโครงข่ายประสาทเทียมตัวที่ q

x_j คือ ข้อมูลเข้าจากชั้นข้อมูลเข้าตัวที่ j

w_{qj} คือ ค่าน้ำหนักเชื่อมโยงระหว่างระบบโครงข่ายประสาทเทียมตัวที่ q กับข้อมูลเข้าตัวที่ j

ฟังก์ชันของผลรวมในระบบโครงข่ายประสาทเทียมชั้นที่ 1 จะผ่านฟังก์ชันการกระตุ้น $a_i(f_q)$ สามารถหาได้จากสมการที่ 2-34

$$a(f_q) = a\left(\sum_{j=0}^m w_{qj} x_j\right) \quad (2-34)$$

โดย $a(f_q)$ คือ ฟังก์ชันการกระตุ้นของฟังก์ชันผลรวมของระบบโครงข่ายประสาทเทียมตัวที่ q

ผลรวมของระบบโครงข่ายประสาทเทียมในชั้นที่ 2 จะผ่านฟังก์ชันการกระตุ้นออกมาเป็นข้อมูลในชั้นข้อมูลออกสามารถหาได้จากสมการที่ 2-35

$$y_i = a\left(\sum_{i=1}^n w_{qi} a(f_q)\right) = a\left(\sum_{i=1}^n w_{qi} a\left(\sum_{j=0}^m w_{qj} x_j\right)\right) \quad (2-35)$$

โดย y_i คือ ข้อมูลออกของระบบโครงข่ายระบบประสาทเทียมตัวที่ i

w_{qi} คือ ค่าน้ำหนักเชื่อมโยงระหว่างระบบโครงข่ายประสาทเทียมตัวที่ q กับข้อมูลเข้าตัวที่ i

ค่าของสัญญาณผิดพลาดจะสามารถหาได้จากสมการที่ 2-36

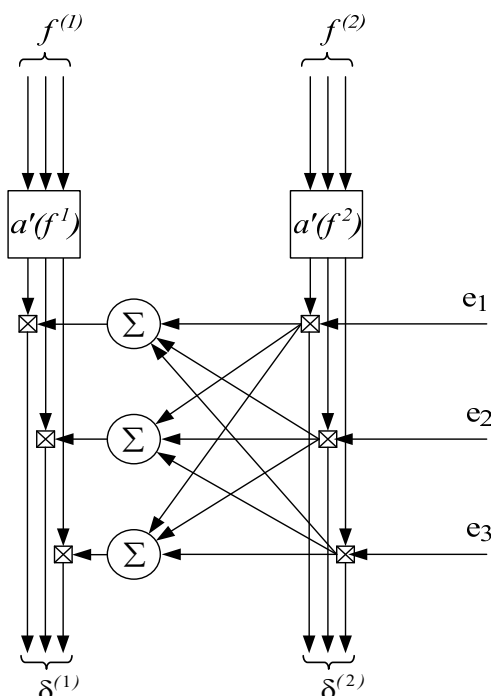
$$e_i = d_i - y_i \quad (2-36)$$

โดย e_i คือ ค่าของสัญญาณผิดพลาดตัวที่ i

d_i คือ ข้อมูลออกที่ i ของระบบโครงข่ายประสาทเทียมที่ต้องการ

2.5.4.2 การคำนวณย้อนหลัง (Backward Computation)

ขั้นตอนการคำนวณย้อนหลัง จะคำนวณผลต่างค่าน้ำหนักเชื่อมโยงที่จะเปลี่ยน โดยจะเริ่มเปลี่ยนค่าน้ำหนักเชื่อมโยงจากชั้นข้อมูลออกกลับไปยังชั้นข้อมูลเข้า กระบวนการนี้จะเกิดหลังจากจบกระบวนการคำนวณไปข้างหน้าแล้ว ภาพแสดงขั้นตอนการคำนวณย้อนหลัง ดังภาพที่ 2-38



ภาพที่ 2-38 แสดงกระบวนการคำนวณย้อนหลัง

การคำนวณค่าน้ำหนักเชื่อมโยงใหม่และค่าน้ำหนักเชื่อมโยงเปลี่ยนแปลง จากสมการที่ 2-37 และสมการที่ 2-38 ตามลำดับ

$$w^{\text{new}} = w^{\text{old}} + \Delta w \quad (2-37)$$

$$\Delta w = \eta \delta y \quad (2-38)$$

โดย Δw คือ ค่าน้ำหนักเชื่อมโยงเปลี่ยนแปลง

η คือ อัตราการเรียนรู้ (Learning Rate)

δ คือ ค่าความผิดพลาด

y คือ ข้อมูลออกระบบโครงข่ายประสาทเทียม

ค่าความผิดพลาดในชั้นข้อมูลออกสามารถคำนวณได้จากสมการที่ 2-39

$$\delta_q^{(L)} = e_q \cdot a'(f^{(L)}) \quad (2-39)$$

โดย $\delta_q^{(L)}$ คือ ค่าความผิดพลาดของชั้นข้อมูลออกที่ L
ของระบบโครงข่ายประสาทเทียมตัวที่ q

e_q คือ ค่าของสัญญาณผิดพลาดของระบบโครงข่ายประสาทเทียมตัวที่ q

$a'(f^{(L)})$ คือ อนุพันธ์ของฟังก์ชันการกระตุ้นชั้นที่ L

ค่าความผิดพลาดในชั้นซ่อนสามารถคำนวณได้จากสมการที่ 2-40

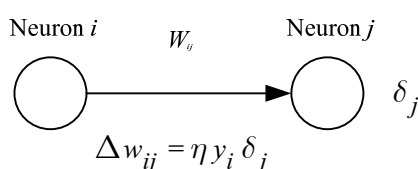
$$\delta_q^{(l)} = a'(f^{(l)}) \cdot \sum \delta^{(l+1)} w^{(l+1)} \quad (2-40)$$

โดย $\delta_q^{(l)}$ คือ ค่าความผิดพลาดของชั้นข้อมูลออกที่ l
ของระบบโครงข่ายประสาทเทียมตัวที่ q

$\delta^{(l+1)}$ คือ ค่าความผิดพลาดของชั้นข้อมูลออกที่ $l+1$ หรือชั้นก่อน

$w^{(l+1)}$ คือ ค่าน้ำหนักเชื่อมโยงชั้นที่ $l+1$

หากปรับค่าน้ำหนักเชื่อมโยงจากระบบโครงข่ายประสาทเทียมที่ i ไปยังระบบประสาทที่ j
แสดงดังภาพที่ 2-39



ภาพที่ 2-39 แสดงการเปลี่ยนแปลงค่าตัวถ่วงน้ำหนักเชื่อมโยง

จากภาพที่ 2-39 เปรียบเทียบกับสมการที่ 2-38 สามารถเขียนสมการในการเปลี่ยนแปลง
ค่าน้ำหนักเชื่อมโยงได้ดังสมการที่ 2-41

$$\Delta w_{ij} = \eta \delta_j y_i \quad (2-41)$$

การปรับค่าน้ำหนักเชื่อมโยงขึ้นกับตัวแปร 3 ตัวคือ อัตราการเรียนรู้ (η) สามารถกำหนดโดย
ผู้ใช้ค่าความผิดพลาด (δ) และข้อมูลออกจากระบบประสาท (y) นอกจากนั้นยังมีค่าคงตัวโมเมนตัม
(Momentum Constant: α) ดังนั้นสมการในการเปลี่ยนแปลงค่าน้ำหนักเชื่อมโยง สามารถเขียนได้
ดังสมการที่ 2-42

$$\Delta w_{ij}(k+1) = \eta \delta_j y_i + \alpha (\Delta w_{ij}(k)) \quad (2-42)$$

โดย α คือ ค่าคงตัวโมเมนตัม

$\Delta w_{ij}(k+1)$ คือ ค่าน้ำหนักเชื่อมโยงที่เปลี่ยนแปลงครั้งต่อไป

$\Delta w_{ij}(k)$ คือ ค่าน้ำหนักเชื่อมโยงเปลี่ยนแปลงครั้งปัจจุบัน

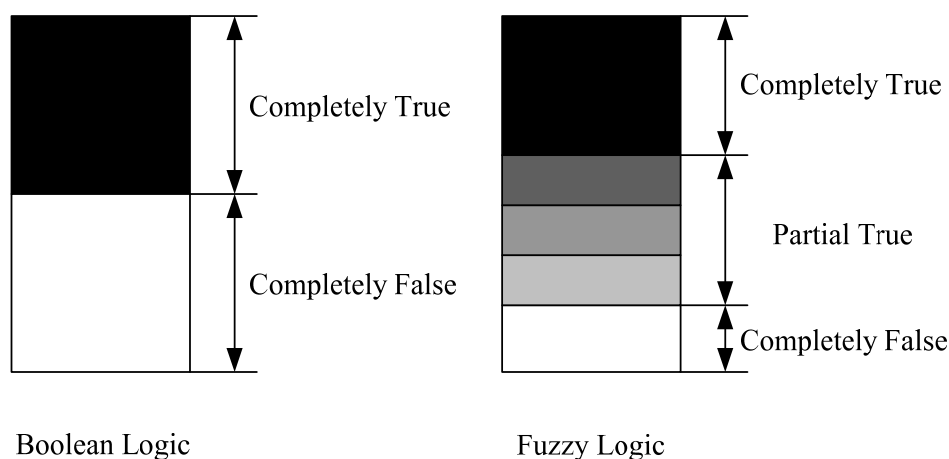
ในการคำนวณซ้ำด้วยวิธีการสอนแบบมีผู้ฝึกสอนจะทำเป็นจำนวน N รอบเวลา (Epoch) จนกระทั่งค่าน้ำหนักเชื่อมโยงมีการลู่เข้า โดยสามารถสังเกตได้จากการลู่เข้าของค่าเฉลี่ยของผลรวมค่าผิดพลาดกำลังสอง (Mean of Sum of Squared Error: MSE) ดังสมการที่ 2-43

$$MSE = \frac{\sum_{j=1}^N \sum_{i=1}^q (e_i(j))^2}{N} \quad (2-43)$$

โดย $e_i(j)$ คือ สัญญาณผิดพลาดจากระบบประสาทที่ i ในการทำซ้ำที่ j

2.6 ทฤษฎีฟัซซี่ (พยุ่ง, 2548)

ฟัซซี่ลอจิก (Fuzzy Logic) หรือตรรกศาสตร์แบบคลุมเครือ ได้ถูกคิดค้นโดย L.A. Zadeh ในปี 1965 โดยฟัซซี่ลอจิกเป็นศาสตร์การคำนวณแขนงใหม่ที่มีบทบาทในการวิจัยด้านคอมพิวเตอร์ และนำมาประยุกต์มาใช้ในการงานด้านต่างๆ เช่น การแพทย์ การทหาร ธุรกิจ และอุตสาหกรรม เป็นต้น ฟัซซี่ลอจิกเป็นเครื่องมือที่ช่วยในการตัดสินใจภายใต้ความไม่แน่นอนของข้อมูล โดยยอมให้มีความยืดหยุ่นได้ใช้หลักการเปลี่ยนแบบวิธีคิดที่ซับซ้อนของมนุษย์ ฟัซซี่ลอจิกมีลักษณะที่พิเศษกว่าตรรกะแบบจริงเท็จหรือบูลีนลอจิก (Boolean Logic) โดยมีแนวคิดที่ต่อขยายในส่วนของความจริง (Partial True) โดยค่าความจริงจะอยู่ระหว่างจริง (Completely True) กับเท็จ (Completely False) แสดงการเปรียบเทียบระหว่างบูลีนลอจิกกับฟัซซี่ลอจิก ดังภาพที่ 2-40



ภาพที่ 2-40 แสดงการเปรียบเทียบระหว่างบูลีนลอจิกกับฟัซซี่ลอจิก

2.6.1 ฟัซซี่เซต (Fuzzy Set)

ฟัซซี่เซต เป็นเซตที่อธิบายสิ่งที่มีความคลุมเครือและไม่สามารถกำหนดขอบเขตอย่างชัดเจน โดยสมาชิกภายในเซตจะมีอัตราความเป็นสมาชิกเซตอยู่ในช่วงระหว่าง 0 ถึง 1 โดยฟัซซี่เซตจะมีขอบเขตแบบฟัซซี่ไม่ใช่เปลี่ยนแปลงทันทีทันใดจากขาวเป็นดำ

2.6.2 ความสัมพันธ์ของฟัซซี่ (Fuzzy Relation)

ความสัมพันธ์ของฟัซซี่ เป็นความสัมพันธ์ระหว่างฟังก์ชันความเป็นสมาชิก 2 คู่ขึ้นไปโดยมีอัตราความเป็นสมาชิก (Degree of Membership) อยู่ในช่วง 0 ถึง 1 โดยกำหนดค่านิยามดังนี้

ความสัมพันธ์ของฟัซซี่ ถ้า U หรือ V เป็นเอกภพสัมพัทธ์ต่อเนื่องนับไม่ได้ (Uncountable Continuous Universes) และ $\mu_R : U \times V \rightarrow [0,1]$ จะได้ว่า

$$R = \int_{U \times V} \mu_R(u, v) / (u, v) \quad (2-44)$$

โดย (u, v) คือ คู่อันดับใด ๆ ในเซตของผลคูณคาร์ทีเซียน $U \times V$ โดยที่ $u \in U$ และ $v \in V$

R คือ ความสัมพันธ์ของฟัซซี่

สัญลักษณ์ อินทิกรัล แทนเซตของ $\mu_R(u, v) / (u, v)$ ทั้งหมดใน $U \times V$ และถ้า U และ V เป็นเอกภพสัมพัทธ์เชิงจำนวนนับได้ (Countable Discrete Universe)

$$R = \sum_{U \times V} \mu_R(u, v) / (u, v) \quad (2-45)$$

โดย $\mu_R(u, v) = \mu_{U \times V}(u, v) = \min(\mu_U(u), \mu_V(v))$

ยกตัวอย่างเช่น ถ้าความเป็นสมาชิกสัมพันธ์ กำหนดเป็น

$$\mu_R(x, y) = \begin{cases} 1 & x = y \\ 0.8 & \text{เมื่อ } x - y = 1 \\ 0.3 & x - y = 2 \end{cases}$$

สามารถเขียนค่าความสัมพันธ์ฟัซซี่เป็นเมตริกซ์ได้ดังนี้

		y		
		1	2	3
R: x	1	1	0.8	0.3
	2	0.8	1	0.8
	3	0.3	0.8	1

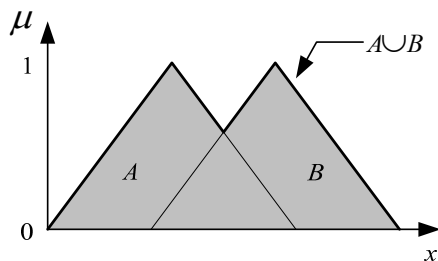
2.6.3 การดำเนินการของฟัซซี่เซต

การดำเนินการของฟัซซี่เซตมีคุณสมบัติเช่นเดียวกับเซตโดยทั่วไปมีการดำเนินการคือ ยูเนียน อินเตอร์เซกชัน และคอมพลิเมนต์

2.6.3.1 ยูเนียน (Union) ของฟัซซี่เซตจะเป็น OR Operation ดังสมการที่ 2-46

$$\mu_{A \cup B}(x) = \mu_A(x) \vee \mu_B(x) \quad (2-46)$$

ภาพของการยูเนียนของฟัซซี่เซต A และ B แสดงดังภาพที่ 2-41

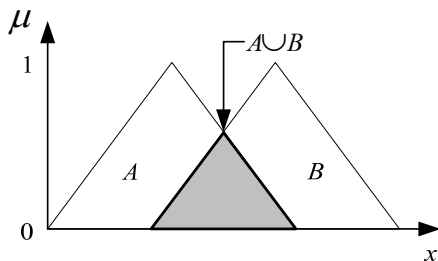


ภาพที่ 2-41 แสดงภาพของการยูเนียนของฟัซซี่เซต A และ B

2.6.3.2 อินเตอร์เซกชัน (Intersection) ของฟัซซี่เซตจะเป็น AND Operation ดังสมการที่ 2-47

$$\mu_{A \cap B}(x) = \mu_A(x) \wedge \mu_B(x) \quad (2-47)$$

ภาพของการยูเนียนของฟัซซี่เซต A และ B แสดงดังภาพที่ 2-42

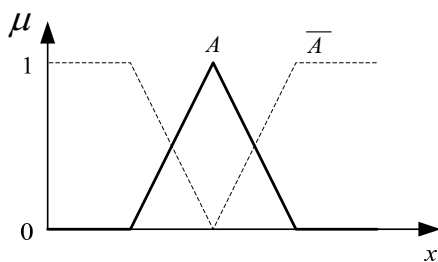


ภาพที่ 2-42 แสดงภาพของการอินเตอร์เซกชันของฟัซซี่เซต A และ B

2.6.3.3 คอมพลิเมนต์ (Complement) ของฟัซซี่เซต ในสมการที่ 2-48

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (2-48)$$

ภาพของการคอมพลิเมนต์ของฟัซซี่เซต A แสดงดังภาพที่ 2-43



ภาพที่ 2-43 แสดงภาพของการคอมพลิเมนต์ของฟัซซี่เซต A

2.6.4 ฟังก์ชันความเป็นสมาชิก (Membership Function)

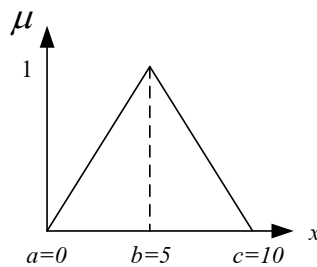
เป็นฟังก์ชันที่มีการกำหนดระดับความเป็นสมาชิกของตัวแปรที่ต้องการใช้งาน โดยเริ่มจากการแทนที่กับตัวแทนที่มีความคลุมเครือ ดังนั้นส่วนสำคัญต่อการดำเนินการของพีชพีเพราะรูปร่างของฟังก์ชันความเป็นสมาชิกมีความสำคัญต่อกระบวนการคิดและแก้ไขปัญหา โดยฟังก์ชันความเป็นสมาชิกจะไม่สมมาตรกันหรือสมมาตรกันทุกประการก็ได้ ชนิดของฟังก์ชันความเป็นสมาชิกที่ใช้งานทั่วไปมีหลายชนิด แต่ในที่นี้จะกล่าวเพียง 6 ชนิดดังนี้

2.6.4.1 ฟังก์ชันสามเหลี่ยม (Triangular Membership Function)

ฟังก์ชันสามเหลี่ยมมีพารามิเตอร์ทั้งหมด 3 ค่า คือ $\{a, b, c\}$ ดังสมการที่ 2-49

$$\text{triangular}(x : a, b, c) = \begin{cases} 0 & , \quad x < a \\ (x - a) / (b - a) & , \quad a \leq x \leq b \\ (c - x) / (c - b) & , \quad b \leq x \leq c \\ 0 & , \quad x > c \end{cases} \quad (2-49)$$

และกราฟของฟังก์ชันสามเหลี่ยมแสดงดังภาพที่ 2-44



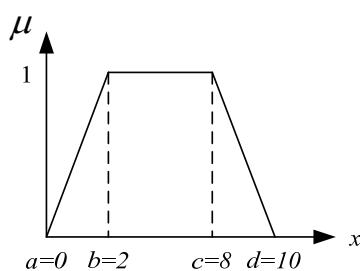
ภาพที่ 2-44 แสดงกราฟของฟังก์ชันสามเหลี่ยม

2.6.4.2 ฟังก์ชันสี่เหลี่ยมคางหมู (Trapezoidal Membership Function)

ฟังก์ชันสี่เหลี่ยมคางหมูมีพารามิเตอร์ทั้งหมด 4 ค่าคือ $\{a, b, c, d\}$ ดังสมการที่ 2-50

$$\text{trapezoidal}(x : a, b, c, d) = \begin{cases} 0 & , \quad x < a \\ (x - a) / (b - a) & , \quad a \leq x < b \\ 1 & , \quad b \leq x < c \\ (d - x) / (d - c) & , \quad c \leq x < d \\ 0 & , \quad x \geq d \end{cases} \quad (2-50)$$

และกราฟของฟังก์ชันสี่เหลี่ยมคางหมูแสดงดังภาพที่ 2-45



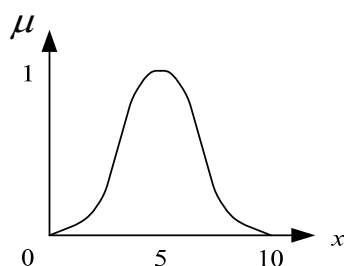
ภาพที่ 2-45 แสดงกราฟของฟังก์ชันสี่เหลี่ยมคางหมู

2.6.4.3 ฟังก์ชันเกาส์เซียน (Gaussian Membership Function)

ฟังก์ชันเกาส์เซียนมีพารามิเตอร์ 2 ค่า คือ $\{m, \sigma\}$ ซึ่ง m หมายถึงค่าเฉลี่ย และ σ หมายถึงค่าเบี่ยงเบนมาตรฐาน ดังสมการที่ 2-51

$$gaussian(x : m, \sigma) = \left(-\frac{(x-m)^2}{\sigma^2} \right) \quad (2-51)$$

และกราฟของฟังก์ชันเกาส์เซียนแสดงดังภาพที่ 2-46



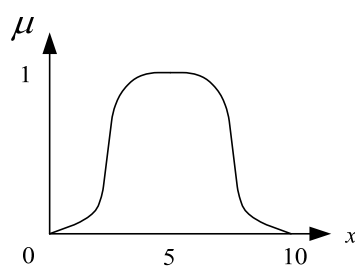
ภาพที่ 2-46 แสดงกราฟของฟังก์ชันเกาส์เซียน

2.6.4.4 ฟังก์ชันระฆังคว่ำ (Bell Shaped Membership Function)

ฟังก์ชันรูประฆังคว่ำมีพารามิเตอร์ทั้งหมด 3 ค่า คือ $\{a, b, c\}$ ดังสมการที่ 2-52

$$bell\ shaped(x : a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (2-52)$$

และกราฟของฟังก์ชันระฆังคว่ำแสดงดังภาพที่ 2-47



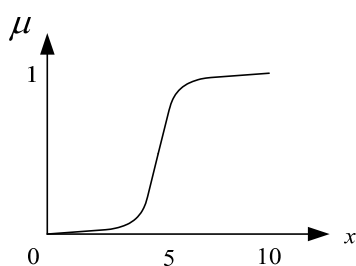
ภาพที่ 2-47 แสดงกราฟของฟังก์ชันระฆังคว่ำ

2.6.4.5 ฟังก์ชันตัวเอส (Smooth Membership Function)

ฟังก์ชันรูปตัวเอสมีพารามิเตอร์ทั้งหมด 2 ค่า คือ $\{a, b\}$ ดังสมการที่ 2-53

$$S(x : a, b) = \begin{cases} 0 & , \quad x < a \\ 2\left(\frac{x-b}{b-a}\right)^2 & , \quad a \leq x \leq \frac{a+b}{2} \\ 1 - 2\left(\frac{x-b}{b-a}\right)^2 & , \quad \frac{a+b}{2} \leq x < b \\ 1 & , \quad x \geq b \end{cases} \quad (2-53)$$

