

ระบบช่วยเดินโดยใช้ชุด LOWER LIMB EXOSKELETON สั่งการโดยใช้
BACKPROPAGATION NEURAL NETWORK

THE WALKING ASSISTANCE SYSTEM USING THE LOWER LIMB
EXOSKELETON SUIT COMMANDED BY BACKPROPAGATION NEURAL
NETWORK

โอบนิธิ กรัณฑรัตน์

OBNITHI KARANTARAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2562

KMITL-2019-EN-M-040-022

ระบบช่วยเดินโดยใช้ชุด LOWER LIMB EXOSKELETON สั่งการโดยใช้
BACKPROPAGATION NEURAL NETWORK

THE WALKING ASSISTANCE SYSTEM USING THE LOWER LIMB
EXOSKELETON SUIT COMMANDED BY BACKPROPAGATION NEURAL
NETWORK

โอบนิธิ กรัณท์รัตน

OBNITHI KARANTARAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2562

KMITL-2019-EN-M-040-022

THE WALKING ASSISTANCE SYSTEM USING THE LOWER LIMB
EXOSKELETON SUIT COMMANDED BY BACKPROPAGATION NEURAL
NETWORK

OBNITHI KARANTARAT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRONICS ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2019

KMITL-2019-EN-M-040-022

COPYRIGHT 2019

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

หัวข้อวิทยานิพนธ์	ระบบช่วยเดินโดยใช้ชุด Lower Limb Exoskeleton สั่งการโดยใช้ Backpropagation Neural network
นักศึกษา	นายโอบนิธิ กรัณทรัตน
รหัสประจำตัว	60601068
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์
พ.ศ.	2562
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.ยุทธนา คิดใจเดียว

บทคัดย่อ

ในปัจจุบันมีผู้คนจำนวนมากที่มีปัญหาในการเดิน ในงานวิจัยนี้มีเป้าหมายในการพัฒนาและแก้ปัญหาเรื่องการเดิน โดยการแนะนำระบบที่ช่วยในการเดินที่สามารถจดจำท่าทางได้ 3 ท่า คือ ท่าเดิน, ท่านั่ง และท่านอน ระบบที่ช่วยในการเดินนี้ประกอบด้วย 3 ส่วนหลัก คือ การจำแนกท่าทางการเดิน, การฝึกสอนท่าทางให้กับชุด Exoskeleton และ Exoskeleton suit system ในการจำแนกลักษณะของสัญญาณ sEMG ในงานวิจัยนี้ใช้ Conjugate Gradient Backpropagation Neural Network โดยจะทำการจำแนกหลังจากผ่านกระบวนการ Feature extraction แล้ว หลังจากนั้นเอาที่พุดที่ได้จากการจำแนกของ Backpropagation Neural Network จะถูกนำไปใช้เพื่อสั่งให้ชุดเคลื่อนที่ในท่าทางตามที่ชุดได้จดจำไว้ โดยวิธีการจดจำท่าทางของชุดที่ได้เสนอในงานวิจัยนี้นั้นมี 2 วิธี คือ การโปรแกรมท่าทางไว้ล่วงหน้า และการใช้ Levenberg-Marquardt backpropagation เมื่อได้ฝึกสอนท่าทางต่างๆให้กับชุดแล้ว ก็สามารถให้ชุดนำท่าทางที่เรียนรู้ออกมาใช้งานได้ นอกจากนี้ในงานวิจัยนี้ยังได้ใช้ PID controller เพื่อควบคุมมอเตอร์กระแสตรง และการออกแบบกลไกแบบ Four Bar Linkages Mechanisms ที่สามารถลดจำนวนมอเตอร์ที่ใช้งานและเพิ่มเสถียรภาพในช่วง Stance Phase ของการเดินอีกด้วย จากผลการทดลอง สรุปได้ว่า วิธีการ Levenberg-Marquardt Backpropagation เป็นวิธีการที่สอนท่าทางให้กับชุดแล้วมีความคลาดเคลื่อนน้อยที่สุด คือ มีค่าเฉลี่ยของความคลาดเคลื่อนอยู่ที่ 4.418 ในขณะที่วิธีการ Curve fitting ทั่วไปมีความคลาดเคลื่อน 12.71 และสำหรับการจำแนกท่าทาง การใช้ feature ทั้งหมดใน time domain จะทำให้ recognition rate สูงถึง 99.39%

Thesis	The Walking Assistance System using The Lower Limb Exoskeleton Suit Commanded by Backpropagation Neural Network
Student	Mr. Obnithi Karantarat
Student ID.	60601068
Degree	Master of Engineering
Program	Electronics Engineering
Year	2019
Thesis Advisor	Asst. Prof. Dr. Yuttana Kitjaidure

ABSTRACT

Currently there are many people who have walking problems. This research aims to develop and solve these problems by introducing walking assistance system which can recognize 3 types of gestures, that include walking, sitting and standing. Our system is divided into 3 main parts including classification, posture training for the exoskeleton and exoskeleton suit systems. Conjugate Gradient Backpropagation Neural Networks have been used to classify sEMG signals of lower limb postures after extracting the features. Then the output of classification has been used to command the exoskeleton suit to perform the gesture according to the results of the recognition. This research aims to present 2 methods for recognizing the suit gesture: fixed programmed gestures and training gestures. In addition, our work uses PID controller to control the DC motors. In order to reduce the number of motors and increase stability of the system, the of Four Bar Linkages Mechanisms of Lower Limb Exoskeleton suit are used. The results from the training gesture method have shown that the Levenberg-Marquardt method provides the lowest average error of 4.418, while the general method of curve fitting gives 12.71. The results from classify sEMG signals of lower limb postures show that all features in time domain provided the highest recognition rate at 99.39%.

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงได้ด้วยดี ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ผศ.ดร.ยุทธนา คิทธิง
เดี่ยว ที่ถ่ายทอดความรู้และประสบการณ์ ให้คำปรึกษา คำแนะนำ ตลอดจนชี้แนะแนวทางสำหรับ
การแก้ปัญหาต่างๆให้กับข้าพเจ้า

ขอขอบคุณการสนับสนุนทุนการศึกษาในการเรียนระดับปริญญาโท จาก รศ.ดร.คมสัน มาลีสี
คณบดี คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ขอขอบคุณนางสาวพิมพ์พร เหมยน้อย และนายอัฒนา แซงโตะ ที่ให้แนวคิดและมุมมองในการทำ
วิจัย ขอขอบคุณนางสาวอริยา ทองทวี ที่ช่วยตรวจทานการใช้ภาษาอังกฤษของข้าพเจ้า

และสุดท้ายขอขอบคุณครอบครัวของข้าพเจ้า ทั้งบิดา มารดา คุณย่า และน้องสาวสำหรับการ
สนับสนุนในทุกด้าน

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์นี้ ข้าพเจ้าขอมอบให้บิดา มารดา ซึ่งเป็นที่รัก
และเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้และถ่ายทอด
ประสบการณ์ที่ดีให้แก่ข้าพเจ้า หากมีความผิดพลาดประการใดในวิทยานิพนธ์เล่มนี้ ข้าพเจ้าขอน้อมรับ
แต่เพียงผู้เดียว

โอบนิธิ ภัทรรัตน์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์.....	1
1.3 สมมติฐานของการศึกษา.....	1
1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย.....	1
1.5 ขอบเขตของการศึกษา.....	1
1.6 โครงสร้างของวิทยานิพนธ์.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง.....	3
2.1 วรรณกรรมหรืองานวิจัยที่เกี่ยวข้อง (Literature review).....	3
2.2 Gait analysis.....	3
2.3 PID Controller.....	5
2.4 สัญญาณ Surface Electromyography (sEMG) และกล้ามเนื้อส่วนล่าง.....	6
2.5 Feature Extraction.....	8
2.6 การจำแนกท่าทางโดยใช้ Backpropagation Neural Network.....	9
2.7 Levenberg-Marquardt backpropagation neural network (LM-BPNN).....	10
บทที่ 3 วิธีดำเนินงานวิจัย.....	12
3.1 Lower Limb Exoskeleton.....	12
3.2 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง.....	15
3.2.1 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง แบบที่ไม่มีโหลดทางกล.....	15
3.2.2 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง แบบที่มีโหลดทางกล.....	16

3.3 การทำงานของ Backpropagation neural networks เพื่อใช้ในการจำแนกท่าทาง.....	16
3.4 การสอนท่าทางให้ชุด Exoskeleton ในแต่ละข้อต่อ.....	17
3.4.1 วิธีการ Curve fitting.....	18
3.4.1.1 วิธีการ Fitting Gaussian Model.....	18
3.4.1.2 วิธีการ Sum of Sine.....	18
3.4.1.3 การประเมินความถูกต้องของ Curve fitting.....	19
3.4.2 วิธีการ Backpropagation Neuron Network แบบ Levenberg-Marquardt.....	20
บทที่ 4 การทดลองและผลการทดลอง	21
4.1 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง	21
4.1.1 การทดลองกับมอเตอร์ที่ไม่มีโหลดทางกล.....	21
4.1.2 การทดลองกับมอเตอร์ที่มีโหลดทางกล.....	22
4.2 Backpropagation Neural Network และ Walking Assistance system สำหรับการจำแนกท่าทาง	25
4.3 การสอนท่าทางให้ชุด Exoskeleton ในแต่ละข้อต่อในตัวควบคุมการเคลื่อนที่.....	30
4.3.1 วิธีการ Curve Fitting	31
4.3.2 วิธีการ Sum of Sine.....	33
4.3.3 Backpropagation Neuron Network แบบ Levenberg-Marquardt	34
4.4 การนำ Fitting Backpropagation Neuron Network ไปใช้งานกับชุด Lower Limb Exoskeleton.....	40
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ.....	43
เอกสารอ้างอิง	45
ภาคผนวก ก. คำสั่งในการเขียนโปรแกรม MATLAB.....	47
ภาคผนวก ข. ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่.....	49
ประวัติผู้เขียน.....	62

สารบัญตาราง

ตารางที่	หน้า
3.1 เปรียบเทียบช่วงมุมมองศา (Degree) สูงสุดและต่ำสุดใน 1 รอบการเดินของ ข้อต่อส่วนต่างๆ ระหว่างมนุษย์กับชุด Exoskeleton ที่ออกแบบ	13
4.1 เปอร์เซ็นต์การตอบถูกของการจำแนกท่าทาง (Percent Correct Classification).....	27
4.2 ค่าสัมประสิทธิ์ของสมการ Gaussian Model 8	32
4.3 ค่าสัมประสิทธิ์ของสมการ Sum of Sine.....	33
4.4 เปรียบเทียบค่า Error ของ Curve Fitting และ Fitting Backpropagation Neuron Network สำหรับมุมมองศาของหัวเข่าในท่าทางการเดิน.....	39

สารบัญรูป

รูปที่	หน้า
2.1 Gait Analysis	4
2.2 Joint angle of hip, knee and ankle.....	4
2.3 บล็อกไดอะแกรมตัวควบคุม PID ของ Exoskeleton actuator	5
2.4 สัญญาณ sEMG จากกล้ามเนื้อ rectus femoris เมื่อมนุษย์กำลังเดิน	6
2.5 กล้ามเนื้อต้นขาด้านหน้า.....	7
2.6 กล้ามเนื้อต้นขาด้านหลัง	7
2.7 feed forward neural network.....	9
3.1 Exoskeleton system diagram	12
3.2 การออกแบบกลไก Four bar linkages mechanisms ของ Lower limb exoskeleton suit.....	13
3.3 มุมอ้างอิงของขามนุษย์	14
3.4 ข้อมูลสัดส่วนร่างกายมนุษย์ (Anthropometric data) ของคนที่กำลังยืน	14
3.5 มอเตอร์กระแสตรงที่ต่อกับโพลดทางกล	15
3.6 Block diagram การทำงานของ Backpropagation neural networks เพื่อใช้ในการจำแนกท่าทาง.....	16
3.7 บล็อกไดอะแกรมของตัวควบคุมการเคลื่อนที่ (Trajectory controllers) ในช่วงฝึกสอน และช่วงใช้งาน	17
4.1 การทดลองการควบคุม PID โดยใช้สมการมอเตอร์ (3.4)	21
4.2 กราฟเปรียบเทียบระหว่างผลการตอบสนองของแบบจำลองมอเตอร์กระแสตรงที่ไม่ใช่ PID และกราฟผลการตอบสนองของแบบจำลองมอเตอร์กระแสตรงที่ใช้ PID	21
4.3 การใช้โมเดลมอเตอร์ทางคณิตศาสตร์ โดยใช้สมการ (3.6) กับตัวควบคุม PID.....	22
4.4 ผลการตอบสนองของ DC motor model ส่วนสะพาน โดยใช้ตัวควบคุม PID.....	22
4.5 ผลการตอบสนองของ DC motor model ที่ข้อเท้า โดยใช้ตัวควบคุม PID	23
4.6 DC motor ที่ทดลองใน Simulink	23
4.7 ผลการตอบสนองทางเวลาของแบบจำลองมอเตอร์กระแสตรงที่สะพานโดยใช้ตัวควบคุมPID	24
4.8 ผลการตอบสนองทางเวลาของแบบจำลองมอเตอร์กระแสตรงที่ข้อเท้าโดยใช้ตัวควบคุมPID	24

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.9 ชุด Exoskeleton ขยับตามที่ได้กำหนดค่าไว้.....	25
4.10 Backpropagation neural network ใน MATLAB.....	25
4.11 Confusion Matrix ของตัวจำแนกท่าทาง Backpropagation Neural Network.....	26
4.12 Exoskeleton และ Neural Network ที่ทดลองใน Simulink.....	28
4.13 Exoskeleton และ BP Neural Network ที่ทำการทดลองใน Simulink แบบที่ใช้มอเตอร์สองตัว สำหรับกลไกแบบ Four Bar Linkages	28
4.14 Error Histogram ของ Classification Backpropagation Neural Network.....	29
4.15 กราฟ Performance ของตัวจำแนกท่าทาง Backpropagation Neural Network.....	29
4.16 Training state ของตัวจำแนกท่าทาง Backpropagation Neural Network	30
4.17 มุมองศาของหัวเข้าในการเดินแบบต่อเนื่อง.....	31
4.18 เอาท์พุทของวิธีการ Curve Fitting แบบ Gaussian Model เทียบกับมุมมองศาของหัวเข้าในการเดินแบบต่อเนื่อง.....	32
4.19 เอาท์พุทของวิธีการ Curve Fitting แบบ Sum of Sine เทียบกับมุมมองศาของหัวเข้าในการเดินแบบต่อเนื่อง.....	34
4.20 Fitting Backpropagation Neuron Network แบบ Levenberg-Marquardt	34
4.21 กราฟ Performance ของ Fitting BPNN แบบ Levenberg-Marquardt.....	35
4.22 Training State ของ Fitting BPNN แบบ Levenberg-Marquardt.....	35
4.23 กราฟ Training Regression ของ Fitting BPNN แบบ Levenberg-Marquardt.....	36
4.24 การ Fit data ของมุมมองศาของหัวเข้าในการเดินแบบต่อเนื่อง โดยใช้ Fitting Backpropagation Neuron Network.....	37
4.25 การ Fit data มุมองศาของสะโพกในการเดิน โดยใช้ Fitting Backpropagation Neuron Network.....	37
4.26 การ Fit data มุมองศาของหัวเข้าในการเดิน โดยใช้ Fitting Backpropagation Neuron Network.....	38
4.27 การ Fit data มุมองศาของข้อเท้าในการเดิน โดยใช้ Fitting Backpropagation Neuron Network.....	38

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.28 การนำ Fitting BPNN ไปใช้กับมอเตอร์ของ ชุด Lower limb Exoskeleton 1 ข้อต่อ (มอเตอร์หัวเข่า).....	40
4.29 ค่า Output ของมอเตอร์หัวเข่า.....	40
4.30 ค่าผลต่างระหว่าง Output จาก Fitting backpropagation neuron network กับ Output ของมอเตอร์หัวเข่า	41
4.31 ระบบทั้งหมดของชุด Lower Limb Exoskeleton	42
5.1 Lower Limb Exoskeleton ต้นแบบ: Electro Armor 6.....	44

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันมีผู้คนจำนวนมากที่มีปัญหาในด้านการเดินจากสาเหตุต่างๆ เช่น โรคข้อเข่าเสื่อม, ผู้ที่มีน้ำหนักตัวมาก เป็นต้น หนึ่งในวิธีที่จะมาช่วยแก้ปัญหาในการเดิน คือ lower limb exoskeleton ซึ่งเป็นอุปกรณ์ที่ใช้ในการช่วยประคองและผ่อนแรงในการเดิน

1.2 ความมุ่งหมายและวัตถุประสงค์

ในงานวิจัยนี้ได้ออกแบบระบบช่วยเดินสำหรับผู้ที่มีปัญหาด้านการเดิน โดยพัฒนาระบบช่วยเดินโดยใช้การเรียนรู้ของเครื่อง (Machine learning) เพื่อให้การทำงานของชุดมีความฉลาดมากขึ้น สามารถจำแนกและเรียนรู้ท่าทางของผู้สวมใส่ได้

1.3 สมมติฐานของการศึกษา

นำวิธีการเรียนรู้ของเครื่องมาใช้พัฒนาชุด lower limb exoskeleton ให้มีประสิทธิภาพดียิ่งขึ้น โดยจะใช้โครงข่ายประสาทเทียม (Artificial neural network) มาจำแนกท่าทางที่ผู้สวมใส่ต้องการ และนำโครงข่ายประสาทเทียมมาใช้ในการเรียนรู้ท่าทางต่างๆของร่างกายส่วนล่าง

1.4 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

ทฤษฎีที่จำเป็นต้องใช้สำหรับงานวิจัยนี้ประกอบด้วย 3 ส่วน คือ 1. ระบบการควบคุม PID และมอเตอร์ 2. โครงข่ายประสาทเทียมสำหรับการจำแนกประเภท และ 3. โครงข่ายประสาทเทียมสำหรับการเรียนรู้ท่าทาง

1.5 ขอบเขตของการศึกษา

งานวิจัยนี้ได้แบ่งการพัฒนาเป็น 3 ส่วน นั่นคือ

1. งานวิจัยนี้จึงได้พัฒนาชุด exoskeleton ในส่วนล่าง หรือ lower limb exoskeleton โดยประยุกต์ใช้หลักการของ four bar linkages เพื่อลดจำนวนของมอเตอร์ลง ให้เหลือเพียงข้างละ 1 ตัวในส่วนของเขา ซึ่งกลไกที่ออกแบบนี้ ยังเพิ่มเสถียรภาพในช่วง Stance phase อีกด้วย และใช้การควบคุมแบบ PID เพื่อให้ชุดเคลื่อนไหวได้อย่างนุ่มนวล ไม่กระตุก

2. ในงานวิจัยนี้จะพัฒนาโดยใช้ชุดข้อมูล (dataset) ของสัญญาณ sEMG ในการฝึกสอน (train) Backpropagation neural network แนวคิดในการใช้งานวิธีนี้ คือ เมื่อชุดสามารถจดจำสัญญาณที่ได้รับมาได้ว่าเป็นท่าทางใด มอเตอร์ของชุดก็จะทำการเคลื่อนไหวตามท่าทางนั้น
3. การสอนให้ชุดหัดเดิน โดยใช้ Backpropagation neural network ในการเรียนรู้ท่าทางการเดินเฉพาะบุคคลจากมุมมองศา

1.6 โครงสร้างของวิทยานิพนธ์

โครงสร้างของวิทยานิพนธ์นี้ คือ ในบทที่ 2 จะเป็นการศึกษาทฤษฎีและงานวิจัยต่างๆที่เกี่ยวข้อง เช่น งานวิจัยที่ได้มีการเสนอมาก่อนหน้า, ระบบการควบคุม PID, แบบจำลองของมอเตอร์, โครงข่ายประสาทเทียมสำหรับการจำแนกรูปแบบ (Pattern classification) และโครงข่ายประสาทเทียมสำหรับการเรียนรู้ท่าทาง จากนั้นก็นำหลักการทั้งหมดมาประยุกต์ใช้กับการออกแบบชุด Lower limb exoskeleton ในบทที่ 3 แล้วทำการทดสอบการทำงานของระบบที่ได้ทำการออกแบบไว้ในบทที่ 4 และสรุปผลทั้งหมดในบทที่ 5

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 วรรณกรรมหรืองานวิจัยที่เกี่ยวข้อง (Literature review)

ชุด Lower limb exoskeleton ได้ถูกนำเสนอในงานวิจัยต่างๆ ก่อนหน้า เช่น ในงานวิจัย Simulation Research of Exoskeleton Suit Based on Neural Network Sensitivity Amplification Control [1] จะนำเสนอเฉพาะการใช้ Backpropagation neural network ในส่วน Swing phase เท่านั้น นอกจากนี้ยังมี Lower limb exoskeleton ที่มีการพัฒนามาก่อนหน้า เช่น BLEEX [2] [3], BART LAB LL-EXO1 [4] และ HAL [5]

ชุด BLEEX ถูกพัฒนาโดย The Defense Advanced Research Project Agency (DARPA) ใช้วิธีการ Sensitivity Amplification Control (SAC) ในการควบคุมชุด คือ ผู้สวมใส่ต้องออกแรงขยับชุดเล็กน้อย ก่อนที่ชุดจะออกแรงในส่วนที่เหลือให้ ทำให้สามารถทนแรงได้

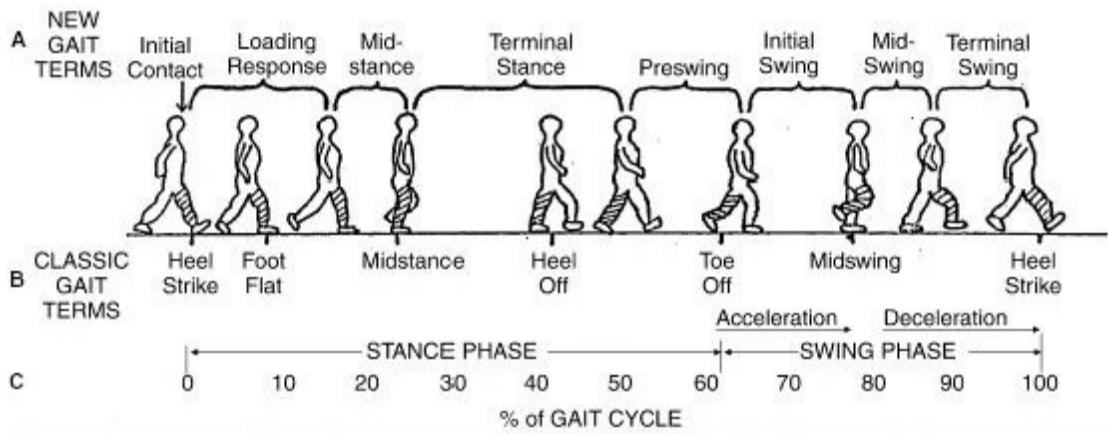
งานวิจัย [4] ได้พัฒนาชุด Lower limb exoskeleton ที่มีชื่อว่า BART LAB LL-EXO1 ก็ได้นำเสนอชุดที่ออกแบบโดยใช้กลไกสปริง และใช้การโปรแกรมท่าทางล่วงหน้า