และกราฟของฟังก์ชันตัวเอสแสดงดังภาพที่ 2-48



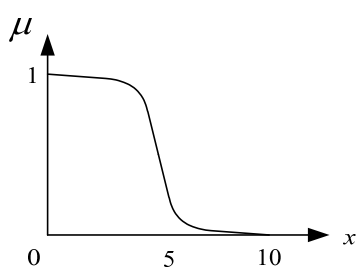
ภาพที่ 2-48 แสดงกราฟของฟังก์ชันตัวเอส

2.6.4.6 ฟังก์ชันตัวแซด (Z-Membership Function)

ฟังก์ชันรูปตัวแซดมีพารามิเตอร์ทั้งหมด 2 ค่า คือ $\{a, b\}$ ดังสมการที่ 2-54

$$Z(x : a, b) = \begin{cases} 0 & , \quad x < a \\ 1 - 2\left(\frac{x-b}{b-a}\right)^2 & , \quad a \leq x \leq \frac{a+b}{2} \\ 2\left(\frac{x-b}{b-a}\right)^2 & , \quad \frac{a+b}{2} \leq x < b \\ 1 & , \quad x \geq b \end{cases} \quad (2-54)$$

และกราฟของฟังก์ชันตัวแซดแสดงดังภาพที่ 2-49

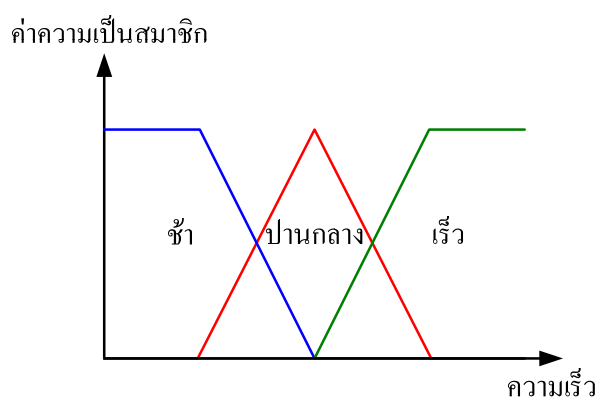


ภาพที่ 2-49 แสดงกราฟของฟังก์ชันตัวแซด

การเลือกฟังก์ชันของความเป็นสมาชิก จะต้องเลือกตามความเหมาะสมความครอบคลุมของข้อมูลที่จะรับเข้ามาโดยสามารถที่ทับซ้อนกันเพื่อให้การดำเนินงานราบเรียบซึ่งมีความเป็นสมาชิกหลายค่าได้และฟังก์ชันความเป็นสมาชิกเปลี่ยนแปลงแก้ไขให้เหมาะกับงานที่กำลังปฏิบัติงานหรือตามความต้องการ

2.6.5 ตัวแปรภาษา (Linguistic Variable)

ตัวแปรภาษา เป็นตัวแปรที่กำหนดย่านหรือปริมาณทางฟิสิกส์ ที่กำหนดแทนภาษามนุษย์ เช่น อากาศร้อน อุณหภูมิสูง รถมอเตอร์เร็วสูงหรือความดันระดับปานกลาง เป็นต้น โดยปริมาณเหล่านี้จะใช้เป็นตัวกำหนดชื่อของฟังก์ชันความเป็นสมาชิกของฟัซซี่



ภาพที่ 2-50 แสดงตัวแปรภาษากำหนดย่านความเร็ว

จากภาพที่ 2-50 ประกอบด้วยฟังก์ชันความเป็นสมาชิกฟัซซี่ของความเร็ว 3 ฟังก์ชัน คือ ช้า ปานกลาง และเร็ว โดยการกำหนดคุณสมบัติของตัวแปรตัวแปรภาษาใช้เป็น (X, T, U, M)

X คือ ปริมาณ หรือตัวแปรทางฟิสิกส์ที่ใช้แทน เช่น ความเร็ว เป็นต้น

T คือ เซตของตัวแปรทั้งหมดของตัวแปร X เช่น ช้า ปานกลาง และเร็ว

U คือ เอกภพสัมพัทธ์ของ X

M คือ ความสัมพันธ์ของ T แต่ละค่า

ค่าตัวแปรภาษานี้เป็นสิ่งสำคัญ และการจำแนกปริมาณทางฟิสิกส์ ด้วยความรู้สึกรับรู้ของมนุษย์ หรือประสบการณ์จากผู้เชี่ยวชาญเพื่อนำไปประเมินผลจากเหตุการณ์ต่างๆที่เกิดขึ้นหรือเกิดในอดีต

2.6.6 กฎของฟัซซี่ (Fuzzy Rule)

กฎของฟัซซี่เป็นการสรุปจากเหตุและผลตามฐานความคิดของมนุษย์ กฎพื้นฐานที่นิยมในการประยุกต์ใช้งานมากที่สุดได้แก่ กฎของฟัซซี่แบบถ้า-แล้ว (Fuzzy if-then Rule) สามารถแสดงในรูปแบบประโยค

ถ้า ข้อตั้ง (ข้อนำ) ดังนั้น ข้อยุติ (ข้อตาม)

If premise (Antecedent) Then conclusion (Consequent)

หากทราบความจริง (ข้อตั้ง ข้อสมมุติฐาน หรือข้อนำ) แล้วสามารถหาข้อสรุปความจริง
 อีกอย่างหนึ่งที่เรียกว่าข้อยุติ ซึ่งเป็นการแสดงประสบการณ์ของมนุษย์และองค์ความรู้ในรูปแบบ
 ประโยคภาษามนุษย์ที่ใช้สื่อสารทั่วไป กฎของฟัซซี่เป็นสิ่งที่มีความจำเป็นในการจัดรูปแบบของ
 ระบบที่ซับซ้อนซึ่งสามารถสังเกตได้โดยมนุษย์ เพราะระบบเหล่านี้สามารถแสดงด้วยตัวแปรภาษา
 สามารถแสดงได้ด้วยฟัซซี่เซตและตัวเชื่อมตรรกะของเซต ยกตัวอย่างการสร้างของการจัดกลุ่ม
 ข้อมูลของสมาชิกจากค่า x_1 และ x_2

กฎข้อ 1: ถ้า x_1 มีค่า low และ x_2 มีค่า low แล้ว ข้อมูล (x_1, x_2) เป็นกลุ่ม C_1

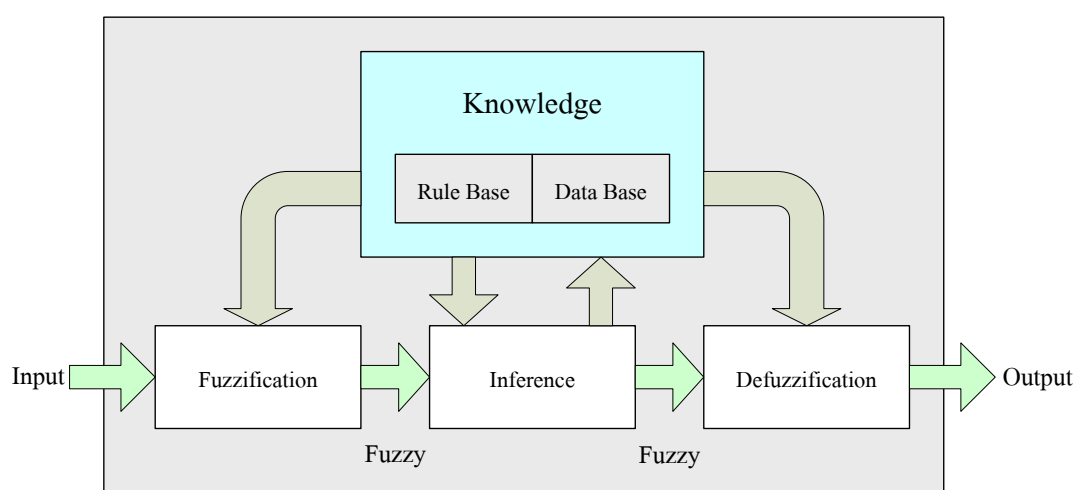
กฎข้อ 2: ถ้า x_1 มีค่า low และ x_2 มีค่า high แล้ว ข้อมูล (x_1, x_2) เป็นกลุ่ม C_2

กฎข้อ 3: ถ้า x_1 มีค่า high และ x_2 มีค่า low แล้ว ข้อมูล (x_1, x_2) เป็นกลุ่ม C_3

กฎข้อ 4: ถ้า x_1 มีค่า high และ x_2 มีค่า high แล้ว ข้อมูล (x_1, x_2) เป็นกลุ่ม C_4 เป็นต้น

2.6.7 โครงสร้างพื้นฐานของการประมวลผลระบบฟัซซี่

โครงสร้างพื้นฐานของการประมวลผลระบบฟัซซี่ มีองค์ประกอบ 4 ส่วนดังนี้



ภาพที่ 2-51 แสดงโครงสร้างพื้นฐานของการประมวลผลระบบฟัซซี่

ส่วนการแปลงอินพุตทั่วไปเป็นอินพุตแบบฟัซซี่ (Fuzzification) ในรูปแบบฟัซซี่เซตหรือ
 ตัวแปรภาษา (Linguistic Variable)

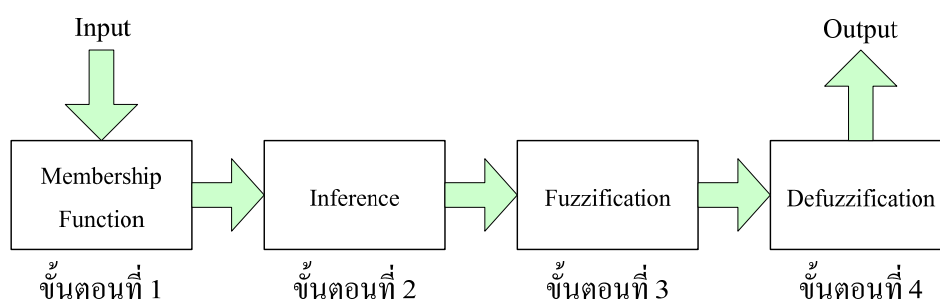
ฐานความรู้ (Knowledge Base) เป็นส่วนที่จัดเก็บรวบรวมข้อมูลในการควบคุมประกอบด้วย
 2 ส่วนคือ ฐานกฎ (Rule Base) ส่วนของการกำหนดวิธีการควบคุม ซึ่งได้จากผู้เชี่ยวชาญในรูปแบบ
 ของชุดข้อมูลแบบกฎของภาษา (Linguistic Rule) และฐานข้อมูล (Database) เป็นการเตรียมส่วน
 ที่จำเป็นเพื่อใช้ในการกำหนดกฎการควบคุม และการจัดการข้อมูลของตรรกศาสตร์ฟัซซี่

การตีความ (Inference Engine) เป็นส่วนที่ทำหน้าที่ตรวจสอบข้อเท็จจริงและกฎ เพื่อใช้ในการตีความหาเหตุผล เหมือนกลไกสำหรับควบคุมการใช้ความรู้ในการแก้ไขปัญหา รวมทั้งการกำหนดวิธีการของการตีความเพื่อหาคำตอบ

การแปลงเอาต์พุตแบบฟัซซี่เป็นเอาต์พุตทั่วไป (Defuzzification) เป็นส่วนการทำการแปลงข้อมูลที่อยู่ในรูปแบบฟัซซี่ให้เป็นค่าที่สรุปผลหรือค่าการควบคุมระบบ

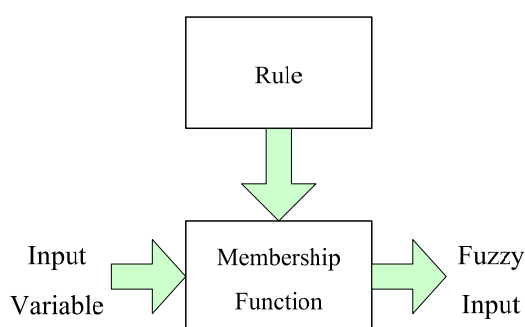
2.6.7.1 ขั้นตอนการประมวลผลของระบบฟัซซี่

ขั้นตอนการประมวลผลของระบบฟัซซี่มีรูปแบบการทำงานแบ่งได้ 4 ส่วน ดังภาพที่ 2-52



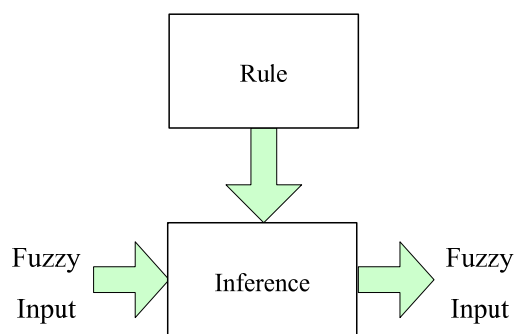
ภาพที่ 2-52 แสดงขั้นตอนการประมวลผลของระบบฟัซซี่

ก) ขั้นตอนที่ 1 เป็นการแปลงการอินพุตแบบตัวแปรฟัซซี่ โดยจะสร้างฟังก์ชันความเป็นสมาชิก โดยฟังก์ชันจะมีลักษณะเป็นการกำหนดภาษาสามัญ เพื่อให้เป็นอินพุตแบบตัวแปรฟัซซี่ แสดงดังภาพที่ 2-53



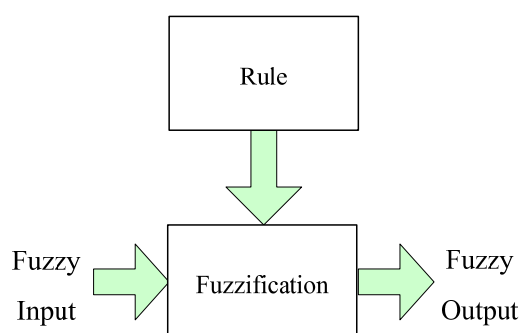
ภาพที่ 2-53 แสดงขั้นตอนที่ 1 ของการประมวลผลของระบบฟัซซี่

ข) ขั้นตอนที่ 2 เป็นการสร้างความสัมพันธ์ระหว่างอินพุตกับเอาต์พุตที่อาศัยหลักการของการหาเหตุผล อาจสร้างการเก็บข้อมูล การคาดการณ์จากการตัดสินใจ โดยเขียนเป็นกฎการควบคุมระบบซึ่งมีลักษณะอยู่ในรูปแบบ ถ้า (If) และ (And) หรือ (Or) ซึ่งเป็นภาษาสามัญ นำกฎทั้งหมดมาประมวลผล เพื่อการหาตัดสินใจที่เหมาะสม แสดงดังภาพที่ 2-54



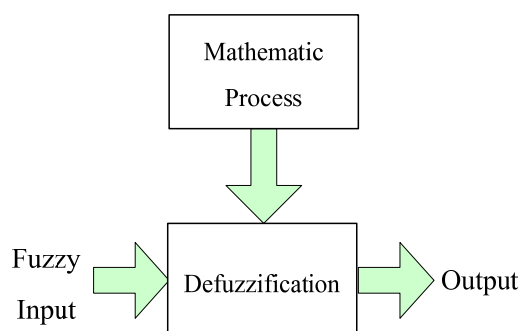
ภาพที่ 2-54 แสดงขั้นตอนที่ 2 ของการประมวลผลของระบบฟัซซี่

ค) ขั้นตอนที่ 3 เป็นการหาฟัซซี่เอาต์พุต โดยการนำกฎการควบคุมที่สร้างขึ้นในขั้นตอนที่ 2 มาประมวลผลกับฟัซซี่อินพุต โดยใช้วิธีการทางคณิตศาสตร์ เพื่อนำค่าที่ได้มาประมวลผล ดังภาพที่ 2-55 วิธีการทำเป็นค่าคลุมเครือ (Fuzzification) วิธีการที่นิยมใช้ในการหาค่าผลลัพธ์เลือกใช้ Max-Min Method และ Max-Dot Method



ภาพที่ 2-55 แสดงขั้นตอนที่ 3 ของการประมวลผลของระบบฟัซซี่

ง) ขั้นตอนที่ 4 ขั้นตอนการสรุปเหตุผลฟัซซี่ โดยจะเปลี่ยนฟัซซี่เอาต์พุตด้วยวิธีการทางคณิตศาสตร์ เช่น วิธีการหาจุดศูนย์กลาง (Central of Gravity) เพื่อนำค่าที่ได้มาใช้ในการตัดสินใจเพื่อควบคุมระบบ



ภาพที่ 2-56 แสดงขั้นตอนที่ 4 ของการประมวลผลของระบบฟัซซี่

วิธีการหาค่าฟัซซี่ให้เป็นค่าปกติ (Defuzzification) วิธีการที่เป็นเทคนิคการเลือกค่าสูงสุดหรือสรุปหาเหตุผลจากหลาย ๆ เซตมาเพียงค่าเดียว

วิธีการหาจุดศูนย์กลางถ่วง (Central of Gravity: COG) เป็นวิธีการเฉลี่ยผลที่ได้จากการตีความหาเหตุที่นิยมใช้ในปัจจุบัน จะหาได้จากการประมาณค่าจากสมการที่ 2-55

$$COG = \frac{\sum_{i=1}^N \alpha_i W_i}{\sum_{i=1}^N \alpha_i} \quad (2-55)$$

โดย COG คือ ค่าของจุดศูนย์กลางถ่วง (Central of Gravity)

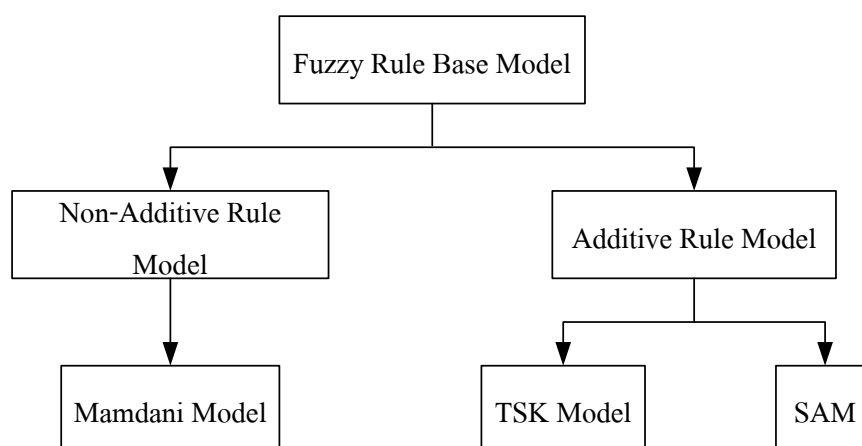
α_i คือ ค่าฟัซซี่ของเอาต์พุตในเซตฟัซซี่ตำแหน่งที่ i

W_i คือ พื้นที่ใต้โค้งของฟัซซี่เซตตำแหน่งที่ i

N คือ จำนวนของฟัซซี่เซต

2.6.8 ชนิดของระบบกฎฟัซซี่

ระบบกฎฟัซซี่ที่ใช้ในการประมาณฟังก์ชันมี 3 รูปแบบได้แก่ แบบแมนดานิ (Mamdani) แบบ Takagi-Sugeno-Kang (TSK) และแบบบวกมาตรฐาน (Standard Additive Model :SAM) โดยแบบแมนดานิจะรวมผลการอนุมาน (Inference) ของกฎด้วยวิธีการซ้อนทับ (Superimposition) จากกฎหลายๆข้อโดยไม่เป็นแบบบวกกันจึงเรียกระบบแบบนี้ว่าเป็น Non-Additive Rule Model แต่สำหรับแบบ TSK และแบบ SAM มีการอนุนามแบบรวมค่าน้ำหนัก (Weighted Sum) จากหลายๆกฎ เพื่อรวมเป็นข้อสรุปสุดท้ายจึงเรียกระบบแบบนี้ว่า Additive Rule Model ในวิทยานิพนธ์นี้จะกล่าวเฉพาะกฎฟัซซี่แบบแมนดานิ ส่วนการจัดกลุ่มของระบบกฎแบบฟัซซี่แสดงดังภาพที่ 2-57



ภาพที่ 2-57 แสดงระบบกฎฟัซซี่

2.6.8.1 ระบบกฎฟัซซีของแมมดานิ

ระบบกฎฟัซซีแบบแมมดานิ เป็นระบบใช้มากที่สุดในทางปฏิบัติเป็นระบบที่ใช้ตัวแปรภาษาทั้งในข้อตั้งและข้อตามเพื่อจัดเทียบฟังก์ชัน

กฎที่ 1: If (x_1 is A_{11}) AND (x_2 is A_{12}) AND AND (x_n is A_{1n}) THEN y is C_1

กฎที่ 2: If (x_1 is A_{21}) AND (x_2 is A_{22}) AND AND (x_n is A_{2n}) THEN y is C_2

⋮

⋮

กฎที่ L: If (x_1 is A_{L1}) AND (x_2 is A_{L2}) AND AND (x_n is A_{Ln}) THEN y is C_L

เมื่อ $x_j, j = 1, \dots, n$, เป็นตัวประกอบที่ j ของตัวแปรอินพุต x , y เป็นตัวแปรเอาต์พุต, A_{ij} เป็นพจน์ภาษาของข้อตั้ง (Consequence Linguistic Term) หรือเป็นฟังก์ชันความเป็นสมาชิกของข้อตั้ง (Antecedent Membership Function) ในกฎที่ $i, i = 1, \dots, L$, C_i เป็นพจน์ภาษาของข้อตามหรือฟังก์ชันความเป็นสมาชิกของข้อตาม (Consequent Membership Function) ของกฎที่ i

กำหนดให้ A เป็นเซตใหม่ สำหรับกฎข้อที่ i เมื่อ $i = 1, 2, \dots, L$

$$A = A_{i1} \cap A_{i2} \cap \dots \cap A_{in}$$

แสดงในรูปฟังก์ชันความเป็นสมาชิกได้เป็น

$$\mu_A(x) = \min(\mu_{A_{i1}}(x_1), \mu_{A_{i2}}(x_2), \dots, \mu_{A_{in}}(x_n))$$

หากมีอินพุตเข้ามาในรูป

$$(x_1 = x'_1), (x_2 = x'_2), \dots, (x_n = x'_n)$$

โดยที่ $x'_1, x'_2, \dots, x'_n$ เป็นค่าอินพุตใดๆ จะได้ค่าฟัซซี ในส่วนของข้อตั้งเป็น

$$\alpha_i = \min(\mu_{A_{i1}}(x'_1), \mu_{A_{i2}}(x'_2), \dots, \mu_{A_{in}}(x'_n))$$

ค่าเอาต์พุตของกฎแต่ละข้อของระบบฟัซซีแบบแมมดานิ ที่เป็นค่าฟัซซีสามารถหาได้จาก

$$\mu_{C_i}(y) = \alpha_i \vee \mu_{C_i}(y)$$

ค่าเอาต์พุตของระบบเป็นผลรวมจากเอาต์พุตจากกฎแต่ละข้อโดยใช้สมการ

$$\mu_C(y) = \max(\mu_{C_1}(y), \mu_{C_2}(y), \dots, \mu_{C_L}(y))$$

ฟัซซีเอาต์พุต สามารถแปลงเป็นค่าปกติได้ด้วยวิธีเฉลี่ยน้ำหนัก

$$y^* = \frac{\sum \mu_C(\bar{y}) \times \bar{y}}{\sum \mu_C(\bar{y})}$$

เมื่อ $\bar{y}$ เป็นค่า Centriod ของฟังก์ชันความเป็นสมาชิกที่สมมาตร

2.6.8.2 วิธีการอนุมานแบบแมมดานิ

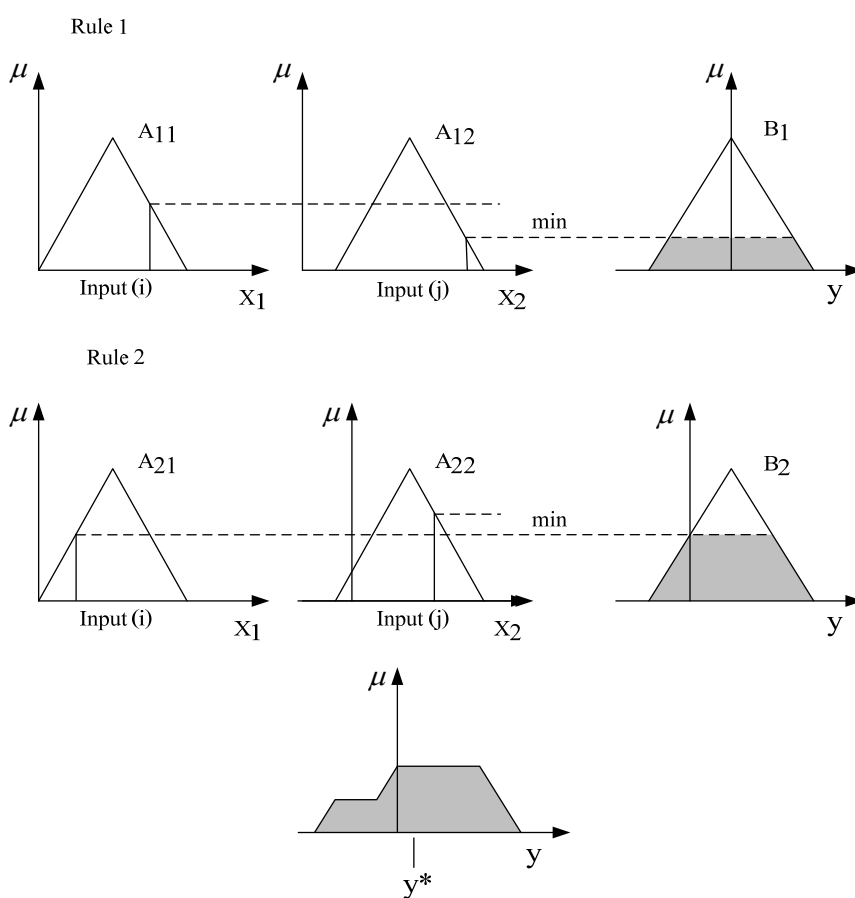
หากให้ระบบฟuzzyแบบแมมดานิ มี 2 อินพุต x_1, x_2 และมี 1 เอาต์พุต y ซึ่งมีกฎฟuzzyเป็น

If (x_1 is A_{k1}) AND (x_2 is A_{k2}) THEN y is B_k สำหรับ $k = 1, 2, \dots, r$

ผลรวมเอาต์พุตหาได้ด้วยวิธีการจัดองค์ประกอบแบบค่าสูงสุด-ต่ำสุด (Max-Min Composition) และวิธีการจัดองค์ประกอบแบบค่าสูงสุดผลคูณ (Max-Product Composition)

ก) วิธีการจัดองค์ประกอบแบบค่าสูงสุด-ต่ำสุด (Max-Min Composition)

$$\mu_{B_k}(y) = \max \left[\min \left(\mu_{A_{k1}}(\text{Input}(i)), \mu_{A_{k2}}(\text{Input}(j)) \right) \right] \text{ สำหรับ } k = 1, 2, \dots, r$$

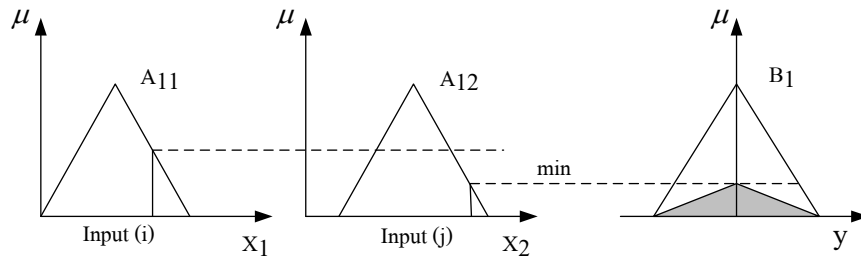


ภาพที่ 2-58 แสดงกราฟฟิกของวิธีการจัดองค์ประกอบแบบค่าสูงสุด-ต่ำสุด

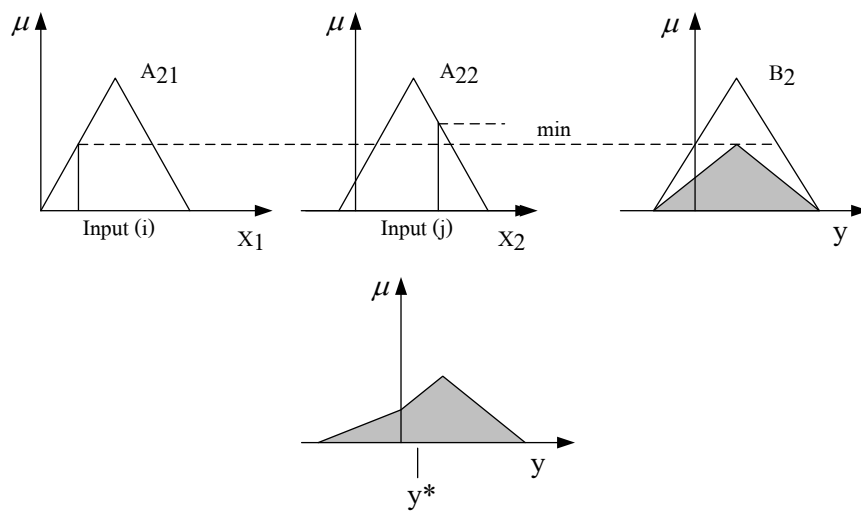
ข) วิธีการจัดองค์ประกอบแบบค่าสูงสุดผลคูณ (Max-Product Composition)

$$\mu_{B_k}(y) = \max(\mu_{A_{k1}}(\text{Input}(i)) \cdot \mu_{A_{k2}}(\text{Input}(j))) \quad \text{สำหรับ } k = 1, 2, \dots, r$$

Rule 1



Rule 2



ภาพที่ 2-59 แสดงกราฟฟิคของวิธีการจัดองค์ประกอบแบบค่าสูงสุดผลคูณ

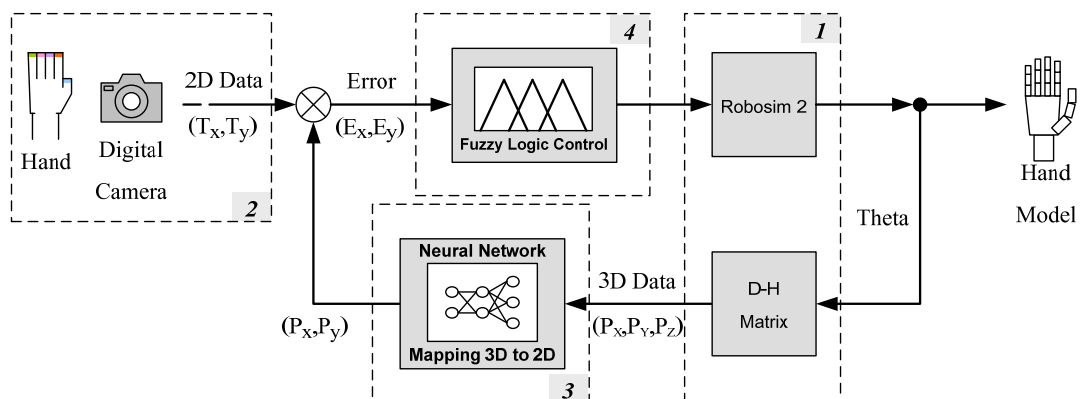
บทที่ 3

การวิเคราะห์และการออกแบบ

การวิเคราะห์และการออกแบบ ในวิทยานิพนธ์นี้ผู้วิจัยได้วิเคราะห์ลักษณะทางกายภาพของมือมนุษย์ ออกแบบแบบจำลองมือกลด้วยโปรแกรมจำลองหุ่นยนต์ในระบบ 3 มิติ และควบคุมแบบจำลอง โดยผู้วิจัยได้จัดแบ่งเนื้อหาในการวิเคราะห์ และออกแบบ เป็น 4 ส่วนดังต่อไปนี้

- 3.1 การวิเคราะห์โครงสร้างมือมนุษย์ และออกแบบแบบจำลองมือ
- 3.2 การออกแบบระบบประมวลผลภาพ
- 3.3 การออกแบบระบบหาดำแหน่ง 2 มิติ จากตำแหน่ง 3 มิติ
- 3.4 การวิเคราะห์การเคลื่อนไหวและออกแบบระบบควบคุม

ส่วนแรกเป็นการวิเคราะห์โครงสร้างมือมนุษย์และออกแบบแบบจำลองมือ โดยส่วนนี้จะเป็นการวิเคราะห์โครงสร้างมือ ออกแบบแบบจำลองมือและแสดงผลการติดตามท่าทางด้วยโปรแกรมจำลองหุ่นยนต์ 3 มิติ ส่วนที่สองเป็นการออกแบบระบบประมวลผลภาพ เพื่อหาดำแหน่งของปลายนิ้วแบบ 2 มิติ จากภาพถ่าย ส่วนที่สามเป็นการออกแบบระบบโครงข่ายประสาทเทียมเพื่อใช้ในการแปลงตำแหน่ง 3 มิติ จากแบบจำลองเป็นตำแหน่ง 2 มิติ โดยตำแหน่ง 2 มิติ ที่ได้จะนำไปเปรียบเทียบกับตำแหน่งจริงจากระบบประมวลผลภาพต่อไป ส่วนสุดท้ายเป็นการวิเคราะห์และออกแบบระบบควบคุมด้วยระบบฟัซซี่ จากทั้งหมดสามารถแสดงเป็นไดอะแกรมดังภาพที่ 3-1



ภาพที่ 3-1 แสดงไดอะแกรมของงานวิจัย

3.1 การวิเคราะห์โครงสร้างมือนมนุษย์ และออกแบบแบบจำลองมือ (Lien C. C., 1998)

การวิเคราะห์โครงสร้างมือนมนุษย์และออกแบบแบบจำลองมือ สามารถแบ่งเนื้อหาออกได้เป็น 2 ส่วนด้วยกันคือ การวิเคราะห์โครงสร้างมือนมนุษย์ โดยอาศัยหลักจลนศาสตร์ในการหาเมตริกซ์การแปลง (Transformation Matrix) และพารามิเตอร์ของ D-H (Denavit-Hartenberg Parameters) เพื่อหาดำแหน่งของปลายนิ้วของแบบจำลองมือ การออกแบบแบบจำลองมืออาศัยข้อมูลจากการวิเคราะห์โครงสร้างมือ เพื่อสร้างแบบจำลองมือแสดงผลด้วยโปรแกรมโรโบซิม 2 (Robosim 2)

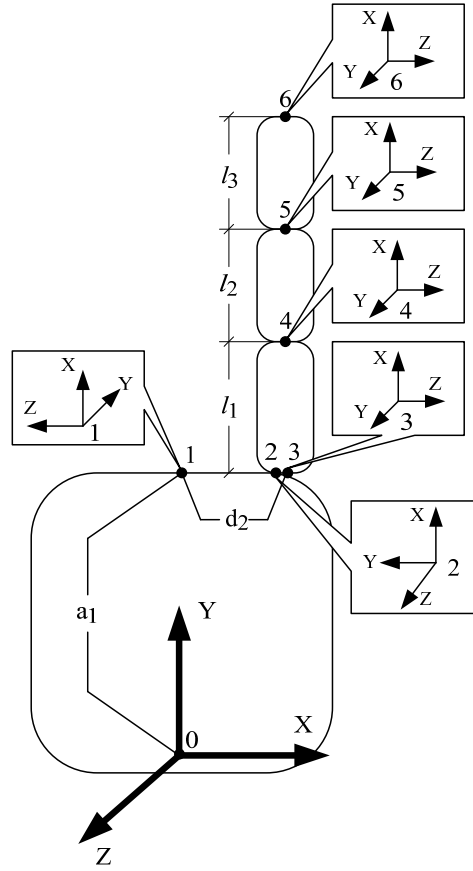
3.1.1 การวิเคราะห์โครงสร้างมือนมนุษย์

การวิเคราะห์โครงสร้างมือนมนุษย์ได้อาศัยหลักจลนศาสตร์ โดยทั่วไปได้ใช้พารามิเตอร์ของ D-H (Denavit-Hartenberg Parameters) ในการเขียนเมตริกซ์การแปลง (Transformation Matrix) เพื่อทราบตำแหน่งปลายของนิ้วมือ ดังสมการที่ 3-1

$$T_{NA} = Trans(0, 0, d_i) Rot_z(\theta_i) Rot_x(\alpha_i) Trans(a_i, 0, 0) \quad (3-1)$$

จากสมการที่ 3-1 สามารถแยกการวิเคราะห์จลนศาสตร์แบบไปข้างหน้าได้ 2 แบบ แบ่งเป็นจลนศาสตร์แบบไปข้างหน้าของนิ้วมือทั้ง 4 และจลนศาสตร์แบบไปข้างหน้าของนิ้วหัวแม่มือ

3.1.1.1 จลนศาสตร์แบบไปข้างหน้าของนิ้วมือทั้ง 4 จลนศาสตร์ของมือนมนุษย์เมื่อเทียบกับหุ่นยนต์หรือแขนกลแล้วจะมีลักษณะคล้ายกัน คือมีการต่อในลักษณะเป็นลูกโซ่โดยที่ในแต่ละนิ้วของนิ้วมือทั้ง 4 นั้นมีมุมหมุนอิสระจำนวนทั้งสิ้น 4 มุม โดย 2 มุมอยู่ที่จุดต่อระหว่างกระดูกฝ่ามือกับกระดูกนิ้ว (MCP) ซึ่งเป็นมุมแบบตามการพับงอ (F/E) และตามการกางของนิ้วมือ (Abd/Add) ส่วนที่จุดต่อระหว่างกระดูกนิ้วของนิ้วมือ (PIP) และจุดต่อระหว่างกระดูกนิ้วกับกระดูกนิ้วส่วนปลาย (DIP) มีมุมหมุนอิสระจุดละ 1 มุม ตามลำดับ โดยเป็นมุมแบบตามการพับงอ (F/E) ดังภาพที่ 3-2



ภาพที่ 3-2 แสดงการวิเคราะห์โครงสร้างของแบบจำลองสำหรับนิ้วมือทั้ง 4

จากข้อมูลข้างต้นสามารถเขียนเมตริกซ์การแปลง (Transformation Matrix) โดยใช้พารามิเตอร์ของ D-H เริ่มจากฐานหรือฝ่ามือจนถึงปลายนิ้วมือ ซึ่งประกอบด้วยก้านโง (Link) และจุดต่อทั้งสิ้น 6 ตำแหน่ง (Chua C. S., 2002) ดังนั้นเมตริกซ์การแปลงต่อนิ้วของนิ้วมือทั้ง 4 สามารถเขียนได้ตามสมการที่ 3-2 ถึงสมการที่ 3-7

$$T_{01} = Rot_{z_0}(\frac{\pi}{2})Rot_{x_0}(-\frac{\pi}{2})Trans(a_1, 0, 0) \quad (3-2)$$

$$T_{12} = Trans(0, 0, d_2)Rot_{x_1}(\frac{\pi}{2}) \quad (3-3)$$

$$T_{23} = Rot_{z_2}(\theta_1)Rot_{x_2}(\frac{\pi}{2}) \quad (3-4)$$

$$T_{34} = Rot_{z_3}(\theta_2)Trans(l_1, 0, 0) \quad (3-5)$$

$$T_{45} = Rot_{z_4}(\theta_3)Trans(l_2, 0, 0) \quad (3-6)$$

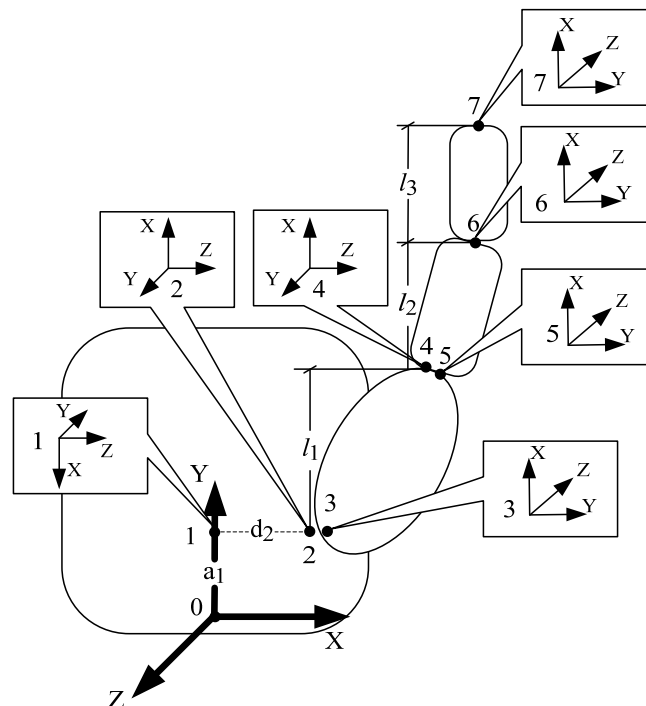
$$T_{56} = Rot_{z_5}(\theta_4)Trans(l_3, 0, 0) \quad (3-7)$$

เมื่อ l_1 , l_2 และ l_3 คือ ความยาวของก้านโงหรือความยาวของนิ้วมือ จากสมการที่ 3-2 ถึงสมการที่ 3-7 สามารถนำมาเขียนพารามิเตอร์ของ D-H ได้ดังตารางที่ 3-1

ตารางที่ 3-1 พารามิเตอร์ของ D-H สำหรับนิ้วมือทั้ง 4

Frame	Geometry	d	θ	α	a
0	Palm	0	0	0	0
1		0	$\pi/2$	$-\pi/2$	a_1
2		d_2	0	$\pi/2$	0
3		0	θ_1	$\pi/2$	0
4	Link1	0	θ_2	0	l_1
5	Link2	0	θ_3	0	l_2
6	Link3	0	θ_4	0	l_3

3.1.1.2 จลนศาสตร์แบบไปข้างหน้าของนิ้วหัวแม่มือ นิ้วหัวแม่มือนั้นมีมุมหมุนอิสระจำนวนทั้งสิ้น 5 มุม โดยจุดต่อระหว่างกระดูกข้อมือกับกระดูกฝ่ามือ (CMC) และจุดต่อระหว่างกระดูกฝ่ามือกับกระดูกนิ้ว (MCP) มีมุมหมุนอิสระจุดละ 2 มุม ตามลำดับซึ่งเป็นมุมหมุนตามการพับงอ (F/E) และตามการกางของนิ้วมือ (Abd/Add) ส่วนจุดต่อระหว่างกระดูกนิ้วกับกระดูกนิ้วส่วนปลายของนิ้วหัวแม่มือ (IP) มีมุมหมุนอิสระ 1 มุม โดยเป็นมุมแบบตามการพับงอ (F/E) ดังภาพที่ 3-3



ภาพที่ 3-3 แสดงการวิเคราะห์โครงสร้างของแบบจำลองของนิ้วหัวแม่มือ

จากข้อมูลข้างต้นสามารถเขียนเมตริกซ์การแปลง (Transformation Matrix) โดยใช้พารามิเตอร์ของ D-H (Chua C. S., 2002) โดยเริ่มจากฐานหรือฝ่ามือจนถึงปลายนิ้วหัวแม่มือ ประกอบด้วยก้านโยง (Link) และจุดต่อทั้งสิ้น 7 ตำแหน่ง ดังนั้นเมตริกซ์การแปลงของนิ้วหัวแม่มือสามารถเขียนดังสมการที่ 3-8 ถึงสมการที่ 3-14

$$T_{01} = Rot_{z_0}(\frac{-\pi}{2})Rot_{x_0}(\frac{-\pi}{2})Trans(a_1, 0, 0) \quad (3-8)$$

$$T_{12} = Trans(0, 0, d_2)Rot_{z_1}(\pi) \quad (3-9)$$

$$T_{23} = Rot_{z_2}(\theta_1)Rot_{x_2}(\frac{\pi}{2}) \quad (3-10)$$

$$T_{34} = Rot_{z_3}(\theta_2)Rot_{x_3}(\frac{-\pi}{2})Trans(l_1, 0, 0) \quad (3-11)$$

$$T_{45} = Rot_{z_4}(\theta_3)Rot_{x_4}(\frac{\pi}{2}) \quad (3-12)$$

$$T_{56} = Rot_{z_5}(\theta_4)Trans(l_2, 0, 0) \quad (3-13)$$

$$T_{67} = Rot_{z_6}(\theta_5)Trans(l_3, 0, 0) \quad (3-14)$$

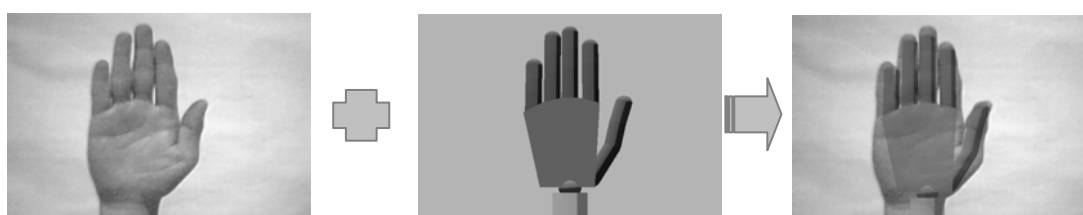
เมื่อ l_1 , l_2 และ l_3 คือ ความยาวของก้านโยงหรือความยาวของนิ้วมือ จากสมการที่ 3-8 ถึงสมการที่ 3-14 สามารถนำมาเขียนพารามิเตอร์ของ D-H ได้ดังตารางที่ 3-2