ชุด HAL-5 ได้ถูกพัฒนาโดยศาสตราจารย์ Yoshiyuki Sankai จาก University of Tsukuba และเป็นผู้ก่อตั้งบริษัท Cyberdyne Inc. เพื่อใช้ในทางการแพทย์ ชุด HAL-5 มีความสามารถ เช่น มีใช้สัญญาณไฟฟ้าจากกล้ามเนื้อ (sEMG) ในการควบคุมชุด Exoskeleton แล้วให้ชุดขยับตามท่าทางที่ได้โปรแกรมไว้ล่วงหน้า [5]

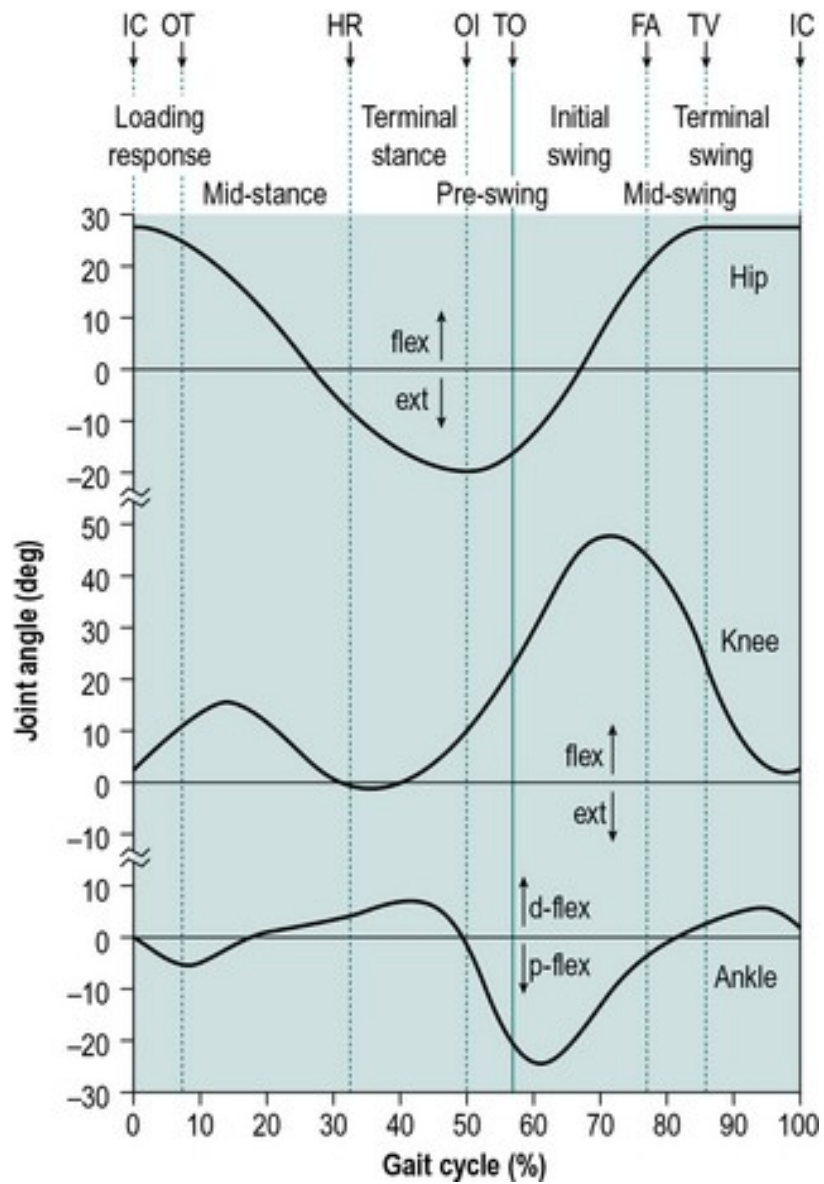
จากงานวิจัยทั้งหมดที่ได้กล่าวมาก่อนหน้า ในการพัฒนางานวิจัยนี้จะมีต่อยอดจากชุด Exoskeleton อื่นๆ เช่น การทำงานของชุดที่ใช้ตัวควบคุม PID และกลไกแบบ Four bar linkage [6] , ส่วนของระบบจําแนกรูปแบบเพื่อให้สามารถจําแนกท่าทางของชุดที่ผู้สวมใส่ต้องการได้ [7] และยังมีโหมดการสอนให้ชุดเคลื่อนไหวได้โดยไม่ต้องโปรแกรมท่าทางล่วงหน้า แต่ให้ผู้สวมใส่สามารถฝึกเดินไปพร้อมกับชุดได้

2.2 Gait analysis

ในหนึ่งรอบการเดินของมนุษย์สามารถแบ่งได้เป็นสองช่วงหลักๆ นั่นคือ stance phase คิดเป็น 60% ในหนึ่งรอบการเดิน และ swing phase คิดเป็น 40% ในหนึ่งรอบการเดิน ช่วง stance หมายถึง ช่วงที่เท้าสัมผัสพื้น และช่วง swing หมายถึง ช่วงที่เท้าไม่สัมผัสพื้น ดังที่แสดงในรูปที่ 2.1 และในรูปที่ 2.2 จะแสดงมุมมองของข้อต่อต่างๆที่ใช้ในการเคลื่อนที่ของมนุษย์ ซึ่งประกอบด้วย สะโพก, เข่า และข้อเท้า มุมอ้างอิงจะมาจากขาที่แรงงาในรูปที่ 2.1 ในหนึ่งรอบการเดิน



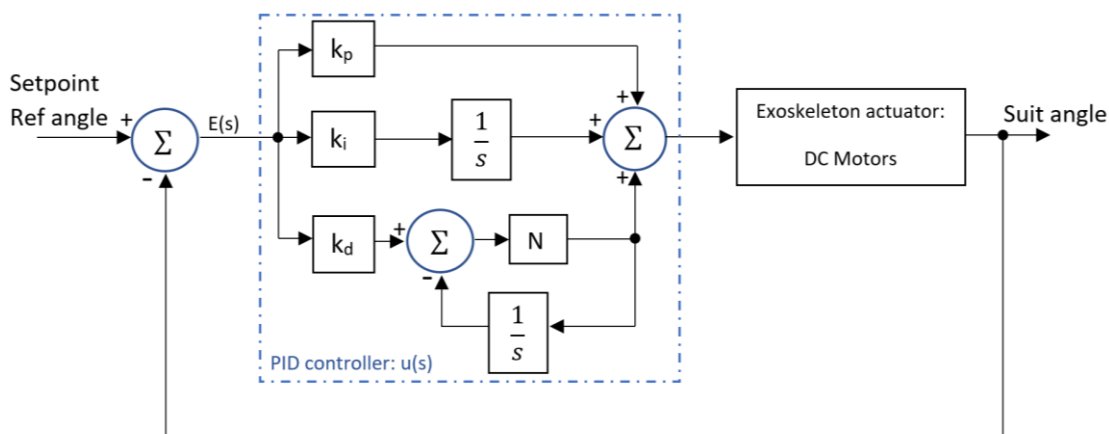
รูปที่ 2.1 Gait Analysis [8]



รูปที่ 2.2 Joint angle of hip, knee and ankle [9]

2.3 PID Controller

ตัวควบคุมแบบ PID ดังที่ได้แสดงในรูปที่ 2.3 คือระบบควบคุมแบบป้อนกลับที่ใช้ error $E(s)$ เป็นอินพุต ค่า error คือผลต่างระหว่างมุมที่ต้องการ (setpoint) กับมุมของชุด (feedback)



รูปที่ 2.3 บล็อกไดอะแกรมตัวควบคุม PID ของ Exoskeleton actuator

ตัวควบคุม PID ประกอบด้วย 3 เทอม คือ proportional term (P), integration term (I) และ derivative term (D) อย่างไรก็ตาม ตัวควบคุมมีความไวต่อสัญญาณรบกวน (noise) จึงต้องมีการใช้ filter เพื่อที่จะลดผลของสัญญาณรบกวนดังกล่าว เราสามารถปรับค่า filter coefficients (N) เพื่อที่จะลด noise เราใช้สมการที่ (2.1) และ (2.2) ในการออกแบบตัวควบคุมแบบ PID โดยการปรับค่า K_p , K_i , K_d และ N

$$E(s) = \text{setpoint} - \text{feedback} \quad (2.1)$$

$$u(s) = E(s) \left(K_p + K_i \frac{1}{s} + K_d \frac{N}{1 + N \frac{1}{s}} \right) \quad (2.2)$$

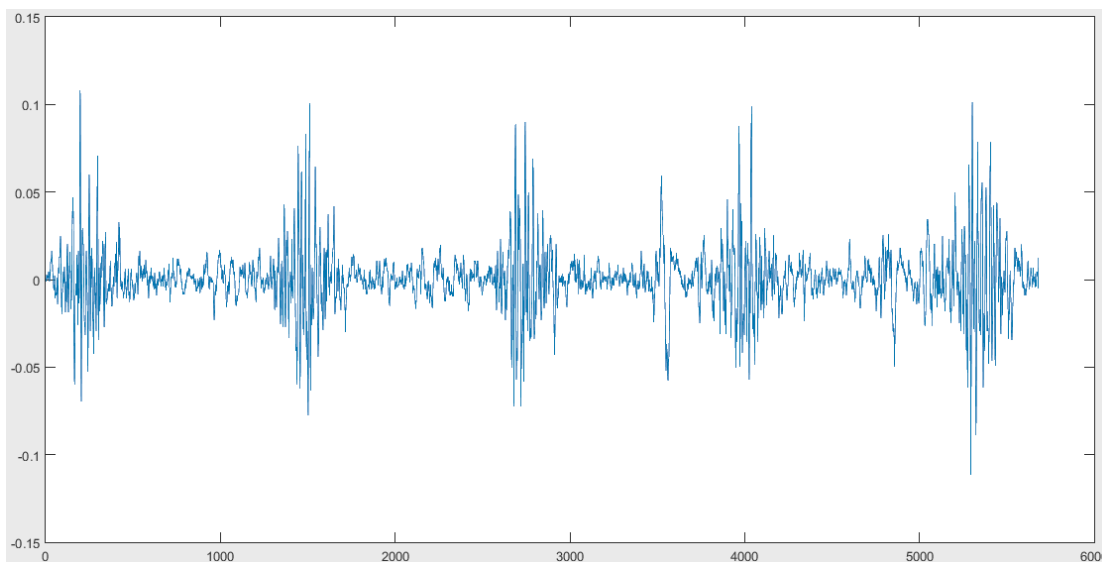
โดย K_p คือ proportional gain

K_i คือ integration gain

และ K_d คือ derivative gain

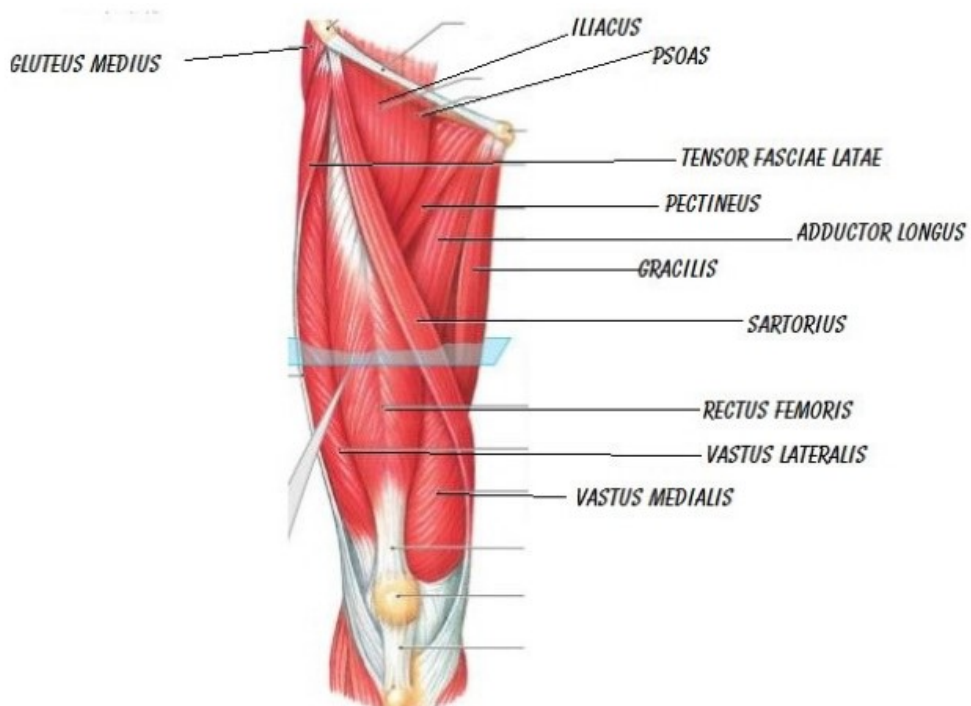
2.4 สัญญาณ Surface Electromyography (sEMG) และกล้ามเนื้อส่วนล่าง

สัญญาณ Electromyography ที่ได้แสดงในรูปที่ 2.4 คือ สัญญาณไฟฟ้าที่วัดได้จากกล้ามเนื้อเมื่อกล้ามเนื้อมีการทำงาน [10] โดย surface EMG: sEMG คือ การวัดสัญญาณ EMG ที่ผิวหนัง เราใช้ dataset ของ sEMG เพื่อใช้ในการฝึกสอนโครงข่าย สำหรับการจำแนกท่าทางของร่างกายส่วนล่าง

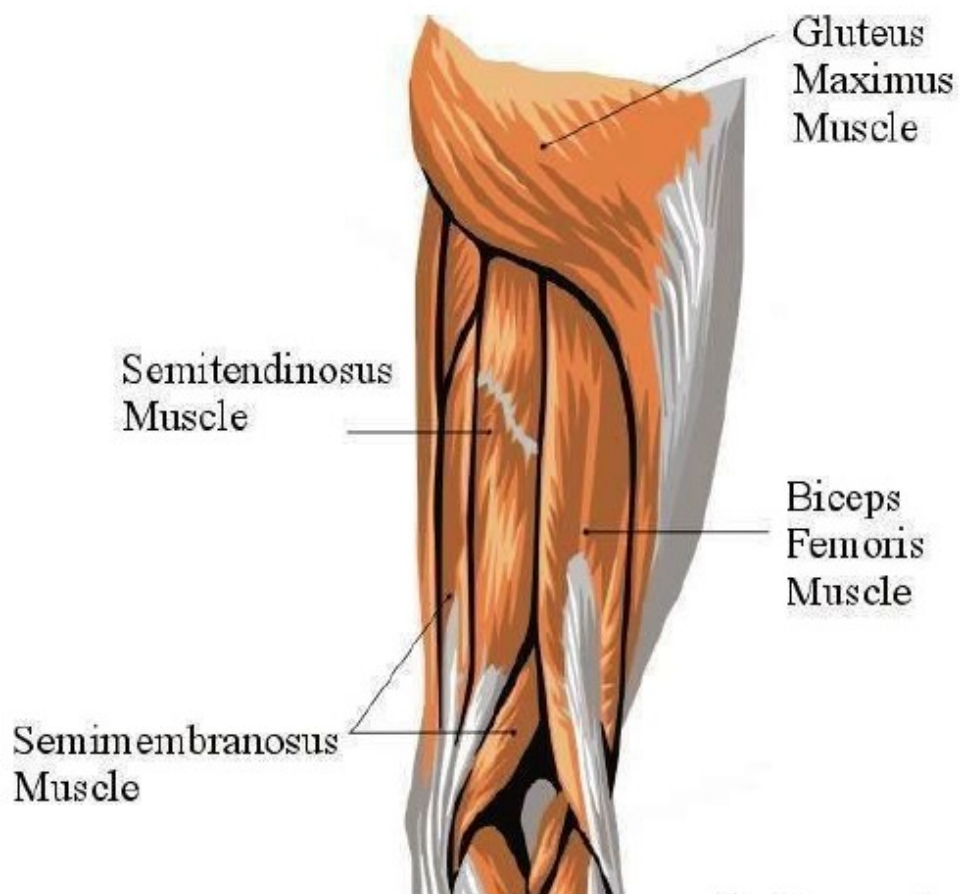


รูปที่ 2.4 สัญญาณ sEMG จากกล้ามเนื้อ rectus femoris เมื่อมนุษย์กำลังเดิน

สัญญาณ sEMG ที่ใช้ในงานนี้ประกอบด้วย กล้ามเนื้อต้นขาด้านหน้า (muscle of the anterior thigh), กล้ามเนื้อต้นขาด้านหลัง (muscle of the posterior thigh) และมุมมองการเคลื่อนที่ของหัวเข่าที่วัดมาจาก MWX8 Data log Biometrics กล้ามเนื้อต้นขาด้านหน้า ทำหน้าที่งอสะโพกและเหยียดเข่า ประกอบด้วย Rectus Femoris (RF) และ Vastus Medialis (VM) ส่วนกล้ามเนื้อต้นขาด้านหลัง จะทำหน้าที่เหยียดสะโพกและงอเข่า ประกอบด้วย Biceps Femoris (BF) และ Semitendinosus (ST) ดังรูปที่ 2.5 และรูปที่ 2.6 ตามลำดับ



รูปที่ 2.5 กล้ามเนื้อต้นขาด้านหน้า [11]



รูปที่ 2.6 กล้ามเนื้อต้นขาด้านหลัง [11]

2.5 Feature Extraction

Feature extraction คือกระบวนการที่จะแปลงข้อมูลและดึงลักษณะสำคัญของสัญญาณออกมา [12] และลดปริมาณข้อมูลที่ใช้ในการฝึกสอนโครงข่าย ในงานวิจัยนี้ใช้ time domain และ frequency domain ในการทำ Feature Extraction โดย time domain จะประกอบด้วย Root Mean Square, Mean Absolute Value, Simple Square Integral, Variance of EMG และ Integrated Absolute Value ส่วน frequency domain จะประกอบด้วย Mean Frequency และ Median Frequency

Root Mean Square (RMS)

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (2.3)$$

Mean Absolute Value (MAV)

$$MAV = \frac{1}{N} \sum_{i=1}^N |x_i| \quad (2.4)$$

Simple Square Integral (SSI)

$$SSI = \sum_{i=1}^N |x_i|^2 \quad (2.5)$$

Variance of EMG (VAR)

$$VAR = \frac{1}{N-1} \sum_{i=1}^N x_i^2 \quad (2.6)$$

Integrated Absolute Value (IAV)

$$IAV = \sum_{i=1}^N |x_i| \quad (2.7)$$

Mean Frequency

$$MNF = \frac{\sum_{j=1}^M f_j P_j}{\sum_{j=1}^M P_j} \quad (2.8)$$

Median Frequency

$$\sum_{j=1}^{MDF} P_j = \sum_{j=MDF}^M P_j = \frac{1}{2} \sum_{j=1}^M P_j \quad (2.9)$$

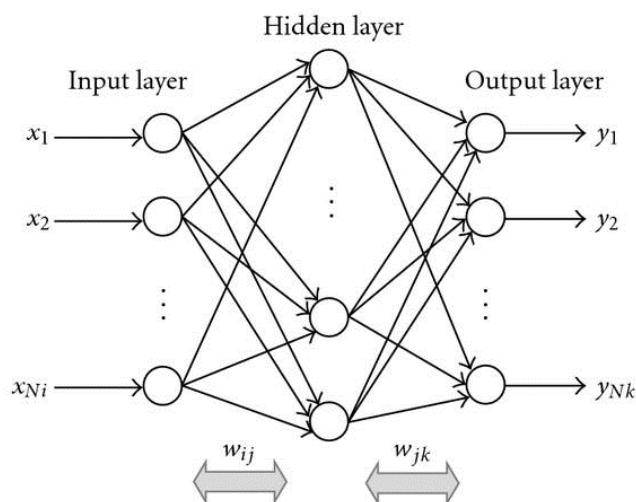
เมื่อ x_i คือสัญญาณ sEMG

และ P_j คือ power spectrum ของสัญญาณ sEMG

2.6 การจำแนกท่าทางโดยใช้ Backpropagation Neural Network

ตัวจำแนกรูปแบบที่ใช้คือ Backpropagation neural network (BPNN) ซึ่งเป็นวิธีการหนึ่งสำหรับการเรียนรู้ของเครื่องแบบ supervised โดยใช้การคำนวณ errors จากความแตกต่างของ Output และ target โดยใช้ gradient descent

Backpropagation neural network จะถูกฝึกสอนเพื่อที่จะจำแนกประเภทของ input ตาม target classes อินพุตของ Backpropagation neural network คือ สัญญาณ sEMG ที่ได้หลังจากการทำ feature extraction และเอาต์พุตของ Backpropagation neural network จะถูกใช้ในการสั่งตัวควบคุมการเคลื่อนที่ ให้ควบคุมท่าทางของชุด Exoskeleton



รูปที่ 2.7 feed forward neural network [13]

2.7 Levenberg-Marquardt backpropagation neural network (LM-BPNN)

Levenberg-Marquardt backpropagation เป็น Backpropagation neural network อีกรูปแบบหนึ่งที่นำมาใช้ในการสอนให้ชุด Exoskeleton เรียนรู้ท่าทางการเดินจากมุมมองตา