ตารางที่ 3-2 พารามิเตอร์ของ D-H ของนิ้วหัวแม่มือ

Frame	Geometry	d	θ	α	a
0	Palm	0	0	0	0
1		0	$-\pi/2$	$-\pi/2$	a_1
2		d_2	π	0	0
3		0	θ_1	$\pi/2$	0
4	Link1	0	θ_2	$-\pi/2$	l_1
5		0	θ_3	$\pi/2$	0
6	Link2	0	θ_4	0	l_2
7	Link3	0	θ_5	0	l_3

3.1.2 การออกแบบแบบจำลองมือ

การออกแบบแบบจำลองมือได้นำข้อมูลจากการวิเคราะห์โครงสร้างมือนมนุษย์ร่วมกับการออกแบบมูมหมุนที่บริเวณข้อต่อของนิ้วมือ ส่วนการสร้างแบบจำลองมือได้ใช้โปรแกรมจำลองหุ่นยนต์ 3 มิติ โรโบซิม 2 (Robosim 2) ในการแสดงผลโดยการออกแบบขนาดของแบบจำลองมือได้ใช้วิธีการเปรียบเทียบขนาดมือจากภาพถ่าย กับภาพแบบจำลองมือบนหน้าจอของโปรแกรมจำลองหุ่นยนต์ 3 มิติ ให้มีขนาดของมือใกล้เคียงกัน โดยกำหนดขนาดของภาพทั้ง 2 ให้เท่ากัน ภาพที่ใช้มีขนาด 640×480 พิกเซล แสดงดังภาพที่ 3-4



ก) ภาพถ่ายมือจริง

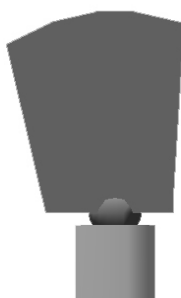
ข) ภาพจากแบบจำลองมือ

ค) ภาพจากการเปรียบเทียบ

ภาพที่ 3-4 แสดงการเปรียบเทียบขนาดของภาพถ่ายมือจริงกับแบบจำลองมือ

ในการออกแบบแบบจำลองมือได้ศึกษาเกี่ยวกับโครงสร้างมือโดยแบ่งการออกแบบออกเป็น 3 ส่วนด้วยกันคือ แบบจำลองบริเวณฝ่ามือ แบบจำลองนิ้วทั้ง 4 และแบบจำลองนิ้วหัวแม่มือ

3.1.2.1 การออกแบบแบบจำลองบริเวณฝ่ามือ ได้แบ่งการออกแบบเป็นส่วนย่อยๆ โดยใช้รูปทรงเรขาคณิตเป็นโครงสร้างพื้นฐานเช่น รูปทรงกล่องสี่เหลี่ยม ทรงกลม และทรงกระบอก โดยมีฐานของแบบจำลองเป็นทรงกระบอก ส่วนข้อมือได้ใช้รูปทรงกลม และส่วนของฝ่ามือได้ใช้รูปทรงกล่องสี่เหลี่ยมมาประกอบเข้าด้วยกัน ซึ่งแบบจำลองบริเวณฝ่ามือนั้นจะเป็นจุดต่อที่ไม่สามารถเคลื่อนไหวได้ (Fixed Link) หรือเป็นฐาน (Base) โดยเป็นฐานของข้อต่อแบบขนาน (Parallel Agent Joint) แสดงดังภาพที่ 3-5



ภาพที่ 3-5 แสดงแบบจำลองบริเวณฝ่ามือ

3.1.2.2 การออกแบบแบบจำลองนิ้วทั้ง 4 สามารถแบ่งการออกแบบเป็นส่วนย่อยๆ โดยใช้รูปทรงเรขาคณิตเป็นโครงสร้างพื้นฐานเช่น ทรงกระบอก ทรงกลม และครึ่งทรงกลมเป็นต้น แต่ละจุดต่อนิ้วจะใช้ทรงกลม นิ้วมือส่วนต่างๆจะใช้ทรงกระบอกและบริเวณปลายนิ้วได้ใช้ครึ่งทรงกลม จุดหมุนหรือจุดต่อของแบบจำลองมือของนิ้วมือทั้ง 4 มีจำนวน 3 จุดด้วยกันโดยเป็นจุดต่อที่มีมุมหมุนอิสระ 2 มุม อยู่ที่จุดต่อ MCP โดยมีลักษณะการเคลื่อนที่แบบตามการพับงอ (F/E) และการกางของนิ้วมือ (Abd/Add) ส่วนจุดต่อ PIP และจุดต่อ DIP มีมุมหมุนอิสระจุดละ 1 มุม โดยมีลักษณะการเคลื่อนที่แบบตามการพับงอของนิ้ว (F/E) ดังภาพที่ 3-6 ถึง ภาพที่ 3-7

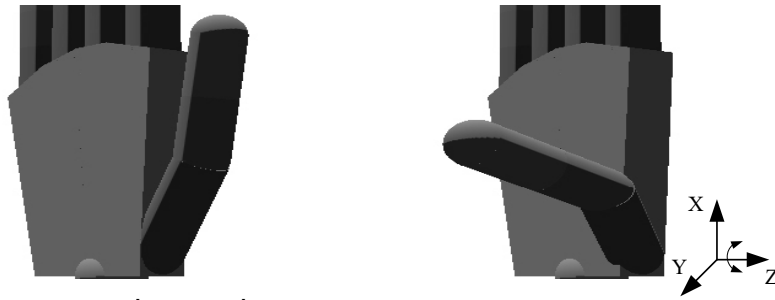


ภาพที่ 3-6 แสดงการเคลื่อนไหวที่จุดต่อ MCP ในลักษณะการกาง (F/E)

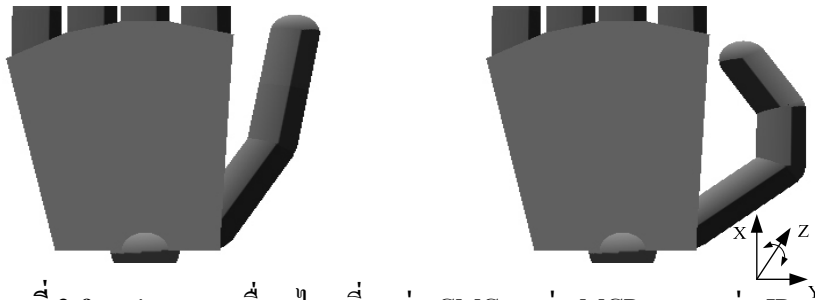


ภาพที่ 3-7 แสดงการเคลื่อนไหวที่จุดต่อ MCP จุดต่อ PIP และจุดต่อ DIP ในลักษณะการพับงอ (Abd/Add)

3.1.2.3 การออกแบบแบบจำลองนิ้วหัวแม่มือ แบ่งการออกแบบเป็นส่วนย่อยๆโดยใช้รูปทรงเรขาคณิตเป็นโครงสร้างพื้นฐานเช่น ทรงกระบอก ทรงกลม และครึ่งทรงกลม (Dome) โดยแต่ละจุดต่อนิ้วจะใช้ทรงกลม นิ้วส่วนต่างๆจะใช้ทรงกระบอก และบริเวณปลายนิ้วหัวแม่มือใช้ครึ่งทรงกลม จุดหมุนหรือจุดต่อของแบบจำลองมือของนิ้วหัวแม่มือมีจำนวน 3 จุด โดยเป็นจุดต่อที่มีมุมหมุนอิสระ 2 มุม อยู่ที่จุดต่อ CMC และจุดต่อ MCP โดยมีลักษณะการเคลื่อนที่แบบตามการพับงอ (F/E) และการกางของนิ้วมือ (Abd/Add) ส่วนจุดต่อ IP โดยมีลักษณะการเคลื่อนที่แบบตามการพับงอของนิ้ว (F/E) ดังภาพที่ 3-8 ถึง ภาพที่ 3-9

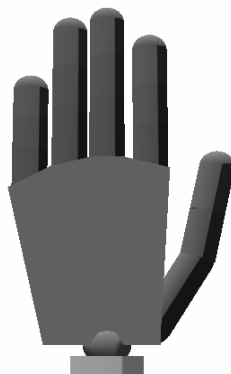


ภาพที่ 3-8 แสดงการเคลื่อนไหวที่จุดต่อ CMC และจุดต่อ MCP ในลักษณะการกาง (Abd/Add)



ภาพที่ 3-9 แสดงการเคลื่อนไหวที่จุดต่อ CMC จุดต่อ MCP และจุดต่อ IP
ในลักษณะการพับงอ (F/E)

จากการออกแบบแบบจำลองมือด้วยโปรแกรมโรโบซิม 2 ผู้วิจัยได้สร้างแบบจำลองมือ โดยมีจำนวนของมุมหมุนอิสระทั้งสิ้น 21 มุม โดยตำแหน่งของมุมมองของโปรแกรมโรโบซิม 2 ซึ่งแสดงผลบนหน้าจออยู่ที่ตำแหน่งพิกัด $[0 \ 155 \ 0]$ ของระบบ 3 มิติ $[X \ Y \ Z]$ และมองไปยังตำแหน่ง $[0 \ 0 \ 30]$ แสดงดังภาพที่ 3-10



ภาพที่ 3-10 แสดงแบบจำลองมือ

3.2 การออกแบบระบบประมวลผลภาพ

การได้มาซึ่งภาพเพื่อใช้ในการประมวลผลภาพนั้นได้มาจากการถ่ายภาพของมือข้างขวาโดยถ่ายจากด้านหน้าของฝ่ามือ ซึ่งมือนั้นสวมถุงมือแพทย์ที่ทำสัญลักษณ์ด้วยสีที่บริเวณปลายนิ้วมือ

ทั้ง 5 นิ้ว มีฉากหลังเป็นสีดำและตั้งกล้องห่างจากมือประมาณ 30 เซนติเมตร โดยภาพถ่ายมือ มีขนาด 640×480 พิกเซล แสดงดังภาพที่ 3-11



ภาพที่ 3-11 แสดงภาพถ่ายมือ

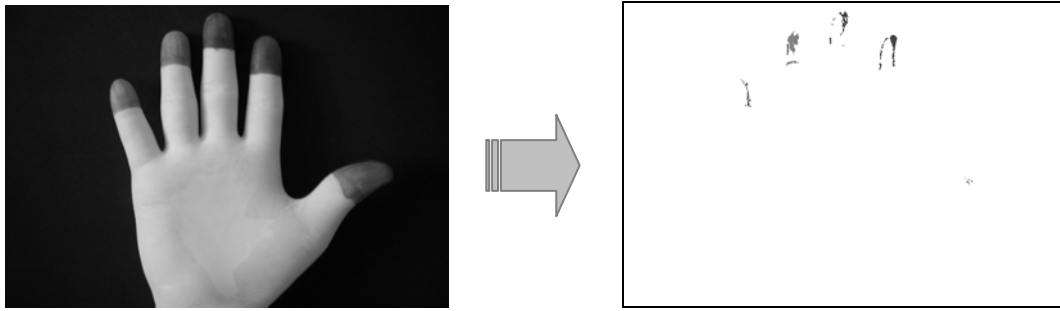
3.2.1 การหาตำแหน่งของสี

การหาตำแหน่งของปลายนิ้วมือจากภาพถ่ายซึ่งเป็นระบบ 2 มิติ สามารถทำได้โดยการหาตำแหน่งของสีซึ่งเกิดจากการทำสัญลักษณ์ไว้ที่ปลายนิ้วมือ การหาตำแหน่งของสีได้ใช้วิธีการตัดค่าสีแบบ RGB โดยในหนึ่งพิกเซลจะประกอบไปด้วยชั้นของแม่สี สีแดง สีเขียว และสีน้ำเงินเป็นหลัก จากนั้นทำการเขียนอัลกอริทึมในโปรแกรมแมทแล็บ (Matlab) เพื่อตรวจจับสีที่บริเวณปลายนิ้วมือทั้ง 5 นิ้ว ได้แก่ สีฟ้าที่นิ้วหัวแม่มือ สีแดงที่นิ้วชี้ สีม่วงที่นิ้วกลาง สีชมพูที่นิ้วนาง และสีเขียวที่นิ้วก้อย โดยมีขอบเขตหรือช่วงของการตัดค่าสี RGB ของแต่ละนิ้ว ดังตารางที่ 3-3

ตารางที่ 3-3 ขอบเขตของการตัดค่าสี RGB ของสีต่างๆของแต่ละนิ้ว

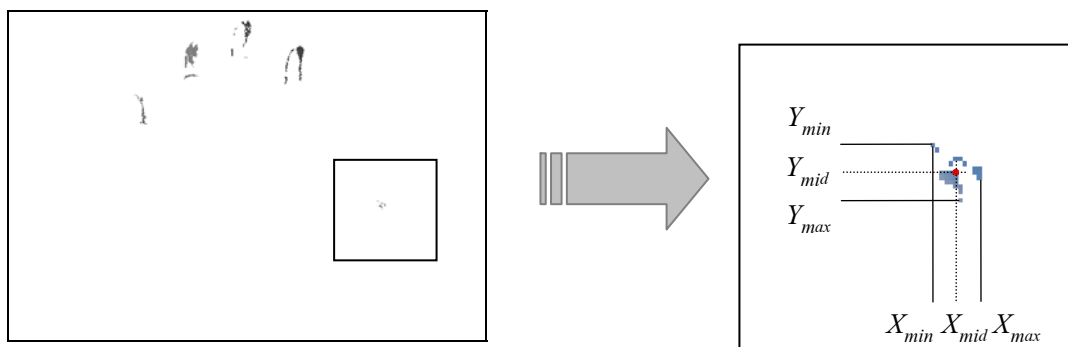
	R _{MAX}	R _{MIN}	G _{MAX}	G _{MIN}	B _{MAX}	B _{MIN}
นิ้วหัวแม่มือ (สีฟ้า)	135	85	175	125	225	175
นิ้วชี้ (สีแดง)	213	187	43	17	43	17
นิ้วกลาง (สีม่วง)	114	86	56	28	132	104
นิ้วนาง (สีชมพู)	255	230	100	60	180	140
นิ้วก้อย (สีเขียว)	74	46	134	106	94	66

เมื่อโปรแกรมทำการตรวจจับสีจากรูปภาพ หากในพิกเซลใดมีสีของค่าสี RGB อยู่ในช่วงที่กำหนดจะเก็บตำแหน่งและค่าสีในพิกเซลนั้นไว้ หากไม่อยู่ในช่วงจะกำหนดให้พิกเซลนั้นมีค่าเป็นสีดำหรือเท่ากับศูนย์ ทำเช่นนั้นจนตรวจจับครบทุกสีบนปลายนิ้วผลของการตรวจจับสีต่างๆสามารถแสดงดังภาพที่ 3-12



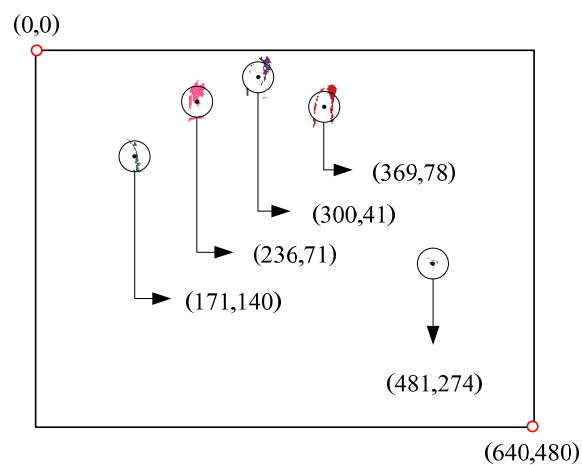
ภาพที่ 3-12 แสดงผลการตรวจจับสีที่ปลายนิ้วมือทั้ง 5 นิ้ว

จากผลการตรวจจับสีที่ปลายนิ้วมือทั้ง 5 นิ้ว สามารถหาตำแหน่งของกลุ่มสีที่ตรวจจับได้ โดยคิดจากกึ่งกลางของตำแหน่งสีในแนวแกน Y และกึ่งกลางของตำแหน่งสีในแนวแกน X ทำให้ได้จุดตัดระหว่างแกน XY ซึ่งเป็นตำแหน่งสีที่ปลายนิ้วโดยประมาณ ดังภาพที่ 3-13



ภาพที่ 3-13 แสดงผลการหาตำแหน่งสีของปลายนิ้วหัวแม่มือ

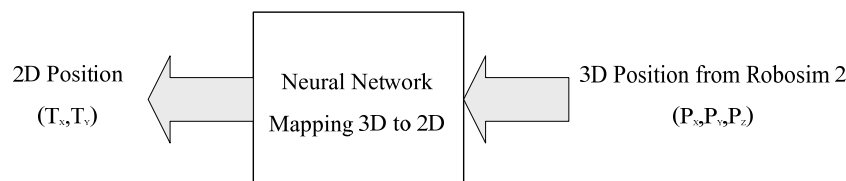
ตำแหน่งสีที่ปลายนิ้วโดยประมาณจะถูกเก็บเป็นตำแหน่งพิกเซลตามที่โปรแกรมได้คำนวณ



ภาพที่ 3-14 แสดงผลการหาตำแหน่งสีที่ปลายนิ้ว

3.3 การออกแบบระบบหาดำแหน่ง 2 มิติ จากตำแหน่ง 3 มิติ

ตำแหน่งปลายนิ้วของแบบจำลองมือในโปรแกรมโรโบซิม 2 เป็นตำแหน่งแบบ 3 มิติ ซึ่งสามารถหาดำแหน่งแบบ 3 มิติ นี้ได้ด้วยเมตริกซ์การแปลงของจลนศาสตร์แบบไปข้างหน้า ส่วนตำแหน่งปลายนิ้วของแบบจำลองมือในแบบ 2 มิติ ที่ปรากฏบนหน้าจอสามารถหาได้ด้วยการใช้ระบบโครงข่ายประสาทเทียม เข้ามาช่วยในการหาดำแหน่ง 2 มิติ บนหน้าจอ



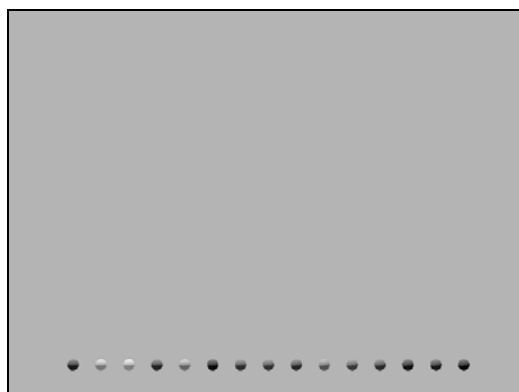
ภาพที่ 3-15 แสดงไดอะแกรมของการแปลงตำแหน่ง 3 มิติ เป็น 2 มิติ

การออกแบบระบบโครงข่ายประสาทเทียมได้แบ่งเป็นส่วนย่อยๆดังนี้คือ การเก็บข้อมูล และการออกแบบระบบโครงข่ายประสาทเทียม

3.3.1 การเก็บข้อมูล

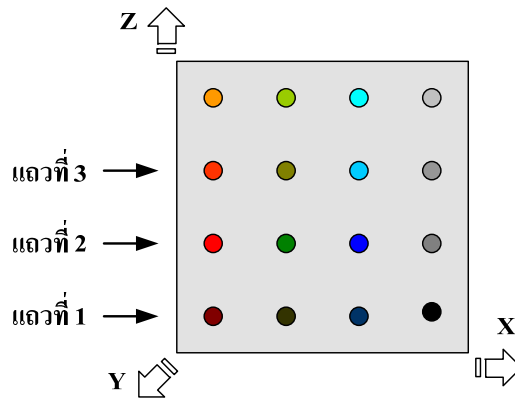
การเก็บข้อมูลเพื่อใช้เป็นข้อมูลในการฝึกฝนระบบโครงข่ายประสาทเทียมจะเก็บข้อมูล 2 ส่วนคือ ส่วนข้อมูลเข้าซึ่งเป็นข้อมูลในจุดต่างๆในระบบ 3 มิติ และส่วนที่ 2 เป็นส่วนข้อมูลออกซึ่งเป็นข้อมูลในจุดต่างๆในระบบ 2 มิติ โดยข้อมูลส่วนนี้ต้องมีความสัมพันธ์กับข้อมูลส่วนแรก

3.3.1.1 การหาชุดข้อมูลตำแหน่ง 3 มิติ บนโปรแกรมโรโบซิม 2 สามารถทำได้ด้วยการสร้างทรงกลมสีขนาด 1 หน่วย วางบนบริเวณต่างๆในระบบ 3 มิติ บนโปรแกรมโรโบซิม 2 โดยจัดทรงกลมดังกล่าวเป็นแถว และกำหนดตำแหน่งในบริเวณต่างๆในระบบ 3 มิติ ทำให้ทราบตำแหน่งแบบ 3 มิติ แสดงดังภาพที่ 3-16



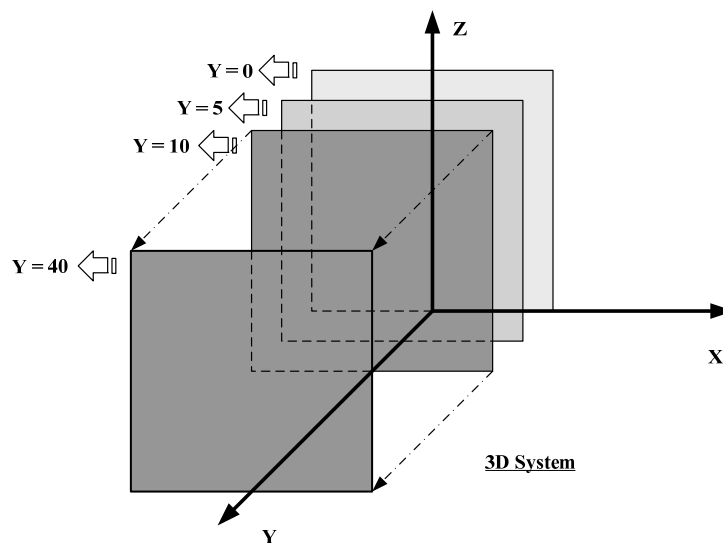
ภาพที่ 3-16 แสดงทรงกลมขนาด 1 หน่วย บนระบบ 3 มิติ บนโปรแกรมโรโบซิม 2

ในการเก็บข้อมูล 3 มิติ จะเก็บทีละแถวเป็นระนาบ ZX โดยกำหนดให้แกน Y คงที่ แสดงดังภาพที่ 3-17



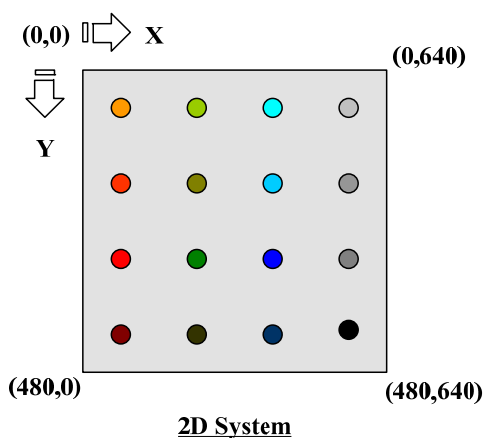
ภาพที่ 3-17 แสดงลักษณะการเก็บข้อมูลในระนาบ ZX

ดังนั้นจะได้ข้อมูลของระบบ 3 มิติ เป็นชั้นตามแกน Y ซ้อนกัน ที่ Y เท่ากับ 0, 5, 10, 15, 20, 25, 30, 35 และ 40 ดังภาพที่ 3-18



ภาพที่ 3-18 แสดงลักษณะข้อมูล 3 มิติ ที่จัดเก็บเป็นชั้นของแกน Y ตั้งแต่ 0 ถึง 40

3.3.1.2 การหาชุดข้อมูลตำแหน่ง 2 มิติ ภาพที่ปรากฏบนหน้าจอแสดงผลบนโปรแกรม โรโบซิม 2 เป็นข้อมูลแบบ 2 มิติ สามารถหาตำแหน่ง 2 มิติ บนหน้าจอด้วยการหาดำแหน่งของสี จากประมวลผลภาพ โดยภาพบนหน้าจอมีขนาด 480×640 และมีพิกัด XY แสดงดังภาพที่ 3-19

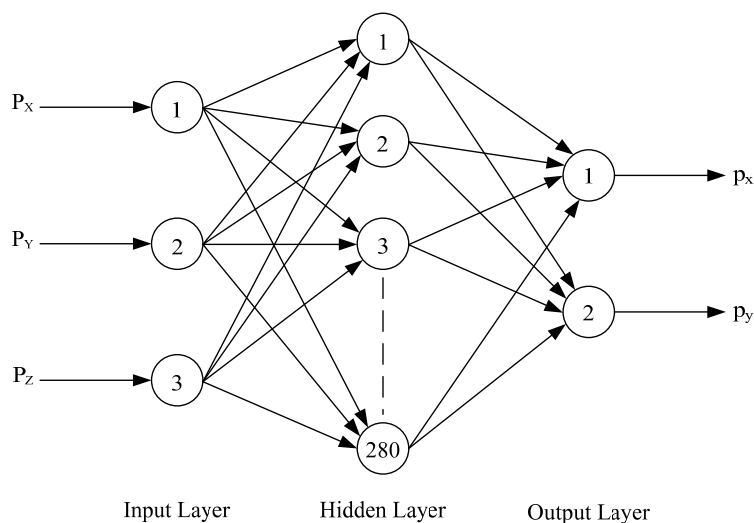


ภาพที่ 3-19 แสดงพิกัด XY ในระบบ 2 มิติ

จำนวนข้อมูลที่ได้ทั้งสิ้น 494 ชุด โดยใช้ตำแหน่ง มุมมอง และการซูมเท่ากับตำแหน่ง มุมมอง และการซูมของแบบจำลองมือที่ต้องใช้งาน

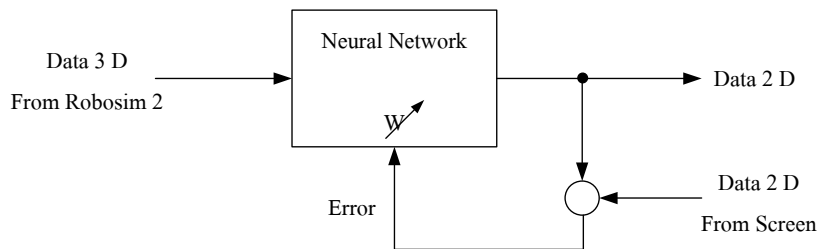
3.3.2 การออกแบบระบบโครงข่ายประสาทเทียม

การออกแบบระบบโครงข่ายประสาทเทียม เพื่อหาความสัมพันธ์ระหว่างตำแหน่ง 3 มิติ จากโปรแกรมโรโบซิม 2 กับตำแหน่ง 2 มิติ บนหน้าจอได้ใช้โครงสร้างของระบบโครงข่ายประสาทเทียมแบบป้อนไปข้างหน้าหลายชั้น โดยมีชั้นข้อมูลเข้าจำนวน 3 ปมข้อมูล (ข้อมูลตำแหน่ง 3 มิติ) ในส่วนชั้นซ่อนมีจำนวน 280 ปมข้อมูล และชั้นข้อมูลออกจำนวน 2 ปมข้อมูล (ข้อมูลตำแหน่ง 2 มิติ) สามารถเขียนโครงสร้างของระบบโครงข่ายประสาทเทียมได้ ดังภาพที่ 3-20



ภาพที่ 3-20 แสดงโครงสร้างของระบบโครงข่ายประสาทเทียม

และสามารถเขียนแผนภาพการฝึกฝนระบบโครงข่ายประสาทเทียมได้ดังภาพที่ 3-21



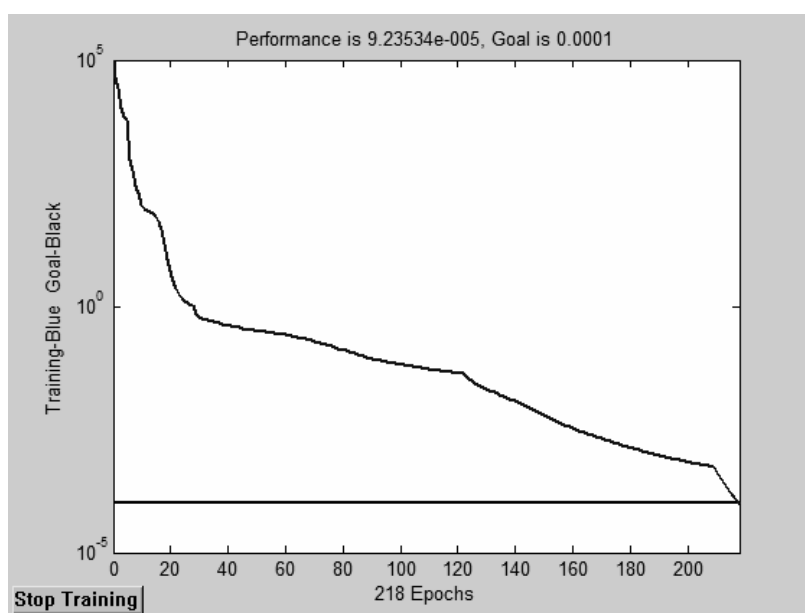
ภาพที่ 3-21 แสดงแผนภาพการฝึกฝนระบบโครงข่ายประสาทเทียม

ใช้คำสั่งในการฝึกฝนระบบโครงข่ายประสาทเทียมด้วยโปรแกรมแมทแลบดังนี้

```

ThreeD = Data3D;                (ข้อมูล 3 มิติจาก Robosim 2)
TwoD = Data2D;                  (ข้อมูล 2 มิติจากการหาดำแหน่งบนหน้าจอ)
s = minmax(ThreeD);
net = newff(s,[3 280 2],{'logsig' 'logsig' 'purelin'},'trainlm');
net.trainParam.show = 25;
net.trainparam.goal = 1e-3;
net.trainparam.epochs = 5000;
net = train(net,ThreeD,TwoD)
  
```

ในการฝึกฝนระบบโครงข่ายประสาทเทียมได้กำหนดให้ Mean Square Error ของระบบเท่ากับ 0.001 การดูเข้าของการฝึกฝนระบบโครงข่ายประสาทเทียมแสดงดังภาพที่ 3-22



ภาพที่ 3-22 แสดงการดูเข้าของระบบโครงข่ายประสาทเทียม

3.4 การวิเคราะห์การเคลื่อนไหวและออกแบบระบบควบคุม

การวิเคราะห์การเคลื่อนไหวและการออกแบบระบบควบคุม แบ่งเนื้อหาเป็น 2 ส่วนคือ การวิเคราะห์การเคลื่อนไหวของนิ้วมือ และออกแบบระบบควบคุม โดยการวิเคราะห์การเคลื่อนไหวของนิ้วมือได้ใช้ข้อกำหนดหรือเงื่อนไขของจุดต่อ ส่วนการออกแบบระบบควบคุมได้ใช้ผลจากการวิเคราะห์ควบคู่กับการควบคุมด้วยระบบฟัซซี่

3.4.1 การวิเคราะห์การเคลื่อนไหว

จากงานวิจัยของ J.Lee และ T.L.Kunil (1995) ได้วิเคราะห์การเคลื่อนไหวของนิ้วมือโดยแบ่งการวิเคราะห์เป็น 2 ส่วนคือ การวิเคราะห์การเคลื่อนไหวของนิ้วมือทั้ง 4 และการเคลื่อนไหวของนิ้วหัวแม่มือ

3.4.1.1 การวิเคราะห์การเคลื่อนไหวของนิ้วทั้ง 4 นิ้ว มีข้อกำหนดหรือเงื่อนไขดังต่อไปนี้ คือ จุดต่อแบบ MCP (θ_2) สามารถเคลื่อนไหวในลักษณะการพับงอ (F/E) โดยมีอัตราส่วนตั้งแต่ 0-0.5 เท่าของจุดต่อแบบ PIP (θ_3) และจุดต่อแบบ PIP (θ_3) จะสามารถเคลื่อนไหวในลักษณะการพับงอ (F/E) ได้เป็นสองในสามเท่าของจุดต่อแบบ DIP (θ_4) ซึ่งสามารถเขียนได้ดังสมการที่ 3-15 และสมการที่ 3-16

$$\theta_2 = k \cdot \theta_3 \quad ; \quad 0 \leq k \leq 0.5 \quad (3-15)$$

$$\theta_4 = \frac{2}{3} \theta_3 \quad (3-16)$$

3.4.1.2 การวิเคราะห์การเคลื่อนไหวของนิ้วหัวแม่มือ มีข้อกำหนดหรือเงื่อนไขดังต่อไปนี้คือ การเคลื่อนไหวในลักษณะการกาง (Abd/Add) ของจุดต่อแบบ CMC (θ_1) กับจุดต่อแบบ MCP (θ_3) แสดงดังสมการที่ 3-17

$$\theta_3 = 2(\theta_1 - \frac{\pi}{6}) \quad (3-17)$$

การเคลื่อนไหวในลักษณะการพับงอ (F/E) ของจุดต่อแบบ CMC (θ_2) กับจุดต่อแบบ MCP (θ_4) ดังสมการที่ 3-18 และจุดต่อแบบ MCP (θ_4) กับจุดต่อแบบ IP (θ_5) ดังสมการที่ 3-19 ตามลำดับ

$$\theta_4 = \frac{7}{5} \cdot \theta_2 \quad (3-18)$$

$$\theta_5 = a \cdot \theta_4 \quad ; \quad a \geq 0 \quad (3-19)$$

3.4.2 การออกแบบระบบควบคุม

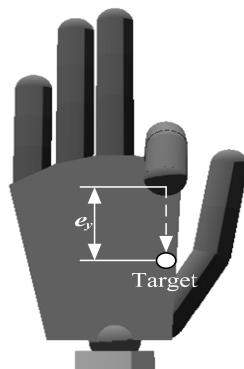
การออกแบบระบบควบคุมได้ใช้ข้อกำหนด หรือเงื่อนไขในการเคลื่อนไหวของนิ้วมือร่วมกับตัวควบคุมแบบฟัซซี่โดยเขียนอัลกอริทึมบนโปรแกรมเมทแลป (Matlab) และเชื่อมต่อกับโปรแกรมโรโบซิม 2 (Robosim 2) ในการออกแบบระบบควบคุมได้แบ่งออกเป็น 2 ส่วนใหญ่คือ การออกแบบระบบควบคุมของนิ้วทั้ง 4 และนิ้วหัวแม่มือ

3.4.2.1 การออกแบบระบบควบคุมของนิ้วทั้ง 4 ได้แบ่งการควบคุมออกเป็น 2 ส่วนคือ การควบคุมการเข้าสู่ตำแหน่งจากการพับงอนิ้ว และการควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้ว

ก) การควบคุมการเข้าสู่ตำแหน่งจากการพับงอนิ้ว ในการควบคุมจะพิจารณา ตัวแปรซึ่งเป็นข้อมูลเข้าสำหรับการควบคุม 2 ค่าด้วยกันคือ ค่าผิดพลาดในแนวแกน Y (e_y) ซึ่งจะบอกตำแหน่งของปลายนิ้ว หากค่าผิดพลาดเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วอยู่ต่ำกว่าตำแหน่งที่ต้องการมาก หากค่าผิดพลาดเป็นลบ (Negative: N) แสดงว่าปลายนิ้วอยู่ต่ำกว่าตำแหน่งที่ต้องการ หากค่าผิดพลาดเป็นบวก (Positive: P) แสดงว่าปลายนิ้วอยู่สูงกว่าตำแหน่งที่ต้องการ หากค่าผิดพลาดเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วอยู่สูงกว่าตำแหน่งที่ต้องการมาก และหากค่าผิดพลาดเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วอยู่ในตำแหน่งที่ต้องการพอดีค่าผิดพลาดในแนวแกน Y สามารถหาได้จากสมการที่ 3-20

$$e_y = T_y - P_y \quad (3-20)$$

โดย e_y คือ ค่าผิดพลาดในแนวแกน Y
 T_y คือ ตำแหน่งในแนวแกน Y ที่ต้องการ
 P_y คือ ตำแหน่งปลายนิ้วในแนวแกน Y ของแบบจำลอง



ภาพที่ 3-23 แสดงการเข้าสู่ตำแหน่งจากการพับงอนิ้ว

ตัวแปรในการควบคุมอีกตัวคือ ผลต่างของค่าผิดพลาดในแนวแกน Y ระหว่างปัจจุบันกับครั้งก่อน (Δe_y) จะเป็นตัวบอกว่าปลายนิ้วกำลังเคลื่อนไหวไปในทิศทางใด หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ลงอย่างรวดเร็ว หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ลง หากค่าเป็นบวก (Positive: P) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ขึ้น หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ขึ้นอย่างรวดเร็ว และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วไม่เกิดการเคลื่อนที่ ผลต่างของค่าผิดพลาดในแนวแกน Y สามารถหาได้จากสมการที่ 3-21

$$\Delta e_y = e_{y(i)} - e_{y(i-1)} \quad (3-21)$$

โดย Δe_y คือ ผลต่างของค่าผิดพลาดในแนวแกน Y
 $e_{y(i)}$ คือ ค่าผิดพลาดในแนวแกน Y ครั้งปัจจุบัน
 $e_{y(i-1)}$ คือ ค่าผิดพลาดในแนวแกน Y ครั้งก่อน

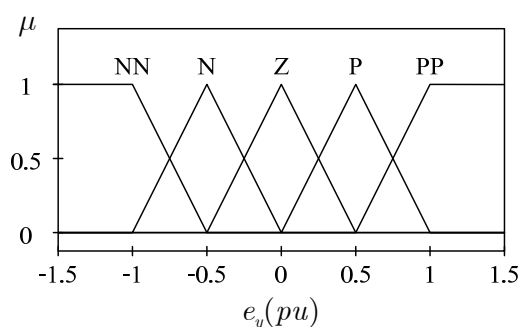
การควบคุมการเข้าสู่ตำแหน่งจากการพังอนี้ สามารถเขียนเป็นกฎของฟัซซี่เพื่อใช้ในการควบคุม ดังตารางที่ 3-4

ตารางที่ 3-4 กฎของฟัซซี่ในการควบคุมการเข้าสู่ตำแหน่งจากการพังอนี้

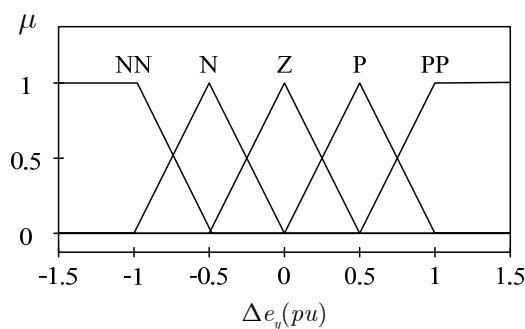
		e_y				
		NN	N	Z	P	PP
Δe_y	NN	NN	NN	Z	P	PP
	N	NN	N	Z	P	PP
	Z	NN	N	Z	P	PP
	P	NN	N	Z	P	PP
	PP	NN	N	Z	PP	PP

ผลที่ได้จากตารางที่ 3-4 เป็นผลต่างของมุม θ_2 ($\Delta\theta_2$) ในการปรับมุม θ_2 ที่จุดต่อ MCP ในแต่ละนิ้วของนิ้วทั้ง 4 หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วเคลื่อนที่ขึ้นอย่างมาก หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิ้วเคลื่อนที่ขึ้น หากค่าเป็นบวก (Positive: P) แสดงว่าปลายนิ้วเคลื่อนที่ลง หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วเคลื่อนที่ลงอย่างมาก และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วไม่เคลื่อนที่

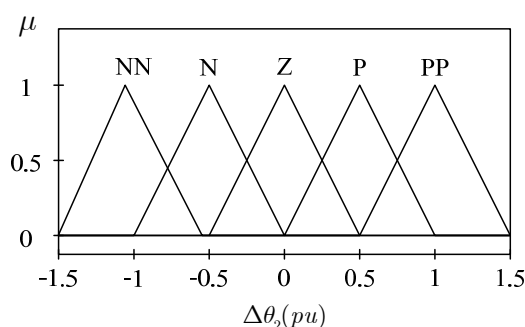
การออกแบบฟังก์ชันความเป็นสมาชิกของ ค่าผิดพลาดในแนวแกน Y (e_y) ผลต่างของค่าผิดพลาดในแนวแกน Y (Δe_y) และผลต่างของมุม θ_2 ($\Delta\theta_2$) ดังภาพที่ 3-24 ถึงภาพที่ 3-26



ภาพที่ 3-24 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน Y (e_y)



ภาพที่ 3-25 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน Y (Δe_y)

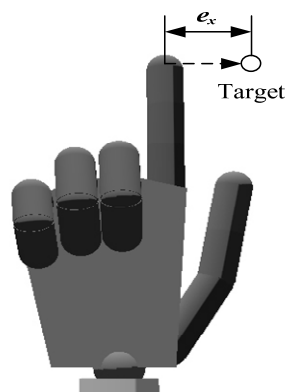


ภาพที่ 3-26 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_2 ($\Delta\theta_2$)

ข) การควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้ว ในการควบคุมจะพิจารณาตัวแปรซึ่งเป็นข้อมูลเข้าสำหรับการควบคุม 2 ค่าด้วยกันคือ ค่าผิดพลาดในแนวแกน X (e_x) ซึ่งจะบอกตำแหน่งของปลายนิ้ว หากค่าผิดพลาดเป็นลบมาก (Negative-Negative: NN) แสดงว่าตำแหน่งที่ต้องการอยู่ทางซ้ายของปลายนิ้วมาก หากค่าผิดพลาดเป็นลบ (Negative: N) แสดงว่าตำแหน่งที่ต้องการอยู่ทางซ้ายของปลายนิ้ว หากค่าผิดพลาดเป็นบวก (Positive: P) แสดงว่าตำแหน่งที่ต้องการอยู่ทางขวาของปลายนิ้ว หากค่าผิดพลาดเป็นบวกมาก (Positive-Positive: PP) แสดงว่าตำแหน่งที่ต้องการอยู่ทางขวาของปลายนิ้วมาก และหากค่าผิดพลาดเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วอยู่ในตำแหน่งที่ต้องการพอดี ค่าผิดพลาดในแนวแกน X สามารถหาได้จากสมการที่ 3-22

$$e_x = T_x - P_x \quad (3-22)$$

โดย e_x คือ ค่าผิดพลาดในแนวแกน X
 T_x คือ ตำแหน่งในแนวแกน X ที่ต้องการ
 P_x คือ ตำแหน่งปลายนิ้วในแนวแกน X ของแบบจำลอง



ภาพที่ 3-27 แสดงการเข้าสู่ตำแหน่งจากการกางนิ้ว

ตัวแปรในการควบคุมอีกตัวคือ ผลต่างของค่าผิดพลาดในแนวแกน X ระหว่างปัจจุบันกับครั้งก่อน (Δe_x) จะเป็นตัวบอกว่าปลายนิ้วกำลังเคลื่อนไหวไปในทิศทางใด หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางขวาอย่างรวดเร็ว หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางขวา หากค่าเป็นบวก (Positive: P) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางซ้าย หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางซ้ายอย่างรวดเร็ว และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าแสดงว่าปลายนิ้วไม่เคลื่อนที่ ผลต่างของค่าผิดพลาดในแนวแกน X สามารถหาได้จากสมการที่ 3-23

$$\Delta e_x = e_{x(i)} - e_{x(i-1)} \quad (3-23)$$

โดย Δe_x คือ ผลต่างของค่าผิดพลาดในแนวแกน X
 $e_{x(i)}$ คือ ค่าผิดพลาดในแนวแกน X ครั้งปัจจุบัน
 $e_{x(i-1)}$ คือ ค่าผิดพลาดในแนวแกน X ครั้งก่อน

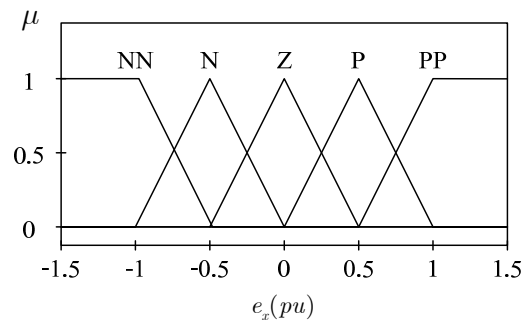
การควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้ว สามารถเขียนเป็นกฎของฟัซซี่เพื่อใช้ในการควบคุม ได้ดังตารางที่ 3-5

ตารางที่ 3-5 กฎของฟัซซี่ในการควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้ว

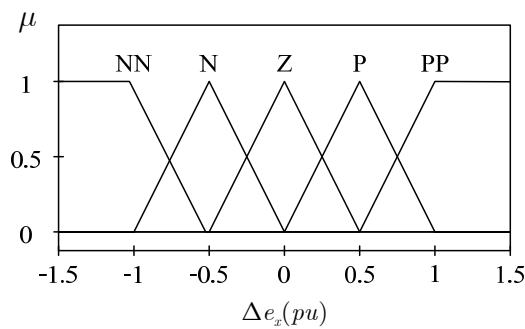
		e_x				
		NN	N	Z	P	PP
Δe_x	NN	PP	PP	Z	NN	NN
	N	PP	P	Z	N	NN
	Z	PP	P	Z	N	NN
	P	PP	P	Z	N	NN
	PP	PP	PP	Z	NN	NN

ผลที่ได้จากตารางที่ 3-5 เป็นผลต่างของมุม θ_1 ($\Delta\theta_1$) ในการปรับมุม θ_1 ที่จุดต่อ MCP ในแต่ละนิวของนิวทั้ง 4 หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิวเคลื่อนไปทางขวาอย่างมาก หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิวเคลื่อนไปทางขวา หากค่าเป็นบวก (Positive: P) แสดงว่าปลายนิวเคลื่อนไปทางซ้าย หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิวเคลื่อนไปทางซ้ายอย่างมาก และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิวไม่เคลื่อนที่