วิธีการของ Levenberg-Marquardt จะเป็นการผสมผสานวิธีการ steepest descent กับวิธีการ Gauss-Newton เข้าด้วยกัน ซึ่งจะทำให้สามารถเข้าสู่ค่าตอบได้เร็วเหมือนวิธีการของ Gauss-Newton แต่ก็มีความเร็วมากกว่า เหมือนกับวิธีการ steepest descent [14] สำหรับวิธีการ steepest descent จะใช้วิธีการปรับค่าถ่วงน้ำหนัก (weight) ดังนี้

$$w_{k+1} = w_k - \alpha g_k \quad (2.10)$$

โดย α เป็นค่า learning constant (step size)

w_k คือ ค่า weight vector

และค่า gradient (g) จะสามารถคำนวณได้ว่า

$$g = J^T e \quad (2.11)$$

สำหรับ Levenberg-Marquardt algorithm ที่ใช้งานใน MATLAB นั้น ได้ถูกออกแบบมาให้ประมาณค่า second-order โดยไม่ต้องคำนวณเมตริกซ์ Hessian เมื่อ performance function อยู่ในรูปของ sum of squares เมตริกซ์ Hessian (H) จะประมาณค่าได้ว่า

$$H = J^T J \quad (2.12)$$

เมื่อ J คือ Jacobian matrix ที่ประกอบด้วยอนุพันธ์อันดับหนึ่งของความผิดพลาดกับ weights และ biases ตามลำดับ, e คือ เวกเตอร์ของค่าความผิดพลาด (vector of network errors)

Jacobian matrix จะสามารถคำนวณโดยใช้วิธีการ backpropagation มาตราฐาน ซึ่งซับซ้อนน้อยกว่าการคำนวณเมตริกซ์ Hessian ซึ่งวิธีการ Levenberg-Marquardt algorithm จะใช้ประมาณค่า Hessian matrix ในวิธีการ Newton-like update สำหรับปรับค่าถ่วงน้ำหนักดังนี้

$$w_{k+1} = w_k - [J^T J + \mu I]^{-1} J^T e \quad (2.13)$$

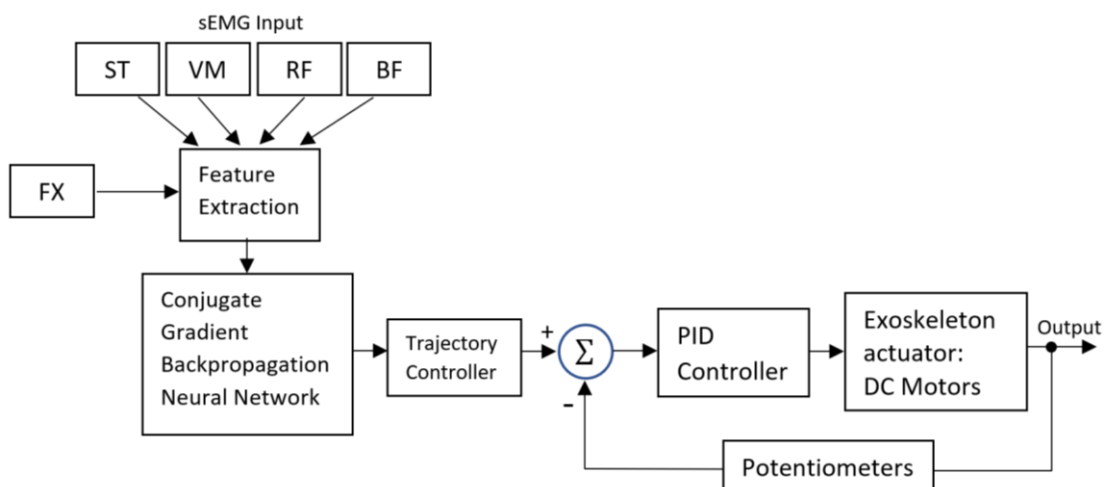
$$\alpha = \frac{1}{\mu} \quad (2.14)$$

เมื่อค่า μ มีค่าเป็นศูนย์ สมการนี้จะกลายเป็นวิธีการ Newton's method ใช้ประมาณค่า Hessian matrix เมื่อ μ มีค่ามากๆ สมการนี้จะกลายเป็น steepest descent ที่มีค่า step size น้อยๆ

วิธีการ Newton's method มีความเร็วในการลู่เข้ามากกว่า และมีความถูกต้องใกล้เคียงกับ error minimum มากกว่า ดังนั้นเป้าหมายคือการเปลี่ยนไปใช้วิธีการ Newton's method ให้เร็วที่สุดเท่าที่เป็นไปได้ ดังนั้นค่า μ จะลดลงก่อนที่จะจบแต่ละขั้นตอน (reduction in performance function) และเพิ่มค่า μ เฉพาะเมื่อตอนต้นของขั้นตอน เพื่อเพิ่ม performance function. ด้วยวิธีการนี้ performance function จะลดลงในแต่ละรอบของการคำนวณ

บทที่ 3 วิธีดำเนินงานวิจัย

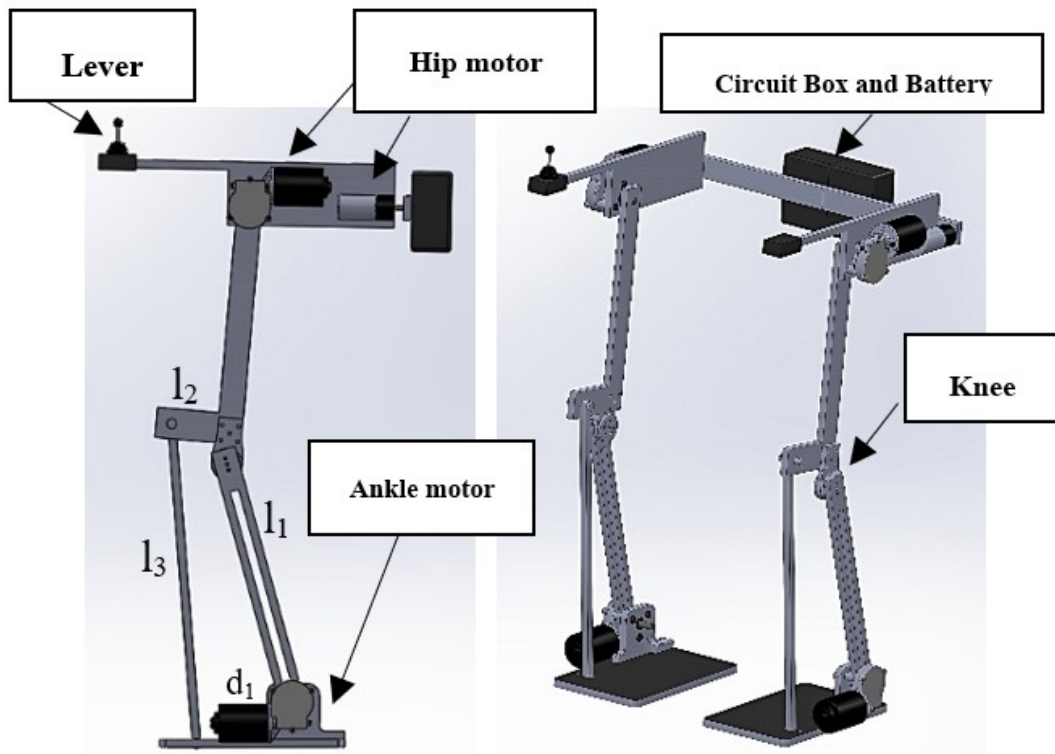
บล็อกไดอะแกรมพื้นฐานของระบบช่วยเดิน ประกอบด้วย สัญญาณอินพุต (sEMG และมุมมองขาของการงอหัวเข่า (Flexion at the knee: Fx)), Feature Extraction, ตัวจำแนกท่าทางโดยใช้ Backpropagation Neural Networks, ตัวควบคุมการเคลื่อนที่ (Trajectory controllers), PID controllers, DC Motors และตัววัดมุม (potentiometers) ดังที่ได้แสดงในรูปที่ 3.1



รูปที่ 3.1 Exoskeleton system diagram

3.1 Lower Limb Exoskeleton

กลไกแบบ Four Bar Linkages เป็นกลไกที่ประกอบด้วยชิ้นส่วนสี่ชิ้น ประกอบด้วย l_1 , l_2 , l_3 และ d_1 กำหนดให้ d_1 เป็น frame, l_1 คือ actuator, l_2 คือ transmission link และ l_3 เป็น load ดังที่ได้แสดงในรูปที่ 3.1 เมื่อ actuator l_1 ทำงาน จะให้ชิ้นส่วน l_3 ขยับด้วย เพราะ l_1 ส่งกำลังไปยัง l_3 ผ่าน l_2 เราออกแบบกลไก Four Bar Linkages เพื่อลดจำนวนมอเตอร์ที่ใช้ใน Exoskeleton suit โดยนำไปใช้บริเวณข้อเท้ากับหัวเข่า เพราะในเวลาเดิน มุมองศาการเคลื่อนที่ของข้อเท้าและหัวเข่าจะมีความสัมพันธ์กัน l_1 คือ แข้ง, d_1 คือเท้า, l_2 คือต้นขา ในงานวิจัยนี้ มุมที่ใช้ในการเดินจะอ้างอิงมาจากรูปที่ 2.2 ซึ่งใช้ในการออกแบบมอเตอร์ที่ข้อเท้า การออกแบบในลักษณะนี้จะทำให้ Center of Gravity (CG) ของชุดอยู่ต่ำกว่าชุด Exoskeleton ทั่วไป ซึ่งสามารถเพิ่มเสถียรภาพในช่วง Stance Phase ได้

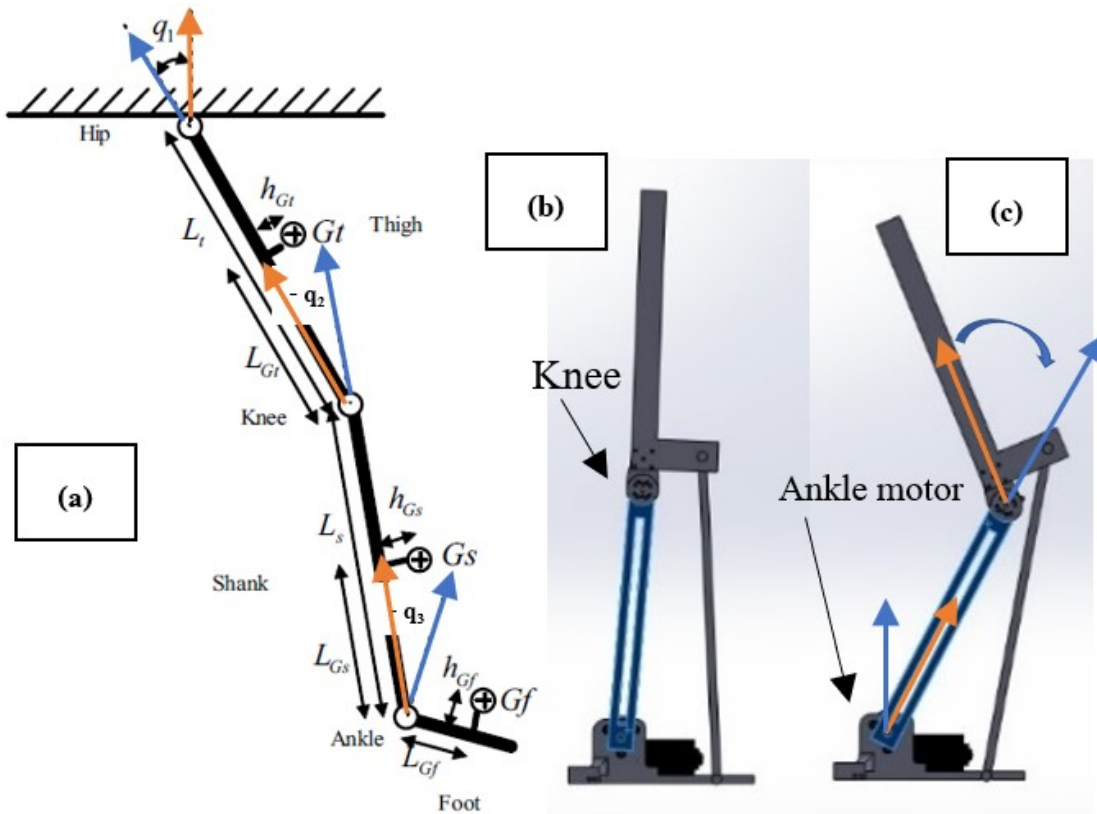


รูปที่ 3.2 การออกแบบกลไก Four bar linkages mechanisms ของ Lower limb exoskeleton suit

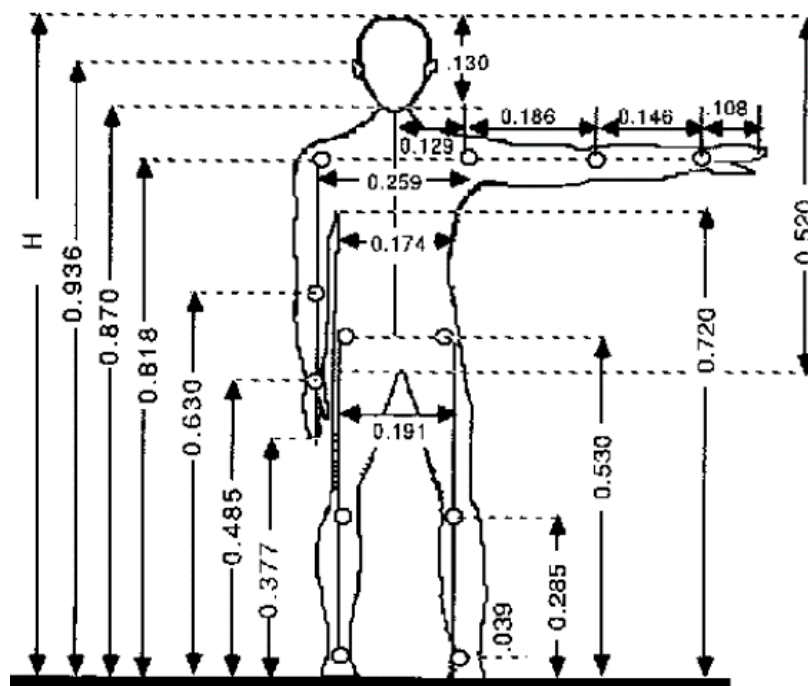
ตารางที่ 3.1 เปรียบเทียบช่วงมุมองศา (Degree) สูงสุดและต่ำสุดใน 1 รอบการเดินของข้อต่อส่วนต่างๆ ระหว่างมนุษย์กับชุด Exoskeleton ที่ออกแบบ

Joint	ค่ามุมที่มากที่สุด (Minimum angle)		ค่ามุมน้อยที่สุด (Maximum angle)	
	มนุษย์	ชุด Exoskeleton	มนุษย์	ชุด Exoskeleton
สะโพก (q_1)	-20	-15	30	30
เข่า (q_2)	-2	3	50	50
ข้อเท้า (q_3)	-25	-10	10	10

ชุด Exoskeleton นี้ได้ออกแบบเพื่อใช้งานกับมนุษย์ จากช่วงมุมองศาของชุดที่ได้แสดงในตารางที่ 3.1 และรูปที่ 3.3 เราใช้ช่วง Stand phase เป็นมุมอ้างอิง เมื่อเท้าขานานกับพื้น, ต้นขาและแข้งตั้งฉากกับพื้น โดยความยาวของต้นขาและแข้งจะคำนวณมาจากข้อมูลสัดส่วนร่างกายมนุษย์ ดังรูปที่ 3.4 [15] การกำหนดช่วงมุมองศาที่ชุดขยับได้ เพื่อไม่ให้มีการบาดเจ็บจากการที่ชุด Exoskeleton ขยับเกินกว่าองศาที่ร่างกายมนุษย์ขยับได้ แต่ยังทำให้ชุด Exoskeleton สามารถเคลื่อนไหวได้ครอบคลุมช่วงการเดิน



รูปที่ 3.3 (a) มุมอ้างอิงของขามนุษย์ [16] ให้การกวาดขึ้นส่วนในทิศทวนเข็มนาฬิกาเป็นบวก จะได้มุมสะโพก(q_1), เข่า($-q_2$), ข้อเท้า($-q_3$), L_t คือความยาวของต้นขา (Thigh), L_s คือความยาวของแข้ง (Shank), แกนอ้างอิงคือ ลูกศรสีส้ม และแกนการแกว่งคือ ลูกศรสีน้ำเงิน และ (b) $-q_2 = -3$ องศา, $-q_3 = 5$ องศา และ (c) $-q_2 = -50$ องศา, $-q_3 = 30$ องศา

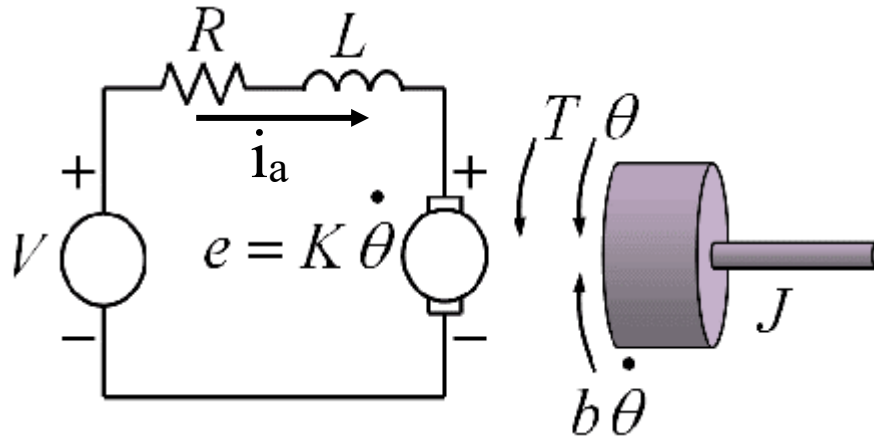


รูปที่ 3.4 ข้อมูลสัดส่วนร่างกายมนุษย์ (Anthropometric data) ของคนที่กำลังยืน [15]

3.2 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง

3.2.1 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง แบบที่ไม่มีโหลดทางกล

วงจรของมอเตอร์กระแสตรงและโหลดทางกลจะเป็นดังรูปที่ 3.5 ซึ่งสมการอนุพันธ์ที่บรรยายพลวัตของส่วนทางกล ให้การเคลื่อนที่ของกลไกเป็นอุดมคติ อธิบายได้โดยสมการที่ (3.1) และสมการความสัมพันธ์ของแรงดันไฟฟ้า ดังสมการที่ (3.2)



รูปที่ 3.5 มอเตอร์กระแสตรงที่ต่อกับโหลดทางกล [17]

$$J\ddot{\theta} + b\dot{\theta} = \tau = k_t i_a \quad (3.1)$$

$$L\dot{i}_a + R i_a = V - k_e \dot{\theta} \quad ; k_e = k_t = k \quad (3.2)$$

ทำการแปลงลาปลาซ แล้วจัดรูปสมการ จะได้

$$\frac{\theta(s)}{V(s)} = \frac{k}{s((Js + b)(Ls + R) + k^2)} \quad (3.3)$$

แบบจำลองของมอเตอร์ไฟฟ้ากระแสตรงจะเป็นดังสมการที่ (3.3) เลือกใช้ DC Wiper Motor 12V เป็นตัวขับ ค่าพารามิเตอร์ที่เป็นคุณสมบัติของ Wiper Motor โดยอ้างอิงจาก [18] แทนค่า $L = 9.6 \times 10^{-10}$ H, $R = 0.6$ Ω , $J = 0.004$ kgm^2 , $b = 0.0185$ $\text{Nm}/(\text{rad}/\text{s})$ และ $k = 0.065$ V/rpm ลงใน (3.3) จะได้ฟังก์ชันถ่ายโอนของมอเตอร์กระแสตรง คือ

$$\frac{\theta(s)}{V(s)} = \frac{1}{(5.9077 \times 10^{-11})s^3 + (0.0369)s^2 + 0.235s} \quad (3.4)$$

3.2.2 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง แบบที่มีโหลดทางกล

ในหัวข้อนี้จะเป็นแบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง แบบที่มีโหลดทางกล โดยมีอัตราทดเกียร์ $r:1$ และ J คือผลรวมแรงเฉื่อยของมอเตอร์และเกียร์ จะได้สมการบรรยายการเคลื่อนที่ของระบบดังสมการ

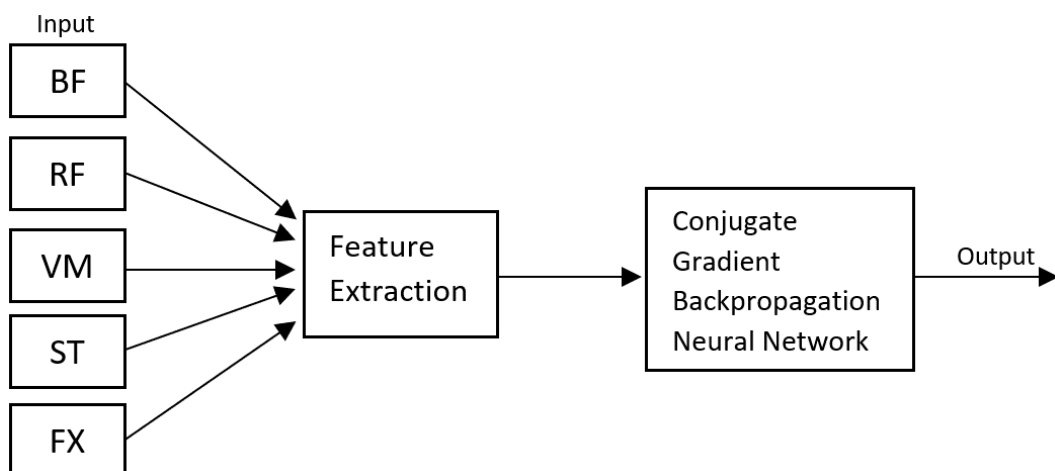
$$J\ddot{\theta} + b\dot{\theta} = \tau_m - \frac{\tau_l}{r} = k_t i_a - \frac{\tau_l}{r} \quad (3.5)$$

ใช้ความสัมพันธ์จากสมการ (3.1) และ (3.2) แทนค่าแล้วจัดรูปสมการ จะได้สมการแบบจำลองของมอเตอร์กระแสตรง 12V ที่ใช้ในการขับเคลื่อนชุด Exoskeleton ดังที่ได้แสดงในสมการ (3.6) ดังนี้

$$\theta \left[(Js^2 + bs) \left(\frac{Ls + R}{k_t} \right) + k_e s \right] + \frac{\tau_l(Ls + R)}{rk_t} = V \quad (3.6)$$