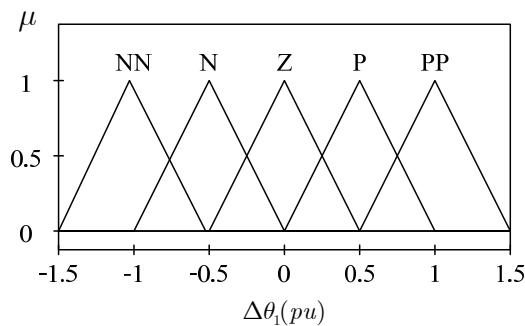
การออกแบบฟังก์ชันความเป็นสมาชิกของ ค่าผิดพลาดในแนวแกน X (e_x) ผลต่างของค่าผิดพลาดในแนวแกน X (Δe_x) และผลต่างของมุม θ_1 ($\Delta\theta_1$) ดังภาพที่ 3-28 ถึงภาพที่ 3-30



ภาพที่ 3-28 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน X (e_x)

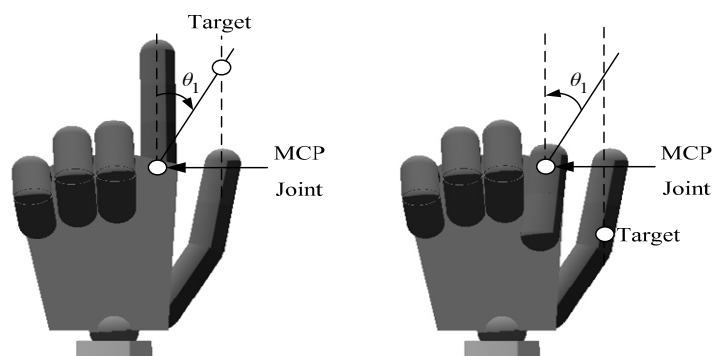


ภาพที่ 3-29 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน X (Δe_x)



ภาพที่ 3-30 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_1 ($\Delta\theta_1$)

หากตำแหน่งของปลายนิ้วอยู่ต่ำกว่าจุดต่อ MCP การเคลื่อนที่ของปลายนิ้วเพื่อเข้าสู่ตำแหน่งจากการกางนิ้ว จะไม่สามารถใช้ตารางที่ 3-5 สามารถอธิบายการเข้าสู่ตำแหน่งจากการกางนิ้วเมื่อตำแหน่งของปลายนิ้วอยู่ต่ำกว่าตำแหน่ง MCP ได้จากภาพที่ 3-31 โดยภาพที่ 3-31 ก) การเข้าสู่ตำแหน่งจากการกางนิ้วที่ตำแหน่งปลายนิ้วอยู่สูงกว่าจุดต่อ MCP สังเกตว่าหากต้องการเคลื่อนตำแหน่งปลายนิ้วให้เข้าสู่ตำแหน่งที่ต้องการจะต้องขยับมุม θ_1 แบบ Negative ซึ่งเป็นผลตามตารางที่ 3-5 แต่จากภาพที่ 3-31 ข) การเข้าสู่ตำแหน่งจากการกางนิ้ว โดยที่ตำแหน่งปลายนิ้วอยู่ต่ำกว่าจุดต่อ MCP สังเกตว่าหากต้องการเคลื่อนตำแหน่งปลายนิ้วให้เข้าสู่ตำแหน่งที่ต้องการจะขยับมุม θ_1 แบบ Negative เดิมดังตารางที่ 3-5 ไม่ได้ แต่จะต้องขยับมุม θ_1 แบบ Positive จึงทำให้ปลายนิ้วเข้าสู่ตำแหน่งได้อย่างถูกต้อง



ก) การเข้าสู่ตำแหน่งจากการกางนิ้ว
ที่ตำแหน่งปลายนิ้วอยู่สูงกว่าจุดต่อ MCP

ข) การเข้าสู่ตำแหน่งจากการกางนิ้ว
ที่ตำแหน่งปลายนิ้วอยู่ต่ำกว่าจุดต่อ MCP

ภาพที่ 3-31 แสดงทิศทางการหมุนของมุม θ_1

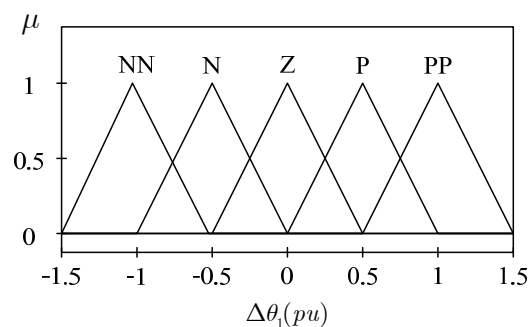
สามารถเขียนเป็นกฎของฟิชชีเพื่อใช้ในการควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้วเมื่อตำแหน่งของปลายนิ้วอยู่ต่ำกว่าจุดต่อ MCP ได้ดังตารางที่ 3-6

ตารางที่ 3-6 กฎของฟิชชีในการควบคุมการเข้าสู่ตำแหน่งจากการกางนิ้ว เมื่อตำแหน่งของปลายนิ้วอยู่ต่ำกว่าจุดต่อ MCP

		e_x				
		NN	N	Z	P	PP
Δe_x	NN	NN	NN	Z	PP	PP
	N	NN	N	Z	P	PP
	Z	NN	N	Z	P	PP
	P	NN	N	Z	P	PP
	PP	NN	NN	Z	PP	PP

ผลที่ได้จากตารางที่ 3-6 เป็นผลต่างของมุม θ_1 ($\Delta\theta_1$) ในการปรับมุม θ_1 ที่จุดต่อ MCP ในแต่ละนิวของนิวทั้ง 4 ในกรณีที่ตำแหน่งของปลายนิวอยู่ต่ำกว่าจุดต่อ MCP หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิวเคลื่อนไปทางซ้ายมาก หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิวเคลื่อนไปทางซ้าย หากค่าเป็นบวก (Positive: P) แสดงว่าปลายนิวเคลื่อนไปทางขวา หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิวเคลื่อนไปทางขวาอย่างมาก และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิวไม่เคลื่อนที่

การออกแบบฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_1 ($\Delta\theta_1$) สำหรับการควบคุมการเข้าสู่ตำแหน่งจากการกางนิวเมื่อตำแหน่งของปลายนิวอยู่ต่ำกว่าจุดต่อ MCP ดังภาพที่ 3-32



ภาพที่ 3-32 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_1 สำหรับการควบคุมการเข้าสู่ตำแหน่งจากการกางนิวเมื่อตำแหน่งของปลายนิวอยู่ต่ำกว่าจุดต่อ MCP

3.4.2.2 การออกแบบระบบควบคุมของนิวหัวแม่มือ แบ่งการควบคุมเป็น 2 ส่วนคือการควบคุมการเข้าสู่ตำแหน่งในแนวแกน X และการควบคุมการเข้าสู่ตำแหน่งแนวแกน Y

ก) การควบคุมการเข้าสู่ตำแหน่งในแนวแกน X จะพิจารณาตัวแปรซึ่งเป็นข้อมูลเข้าสำหรับการควบคุม 2 ค่า คือ ค่าผิดพลาดในแนวแกน X (e_x) จะบอกตำแหน่งของปลายนิว หากค่าผิดพลาดเป็นลบมาก (Negative-Negative: NN) แสดงว่าตำแหน่งที่ต้องการอยู่ทางซ้ายของปลายนิวมาก หากค่าผิดพลาดเป็นลบ (Negative: N) แสดงว่าตำแหน่งที่ต้องการอยู่ทางซ้ายของปลายนิว หากค่าผิดพลาดเป็นบวก (Positive: P) แสดงว่าตำแหน่งที่ต้องการอยู่ทางขวาของปลายนิว หากค่าผิดพลาดเป็นบวกมาก (Positive-Positive: PP) แสดงว่าตำแหน่งที่ต้องการอยู่ทางขวาของปลายนิวมาก และหากค่าผิดพลาดเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิวอยู่ในตำแหน่งที่ต้องการพอดี ค่าผิดพลาดในแนวแกน X สามารถหาได้จากสมการที่ 3-24

$$e_x = T_x - P_x \quad (3-24)$$

โดย e_x คือ ค่าผิดพลาดในแนวแกน X
 T_x คือ ตำแหน่งในแนวแกน X ที่ต้องการ
 P_x คือ ตำแหน่งปลายนิวในแนวแกน X ของแบบจำลอง

ตัวแปรในการควบคุมอีกตัวคือ ผลต่างของค่าผิดพลาดในแนวแกน X ระหว่างปัจจุบัน กับครั้งก่อน (Δe_x) จะเป็นตัวบอกว่าปลายนิ้วกำลังเคลื่อนไหวกวไปทิศทางใด หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางขวาอย่างรวดเร็ว หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางขวา หากค่าเป็นบวก (Positive: P) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางซ้าย หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ไปทางซ้ายอย่างรวดเร็ว และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วไม่เกิดเคลื่อนที่ ผลต่างของค่าผิดพลาดในแนวแกน X สามารถหาได้จากสมการที่ 3-25

$$\Delta e_x = e_{x(i)} - e_{x(i-1)} \quad (3-25)$$

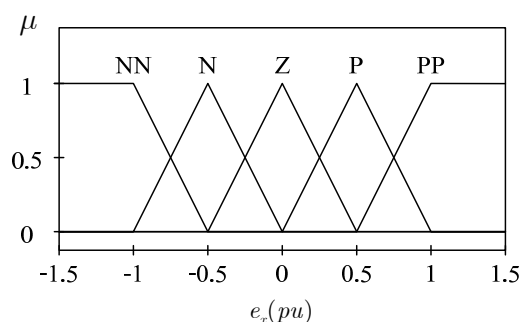
โดย Δe_x คือ ผลต่างของค่าผิดพลาดในแนวแกน X
 $e_{x(i)}$ คือ ค่าผิดพลาดในแนวแกน X ครั้งปัจจุบัน
 $e_{x(i-1)}$ คือ ค่าผิดพลาดในแนวแกน X ครั้งก่อน

ตารางที่ 3-7 กฎของฟิชชีในการควบคุมการเข้าสู่ตำแหน่งในแนวแกน X

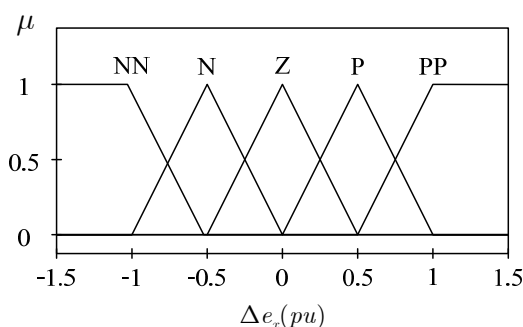
		e_x				
		NN	N	Z	P	PP
Δe_x	NN	NN	NN	Z	P	PP
	N	NN	N	Z	P	PP
	Z	NN	N	Z	P	PP
	P	NN	N	Z	P	PP
	PP	NN	N	Z	PP	PP

ผลที่ได้จากตารางที่ 3-7 เป็นผลต่างของมุม θ_1 ($\Delta\theta_1$) ในการปรับมุม θ_1 ที่จุดต่อ CMC ของ นิ้วหัวแม่มือ หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วเคลื่อนไปทางซ้ายมาก หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิ้วเคลื่อนไปทางซ้าย หากค่าเป็นบวก (Positive: P) แสดงว่าปลายนิ้วเคลื่อนไปทางขวา หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วเคลื่อนไปทางขวามาก และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วไม่เคลื่อนที่

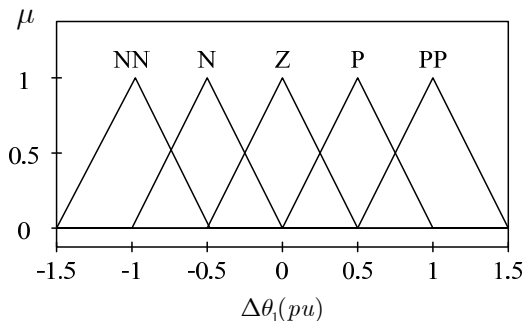
การออกแบบฟังก์ชันความเป็นสมาชิกของ ค่าผิดพลาดในแนวแกน X (e_x) ผลต่างของ ค่าผิดพลาดในแนวแกน X (Δe_x) และผลต่างของมุม θ_1 ($\Delta\theta_1$) ดังภาพที่ 3-33 ถึงภาพที่ 3-35



ภาพที่ 3-33 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน X (e_x)



ภาพที่ 3-34 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน X (Δe_x)



ภาพที่ 3-35 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_1 ($\Delta\theta_1$)

ข) การควบคุมการเข้าสู่ตำแหน่งในแนวแกน Y จะพิจารณาตัวแปรซึ่งเป็นข้อมูลเข้าสำหรับการควบคุม 2 ค่า คือ ค่าผิดพลาดในแนวแกน Y (e_y) จะบอกตำแหน่งของปลายนิ้ว หากค่าผิดพลาดเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วอยู่ต่ำกว่าตำแหน่งที่ต้องการมาก หากค่าผิดพลาดเป็นลบ (Negative: N) แสดงว่าปลายนิ้วอยู่ต่ำกว่าตำแหน่งที่ต้องการ หากค่าผิดพลาดเป็นบวก (Positive: P) แสดงว่าปลายนิ้วอยู่สูงกว่าตำแหน่งที่ต้องการ หากค่าผิดพลาดเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วอยู่สูงกว่าตำแหน่งที่ต้องการมาก และหากค่าผิดพลาดเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วอยู่ในตำแหน่งที่ต้องการพอดี ค่าผิดพลาดในแนวแกน Y สามารถหาได้จากสมการที่ 3-26

$$e_y = T_y - P_y \quad (3-26)$$

โดย e_y คือ ค่าผิดพลาดในแนวแกน Y
 T_y คือ ตำแหน่งในแนวแกน Y ที่ต้องการ
 P_y คือ ตำแหน่งปลายนิ้วในแนวแกน Y ของแบบจำลอง

ตัวแปรในการควบคุมอีกตัว คือ ผลต่างของค่าผิดพลาดในแนวแกน Y ระหว่างปัจจุบันกับครั้งก่อน (Δe_y) จะเป็นตัวบอกว่าปลายนิ้วกำลังเคลื่อนไหวไปในทิศทางใด หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ลงอย่างรวดเร็ว หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิ้วกำลังเคลื่อนที่ลง หากค่าเป็นบวก (Positive: P) แสดงว่าแสดงว่าปลายนิ้วกำลังเคลื่อนที่ขึ้น หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าแสดงว่าปลายนิ้วกำลังเคลื่อนที่ขึ้นอย่างรวดเร็ว และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าแสดงว่าปลายนิ้วไม่เกิดเคลื่อนที่ ผลต่างของค่าผิดพลาดในแนวแกน Y สามารถหาได้จากสมการที่ 3-27

$$\Delta e_y = e_{y(i)} - e_{y(i-1)} \quad (3-27)$$

โดย Δe_y คือ ผลต่างของค่าผิดพลาดในแนวแกน Y
 $e_{y(i)}$ คือ ค่าผิดพลาดในแนวแกน Y ครั้งปัจจุบัน
 $e_{y(i-1)}$ คือ ค่าผิดพลาดในแนวแกน Y ครั้งก่อน

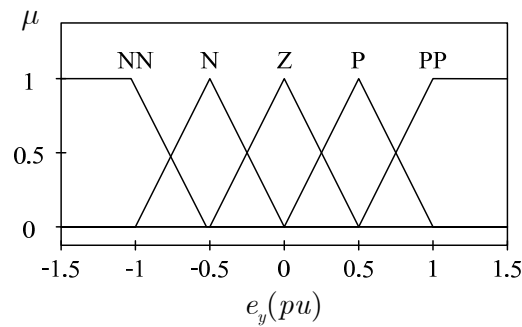
ตารางที่ 3-8 กฎของฟัซซีในการควบคุมการเข้าสู่ตำแหน่งในแนวแกน Y

		e_y				
		NN	N	Z	P	PP
Δe_y	NN	PP	PP	Z	N	NN
	N	PP	P	Z	N	NN
	Z	PP	P	Z	N	NN
	P	PP	P	Z	N	NN
	PP	PP	P	Z	NN	NN

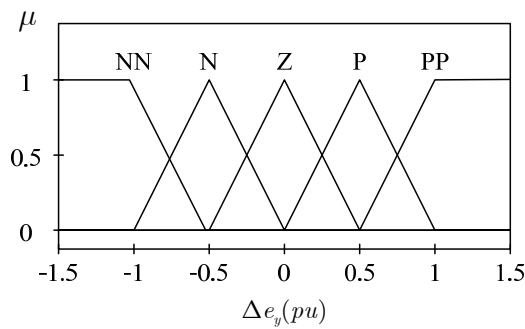
ผลที่ได้จากตารางที่ 3-8 เป็นผลต่างของมุม θ_2 ($\Delta\theta_2$) ในการปรับมุม θ_2 ที่จุดต่อ CMC ของนิ้วหัวแม่มือ หากค่าเป็นลบมาก (Negative-Negative: NN) แสดงว่าปลายนิ้วเคลื่อนลงอย่างมาก หากค่าเป็นลบ (Negative: N) แสดงว่าปลายนิ้วเคลื่อนที่ลง หากค่าเป็นบวก (Positive: P) แสดงว่า

ปลายนิ้วเคลื่อนขึ้น หากค่าเป็นบวกมาก (Positive-Positive: PP) แสดงว่าปลายนิ้วเคลื่อนขึ้นอย่างมาก และหากค่าเป็นศูนย์ (Zero: Z) แสดงว่าปลายนิ้วไม่เคลื่อนที่

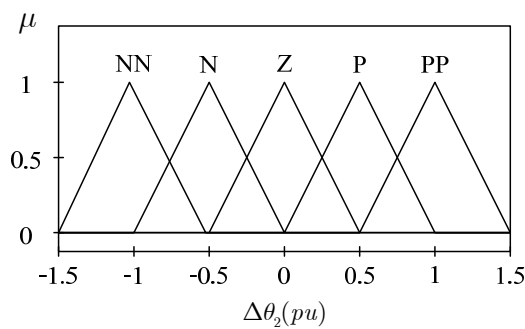
การออกแบบฟังก์ชันความเป็นสมาชิกของ ค่าผิดพลาดในแนวแกน Y (e_y) ผลต่างของค่าผิดพลาดในแนวแกน Y (Δe_y) และผลต่างของมุม θ_2 ($\Delta \theta_2$) ดังภาพที่ 3-36 ถึงภาพที่ 3-38



ภาพที่ 3-36 แสดงกราฟฟังก์ชันความเป็นสมาชิกของค่าผิดพลาดในแนวแกน Y (e_y)



ภาพที่ 3-37 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของค่าผิดพลาดในแนวแกน Y (Δe_y)



ภาพที่ 3-38 แสดงกราฟฟังก์ชันความเป็นสมาชิกของผลต่างของมุม θ_2 ($\Delta \theta_2$)

บทที่ 4

ผลการวิจัย

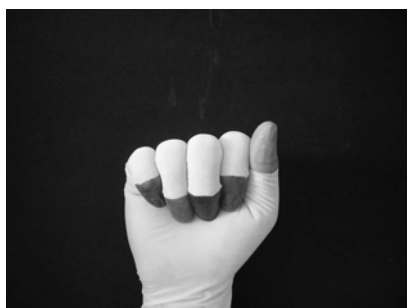
ผลการวิจัยเรื่อง การติดตามท่าทางของมือจากข้อมูลกลศาสตร์ 3 มิติที่ได้จากภาพถ่าย 2 มิติ ผลการวิจัยแบ่งออกเป็น 3 ส่วน ส่วนแรกเป็นผลการประมวลผลภาพ โดยจะนำภาพถ่ายที่ได้ มาหาตำแหน่งของปลายนิ้วมือ ส่วนที่สองเป็นผลการหาตำแหน่ง 2 มิติ จากภาพ 3 มิติ จากแบบจำลองมือซึ่งได้จากโปรแกรมโรโบซิม 2 โดยจะหาตำแหน่ง 2 มิติ ของปลายนิ้ว ของแบบจำลองมือที่แสดงผลบนหน้าจอและส่วนสุดท้ายเป็นผลการติดตามท่าทางจากภาพถ่าย โดยควบคุมแบบจำลองมือให้แสดงท่าทางตามภาพถ่าย ซึ่งแสดงผลการควบคุมด้วยโปรแกรม โรโบซิม 2 ได้ผลดังนี้

4.1 ผลการประมวลผลภาพ

ผลการประมวลผลภาพสามารถแบ่งผลการวิจัยได้ 2 ส่วนคือ การตรวจจับสี และการหาตำแหน่งของสี โดยการวิจัยทั้งสองส่วนนี้ได้ใช้ภาพถ่ายของมือในลักษณะต่างๆซึ่งเป็น สัญญาณภาษามืออเมริกัน (American Sign Language: ASL)

4.1.1 การตรวจจับสี

การทดสอบผลการตรวจจับสีของระบบประมวลผลภาพได้เขียนอัลกอริทึมการตรวจจับสี ด้วยโปรแกรมแมทแล็บ (Matlab) ท่าทางที่ใช้ในการการประมวลผลภาพได้ใช้สัญญาณภาษามือ อเมริกันโดยใช้ท่าทางของพยัญชนะต่างๆ เช่น A, B, C และ D เป็นต้น แสดงดังภาพที่ 4-1



ก) ท่าของพยัญชนะ A



ข) ท่าของพยัญชนะ B

ภาพที่ 4-1 แสดงสัญญาณภาษามืออเมริกันของพยัญชนะ A, B, C และ D



ค) ท่าของพยัญชนะ C



ง) ท่าของพยัญชนะ D

ภาพที่ 4-1 (ต่อ) แสดงสัญญาณภาษามืออเมริกันของพยัญชนะ A, B, C และ D

ผลของโปรแกรมตรวจจับสีสามารถตรวจจับสีที่ต้องการที่บริเวณปลายนิ้วมือทั้ง 5 นิ้ว แสดงดังภาพที่ 4-2



ก) กลุ่มสีที่ปลายนิ้วของพยัญชนะ A



ข) กลุ่มสีที่ปลายนิ้วของพยัญชนะ B



ค) กลุ่มสีที่ปลายนิ้วของพยัญชนะ C

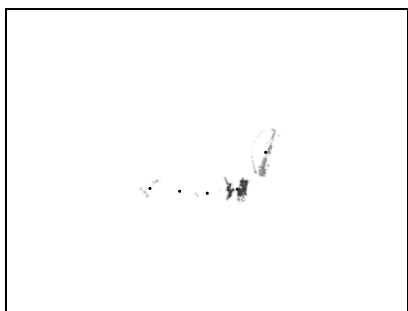


ง) กลุ่มสีที่ปลายนิ้วของพยัญชนะ D

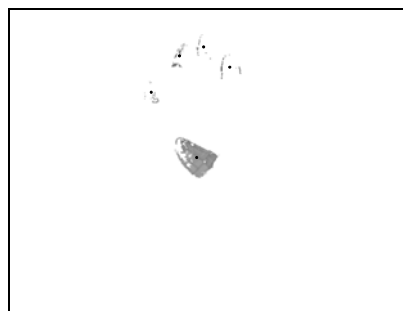
ภาพที่ 4-2 แสดงกลุ่มสีที่บริเวณปลายนิ้วของสัญญาณภาษามืออเมริกัน
ของพยัญชนะ A, B, C และ D

4.1.2 การหาตำแหน่งของสี

การทดสอบหาตำแหน่งของสีจากกลุ่มของสีที่ได้จากการตรวจจับสี ผลการทดสอบสามารถหาตำแหน่งของสัญญาณภาษามืออเมริกัน โดยใช้ท่าทางของพยัญชนะต่างๆ เช่น A, B, C และ D เป็นต้น แสดงดังภาพที่ 4-3



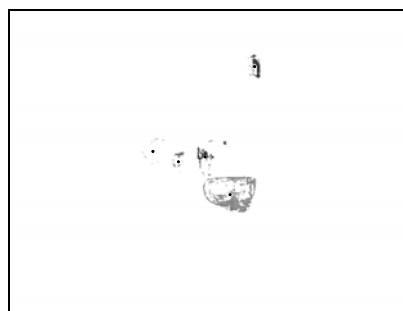
ก) ตำแหน่งสีที่ปลายนิ้วของพยัญชนะ A



ข) ตำแหน่งสีที่ปลายนิ้วของพยัญชนะ B



ค) ตำแหน่งสีที่ปลายนิ้วของพยัญชนะ C



ง) ตำแหน่งสีที่ปลายนิ้วของพยัญชนะ D

ภาพที่ 4-3 แสดงตำแหน่งสีที่บริเวณปลายนิ้วของสัญญาณภาษามืออเมริกัน
ของพยัญชนะ A, B, C และ D

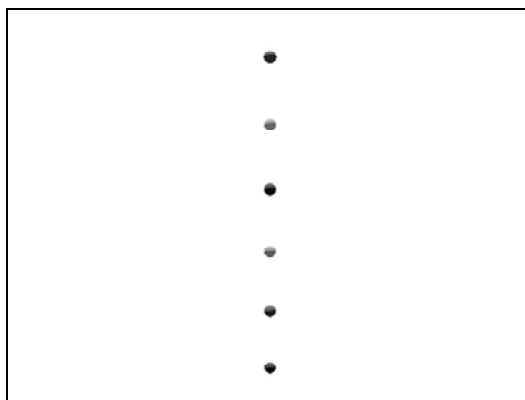
การทดสอบหาตำแหน่งของสี สามารถบอกตำแหน่งสีที่ปลายนิ้วมือทั้ง 5 นิ้วจากภาพที่ 4-3 โดยแสดงเป็นจุดสีคำอยู่บริเวณกลุ่มสีต่างๆ ส่วนตารางที่ 4-1 แสดงตำแหน่งพิกเซลของกลุ่มสีของภาพที่ 4-3 โดยบอกเป็นพิกัด XY ของพิกเซลภาพ

ตารางที่ 4-1 ตำแหน่งสี่บริเวณปลายนิ้วของภาพที่ 4-3

	นิ้วหัวแม่มือ	นิ้วชี้	นิ้วกลาง	นิ้วนาง	นิ้วก้อย
ท่าพยัญชนะ A	(411,226)	(366,286)	(319,292)	(275,290)	(227,285)
ท่าพยัญชนะ B	(298, 234)	(352,89)	(309,57)	(270,71)	(225,130)
ท่าพยัญชนะ C	(379,310)	(371,174)	(314,164)	(270,175)	(222,172)
ท่าพยัญชนะ D	(352,294)	(392,86)	(324,235)	(270,241)	(229,224)

4.2 ผลการหาดำแหน่ง 2 มิติ จากตำแหน่ง 3 มิติ

การทดสอบหาดำแหน่ง 2 มิติ จากตำแหน่ง 3 มิติ ได้ทดสอบโดยการสร้างทรงกลมสีต่างๆ ขนาด 1 หน่วย วางอยู่บนระบบ 3 มิติ ของโปรแกรมโรโบซิม 2 จำนวนทั้งสิ้น 18 ลูก และเก็บภาพจากหน้าต่างแสดงผลของโปรแกรมโรโบซิม 2 เพื่อนำไปคำนวณหาดำแหน่ง 2 มิติ ดังภาพที่ 4-4



ภาพที่ 4-4 แสดงตำแหน่งของทรงกลมจากหน้าต่างแสดงผล

จากตำแหน่งของทรงกลมในระบบ 3 มิติ สามารถหาดำแหน่งในระบบ 2 มิติ บนหน้าต่างแสดงผล เพื่อเปรียบเทียบกับตำแหน่งจริงบนหน้าต่างแสดงผลที่มีขนาดเดียวกันคือขนาด 480×640 พิกเซล โดยใช้ระบบโครงข่ายประสาทเทียมที่ผ่านการเรียนรู้แล้ว ดังกล่าวไว้ในบทที่ 3 สามารถหาดำแหน่ง 2 มิติ บนหน้าต่างแสดงผลได้ดังตารางที่ 4-2

ตารางที่ 4-2 ผลการทดสอบหาดำแหน่ง 2 มิติ จากข้อมูล 3 มิติ ด้วยระบบโครงข่ายประสาทเทียม

ทรงกลมที่ ตำแหน่ง	1	2	3	4	5	6	7	8	9
X	325	321	321	324	322	321	46	140	249
Y	441	371	298	220	140	55	443	370	298

ตารางที่ 4-2 (ต่อ)

ทรงกลมที่ ตำแหน่ง	10	11	12	13	14	15	16	17	18
X	359	139	588	36	128	233	343	466	612
Y	221	471	56	24	91	166	249	340	437

จากตำแหน่ง 2 มิติ จริงที่ปรากฏบนหน้าต่างแสดงผลเมื่อเปรียบเทียบกับตำแหน่ง 2 มิติ ที่ได้จากการประมวลผลของระบบโครงข่ายประสาทเทียม สามารถหาค่าผิดพลาดในแต่ละตำแหน่ง แสดงดังตารางที่ 4-3

ตารางที่ 4-3 แสดงค่าผิดพลาดของตำแหน่ง 2 มิติ จากระบบประสาท กับตำแหน่งจริง

ทรงกลมที่ ค่าผิดพลาด	1	2	3	4	5	6	7	8	9
X Error	-4	0	0	-3	-1	0	-1	-6	0
Y Error	2	1	-1	0	-1	0	0	2	-1

ทรงกลมที่ ค่าผิดพลาด	10	11	12	13	14	15	16	17	18
X Error	-2	0	-1	-1	0	-3	0	2	-6
Y Error	-1	0	-1	-1	0	0	0	0	6

จากตารางที่ 4-3 พบว่าระบบโครงข่ายประสาทเทียมสามารถบอกตำแหน่ง 2 มิติ โดยประมาณบนหน้าต่างแสดงผลของทรงกลมในระบบ 3 มิติ ได้

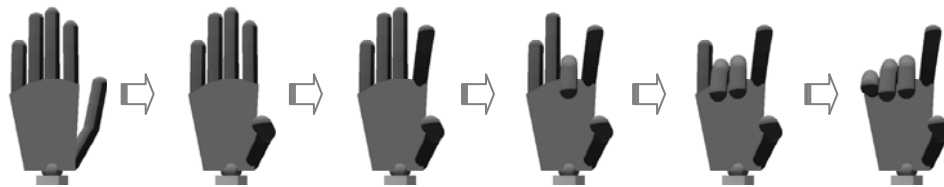
4.3 การทดสอบติดตามท่าทาง

ผลการทดสอบติดตามท่าทางของแบบจำลองมือสามารถแบ่งการทดสอบออกเป็น 2 ส่วนคือ ทดสอบควบคุมด้วยการวนซ้ำ และทดสอบควบคุมด้วยระบบฟัซซี่

4.3.1 ทดสอบควบคุมด้วยการวนซ้ำ

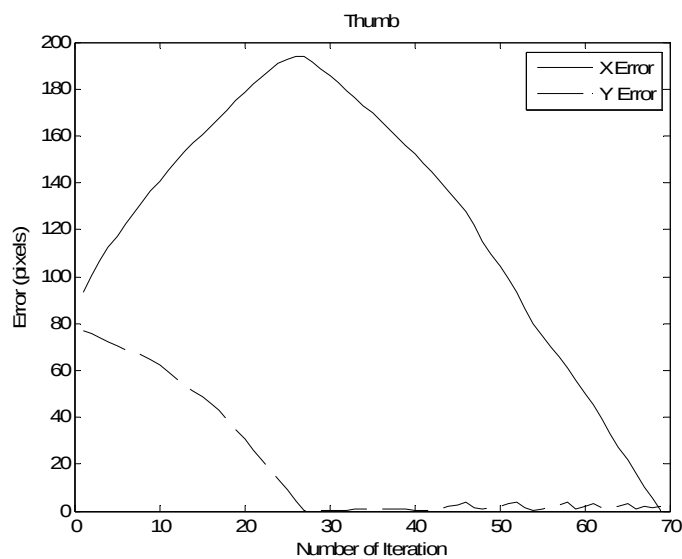
การทดสอบแบบจำลองมือด้วยโปรแกรมโรโบซิม 2 แสดงการติดตามท่าทางด้วยสัญญาณภาษามืออเมริกันโดยกำหนดให้แสดงท่าทางของพยัญชนะ D ผลของการทดสอบการติดตามท่าทางด้วยการวนซ้ำ โดยการปรับมุมหมุนที่จุดต่อในแต่ละส่วนของนิ้วมือ เริ่มจากนิ้วหัวแม่มือจนกระทั่ง

ถึงนิ้วก็ย่อตามลำดับ ขนาดของการปรับมุมหมุนให้เพิ่มหรือลดในแต่ละครั้งจะมีขนาดที่เท่ากัน และในการปรับมุมหมุนแต่ละครั้งจะตรวจจับตำแหน่งปลายนิ้วบนหน้าตาต่างแสดงผลจนตำแหน่งนั้นใกล้เคียงกับตำแหน่งบนภาพถ่าย ดังภาพที่ 4-5

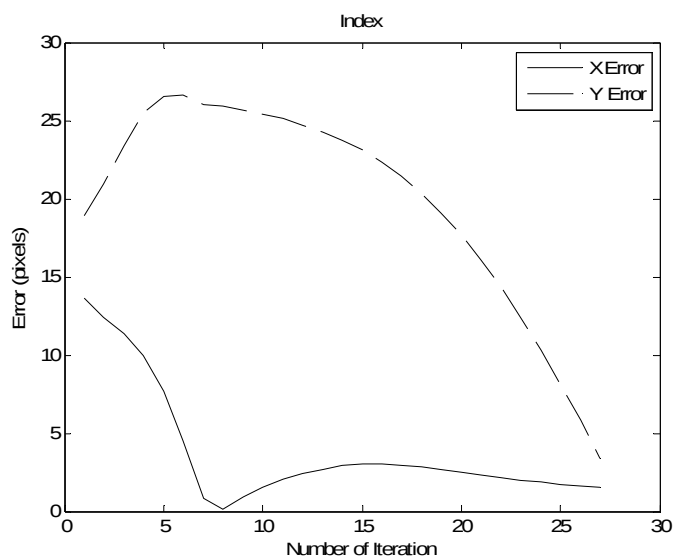


ภาพที่ 4-5 แสดงการปรับมุมหมุนที่จุดต่อของแบบจำลองมือด้วยการวนซ้ำ

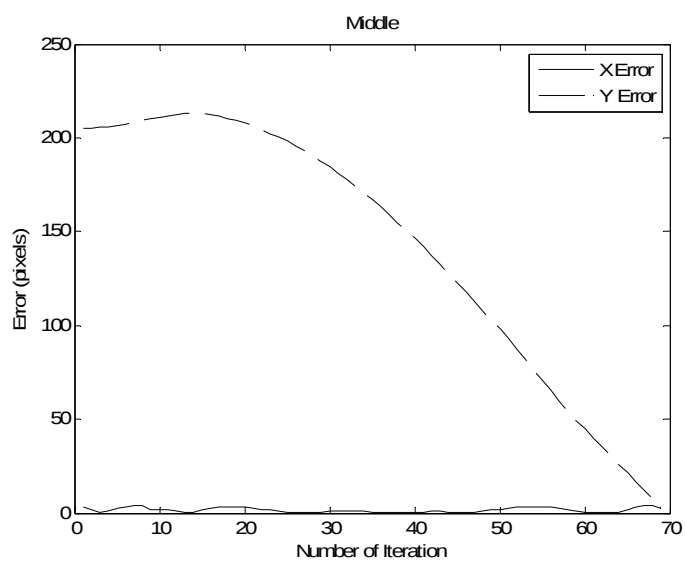
ค่าผิดพลาดในแนวแกน X และ Y ของนิ้วมือทั้ง 5 สามารถนำมาแสดงผลเป็นกราฟได้ ดังภาพที่ 4-6 ถึงภาพที่ 4-10



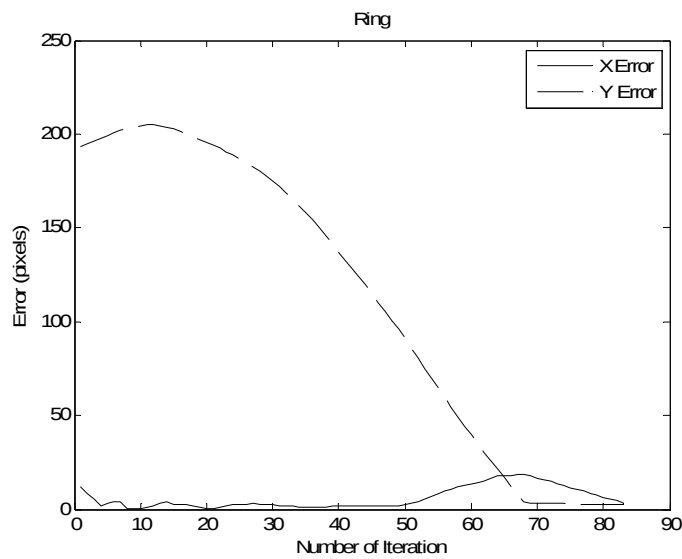
ภาพที่ 4-6 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วหัวแม่มือ



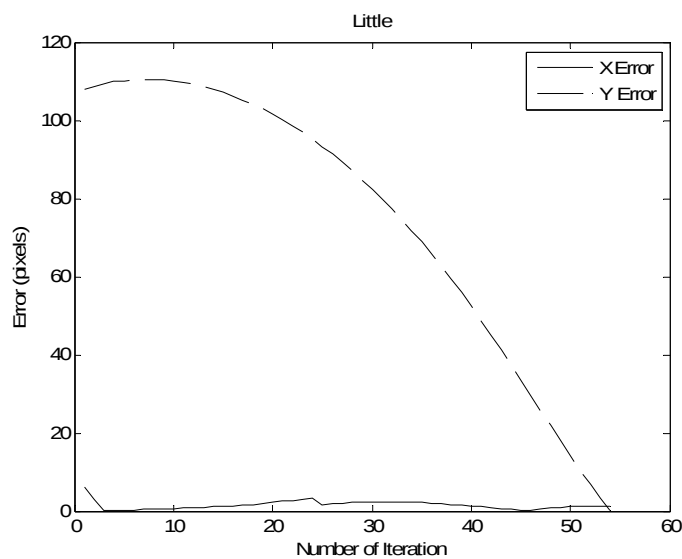
ภาพที่ 4-7 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วชี้



ภาพที่ 4-8 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วกลาง



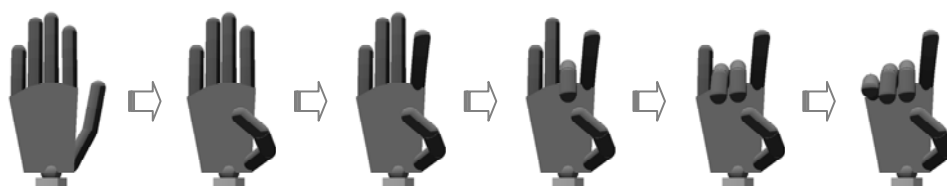
ภาพที่ 4-9 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วนาง



ภาพที่ 4-10 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วก้อย

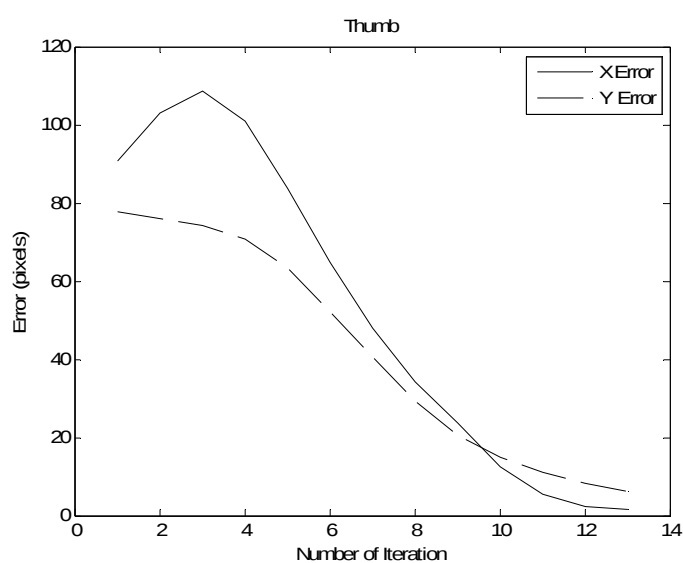
4.3.2 ทดสอบควบคุมด้วยระบบฟัซซี่

การทดสอบแบบจำลองมือด้วยโปรแกรมโรโบซิม 2 แสดงการติดตามท่าทางด้วยสัญญาณภาษามืออเมริกันโดยกำหนดให้แสดงท่าทางของพยัญชนะ D เช่นเดียวกับทดสอบด้วยการวนซ้ำ โดยการปรับมุมหมุนที่จุดต่อของนิ้วมือแต่ละนิ้วนั้น ได้ใช้ระบบฟัซซี่เข้าช่วยในการตัดสินใจเพิ่มหรือลดขนาดของการจับมุมหมุนที่จุดต่อ โดยแสดงผลการเคลื่อนไหวจากนิ้วหัวแม่มือจนถึงนิ้วก้อยตามลำดับ แสดงดังภาพที่ 4-11

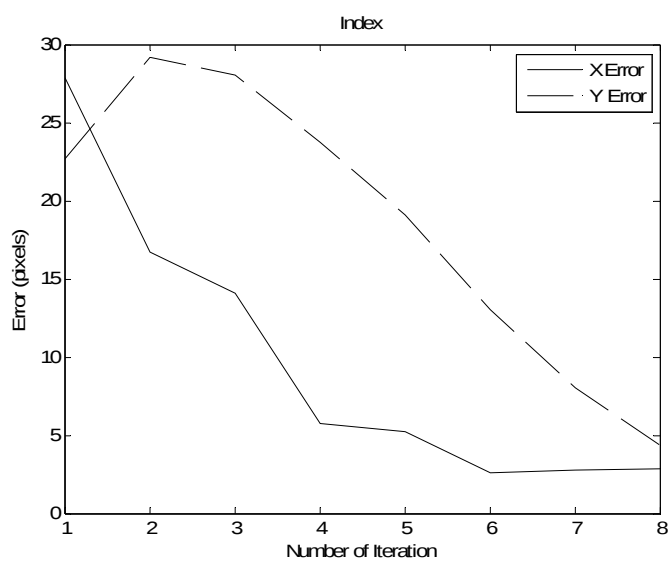


ภาพที่ 4-11 แสดงการปรับมุมหมุนที่จุดต่อของแบบจำลองมือด้วยระบบฟัซซี่

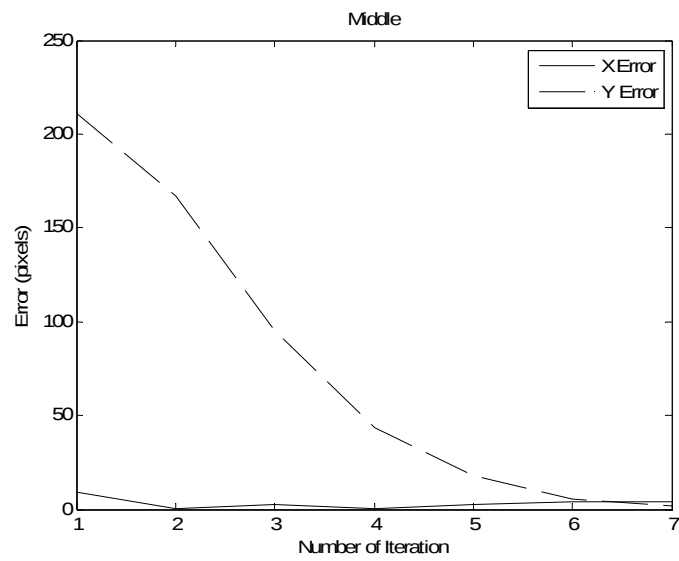
ค่าผิดพลาดในแนวแกน X และ Y ของนิ้วมือทั้ง 5 สามารถนำมาแสดงผลเป็นกราฟดังภาพที่ 4-12 ถึงภาพที่ 4-16



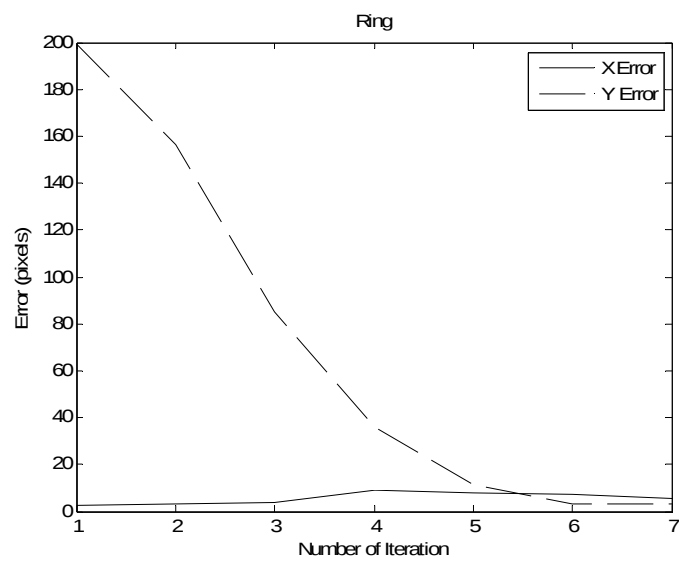
ภาพที่ 4-12 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วหัวแม่มือ