3.3 การทำงานของ Backpropagation neural networks เพื่อใช้ในการจำแนกท่าทาง

ในขั้นตอนแรก สัญญาณอินพุต sEMG จะถูกนำไปคำนวณ Feature extraction เพื่อดึงลักษณะสำคัญของข้อมูลออกมา ก่อนจะนำไปฝึกสอนใน Backpropagation neural network เมื่อฝึกสอน Neural network เสร็จแล้ว เอาท์พุทของ Neural network จะนำไปใช้สั่ง Plant หรือชุด Exoskeleton ต่อไป สำหรับการนำไปใช้งานแบบออนไลน์นั้น สัญญาณอินพุตที่วัดได้จากเซนเซอร์จะถูกนำไปคำนวณ Feature extraction ก่อนเข้า Backpropagation Neural Network แล้วเอาท์พุทของ Neural network จะนำไปใช้สั่ง Plant หรือชุด Exoskeleton ต่อไป

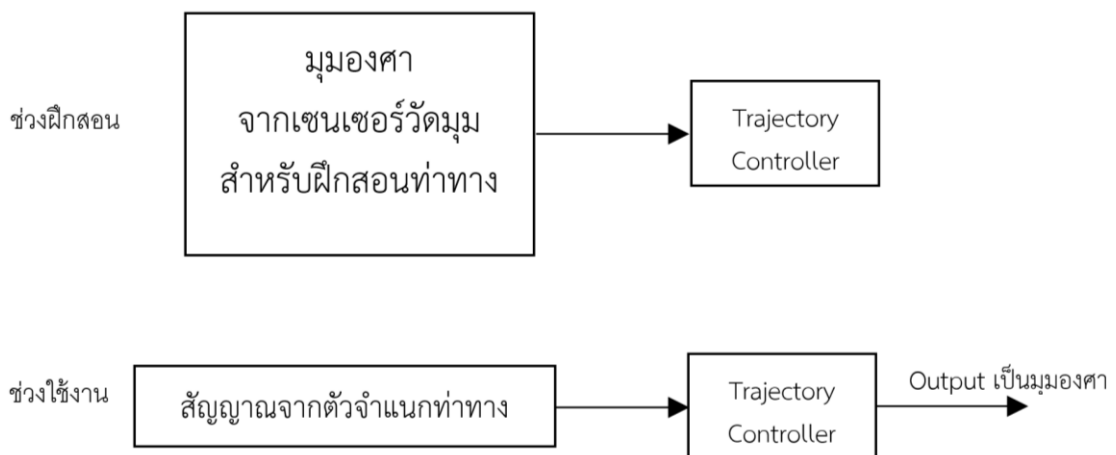


รูปที่ 3.6 Block diagram การทำงานของ Backpropagation neural networks เพื่อใช้ในการจำแนกท่าทาง

3.4 การสอนท่าทางให้ชุด Exoskeleton ในแต่ละข้อต่อ

ในตอนต้นของงานวิจัยนี้ในส่วนของตัวควบคุมการเคลื่อนที่ (Trajectory controllers) จะใช้วิธีการโปรแกรมท่าทางลงไปหน่วยความจำโดยตรง แต่วิธีการนี้ไม่มีความยืดหยุ่น ไม่สามารถปรับท่าทางให้เป็นท่าทางที่เฉพาะกับบุคคลนั้นได้ๆ

ดังนั้น อีกรูปแบบการใช้งานหนึ่งที่จะนำเสนอในงานวิจัยนี้คือ การสอนท่าทางการเคลื่อนไหวของร่างกายส่วนล่างให้กับชุด lower limb exoskeleton วิธีการนี้จะดีกว่าการโปรแกรมท่าทางลงไปหน่วยความจำโดยตรง เพราะว่ามี ความยืดหยุ่นมากกว่า และสามารถสอนท่าทางให้ชุดสามารถจดจำท่าทางการเดินเฉพาะบุคคลได้



รูปที่ 3.7 บล็อกไดอะแกรมของตัวควบคุมการเคลื่อนที่ (Trajectory controllers) ในช่วงฝึกสอนและช่วงใช้งาน

ก่อนจะนำตัวควบคุมการเคลื่อนที่ไปใช้งาน จะต้องนำไปฝึกสอนก่อน ในช่วงฝึกสอนจะเป็นการนำมุมมองไปฝึกสอนให้กับโครงข่าย

ในช่วงใช้งาน เมื่อตัวจำแนกท่าทางส่งผลลัพธ์ของการจำแนกออกมา สัญญาณผลลัพธ์จากตัวจำแนก จะเป็นตัวเปิดหรือปิดการทำงานของตัวควบคุมการเคลื่อนที่ เมื่อตัวควบคุมการเคลื่อนที่ถูกเปิดการทำงานโดยตัวจำแนกท่าทาง ก็จะส่ง Output ออกมาเป็นมุมมองที่ได้ทำการฝึกสอนไป

การสอนท่าทางให้ชุดในงานวิจัยนี้ จะแบ่งออกได้เป็น 2 วิธีหลักๆ คือ การใช้วิธีการ Curve Fitting และการใช้ Backpropagation Neural Network แบบ Levenberg-Marquardt

3.4.1 วิธีการ Curve fitting

การประมาณค่าโดยใช้วิธีการ Curve fitting เพื่อใช้ในการประมาณค่ารูปแบบของมุมมองศา
วิธีการ Curve fitting นั้นมีหลากหลายวิธี ซึ่งวิธีการที่งานวิจัยนี้จะเลือกนำมาใช้ คือ วิธีการ Curve
fitting แบบ Gaussian Model และวิธีการ Sum of Sine

3.4.1.1 วิธีการ Fitting Gaussian Model

สมการ Fitting General model Gauss เป็นดังนี้

$$f(x) = \sum_{i=1}^n a_i e^{\left[-\frac{(x-b_i)^2}{c_i}\right]} \quad (3.6)$$

เมื่อ a คือ amplitude, b คือ centroid (location), c คือ related to the peak width, n
คือ number of peaks to fit และ $1 \leq n \leq 8$ สำหรับงานวิจัยนี้จะใช้ค่า $n = 8$ และจะทำการ
คำนวณแบบวนซ้ำ จนได้ค่า a, b, c ที่ทำให้มีความคลาดเคลื่อนน้อยที่สุด หรือความคลาดเคลื่อนน้อย
กว่าที่กำหนดไว้

3.4.1.2 วิธีการ Sum of Sine

วิธีการ sum of sines เหมาะกับ periodic functions ดังที่แสดงในสมการที่ (3.7)

$$y = \sum_{i=1}^n a_i \sin(b_i x + c_i) \quad (3.7)$$

$$y = a_0 + \sum_{i=1}^n (a_i \cos(iwx) + b_i \sin(iwx)) \quad (3.8)$$

เมื่อ a คือ แอมพลิจูด, b คือ ความถี่, c คือ ค่าคงที่เฟสสำหรับแต่ละเทอมของคลื่นไซน์ และ
ให้ n มีค่าอยู่ในช่วง $1 \leq n \leq 8$ สำหรับงานวิจัยนี้จะใช้ค่า $n = 8$ และจะทำการคำนวณแบบวนซ้ำ
จนได้ค่า a, b, c ที่ทำให้มีความคลาดเคลื่อนน้อยที่สุด หรือความคลาดเคลื่อนน้อยกว่าที่กำหนดไว้
สมการนี้มีความคล้ายกับ Fourier series ในสมการที่ (3.8) ซึ่งข้อแตกต่างที่สำคัญ คือ สมการ
sum of sines นั้นจะมีค่าคงที่เฟส (c_i) และไม่มีเทอมของค่าคงที่ (a_0) เหมือน Fourier series

3.4.1.3 การประเมินความถูกต้องของ Curve fitting

a) Sum of Squares Due to Error

ค่าทางสถิตินี้วัดความเบี่ยงเบนทั้งหมดของค่าการตอบสนองจากการ Fit ค่าการตอบสนอง ถ้าแบบจำลองนี้มีค่าความผิดพลาดแบบสุ่มน้อยมาก ค่า SSE ที่ได้จะมีค่าใกล้เคียงศูนย์ ดังสมการที่ (3.9)

$$SSE = \sum_{i=1}^n w_i (y_i - \hat{y}_i)^2 \quad (3.9)$$

b) R-Square

R-square คือ ค่ากำลังสองของความสัมพันธ์ระหว่างค่า response กับค่า predicted response. ค่า R-square ถูกกำหนดให้เป็นอัตราส่วนของค่า sum of squares of the regression (SSR) และ total sum of squares (SST) โดยค่า SSR เป็นดังสมการนี้

$$SSR = \sum_{i=1}^n w_i (\hat{y}_i - \bar{y})^2 \quad (3.10)$$

SST เรียกอีกอย่างว่าผลรวมของกำลังสองเกี่ยวกับค่าเฉลี่ย (sum of squares about the mean) นั่นคือ

$$SST = \sum_{i=1}^n w_i (y_i - \bar{y})^2 \quad (3.11)$$

เมื่อให้ $SST = SSR + SSE$ ค่า R-square จะได้

$$R \text{ square} = \frac{SSR}{SST} = 1 - \frac{SSE}{SST} \quad (3.12)$$

R-square สามารถมีค่าได้ ๆ ระหว่าง 0 ถึง 1 ถ้าค่า R-square มีค่าใกล้เคียงกับ 1 แสดงว่ามีสัดส่วนของความแปรปรวนมากขึ้น

c) Degrees of Freedom Adjusted R-Square

ค่าทางสถิติตัวนี้จะใช้ R-square ที่กำหนดไว้ข้างต้น และปรับตามระดับองศาอิสระ (the residual degrees of freedom) ซึ่งจะถูกกำหนดเป็นจำนวนค่าการตอบสนอง (response values) n ลบจำนวน fitted coefficients m โดยประมาณจากค่า response values

$$v = n - m \quad (3.13)$$

v คือ จำนวนของข้อมูลอิสระที่เกี่ยวข้องกับจุดข้อมูล n ที่จำเป็นในการคำนวณผลรวมของกำลังสอง

$$adjusted = 1 - \frac{SSE(n-1)}{SST(v)} \quad (3.14)$$

ค่าสถิติ R-square ที่ปรับแล้ว (adjusted R-square) สามารถใช้กับค่าใด ๆ ที่น้อยกว่าหรือเท่ากับ 1 โดยที่ค่าที่ใกล้เคียงกับ 1 มากขึ้น หมายความว่ามีการ fit ที่ดีกว่า

d) Root Mean Squared Error

สถิตินี้เรียกว่าข้อผิดพลาดมาตรฐานที่พอดี (fit standard error) และข้อผิดพลาดมาตรฐานของการถดถอย (standard error of the regression) มันเป็นค่าประมาณของส่วนเบี่ยงเบนมาตรฐานขององค์ประกอบแบบสุ่มในข้อมูลและถูกกำหนดเป็น

$$RMSE = s = \sqrt{MSE} \quad (3.15)$$

เมื่อ MSE คือ mean square error

$$MSE = \frac{SSE}{v} \quad (3.16)$$

เช่นเดียวกับ SSE สำหรับค่า MSE เข้าใกล้ 0 บ่งบอกถึงการ fit ที่มีประโยชน์มากกว่าสำหรับการทำนาย (prediction)

3.4.2 วิธีการ Backpropagation Neuron Network แบบ Levenberg-Marquardt

หลักการออกแบบการทำงานโดยใช้ Backpropagation Neural Network คือ เมื่อมีการรับข้อมูลอินพุตที่เป็นมุมมองขาเข้ามาแล้ว นำไปฝึกสอนให้กับโครงข่าย แล้วทดสอบ (test) เพื่อปรับค่าถ่วงน้ำหนักของโครงข่าย แล้ววนซ้ำจนกระทั่งได้ค่า Error น้อยกว่าค่าที่กำหนดไว้ (เช่น ให้ค่า error ไม่เกิน 10^{-6})

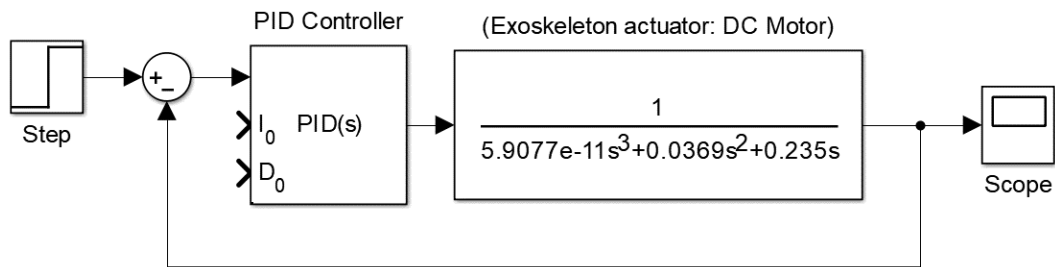
บทที่ 4

การทดลองและผลการทดลอง

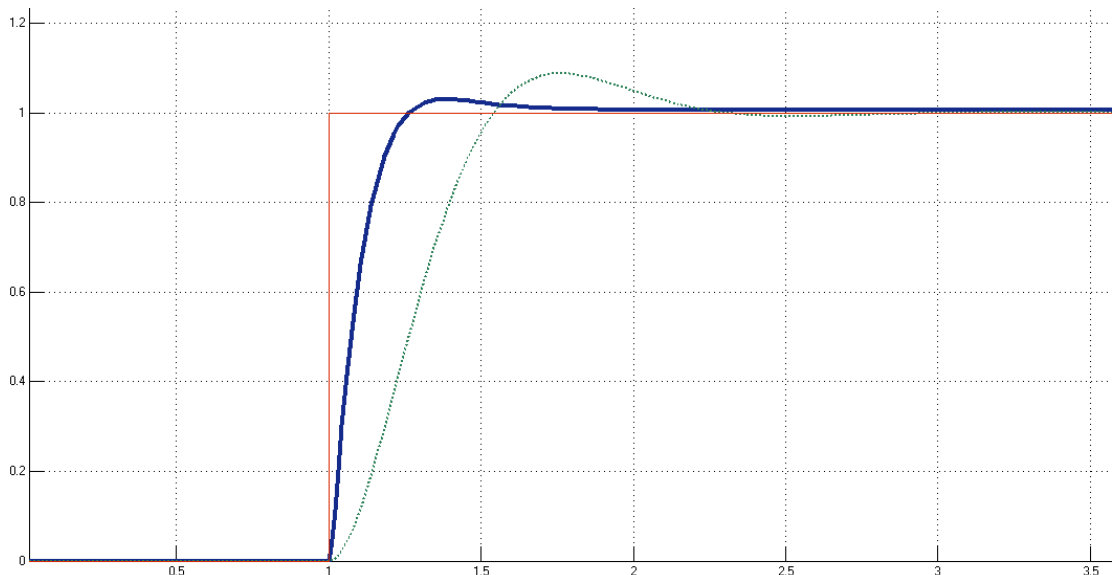
4.1 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง

4.1.1 การทดลองกับมอเตอร์ที่ไม่มีโหลดทางกล

ทำการทดลองกับมอเตอร์ที่ไม่มีโหลดทางกล ใน MATLAB Simulink ดังรูปที่ 4.1 โดยใช้สมการ (3.4) ที่ได้กล่าวไปก่อนหน้านี้ พารามิเตอร์ของมอเตอร์[18] ที่จะนำมาใช้ คือ $L = 9.6 \times 10^{-10}$ H, $R = 0.6 \Omega$, $J = 0.004 \text{ kgm}^2$, $b = 0.0185 \text{ Nm/(rad/s)}$, $r = 7.5$ และ $k = 0.065 \text{ V/rpm}$ ในชุดนี้ใช้มอเตอร์ข้างละ 2 ตัว จากผลการทดลองจะเห็นได้ว่า การควบคุมแบบ PID สามารถตอบสนองได้เร็วกว่า และเกิด Overshoot น้อยกว่า ดังรูปที่ 4.2



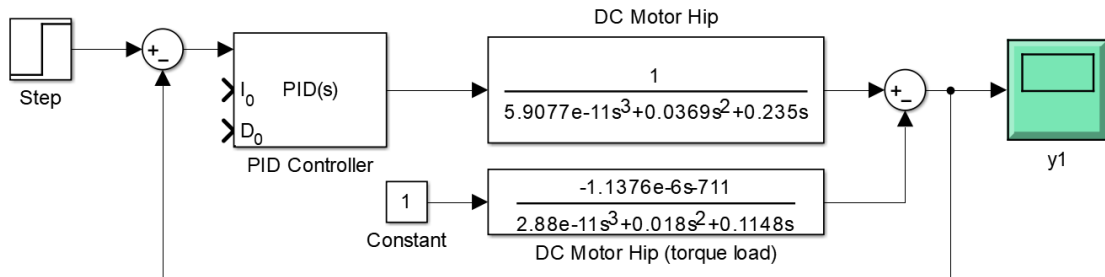
รูปที่ 4.1 การทดลองการควบคุม PID โดยใช้สมการมอเตอร์ (3.4)



รูปที่ 4.2 กราฟเปรียบเทียบระหว่างผลการตอบสนองของแบบจำลองมอเตอร์กระแสตรงที่ไม่ใช้ส่วนควบคุม PID (เส้นประสีเขียว) และกราฟผลการตอบสนองของแบบจำลองมอเตอร์กระแสตรงที่ใช้การควบคุม PID (เส้นสีน้ำเงิน) โดยกราฟเส้นสีแดง คือ อินพุต

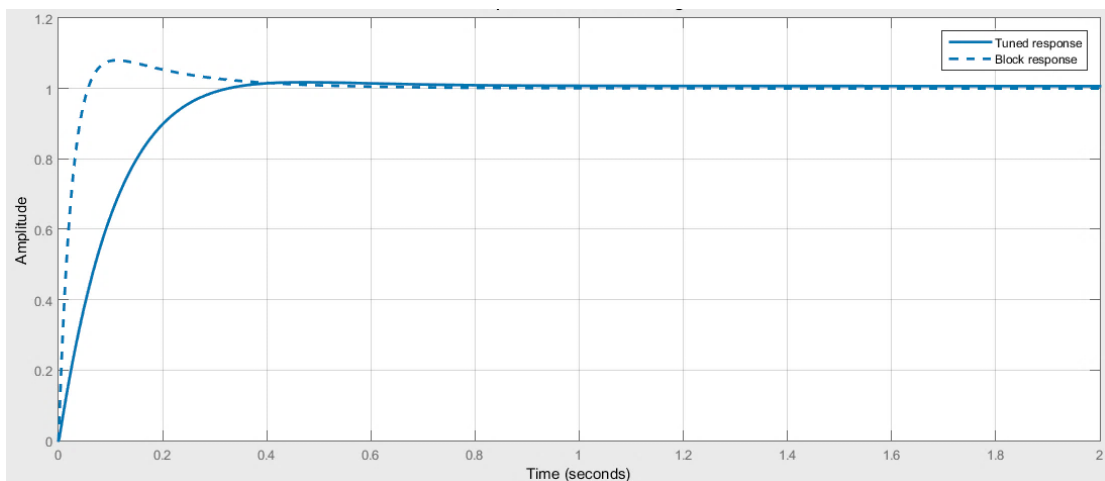
4.1.2 การทดลองกับมอเตอร์ที่มีโหลดทางกล

ทำการทดลองใน MATLAB Simulink ดังรูปที่ 4.3 โดยการสร้างตัวควบคุม PID ของ Exoskeleton actuator โดยใช้แบบจำลองมอเตอร์ในสมการ (3.6) แทนค่าพารามิเตอร์เช่นเดียวกับการทดลองในหัวข้อ 4.1.1

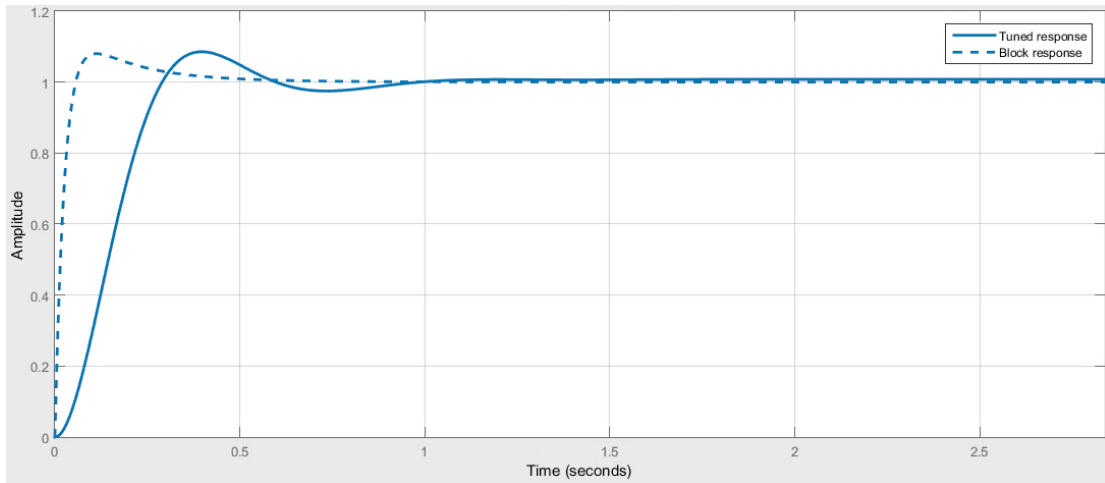


รูปที่ 4.3 การใช้โมเดลมอเตอร์ทางคณิตศาสตร์ โดยใช้สมการ (3.6) กับตัวควบคุม PID

ผลการทดลองดังรูปที่ 4.4 และ 4.5 แสดงให้เห็นถึงผลการตอบสนองของมอเตอร์ เมื่อป้อนอินพุตของมอเตอร์เป็น Unit step โดยเส้นประ (Block response) คือ ผลการตอบสนองของมอเตอร์ก่อนปรับค่าพารามิเตอร์ PID และเส้นทึบ (Tuned response) คือ ผลการตอบสนองของมอเตอร์ที่ปรับค่าพารามิเตอร์ PID แล้ว

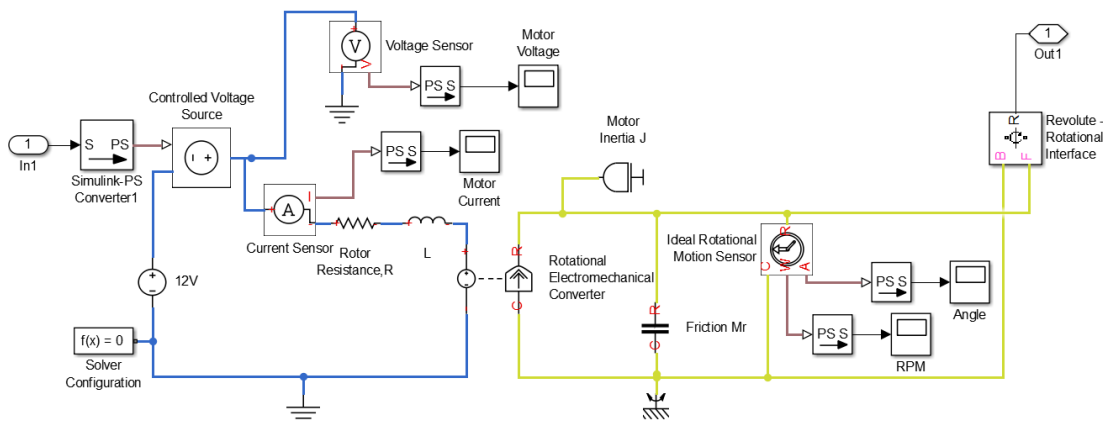


รูปที่ 4.4 ผลการตอบสนองของ DC motor model ส่วนสะโพก โดยใช้ตัวควบคุม PID



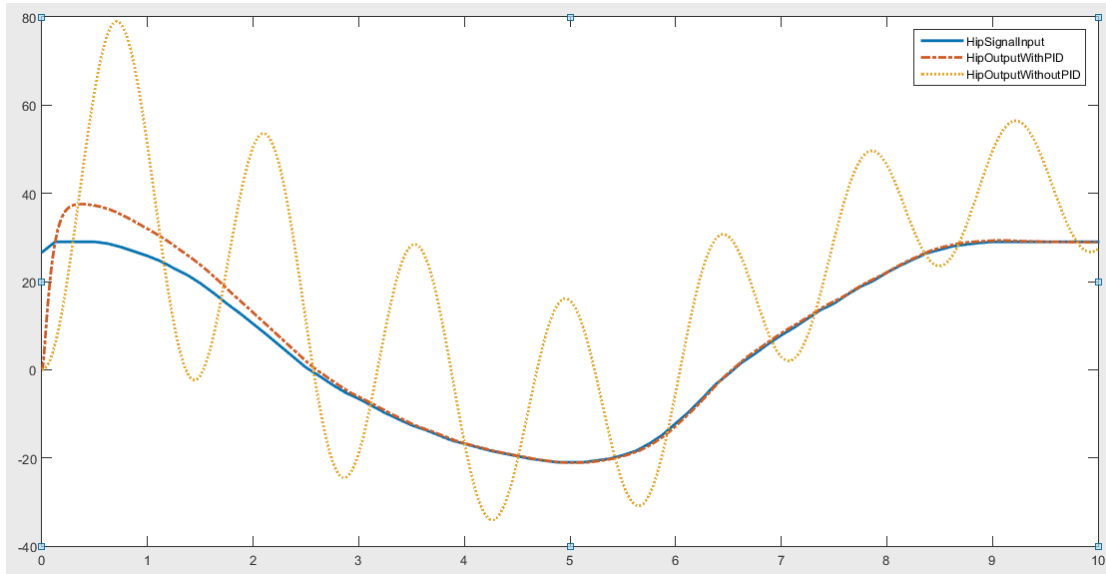
รูปที่ 4.5 ผลการตอบสนองของ DC motor model ที่ข้อเท้า โดยใช้ตัวควบคุม PID

แบบจำลองของมอเตอร์ไฟฟ้ากระแสตรงที่ใช้ตัวควบคุม PID จะมีผลการตอบสนองที่เร็วกว่าและมีโอเวอร์ชูตเกิดน้อยกว่ามอเตอร์ไฟฟ้ากระแสตรงที่ไม่มีตัวควบคุม PID การลดการเกิดโอเวอร์ชูต ก็เพื่อให้เมื่อนำไปใช้ในชุด Exoskeleton ไม่เกิดการกระตุกในการทำงานช่วง Swing phase จากการทดลองปรับค่าพารามิเตอร์ใน MATLAB ใช้ค่า $K_p = 2.69$, $K_i = 0.23$, $K_d = 0.34$ and Filter coefficients (N) = 1143 สำหรับตัวควบคุม PID ของมอเตอร์ส่วนสะโพก และใช้ค่า $K_p = 3.25$, $K_i = 2.42$, $K_d = 1.62$ and Filter coefficients (N) = 5081 สำหรับมอเตอร์ที่ข้อเท้า

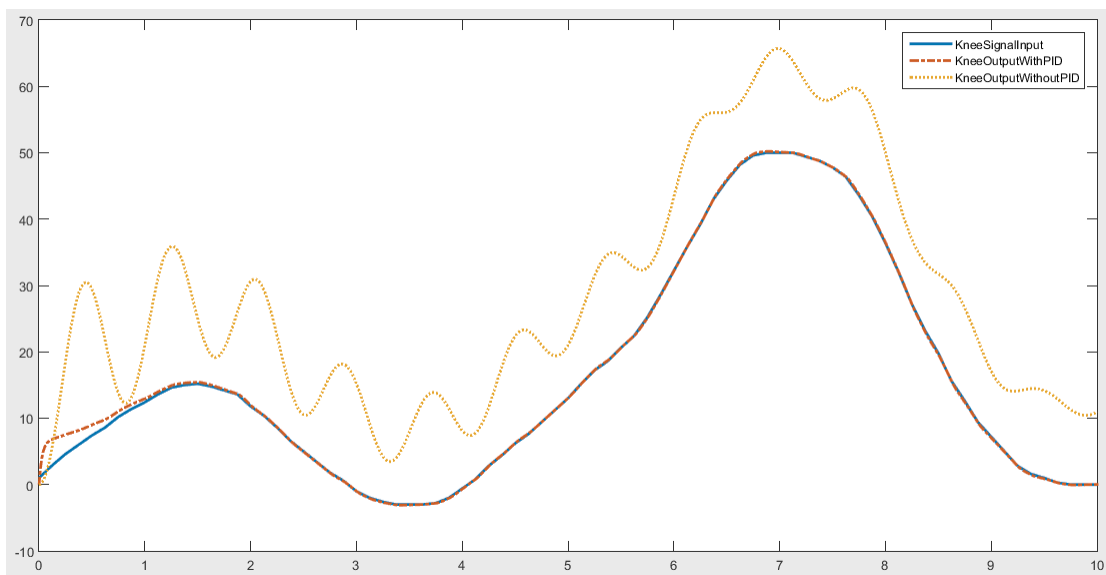


รูปที่ 4.6 DC motor ที่ทดลองใน Simulink

ส่วนผลการทดลองดังรูปที่ 4.7 และรูปที่ 4.8 แสดงให้เห็นถึงผลการตอบสนองของมอเตอร์เมื่อมีการป้อนอินพุตเป็นมุมมองศาที่ใช้ในการเดิน



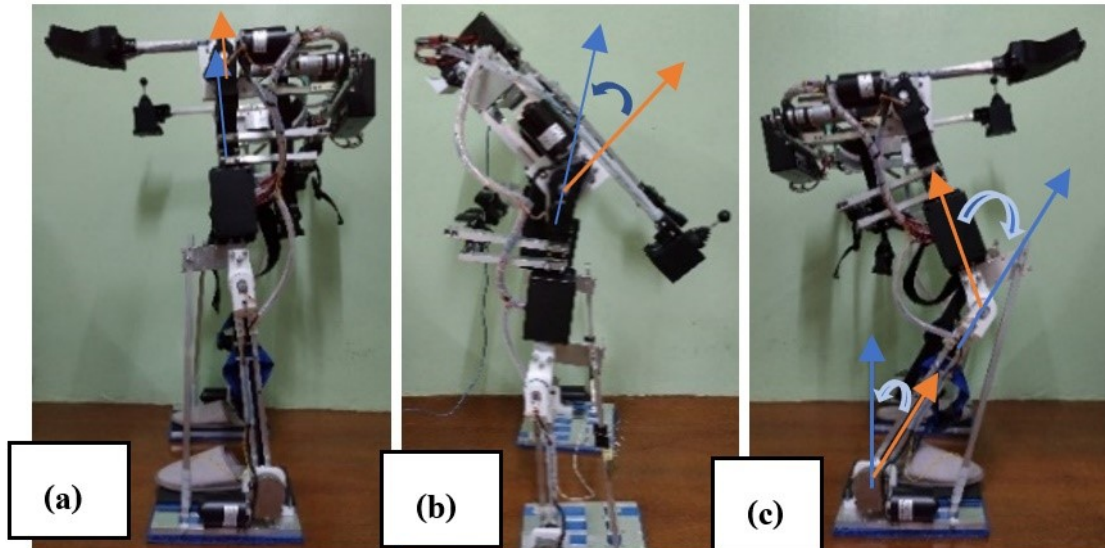
รูปที่ 4.7 ผลการตอบสนองทางเวลาของแบบจำลองมอเตอร์กระแสตรงที่สะโพกโดยใช้ตัวควบคุมPID



รูปที่ 4.8 ผลการตอบสนองทางเวลาของแบบจำลองมอเตอร์กระแสตรงที่ข้อเท้าโดยใช้ตัวควบคุม PID

จากนั้นจึงนำหลักการของ PID Controller และการออกแบบมอเตอร์ที่กล่าวมาก่อนหน้า มาประยุกต์สร้างชุด Exoskeleton สำหรับการควบคุมมอเตอร์ที่จะนำไปใช้งานกับชุดจริงนั้น จะทำการส่งสัญญาณควบคุมมอเตอร์ทุกๆ 0.5 วินาที โดยมีส่วนประกอบต่างๆ คือ 1. Potentiometer ใช้ในการวัดมุมองศาการหมุนของมอเตอร์แต่ละตัว 2. Gyroscope MPU6050 จะทำหน้าที่วัด Zero Moment Point ของชุด Exoskeleton ตรวจจับว่าชุดมีการเอียงจนเสียสมดุลหรือไม่ ในช่วง Terminal Stance phase ถ้าชุดเริ่มมีการโน้มไปข้างหน้ามากเกินไปเกินองศาที่กำหนดไว้ ก็จะทำให้มอเตอร์ในส่วนข้อเท้าและส่วนสะโพกหมุนกลับไปในทางที่กลับมาสมดุลอีกครั้ง 3. ไมโครคอนโทรลเลอร์ Arduino

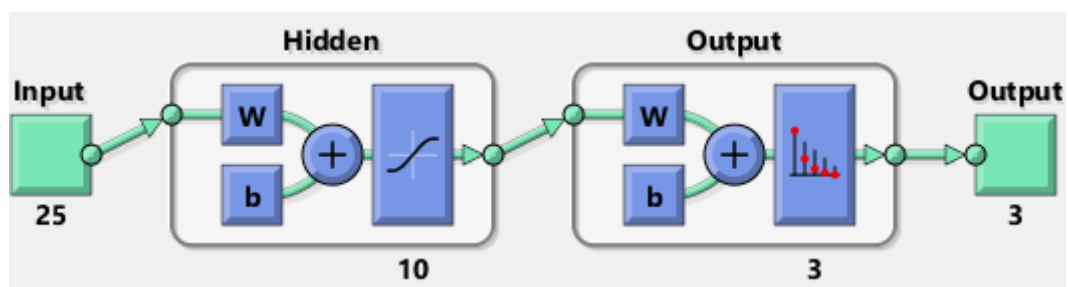
Mega 2560 ใช้ในการประมวลผล และ 4. DC Wiper Motor ขนาด 12V 55rpm 4 ตัว และ DC gear motor จำนวน 2 ตัวในส่วนสะโพก และ 5. ใช้บอร์ดขับเคลื่อนมอเตอร์กระแสตรง เบอร์ MD13S



รูปที่ 4.9 ชุด Exoskeleton ขยับสะโพก (q_1), เข่า (q_2) และข้อเท้า (q_3) ตามที่ได้กำหนดค่าไว้ (a) $q_1 = 0$ องศา, $q_2 = 3$ องศา และ $q_3 = -5$ องศา, (b) $q_1 = -30$ องศา, (c) $q_2 = 50$ องศา และ $q_3 = -30$ องศา

4.2 Backpropagation Neural Network และ Walking Assistance system สำหรับการจำแนกท่าทาง

ในส่วนของตัวจำแนกท่าทางที่ใช้ คือ แบบจำลอง neural network จะทำการควบคุมคำสั่งของชุด โดยจะสั่งท่าทางไปยัง trajectory controller เพื่อที่จะฝึกสอนโครงข่ายให้สามารถจดจำสัญญาณ sEMG ได้ ในวิทยานิพนธ์นี้ใช้ dataset จาก [19] ประกอบด้วยตัวอย่างจากผู้ทดลอง 11 คน ที่ได้รับการวินิจฉัยจากผู้เชี่ยวชาญแล้วว่าส่วนขาปกติ Dataset นี้ประกอบด้วย 3 ท่าทาง คือ ท่าเดิน, ท่านั่ง และท่ายืน โดยใช้ MWX8 Data log Biometrics, 14 bits resolution, sampling rate at 1 kHz ในการเก็บข้อมูล



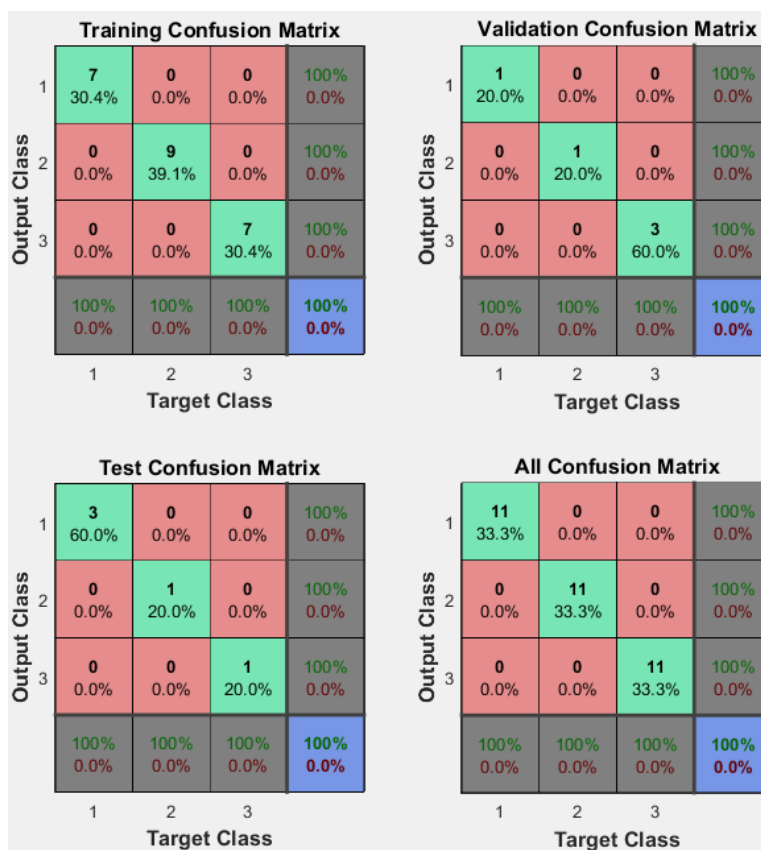
รูปที่ 4.10 Backpropagation neural network ใน MATLAB

ทำการทดลอง เพื่อหาว่าข้อมูลอินพุตที่ผ่าน feature extraction แบบใดที่ฝึกสอนให้กับ Backpropagation Neural Network แล้วจะให้เปอร์เซ็นต์การตอบถูกมากที่สุด โดยจะทำการทดลองหา Confusion Matrix 20 ครั้งแล้วหาค่าเฉลี่ย

จากการทดลองในตารางที่ 4.1 ทำให้สามารถออกแบบให้ Neural Network ที่ใช้ในส่วนนี้มี 25 input เป็น two-layer feed-forward network คือประกอบด้วย hidden layer กับ output layer โดยที่ hidden layer มี 10 node ใช้ sigmoid function, ที่ Output layers มี 3 node ใช้ SoftMax function และทำการฝึกสอนโดยใช้ scaled conjugate gradient backpropagation ในงานวิจัยนี้ ใช้ offline learning neural network

เวลาที่ใช้ในการแปลง Feature extraction สำหรับงานนี้คือ 4.031 วินาที (เวลาที่ใช้ส่วนใหญ่จะเสียไปกับการเปิดไฟล์อินพุต ส่วนเวลาที่ใช้ในการคำนวณ Feature Extraction เช่น rms ใช้ 0.002s, var ใช้ 0.004s) โดยใช้การคำนวณใน MATLAB บนคอมพิวเตอร์ Core i5-7200U 2.50GHz, RAM 12GB ส่วนในการพัฒนาเพื่อไปใช้งานต่อไป อาจจะนำไปใช้งานบน Single Board Computer หรือ Signal Processing Board เพื่อให้การคำนวณค่า Feature extraction สามารถทำได้เร็วขึ้น

Confusion Matrix เป็นตัวชี้วัดที่จะบ่งบอกว่าตัวจำแนกประเภทนั้นมีค่าเปอร์เซ็นต์ที่ถูกต้องและไม่ถูกต้องเท่าไร เปอร์เซ็นต์ของการจำแนกประเภทที่ถูกต้องคือ ช่องสีเขียว และเปอร์เซ็นต์ของการจำแนกประเภทที่ไม่ถูกต้องคือ ช่องสีแดง ในรูปที่ 4.11



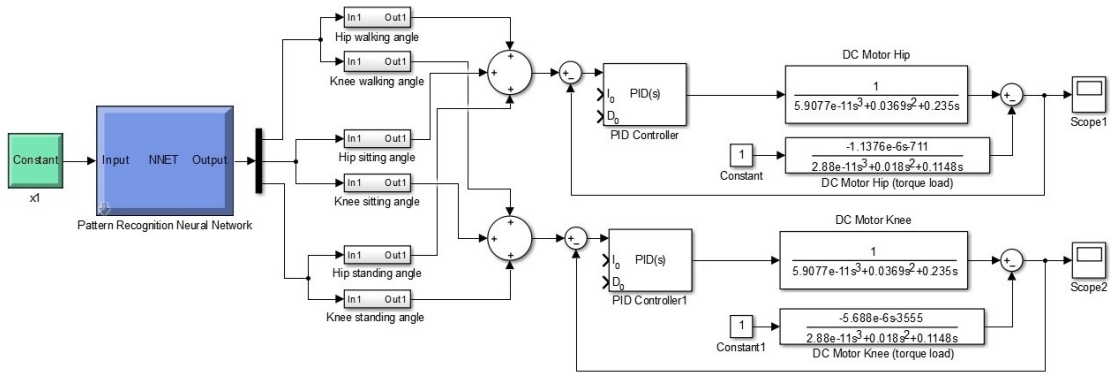
รูปที่ 4.11 Confusion Matrix ของตัวจำแนกทาง Backpropagation Neural Network

ตารางที่ 4.1 เปอร์เซ็นต์การตอบถูกของการจำแนกท่าทาง (Percent Correct Classification)

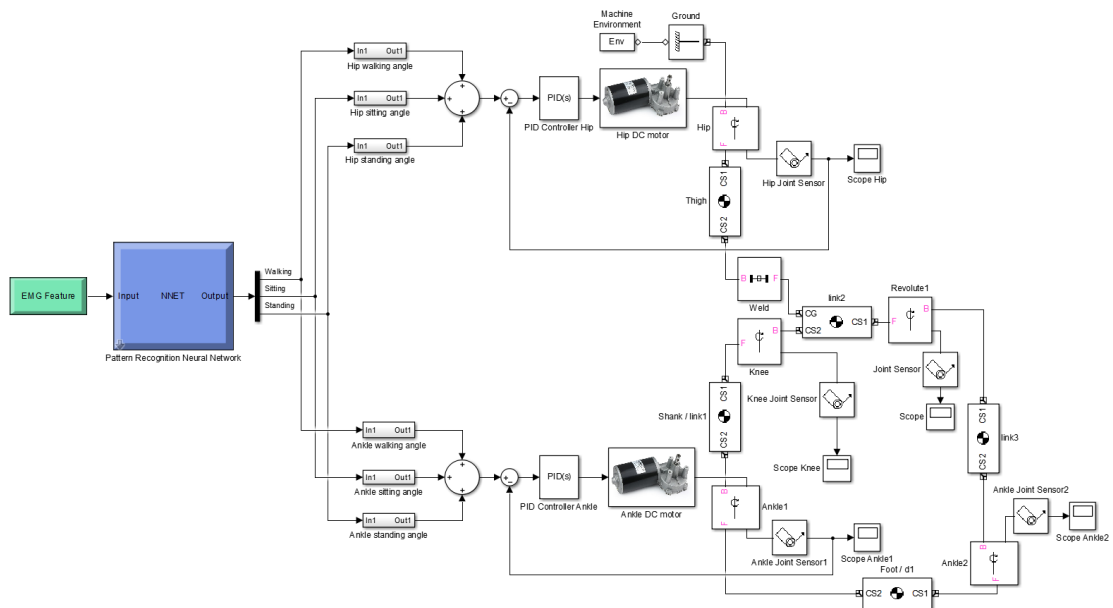
Feature Extraction	เปอร์เซ็นต์การตอบถูกของการจำแนกท่าทาง
RMS (5 input)	91.52
MAV (5 input)	95.25
SSI (5 input)	94.35
VAR (5 input)	78.29
IAV (5 input)	98.39
Mean frequency (5 input)	81.10
Median frequency (5 input)	69.37
All feature in time domain (25 input)	99.39
All feature in frequency domain (10 input)	85.15
All feature extraction (35 input)	95.15

ผลการทดลอง feature extraction ที่ได้แสดงในตารางที่ 4.1 จะเห็นได้ว่าการใช้ All feature in time domain (ใช้เฉพาะ feature ของ time domain ร่วมกัน) มีผล recognition rate ที่ตอบถูกสูงถึง 99.39% ซึ่งเราจะใช้ feature นี้ในชุด lower limb exoskeleton