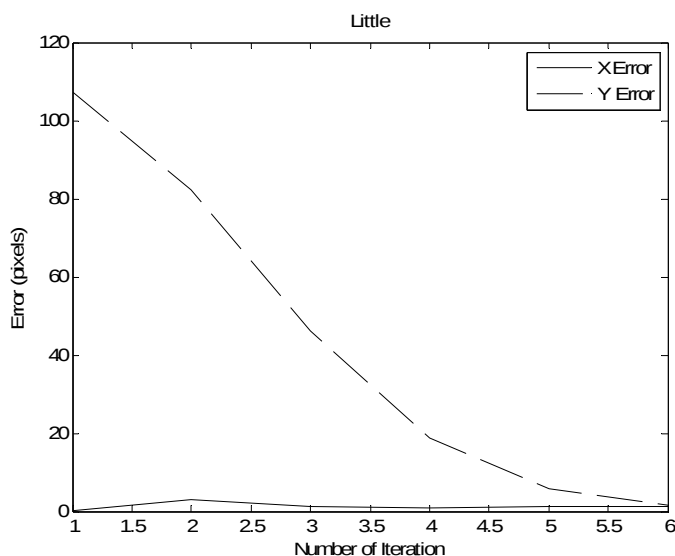
ภาพที่ 4-13 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วชี้



ภาพที่ 4-14 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิวกลาง



ภาพที่ 4-15 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิวนาง

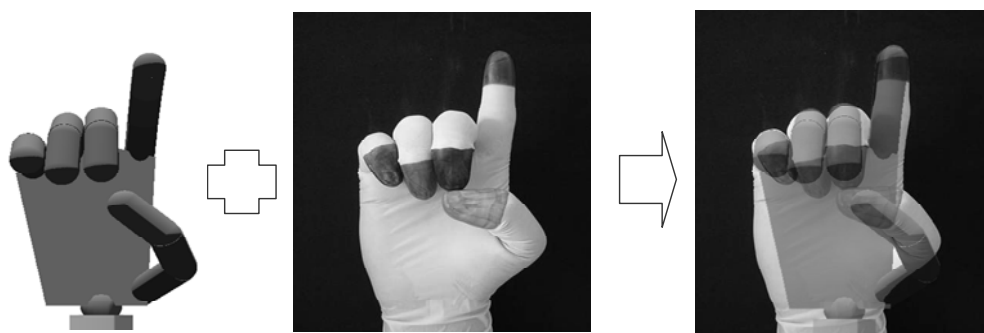


ภาพที่ 4-16 แสดงกราฟของค่าผิดพลาดในแนวแกน X และ Y ของนิ้วก้อย

สรุปผลจากการทดสอบทั้ง 2 แบบพบว่า การควบคุมด้วยระบบฟัซซี่สามารถตัดสินใจได้อย่างแม่นยำ และทำงานได้รวดเร็วกว่าการควบคุมด้วยการวนซ้ำ หากเปรียบเทียบความผิดพลาดด้วยการซ้อนกันของภาพจากทดสอบทั้ง 2 วิธี กับภาพถ่ายมือจริงพบว่า การใช้ระบบฟัซซี่สามารถทำให้แบบจำลองมือขยับปลายนิ้วเข้าสู่ตำแหน่งได้ใกล้เคียงกว่าโดยแสดงดังภาพที่ 4-17 และภาพที่ 4-18 ส่วนการใช้เวลาในการเข้าสู่ตำแหน่งของการควบคุมด้วยระบบฟัซซี่ใช้เวลาน้อยกว่า การควบคุมด้วยการวนซ้ำ ดังแสดงจากภาพที่ 4-6 ถึง 4-10 และภาพที่ 4-12 ถึง 4-16 เนื่องจากการวนซ้ำจะปรับมุมหมุนที่จุดต่อด้วยค่าของมุมที่เพิ่มและลดด้วยช่วงที่คงตลอดจึงใช้เวลานานกว่าระบบฟัซซี่ที่ไม่มีการกำหนดช่วงมุมในการปรับที่ตายตัว



ภาพที่ 4-17 แสดงการซ้อนภาพระหว่างภาพถ่าย กับภาพจากหน้าต่างแสดงผลด้วยการทดสอบควบคุมแบบวนซ้ำ



ภาพที่ 4-18 แสดงการซ้อนภาพระหว่างภาพถ่าย กับภาพจากหน้าต่างแสดงผล
ด้วยการทดสอบควบคุมด้วยระบบพีซี

จากการทดสอบควบคุมด้วยระบบพีซี สามารถติดตามท่าทางสัญญาณภาษามืออเมริกัน ในท่าทางของพยัญชนะต่างๆ ได้จำนวนทั้งสิ้น 18 พยัญชนะ คือ A, B, C, D, E, F, G, H, I, K, L, O, P, Q, U, W, X และ Y โดยจำนวนรอบในการเข้าสู่ตำแหน่ง ดังตารางที่ 4-4

ตารางที่ 4-4 แสดงจำนวนรอบในการแสดงท่าทางของสัญญาณภาษามืออเมริกันในแต่ละท่าทาง

สัญญาณภาษามืออเมริกันของพยัญชนะ									
	A	B	C	D	E	F	G	H	I
จำนวนรอบ	17	17	17	16	15	16	18	18	23

สัญญาณภาษามืออเมริกันของพยัญชนะ									
	K	L	O	P	Q	U	W	X	Y
จำนวนรอบ	21	14	15	15	16	28	16	15	14

จากตารางที่ 4-4 พบว่าจำนวนรอบในการแสดงท่าทางเฉลี่ยประมาณ 17 รอบ โดยท่าที่ใช้จำนวนรอบในการแสดงท่าทางนานที่สุดได้แก่ท่าทางของพยัญชนะ U ใช้จำนวนรอบ 28 รอบ และท่าที่ใช้จำนวนรอบในการแสดงท่าทางน้อยที่สุดได้แก่ท่าทางของพยัญชนะ L และ Y ตามลำดับ โดยใช้จำนวนรอบ 14 รอบ

ค่าผิดพลาดในการเข้าสู่ตำแหน่งในแนวแกน X และ Y ของแต่ละนิ้วในแต่ละท่าทางของสัญญาณภาษามืออเมริกัน แสดงดังตารางที่ 4-5

ตารางที่ 4-5 แสดงค่าผิดพลาดในการเข้าสู่ตำแหน่งในแนวแกน X และ Y ของสัญญาณภาษามืออเมริกันในแต่ละท่าทาง

		สัญญาณภาษามืออเมริกันของพยัญชนะ								
		A	B	C	D	E	F	G	H	I
นิ้วหัวแม่มือ	X Error (pixels)	3	4	1	1	3	3	3	4	3
	Y Error (pixels)	4	4	3	3	2	1	1	5	4
นิ้วชี้	X Error (pixels)	2	3	0	4	5	13	3	3	0
	Y Error (pixels)	3	2	4	3	0	0	2	2	2
นิ้วกลาง	X Error (pixels)	1	3	3	4	4	3	3	2	0
	Y Error (pixels)	3	3	1	2	2	2	3	2	4
นิ้วนาง	X Error (pixels)	1	2	2	4	3	0	3	2	1
	Y Error (pixels)	3	3	1	4	4	4	1	3	4
นิ้วก้อย	X Error (pixels)	1	3	3	1	2	8	14	3	5
	Y Error (pixels)	3	3	2	1	1	16	3	2	2

		สัญญาณภาษามืออเมริกันของพยัญชนะ								
		K	L	O	P	Q	U	W	X	Y
นิ้วหัวแม่มือ	X Error (pixels)	3	5	1	4	3	3	5	3	16
	Y Error (pixels)	4	7	4	1	3	5	8	4	4
นิ้วชี้	X Error (pixels)	2	1	4	1	2	0	9	0	3
	Y Error (pixels)	11	3	1	1	1	4	3	2	3
นิ้วกลาง	X Error (pixels)	3	1	3	3	2	3	3	1	1
	Y Error (pixels)	4	3	2	1	20	3	3	3	24
นิ้วนาง	X Error (pixels)	3	4	2	17	3	0	0	3	0
	Y Error (pixels)	2	3	4	75	4	4	3	2	22
นิ้วก้อย	X Error (pixels)	9	3	1	20	3	9	9	4	21
	Y Error (pixels)	4	2	2	23	2	4	3	2	28

จากตารางที่ 4-5 สามารถหาค่าเฉลี่ยของค่าผิดพลาดกำลังสอง (Mean Square Error: MSE) เพื่อแสดงความสามารถในการเข้าสู่ตำแหน่งของสัญญาณภาษามืออเมริกันในแต่ละท่าทาง แสดงดังตารางที่ 4-6

ตารางที่ 4-6 แสดงค่าเฉลี่ยของค่าผิดพลาดกำลังสองในการเข้าสู่ตำแหน่งของสัญญาณภาษามืออเมริกันในแต่ละท่าทาง

สัญญาณภาษามืออเมริกันของพยัญชนะ									
	A	B	C	D	E	F	G	H	I
Mean Square Error	6.8	9.4	5.4	9.8	8.8	52.8	25.6	8.8	9.1

สัญญาณภาษามืออเมริกันของพยัญชนะ									
	K	L	O	P	Q	U	W	X	Y
Mean Square Error	28.5	13.2	7.2	687.2	46.5	18.1	29.6	7.2	257.6

จากตารางที่ 4-6 พบว่าสัญญาณภาษามืออเมริกันของพยัญชนะ P มีความผิดพลาดในการเข้าสู่ตำแหน่งสูงที่สุดโดยมีค่าเฉลี่ยของค่าผิดพลาดกำลังสอง (Mean Square Error) เท่ากับ 687.2 สัญญาณภาษามืออเมริกันที่มีค่าความผิดพลาดในการเข้าสู่ตำแหน่งต่ำที่สุดคือสัญญาณภาษามืออเมริกันของพยัญชนะ C โดยมีค่าเฉลี่ยของค่าผิดพลาดกำลังสอง (Mean Square Error) เท่ากับ 5.4

ตัวอย่างการติดตามท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะต่างๆ ได้แก่ A, B, C, D, E และ F เป็นต้น แสดงดังภาพที่ 4-19 ถึงภาพที่ 4-24



ก) ภาพถ่ายมือจริง



ข) แบบจำลองจาก
มุมมองที่ [0 155 0]

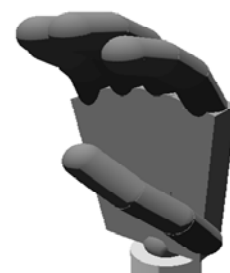


ค) แบบจำลองจาก
มุมมองที่ [-70 100 90]

ภาพที่ 4-19 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ A



ก) ภาพถ่ายมือจริง

ข) แบบจำลองจาก
มุมมองที่ [0 155 0]ค) แบบจำลองจาก
มุมมองที่ [-70 100 90]

ภาพที่ 4-20 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ B



ก) ภาพถ่ายมือจริง

ข) แบบจำลองจาก
มุมมองที่ [0 155 0]ค) แบบจำลองจาก
มุมมองที่ [-70 100 90]

ภาพที่ 4-21 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ C



ก) ภาพถ่ายมือจริง

ข) แบบจำลองจาก
มุมมองที่ [0 155 0]ค) แบบจำลองจาก
มุมมองที่ [-70 100 90]

ภาพที่ 4-22 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ D



ก) ภาพถ่ายมือจริง



ข) แบบจำลองจาก
มุมมองที่ [0 155 0]



ค) แบบจำลองจาก
มุมมองที่ [-70 100 90]

ภาพที่ 4-23 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ E



ก) ภาพถ่ายมือจริง



ข) แบบจำลองจาก
มุมมองที่ [0 155 0]



ค) แบบจำลองจาก
มุมมองที่ [-70 100 90]

ภาพที่ 4-24 แสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ F

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

วิทยานิพนธ์เรื่องการติดตามท่าทางของมือจากข้อมูลกลศาสตร์ 3 มิติ ที่ได้จากภาพถ่าย 2 มิติ ได้ออกแบบและสร้างแบบจำลองมือ ให้แบบจำลองมือแสดงท่าทางตามภาพถ่ายการทำงานได้อาศัยหลักการพื้นฐานจากถุงมือและการจำลองแบบมาประยุกต์ร่วมกัน ได้ใช้ภาพถ่ายฝ่ามือข้างขวา ซึ่งสวมถุงมือทำสัญลักษณ์ด้วยสีที่ปลายนิ้ว แบบจำลองมือถูกสร้างโดยโปรแกรมจำลองหุ่นยนต์ 3 มิติ ที่มีชื่อว่า โรโบซิม 2 (Robosim 2) และควบคุมการขยับปลายนิ้วเข้าสู่ตำแหน่งที่ต้องการด้วยระบบฟัซซี่โดยเขียนอัลกอริทึมในการควบคุมภายใต้โปรแกรมแมทแลป (Matlab)

จากการทดสอบได้ใช้สัญญาณภาษามืออเมริกัน (American Sign Language: ASL) ของพยัญชนะต่างๆจำนวนทั้งสิ้น 24 ท่าทางด้วยกัน และใช้เงื่อนไขหรือข้อกำหนดในการขยับมุมหมุนที่ข้อต่อของนิ้ว (Constraint on Joints) ร่วมด้วยพบว่าสามารถแสดงสัญญาณภาษามืออเมริกันของพยัญชนะต่างๆได้ทั้งสิ้น 18 ท่าทางได้แก่ พยัญชนะ A, B, C, D, E, F, G, H, I, K, L, O, P, Q, U, W, X และ Y จำนวนรอบในการแสดงท่าทางเฉลี่ยประมาณ 17 รอบ ท่าทางของพยัญชนะ U ใช้จำนวนรอบมากที่สุด 28 รอบ และท่าทางของพยัญชนะ L และ Y ใช้จำนวนรอบน้อยที่สุด 14 รอบ การเข้าสู่ตำแหน่งของนิ้วทั้ง 5 ทั้งในแนวแกน X และ Y ได้กำหนดค่าผิดพลาดไม่เกินกว่า 4 พิกเซล ความสามารถในการเข้าสู่ตำแหน่งของสัญญาณภาษามืออเมริกันที่ดีที่สุดได้แก่ท่าทางของพยัญชนะ C โดยมีค่าเฉลี่ยของค่าผิดพลาดกำลังสอง (Mean Square Error) เท่ากับ 5.4

5.2 ข้อดีของงานวิจัย

5.2.1 งานวิจัยนี้ได้ใช้หลักการพื้นฐานจากถุงมือโดยการทำสัญลักษณ์ด้วยสีที่ปลายนิ้ว เพื่อความสะดวกต่อการตรวจจับตำแหน่งสีสำหรับการประมวลภาพ และหาตำแหน่งของปลายนิ้ว

5.2.2 การขยับมุมหมุนที่ข้อต่อของนิ้วได้ใช้เงื่อนไข หรือข้อกำหนดในการขยับมุมหมุนที่ข้อต่อของนิ้วเข้าช่วย เพื่อลดการคำนวณการขยับมุมหมุนที่ข้อต่อนี้จากจำนวน 21 DOFs เหลือ 10 DOFs

5.2.3 การโอนถ่ายตำแหน่ง 3 มิติ จากโปรแกรมโรโบซิม 2 เป็นตำแหน่ง 2 มิติ ที่ปรากฏบนหน้าตาแสดงผล ใช้ระบบโครงข่ายประสาทเทียมในการจดจำตำแหน่งแทนการคำนวณ เพราะหากมีการเปลี่ยนมุมมองของระบบ 3 มิติ ระบบโครงข่ายประสาทเทียมยังสามารถจดจำตำแหน่งระหว่างระบบทั้งสองได้ โดยไม่ต้องฝึกฝนหรือคำนวณใหม่

5.2.4 ใช้ภาพถ่ายเพียงภาพเดียวในการติดตามท่าทาง

5.3 ข้อบกพร่องของงานวิจัย

5.3.1 ข้อมูลจากภาพถ่ายที่นำมาใช้ในการประมวลผลบางภาพไม่สามารถประมวลผลเพื่อหาตำแหน่งของสีที่ปลายนิ้วได้ เนื่องจากความชัดเจน ค่าแสง และสภาพแวดล้อม ส่งผลต่อความผิดพลาดของข้อมูลภาพ

5.3.2 ยังไม่สามารถติดตามท่าทางที่มีการบังคับของสีที่ปลายนิ้วได้ เนื่องจากไม่สามารถหาตำแหน่งของสีที่ถูกบังคับได้

5.3.3 ยังไม่สามารถทำการติดตามท่าทางที่มีลักษณะของนิ้วไขว้กันหรือลักษณะที่มีการฝืนของกล้ามเนื้อได้ เนื่องจากเงื่อนไขหรือข้อกำหนดในการขยับมุมหมุนที่ข้อต่อของนิ้วไม่เหมาะสมกับท่าทางที่มีลักษณะที่มีการฝืนของกล้ามเนื้อ

5.4 ข้อเสนอแนะสำหรับการพัฒนา

5.4.1 ควรมีกระบวนการปรับปรุงข้อมูลของภาพให้มีความชัดเจนก่อนเข้าสู่กระบวนการประมวลผลภาพ

5.4.2 ออกแบบวิธีการประมาณตำแหน่งสีที่ปลายนิ้วมือที่ถูกบังคับ สำหรับกรณีการแสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ S และ V

5.4.3 ควรใช้วิธีการโปรเจกชันแบบมีเงื่อนไข สำหรับลักษณะท่าทางของนิ้วไขว้กันหรือทับซ้อนกัน ในกรณีการแสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ M N R และ T

5.4.4 ควรมีการพัฒนาโปรแกรมให้สามารถแสดงท่าทางแบบเคลื่อนไหว ในกรณีการแสดงท่าทางสัญญาณภาษามืออเมริกันของพยัญชนะ J และ Z

บรรณานุกรม

ภาษาไทย

ณชล ไชยรัตน์. “วิศวกรรมหุ่นยนต์.” เอกสารประกอบการสอนรายวิชา. กรุงเทพมหานคร :

ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ ศูนย์วิจัยและพัฒนาาระบบอัจฉริยะ
สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2545. (อัดสำเนา)

พยุ่ง มีลัจ. “พีชชี และโครงข่ายประสาทเทียม.” เอกสารประกอบการสอน. กรุงเทพมหานคร :

คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2548.
(อัดสำเนา)

พิสุทธิ จันทรัชชชนะกุล. การค้นหาเป้าหมายแบบอัจฉริยะโดยไม่ทราบข้อมูลสภาพแวดล้อมของ

แขนกลแบบข้อต่อยาว. วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า บัณฑิตวิทยาลัย สถาบันเทคโนโลยี
พระจอมเกล้าพระนครเหนือ, 2548.

สถาพร ลักษณะเจริญ. วิศวกรรมหุ่นยนต์. พิมพ์ครั้งที่ 1. กรุงเทพมหานคร : สมาคมส่งเสริม

เทคโนโลยี(ไทย-ญี่ปุ่น), 2548.

สะไกร ทองแสนสุข, สมพงษ์ เกิดคำ. ระบบควบคุมการขับเคลื่อนของรถอัตโนมัติโดยใช้วิธีการ

ประมวลผลทางภาพ. ปรินญาณพนธ์ วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมไฟฟ้า ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2548.

ภาษาอังกฤษ

Chin-Teng Lin and C. S. George Lee. **Neural Fuzzy Systems**. Upper Saddle River : Prentice-Hall, 1996.

Chua, C. S., Guan H. and Ho Y. K. “Model Based 3D Hand Posture Estimation from a Single 2D Image.” **Image and Vision Computing**. 20, (2002) : 191-202.

Haiying Guan and Matthew Turk. **3D Hand Pose Reconstruction with ISOSOM**. Technical Reports, Department of Computer Science, University of California, Santa Barbara, 2005.

- Jingzhou, Y. and Esteban P. P. **Kinematics Human Model**. Technical Reports, Virtual Soldier Research Program Center for Computer-Aided Design College of Engineering, University of Iowa, 2004.
- Koivo, Antti J. **Fundamentals for Control of Robotic Manipulators**. New York : John Wiley & Sons, Inc., 1989.
- Koseeyaporn, P., Cook G. E. and Strauss A. M. "Extendible Simulation Package for Robotics Research and Instruction." **IASTED International Conference Modeling and Simulation**. (2002) : 256-263.
- Kuch, J. J. and Huang T. S. "Vision Based Hand Modeling and Tracking for Virtual Teleconference and Telecollaboration." **ICCV**. (1995) : 666-671.
- Lee, J. and Kunil T. L. "Model-based analysis of hand posture." **IEEE Computer Graphics Applications**. (Sep. 1995) : 77-86.
- Lien, C. C. and Huang C. L. "Model-based articulated hand motion tracking for gesture recognition." **Image and Vision Computing**. 16, (1998) : 121-134.
- Schilling, R. **Fundamentals of Robotics Analysis and Control**. New Jersey : Prentice-Hall, Inc., 1990.
- Sturman, D. J. and Zeltzer D. "A Survey of Glove Based Input." **IEEE Computer Graphics Applications**. (Jan. 1994) : 30-39.

ประวัติผู้วิจัย

ชื่อ : นายไกรวุฒิ หลีกคำ

ชื่อวิทยานิพนธ์ : การติดตามท่าทางของมือจากข้อมูลกลศาสตร์ 3 มิติที่ได้จากภาพถ่าย 2 มิติ

สาขาวิชา : วิศวกรรมไฟฟ้า

ประวัติ

ประวัติส่วนตัว เกิดเมื่อวันที่ 4 มกราคม พ.ศ. 2522 ที่จังหวัดกรุงเทพฯ เป็นบุตรคนแรก ในจำนวนพี่น้อง 3 คน ปัจจุบันอาศัยอยู่บ้านเลขที่ 211/132 หมู่ 9 ถ.ติวานนท์ ต.บางกระสอ อ.เมือง จ.นนทบุรี 11000

ประวัติการศึกษา สำเร็จการศึกษาระดับประกาศนียบัตรวิชาชีพ สาขาช่างไฟฟ้า จากโรงเรียนช่างฝีมือทหาร เมื่อปีการศึกษา 2539 สำเร็จการศึกษาระดับปริญญาตรี หลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิศวกรรมไฟฟ้า ภาควิชาครุศาสตร์ไฟฟ้า คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ ในปีการศึกษา 2544 ขณะศึกษาในระดับปริญญาตรี ได้จัดทำวิทยานิพนธ์เรื่อง “ชุดสาธิตยกลูกเหล็กด้วยสนามแม่เหล็ก”

ประวัติการทำงาน เริ่มทำงานเมื่อ พ.ศ. 2545 ตำแหน่งวิศวกรไฟฟ้า บริษัท ชิสการ์ด จำกัด เป็นเวลา 1 ปี และในปี พ.ศ. 2547 ถึงปัจจุบันทำงานเป็นอาจารย์ประจำแผนกช่างไฟฟ้า ภาควิชาไฟฟ้าและอิเล็กทรอนิกส์ โรงเรียนเทคโนโลยีสยาม