จากนั้น จึงนำ Neural Network ที่ได้มาใช้กับมอเตอร์และกลไกที่ได้ออกแบบไว้ ไปทำการทดลองบน Simulink ดังรูปที่ 4.12 จะเป็นการสร้างระบบที่ใช้แบบจำลองมอเตอร์ตามสมการที่ (3.6) ซึ่งยังเป็นแบบจำลองที่มีโหนดทางกล แต่เป็นโหนดทางกลแบบที่เป็นการประมาณค่าเท่านั้น จึงมีการปรับปรุงแบบจำลองบน Simulink เพื่อให้ได้แบบจำลองมอเตอร์ที่ใกล้เคียงกับมอเตอร์จริง ดังรูปที่ 4.13

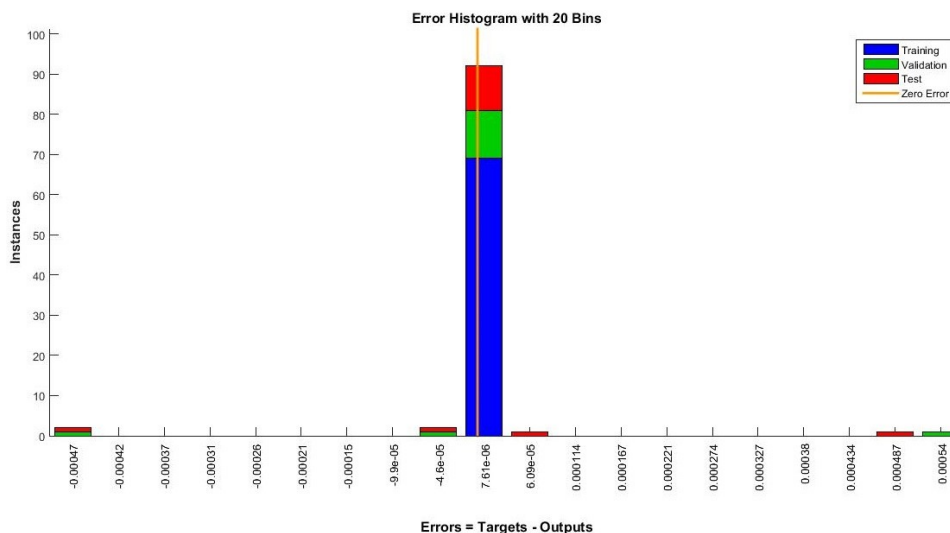


รูปที่ 4.12 Exoskeleton และ Neural Network ที่ทดลองใน Simulink



รูปที่ 4.13 Exoskeleton และ BP Neural Network ที่ทำการทดลองใน Simulink แบบที่ใช้มอเตอร์สองตัว สำหรับกลไกแบบ Four Bar Linkages

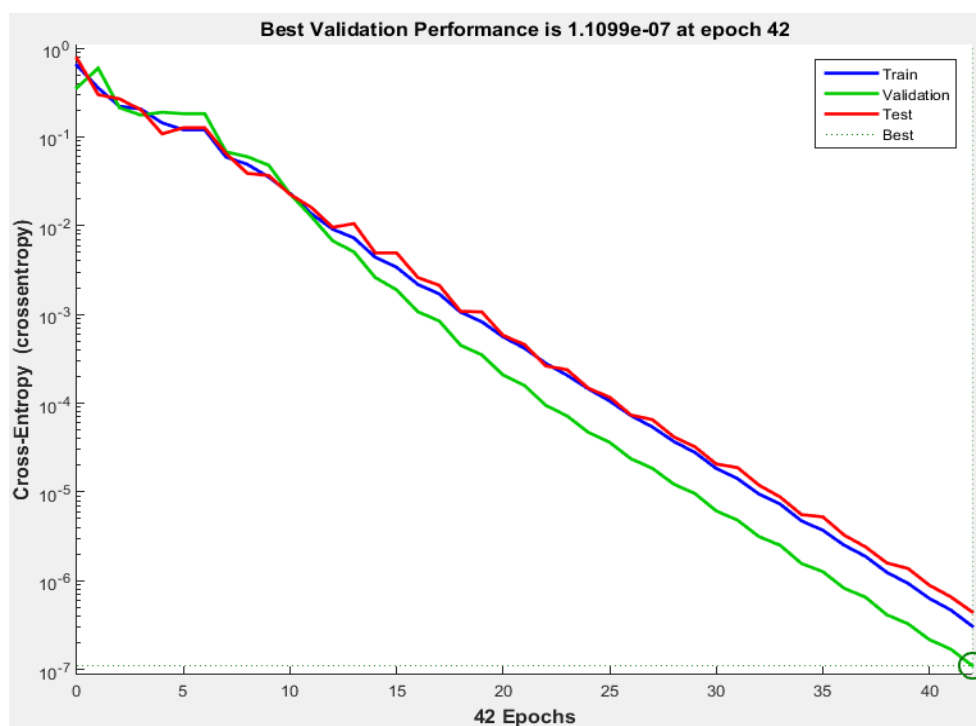
เมื่อทดสอบการใช้งานข้อมูลในช่วง 50% แรกของรอบการเดิน (1/2 Gait cycle), 25% แรกของรอบการเดิน (1/4 Gait cycle) และ 12.5% แรกของรอบการเดิน (1/8 Gait cycle) เมื่อมีสัญญาณเพียงแค่ 12.5% แรกของรอบการเดิน ก็ทำให้ชุด Exoskeleton สามารถจำแนกได้ว่าเป็นท่าทางไหน ไม่จำเป็นต้องรอสัญญาณให้ครบรอบการเดิน



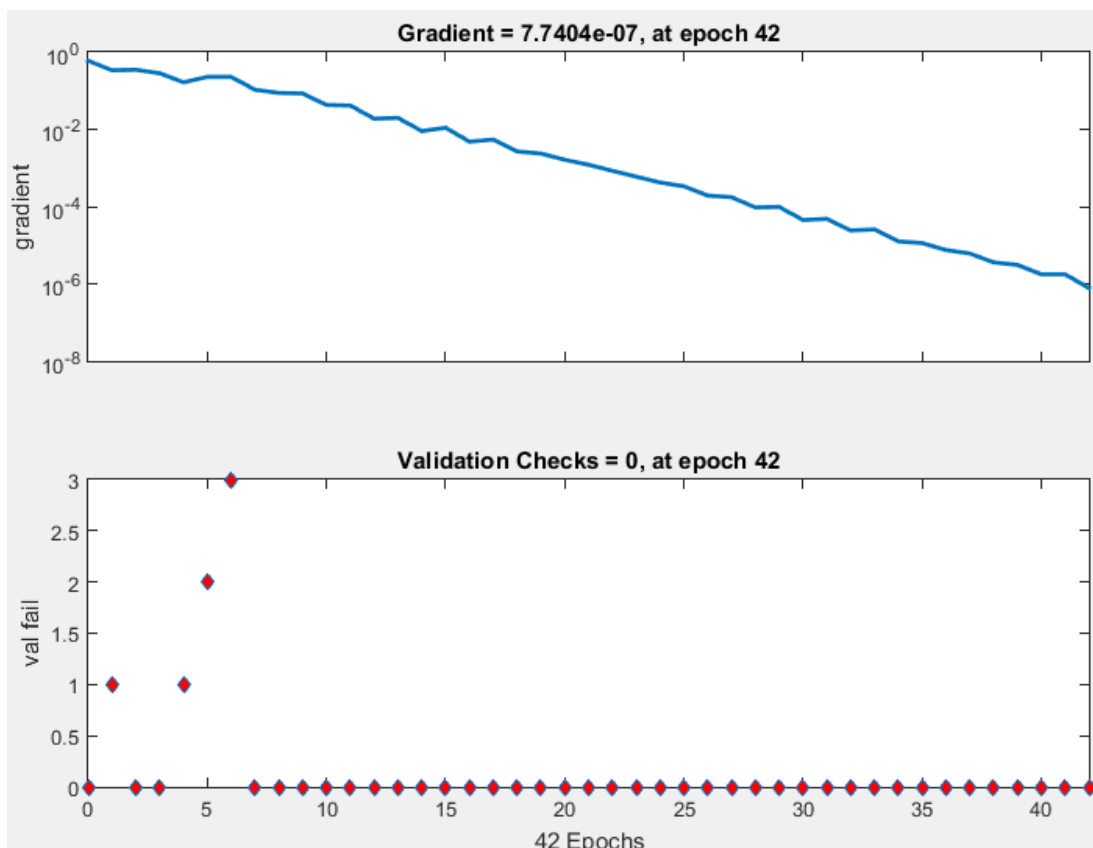
รูปที่ 4.14 Error Histogram ของ Classification Backpropagation Neural Network

สำหรับ Error Histogram ในรูปที่ 4.14 แถบสีฟ้าแสดงถึงข้อมูลการฝึกอบรม (training data), แถบสีเขียวแสดงถึงข้อมูลการตรวจสอบความถูกต้อง (validation data) และแถบสีแดงแสดงถึงข้อมูลการทดสอบ (testing data) Histogram สามารถบ่งบอกถึงค่าผิดปกติ ซึ่งเป็นจุดข้อมูลที่มีความพืดน้อยกว่าข้อมูลส่วนใหญ่อย่างมีนัยสำคัญ

กราฟ Performance ในรูปที่ 4.15 จะถูกวัดโดยใช้ค่า mean squared error ซึ่งควรจะลดลงเมื่อโครงข่ายได้รับการฝึกสอนมากขึ้น ส่วนกราฟ Training State ของ Classification Backpropagation Neural Network ในรูปที่ 4.16 แสดงให้เห็นการลดลงของ Gradient ในการฝึกสอนข้อมูลแต่ละรอบ



รูปที่ 4.15 กราฟ Performance ของตัวจำแนกทำทาง Backpropagation Neural Network

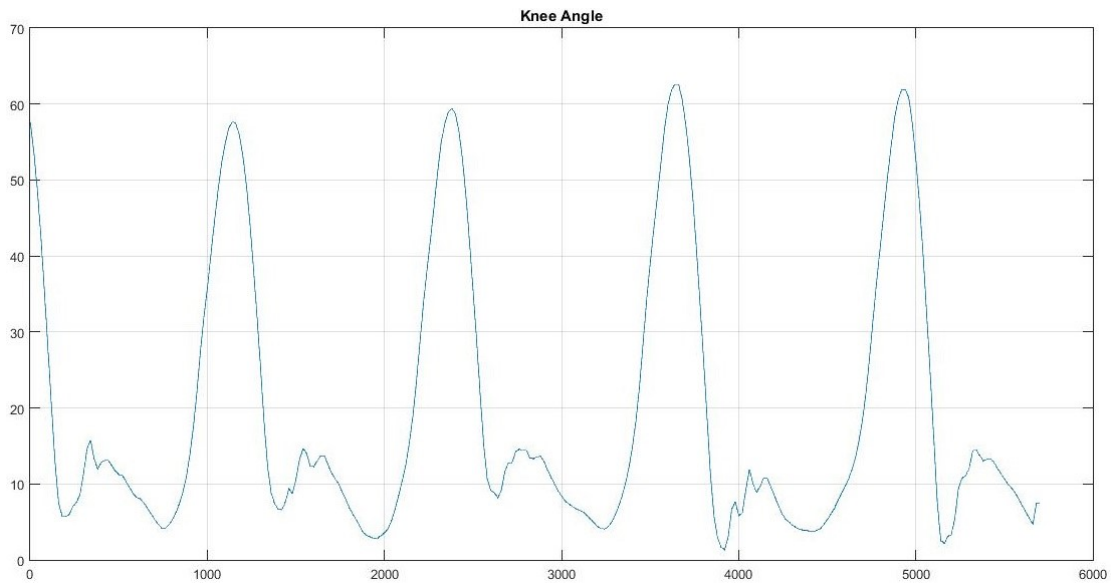


รูปที่ 4.16 Training state ของตัวจำแนกท่าทาง Backpropagation Neural Network

4.3 การสอนท่าทางให้ชุด Exoskeleton ในแต่ละข้อต่อในตัวควบคุมการเคลื่อนที่

จากทั้งหมดที่กล่าวมาก่อนหน้านั้น เป็นการที่ชุด Exoskeleton จะขยับตามท่าทางที่มีการบันทึกมุมมองเอาไว้แล้วล่วงหน้าทั้งสิ้น ในส่วนนี้จะเป็นการทดลองเพื่อพัฒนาการสอนท่าทางให้ชุด Exoskeleton โดยเราไม่จำเป็นจะต้องโปรแกรมมุมมองแค่รูปแบบเดียวอีกต่อไป แต่จะสามารถฝึกสอนให้ชุดสามารถเรียนรู้ท่าทางและเคลื่อนไหวได้ดียิ่งขึ้นต่อไปได้ ในเบื้องต้นจะทำการทดลองกับมุมมองของหัวเข้าในการเดินแบบต่อเนื่อง

การกำหนดค่า Time step ที่ใช้นั้น จะกำหนดให้ใน 1 รอบการเดินนั้นใช้เวลา 5 วินาที ส่วนการใช้มุมมองของหัวเข้าในการเดินแบบต่อเนื่องนั้น จะประกอบด้วยการเล่นหลายรอบ ซึ่งอาจจะมากกว่าหรือน้อยกว่า 5 วินาทีในแต่ละรอบการเดินก็ได้ เพราะคนแต่ละคน ย่อมใช้เวลาในการเล่นไม่เท่ากัน คนเดียวกันอาจจะใช้เวลาเดินในแต่ละช่วงเวลาไม่เท่ากันก็ได้ ค่า Time step ของการใช้มุมมองของหัวเข้าในการเดินแบบต่อเนื่องนั้นจะคิดจากค่า sampling rate ของเครื่องมือวัด นั่นคือ 1kHz ตัวอย่างเช่นในรูปที่ 4.17 ในหนึ่งรอบการเดินมี 1200 ค่า นั่นคือใช้เวลา 1.2 วินาที



รูปที่ 4.17 มุมองศาของหัวเข่าในการเดินแบบต่อเนื่อง

4.3.1 วิธีการ Curve Fitting

วิธีการ Curve Fitting แบบ Gaussian Model ที่ทำการทดลองเป็นดังสมการที่ (4.1)

$$f(x) = \sum_{i=1}^n a_i e^{-\left(\frac{x-b_i}{c_i}\right)^2}; n = 8 \quad (4.1)$$

จะได้ว่า

$$\begin{aligned} f(x) = & a_1 e^{-\left(\frac{x-b_1}{c_1}\right)^2} + a_2 e^{-\left(\frac{x-b_2}{c_2}\right)^2} + a_3 e^{-\left(\frac{x-b_3}{c_3}\right)^2} + a_4 e^{-\left(\frac{x-b_4}{c_4}\right)^2} \\ & + a_5 e^{-\left(\frac{x-b_5}{c_5}\right)^2} + a_6 e^{-\left(\frac{x-b_6}{c_6}\right)^2} + a_7 e^{-\left(\frac{x-b_7}{c_7}\right)^2} + a_8 e^{-\left(\frac{x-b_8}{c_8}\right)^2} \end{aligned} \quad (4.2)$$

นำสมการที่ (4.2) มาเขียนโค้ดใน MATLAB จะได้ว่า

General model Gauss8:

$$\begin{aligned} f(x) = & a_1 * \exp(-((x-b_1)/c_1)^2) + a_2 * \exp(-((x-b_2)/c_2)^2) + a_3 * \exp(-((x-b_3)/c_3)^2) \\ & + a_4 * \exp(-((x-b_4)/c_4)^2) + a_5 * \exp(-((x-b_5)/c_5)^2) + a_6 * \exp(-((x-b_6)/c_6)^2) \\ & + a_7 * \exp(-((x-b_7)/c_7)^2) + a_8 * \exp(-((x-b_8)/c_8)^2) \end{aligned}$$

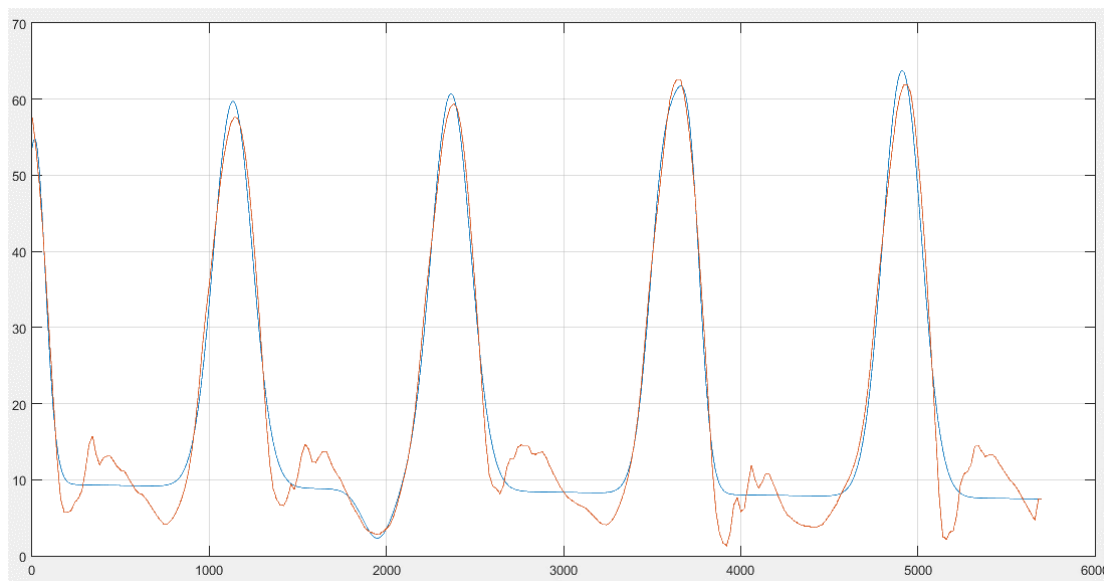
แทนค่าสัมประสิทธิ์จากตารางที่ 4.2 ลงในสมการที่ 4.2

ตารางที่ 4.2 ค่าสัมประสิทธิ์ของสมการ Gaussian Model 8

a1 = 30.49	b1 = 3717	c1 = 89.58
a2 = 56.05	b2 = 4912	c2 = 155
a3 = 52.15	b3 = 2365	c3 = 158.5
a4 = 50.71	b4 = 1134	c4 = 155.7
a5 = 45.36	b5 = 14.25	c5 = 85.24
a6 = -6.445	b6 = 1950	c6 = 119
a7 = 45.81	b7 = 3585	c7 = 132.6
a8 = 1.346e+04	b8 = -3.579e+05	c8 = 1.328e+05

ผลที่ได้: SSE: 5.778e+04, R-square: 0.9681

Adjusted R-square: 0.9679, RMSE: 3.191



รูปที่ 4.18 เวก์พุดของวิธีการ Curve Fitting แบบ Gaussian Model (เส้นสีน้ำเงิน) เทียบกับมุมมองของค่าของหัวเข้าในการเดินแบบต่อเนื่อง (เส้นสีแดง)

4.3.2 วิธีการ Sum of Sine

วิธีการ Sum of sine ที่ทำการทดลองเป็นดังสมการที่ (4.3)

$$f(x) = \sum_{i=1}^n a_i \sin(b_i x + c_i) \quad ; n = 8 \quad (4.3)$$

จะได้ว่า

$$\begin{aligned} f(x) = & a_1 \sin(b_1 x + c_1) + a_2 \sin(b_2 x + c_2) + a_3 \sin(b_3 x + c_3) \\ & + a_4 \sin(b_4 x + c_4) + a_5 \sin(b_5 x + c_5) + a_6 \sin(b_6 x + c_6) \\ & + a_7 \sin(b_7 x + c_7) + a_8 \sin(b_8 x + c_8) \end{aligned} \quad (4.4)$$

นำสมการที่ (4.4) มาเขียนโค้ดใน MATLAB จะได้ว่า

General model Sin8:

$$\begin{aligned} f(x) = & a1*\sin(b1*x+c1) + a2*\sin(b2*x+c2) + a3*\sin(b3*x+c3) \\ & + a4*\sin(b4*x+c4) + a5*\sin(b5*x+c5) + a6*\sin(b6*x+c6) \\ & + a7*\sin(b7*x+c7) + a8*\sin(b8*x+c8) \end{aligned}$$

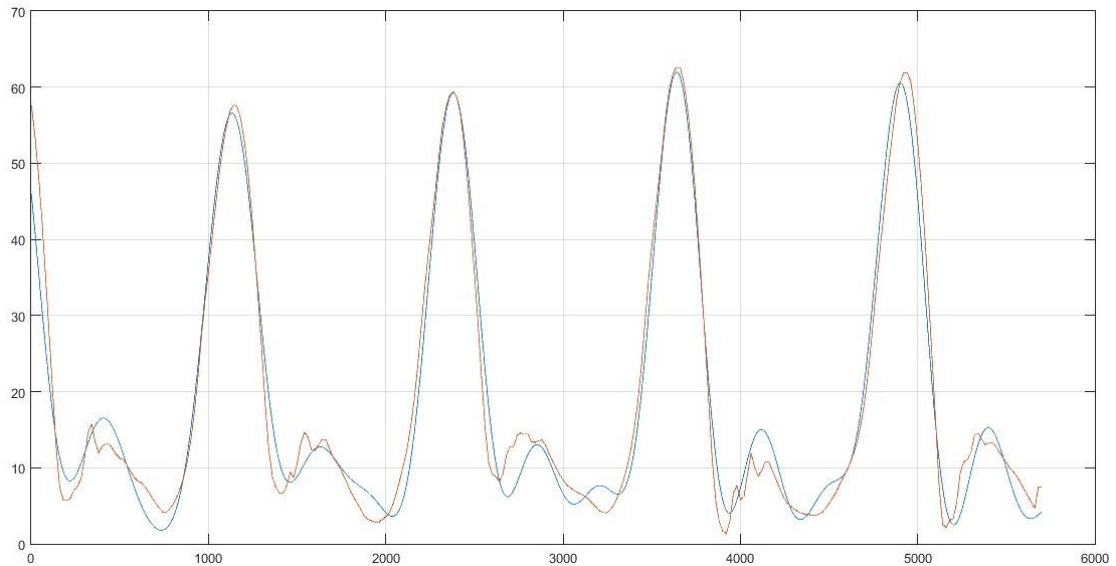
แทนค่าสัมประสิทธิ์จากตารางที่ 4.3 ลงในสมการที่ 4.4

ตารางที่ 4.3 ค่าสัมประสิทธิ์ของสมการ Sum of Sine

a1 = 239.2	b1 = 0.0002445	c1 = 1.728
a2 = - 9.005	b2 = 0.005129	c2 = 1.751
a3 = 13.99	b3 = 0.01003	c3 = - 3.356
a4 = 29.1	b4 = 0.005072	c4 = - 4.347
a5 = 216	b5 = 0.0002657	c5 = 4.836
a6 = 5.371	b6 = 0.01509	c6 = 2.864
a7 = 2.481	b7 = 0.01414	c7 = 0.392
a8 = - 0.7118	b8 = 0.003766	c8 = - 14.89

ผลที่ได้: SSE: 5.963e+04, R-square: 0.9671

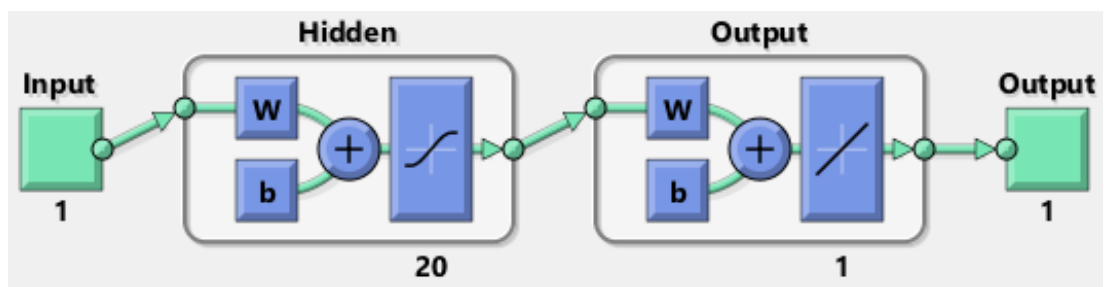
Adjusted R-square: 0.9669, RMSE: 3.241



รูปที่ 4.19 เอกลักษณ์ของวิธีการ Curve Fitting แบบ Sum of Sine (เส้นสีน้ำเงิน) เทียบกับมุมมองคาของหัวเข้าในการเดินแบบต่อเนื่อง (เส้นสีแดง)

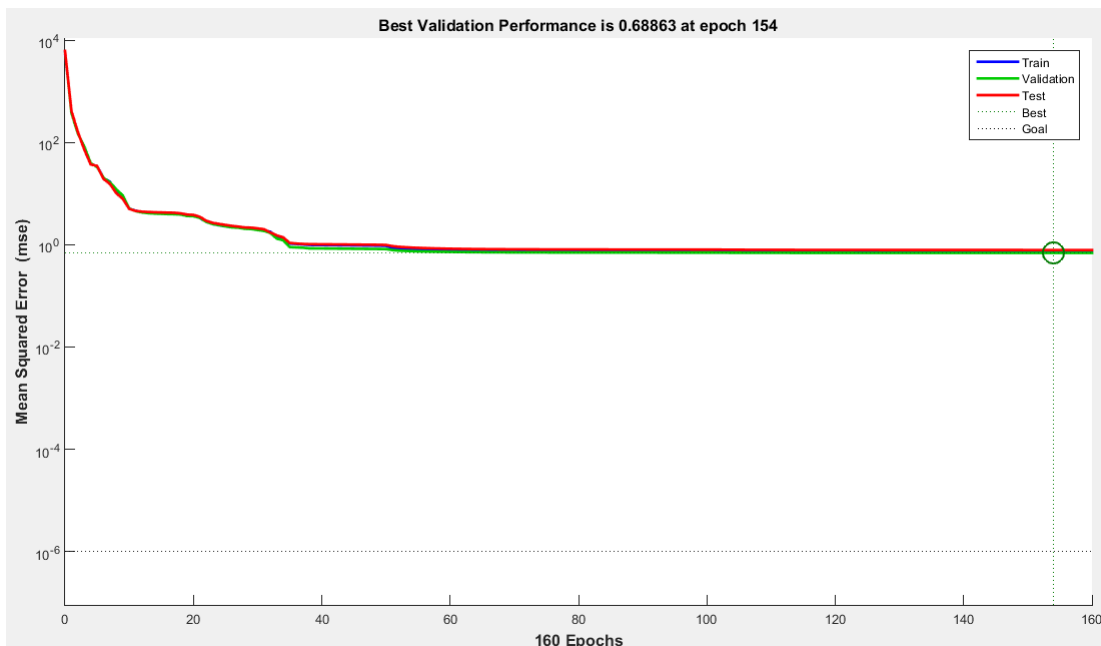
4.3.3 Backpropagation Neuron Network แบบ Levenberg-Marquardt

ในการสร้าง Backpropagation Neuron Network แบบ Levenberg-Marquardt จะใช้คำสั่ง Fitnet ใน MATLAB ดังรูปที่ 4.20 เพื่อใช้ในการ Fit ค่า มีโครงสร้างแบบ 20-1 (1 input, 20 layer)

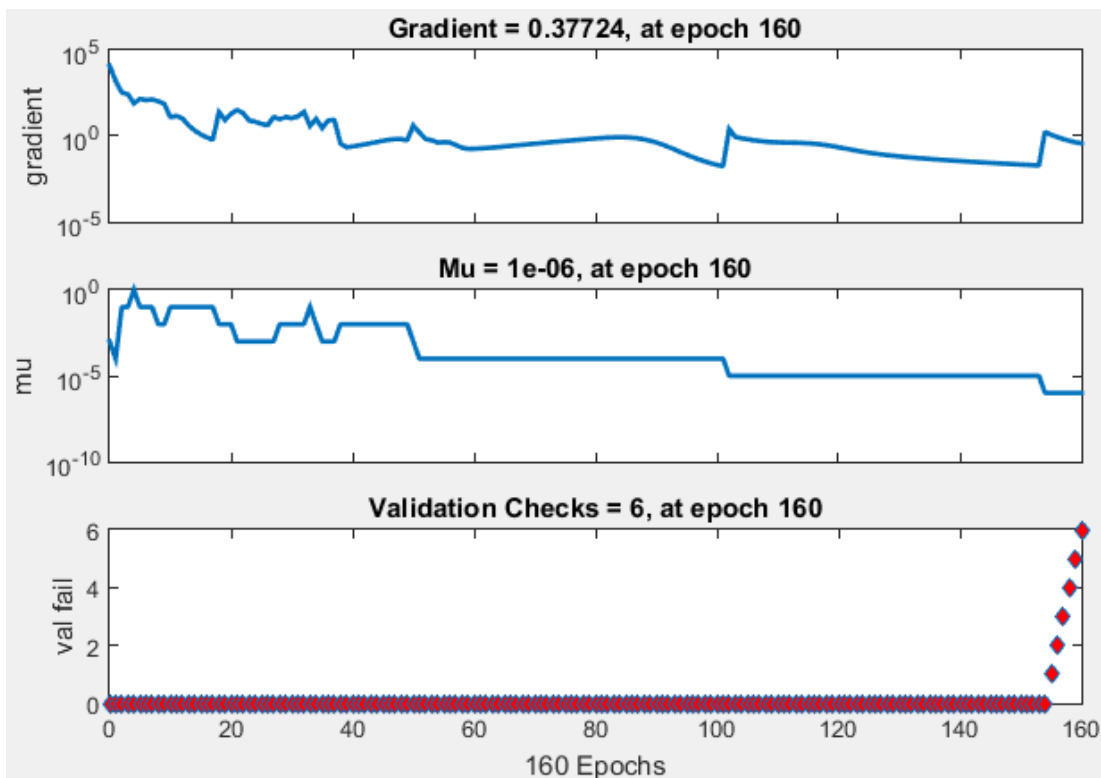


รูปที่ 4.20 Fitting Backpropagation Neuron Network แบบ Levenberg-Marquardt

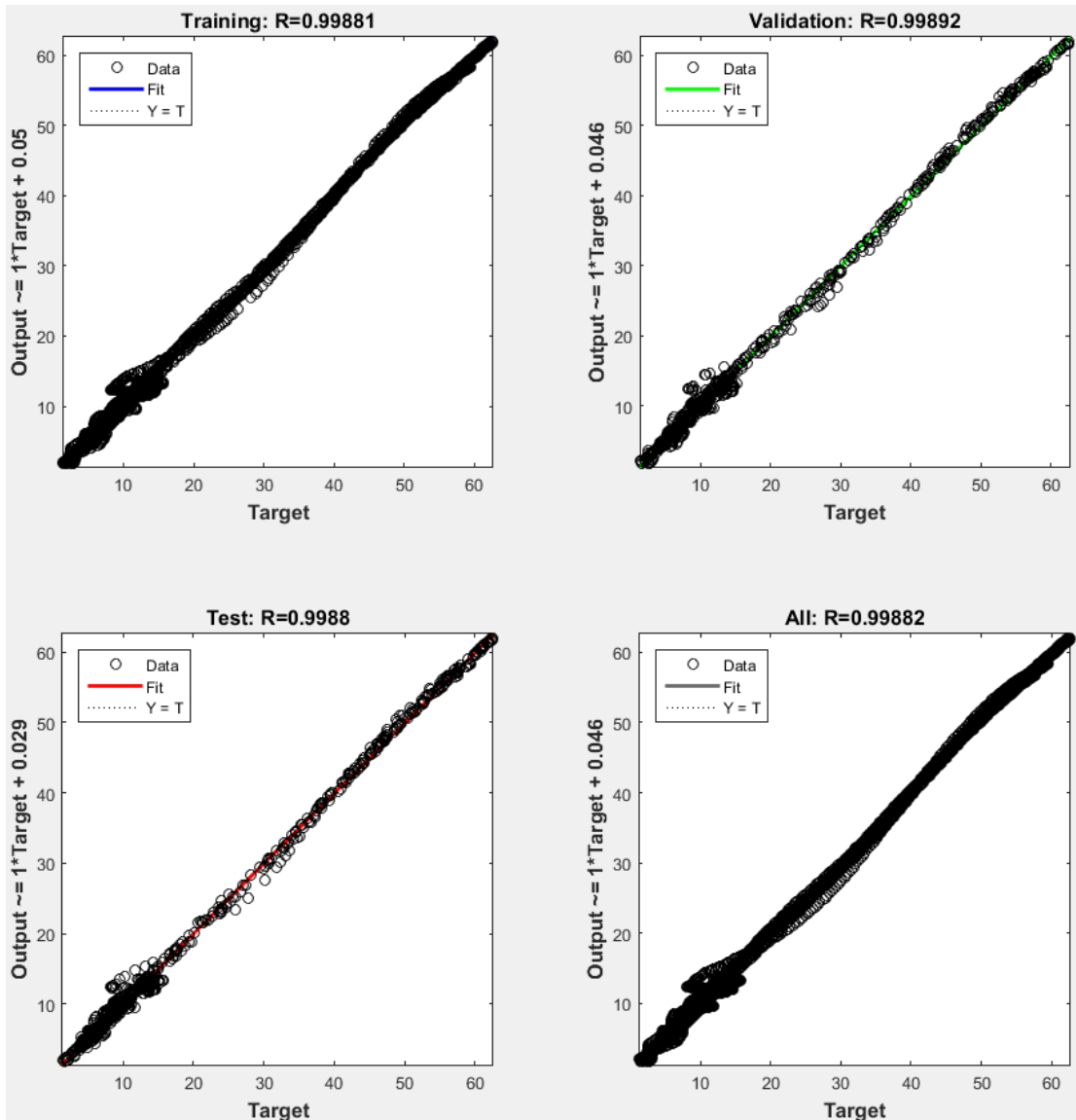
โดยใช้ Hyperbolic tangent sigmoid transfer function ในแต่ละโหนดของ Hidden layer และใช้ Linear transfer function ที่โหนด Output layer



รูปที่ 4.21 กราฟ Performance ของ Fitting BPNN แบบ Levenberg-Marquardt

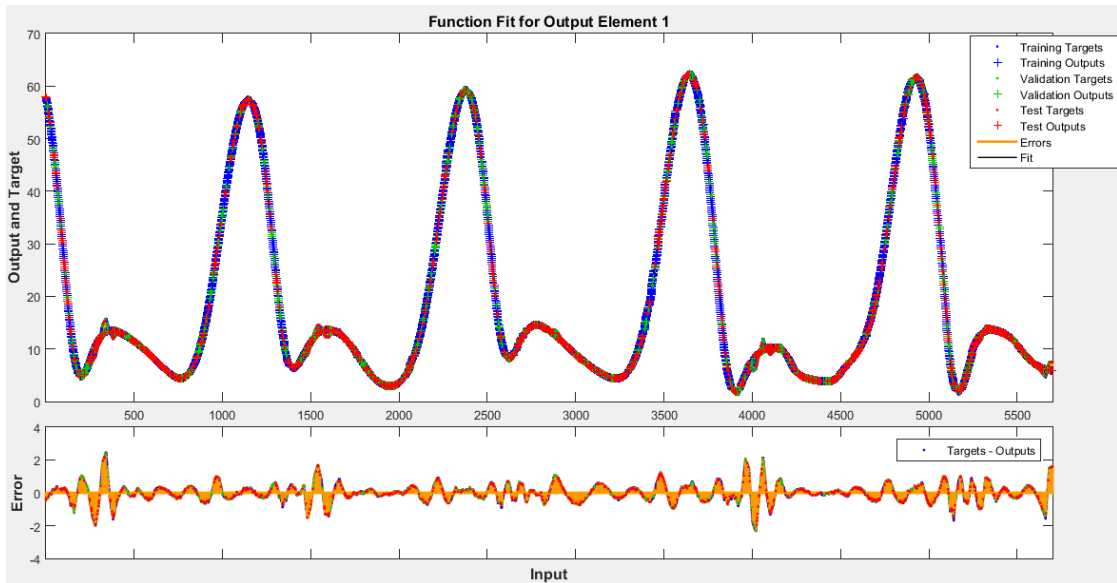


รูปที่ 4.22 Training State ของ Fitting BPNN แบบ Levenberg-Marquardt

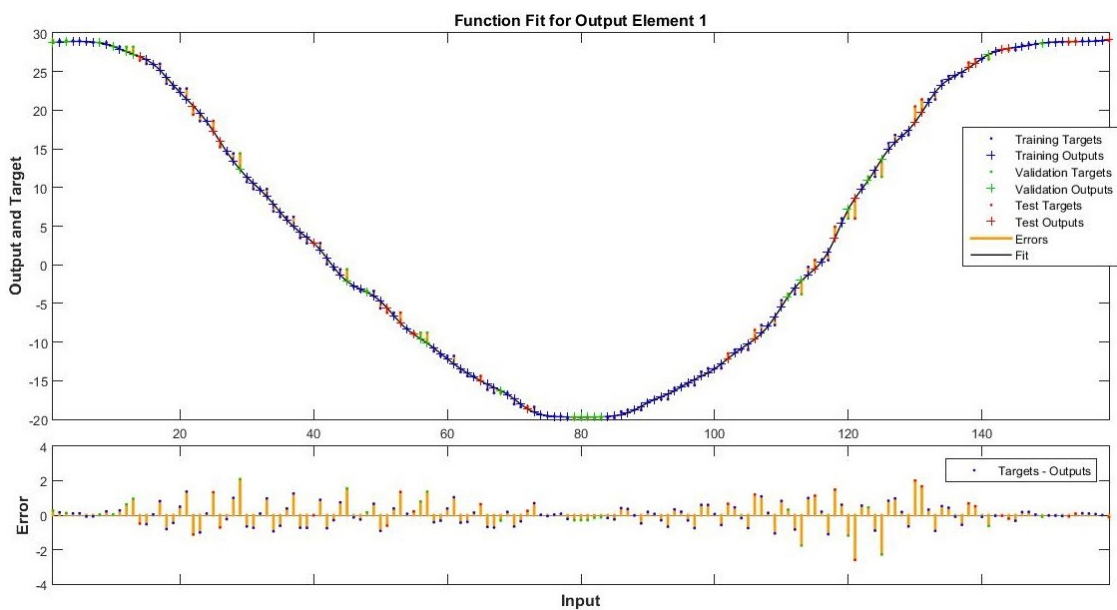


รูปที่ 4.23 กราฟ Training Regression ของ Fitting BPNN แบบ Levenberg-Marquardt

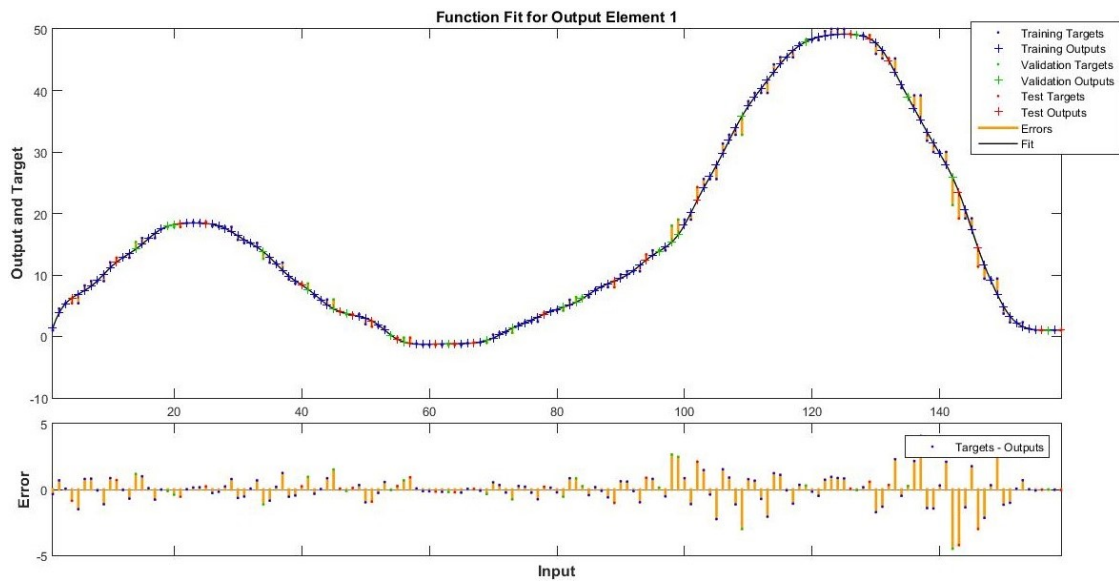
กราฟ Training Regression ในรูปที่ 4.23 เส้นปะในแต่ละจุดแสดงถึง ผลลัพธ์ที่สมบูรณ์แบบ(perfect result) – ผลลัพธ์(outputs) = เป้าหมาย(targets) เส้นทึบแสดงถึงเส้นการถดถอยเชิงเส้น (linear regression) ที่เหมาะสมที่สุดระหว่างเอาต์พุตและเป้าหมาย และค่า R เป็นการบ่งชี้ความสัมพันธ์ระหว่างผลลัพธ์และเป้าหมาย หาก R = 1 แสดงว่ามีความสัมพันธ์เชิงเส้นที่แน่นอนระหว่างเอาต์พุตและเป้าหมาย หาก R ใกล้เคียงกับศูนย์แสดงว่าไม่มีความสัมพันธ์เชิงเส้นระหว่างเอาต์พุตและเป้าหมาย



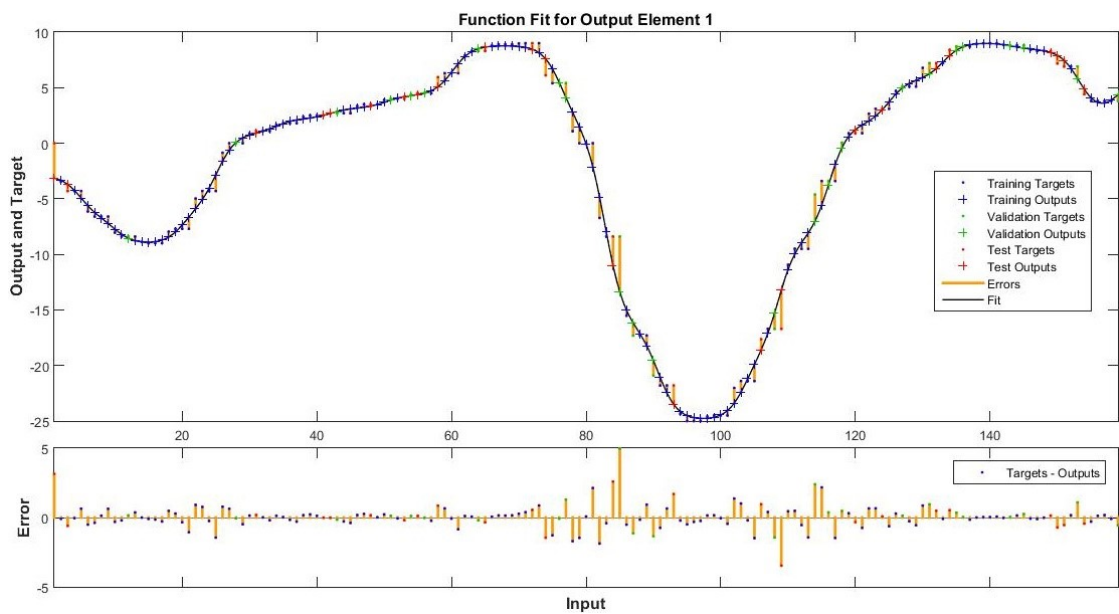
รูปที่ 4.24 การ Fit data ของมุมมองขาของหัวเข้าในการเดินแบบต่อเนื่อง โดยใช้ Fitting Backpropagation Neuron Network



รูปที่ 4.25 การ Fit data มุมมองขาของสะโพกในการเดิน โดยใช้ Fitting Backpropagation Neuron Network



รูปที่ 4.26 การ Fit data มุมมองของหัวเข้าในการเดิน โดยใช้ Fitting Backpropagation Neuron Network



รูปที่ 4.27 การ Fit data มุมมองของข้อเท้าในการเดิน โดยใช้ Fitting Backpropagation Neuron Network

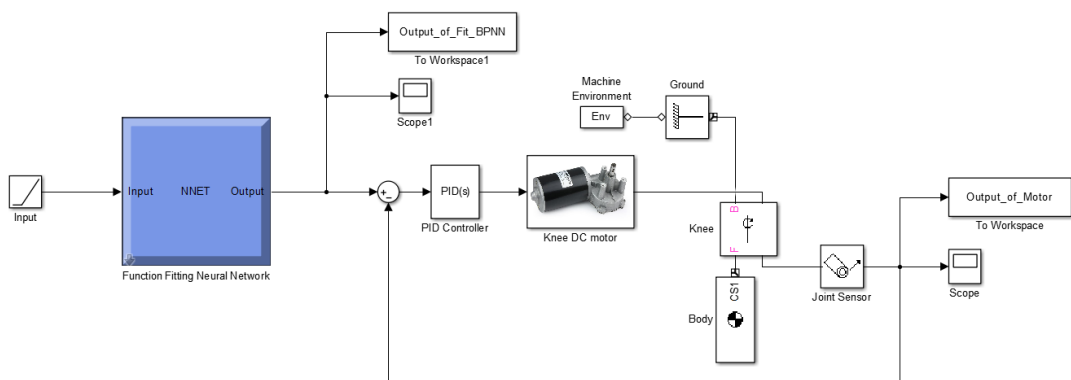
เพื่อหาว่าโมเดลไหนสามารถเรียนรู้ท่าทางได้ดีที่สุด จึงทำการทดลองเปรียบเทียบค่า Error ของแต่ละวิธี โดยจะคิดจากค่าสัมบูรณ์ของผลต่างระหว่าง Target และ Output ที่ได้ ผลการทดลองได้ถูกแสดงในตารางที่ 4.4 ซึ่งจากผลการทดลองจะเห็นได้ว่า Fitting Backpropagation Neuron Network ให้ค่าผิดพลาดน้อยที่สุด

ตารางที่ 4.4 เปรียบเทียบค่า Error ของ Curve Fitting และ Fitting Backpropagation Neuron Network สำหรับมุมมองของหัวเข่าในท่าทางการเดิน

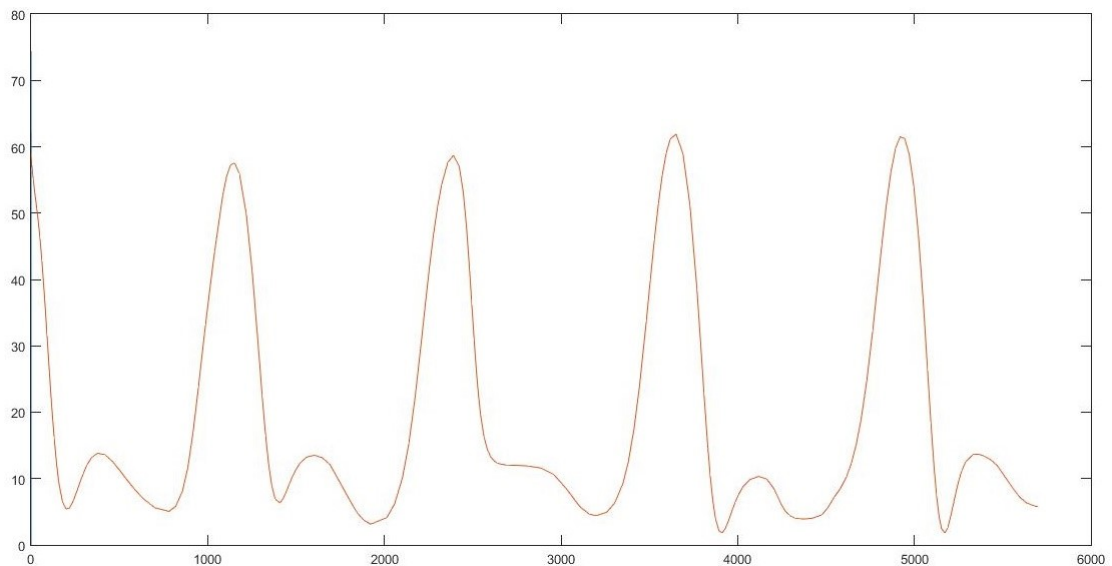
ค่า Error สูงสุดในแต่ละวิธีการ			
	Curve Fitting: General model Gauss8	Curve Fitting: Sum of Sine	Fitting Backpropagation Neuron Network (fitnet)
1	11.5	12.71	2.44
2	11.5	12.71	5.26
3	11.5	12.71	2.61
4	11.5	12.71	6.23
5	11.5	12.71	8.73
6	11.5	12.71	2.64
7	11.5	12.71	5.49
8	11.5	12.71	2.90
9	11.5	12.71	2.29
10	11.5	12.71	5.59
เฉลี่ย	11.5	12.71	4.418

4.4 การนำ Fitting Backpropagation Neuron Network ไปใช้งานกับชุด Lower Limb Exoskeleton

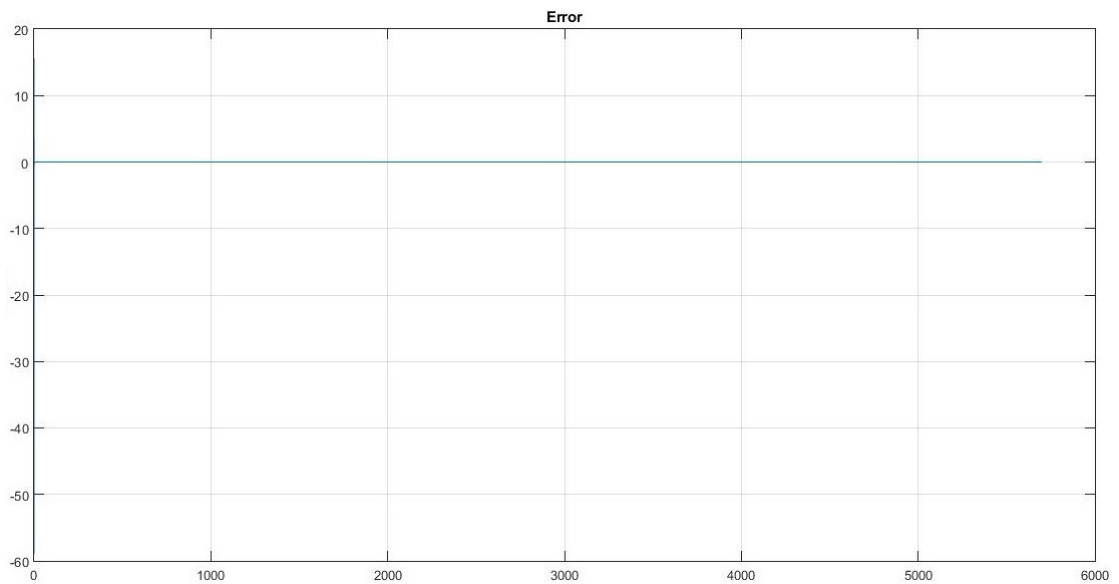
หลังจากที่ได้ทดลองว่าตัวควบคุมการเคลื่อนที่ (Trajectory controllers) ที่มีค่า Error น้อยที่สุด นั่นคือ Fitting Backpropagation Neuron Network จากนั้นจึงนำ Fitting Backpropagation Neuron Network มาใช้เป็น Block เพื่อนำมาต่อใช้งานกับแบบจำลองมอเตอร์และระบบควบคุม PID ที่ได้ออกแบบมาในหัวข้อก่อนหน้า ใน Simulink ดังรูปที่ 4.28



รูปที่ 4.28 การนำ Fitting BPNN ไปใช้กับมอเตอร์ของชุด Lower limb Exoskeleton 1 ข้อต่อ (มอเตอร์หัวเข่า)



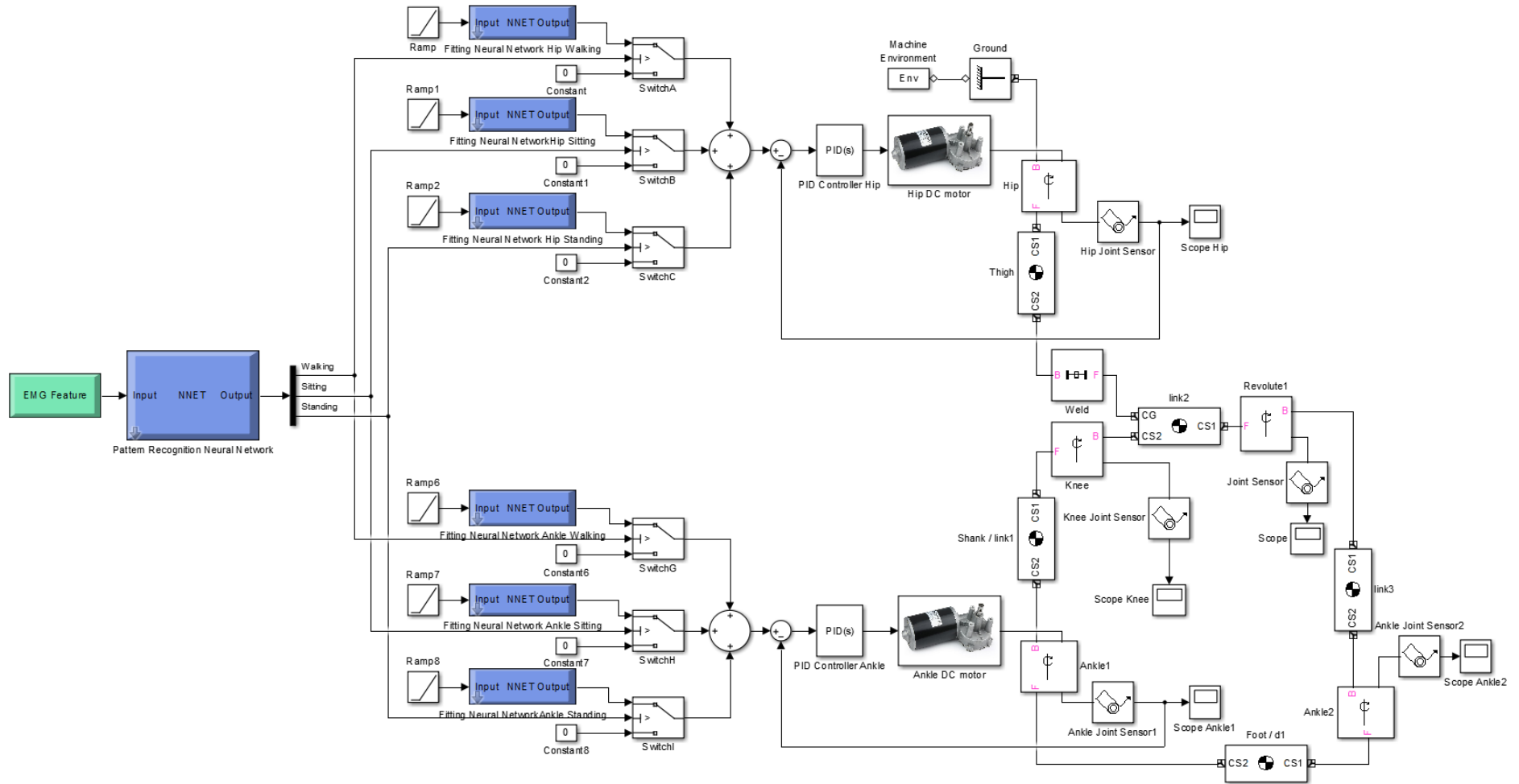
รูปที่ 4.29 ค่า Output ของมอเตอร์หัวเข่า



รูปที่ 4.30 ค่าผลต่างระหว่าง Output จาก Fitting backpropagation neuron network กับ Output ของมอเตอร์หัวเข้า

เมื่อได้รับอินพุตแล้ว ให้ Fitting Backpropagation Neuron Network ส่งค่ามุมมองศาที่โครงข่ายได้เรียนรู้ เพื่อนำไปใช้สั่งให้ชุด Exoskeleton ขยับตาม สำหรับการควบคุมมอเตอร์ที่จะนำไปใช้งานกับชุดจริงนั้น จะทำการส่งสัญญาณควบคุมมอเตอร์ทุกๆ 0.5 วินาที และสามารถใช่วิธีการ Autocorrelation ในการตรวจสอบคาบการเดิน

โดยการทำงานทั้งหมดของระบบชุด Lower limb exoskeleton จะถูกแสดง ดังรูปที่ 4.31



รูปที่ 4.31 ระบบทั้งหมดของชุด Lower Limb Exoskeleton

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

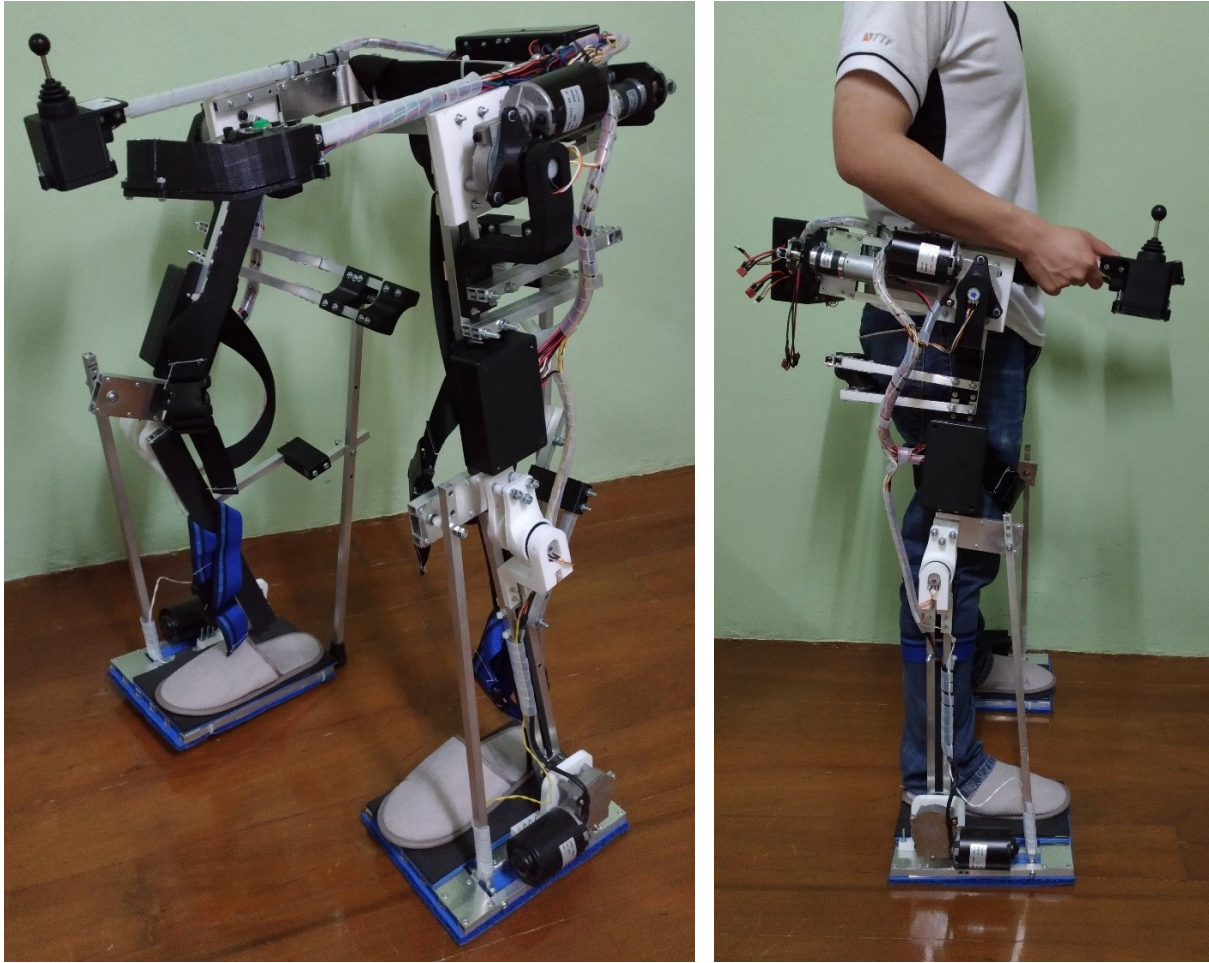
ในงานวิจัยนี้ ได้ออกแบบระบบที่ช่วยในการเดินสำหรับผู้ที่มีปัญหาด้านการเดิน โดยชุด Lower limb exoskeleton จะช่วยประคับประคองและผ่อนแรงในการเดิน

ชุด Lower limb exoskeleton นี้จะแบ่งเป็น 2 โหมดการทำงาน คือ โหมดใช้งาน กับโหมดฝึกสอน ทำทาง

ในส่วนของโหมดใช้งานนั้น จะเริ่มทำงาน เมื่อมีสัญญาณอินพุต คือ สัญญาณ sEMG เข้ามาผ่านกระบวนการ Feature Extraction (ซึ่งประกอบด้วย Time domain และ Frequency domain) จากนั้นก็นำค่าที่ได้ไป Classify ว่าเป็นท่าทางไหน โดยอินพุตของระบบซึ่งมี 3 ท่าทาง ประกอบด้วย ท่าเดิน, ท่านั่ง และ ท่ายืน ใช้ Backpropagation Neural Network (ที่ผ่านการ Train มาแล้ว) ในการจำแนกท่าทาง เมื่อ BPNN สามารถจำแนกท่าทางได้แล้ว ก็จะส่งสัญญาณไปเปิดการทำงานของตัวควบคุมการเคลื่อนที่ตามท่าทางจากสัญญาณที่ neural network จัดจำได้ แล้วตัวควบคุมการเคลื่อนที่ก็จะสั่งให้มอเตอร์หมุนมุมมองตาม และควบคุมการหมุนของมอเตอร์ให้ถูกต้องด้วยการใช้ PID Controller จากผลการทดลอง ได้ recognition rate ของการจำแนก sEMG signal สูงถึง 99.39% โดยใช้เฉพาะทุก feature ใน time domain

ในส่วนของโหมดฝึกสอนทำทางนั้น จะฝึกสอนท่าทางโดยใช้เซนเซอร์วัดมุมเข้าไปฝึกสอน เมื่อเราให้สัญญาณเวลาเข้าไป ตัว Fitting Neural Network (LM-BPNN) ก็จะทำให้เอาท่าพูดออกมาตามเวลา ตามที่ได้ Train ไว้ ทำให้ชุดสามารถเรียนรู้ท่าทางการเดินเฉพาะบุคคลจากมุมมองมาได้

เนื่องจากการคำนวณส่วนใหญ่ได้ทำบนเครื่องคอมพิวเตอร์ จึงใช้เวลาในการคำนวณค่อนข้างมาก สำหรับในการพัฒนาต่อไปในอนาคต อาจจะมีการนำ Single Board Computer มา Optimize การคำนวณต่างๆให้รวดเร็วขึ้น และจะนำวิธีการเหล่านี้ไปใช้งานบนชุดจริงที่ได้มีการออกแบบก่อนหน้าแล้วต่อไป ดังรูปที่ 5.1



รูปที่ 5.1 Lower Limb Exoskeleton ต้นแบบ: Electro Armor 6

เอกสารอ้างอิง

- [1] Zhiyong Yang, Lihua Gui, Xiuxia Yang, Wenjin Gu, " Simulation Research of Exoskeleton Suit Based on Neural Network Sensitivity Amplification Control ", *2008 Chinese Control and Decision Conference (CCDC 2008)*, China, pp.3340-3344, 2008.
- [2] JUSTIN GHAN, RYAN STEGER, H. KAZEROONI, "Control and system identification for the Berkeley lower extremity exoskeleton (BLEEX)", *Advanced Robotics*, Vol. 20, No. 9, pp. 989–1014 (2006).
- [3] Andrew Chu, H. Kazerooni, and Adam Zoss, "On the Biomimetic Design of the Berkeley Lower Extremity Exoskeleton (BLEEX)", *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, pp. 4345–4352, April 2005.
- [4] W. Banchadit, A. Temram, T. Sukwan, P. Owatchaiyapong and J. Suthakorn, "Design and Implementation of a New Motorized-Mechanical Exoskeleton Based on CGA Patternized Control", *International Conference on Robotics and Biomimetics*, Guangzhou, China, pp.1668-1673, December 11-14, 2012.
- [5] Yoshiyuki Sankai, "HAL: Hybrid Assistive Limb based on Cybernetics", Springer-Verlag Berlin Heidelberg 2010, *Robotics Research, STAR 66*, pp. 25–34, 2010.
- [6] โอบนิธิ วรรณรัตน์, ยุทธนา คัดใจเดียว, "การออกแบบสร้างชุด Lower Limb Exoskeleton เพื่อช่วยเดินโดยใช้การควบคุมแบบ PID และกลไกแบบ Four Bar Linkages", การประชุมวิชาการ งานวิจัยและพัฒนาเชิงประยุกต์ ครั้งที่ 10 (Conference proceeding 10th ECTI-CARD-2018), หน้า. 69 - 72, มิถุนายน 2018.
- [7] Obnithi Karantararat, Yuttana Kitjaidure, "The Walking Assistance System using The Lower Limb Exoskeleton Suit Commanded by Backpropagation Neural Network", 2018 11th Biomedical Engineering International Conference (BMEiCON), Chiang Mai, Thailand, Date Added to IEEE Xplore: 14 January 2019.
- [8] Gait. (2018). Retrieved from <https://www.physio-pedia.com/Gait/>.
- [9] Normal gait. (2018). Retrieved from <https://musculoskeletalkey.com/normal-gait/>.
- [10] Didier Gamet, Odette Fokapu, "Electromyography", *Encyclopedia of Biomaterials and Biomedical Engineering*, Informa Healthcare USA, Inc., 2008.
- [11] ดร.ภก.กัญเกียรติ หุดปอ, "กายวิภาคศาสตร์และสรีรวิทยาพื้นฐานสำหรับงานสาธารณสุข" (2019) Retrieved from [http://kpi.msu.ac.th/upload/ag_tor_ref_byval/ag__in_1.2.7_481\(\).pdf](http://kpi.msu.ac.th/upload/ag_tor_ref_byval/ag__in_1.2.7_481().pdf).
- [12] What is Feature Extraction (2018). Retrieved from www.igi-global.com/dictionary/feature-extraction/10960.

เอกสารอ้างอิง(ต่อ)

- [13] feed forward neural network (2019) Retrieved from https://www.researchgate.net/figure/Schematic-diagram-of-a-general-back-propagation-neural-network_fig10_39730973.
- [14] Hao Yu and B. M. Wilamowski, “Levenberg–Marquardt Training” Industrial Electronics Handbook, vol. 5 – Intelligent Systems, 2nd Edition, chapter 12, pp. 12-1 to 12-15, CRC Press 2011.
- [15] anthropometry. (2018). Retrieved from www.slideshare.net/vimal333/anthropometry-15401067.
- [16] Zhiyong Yang, Lihua Gui, Xiuxia Yang, Wenjin Gu, Yuanshan Zhang, "Simulation Research of Exoskeleton Suit Based on Sensitivity Amplification Control", Proceedings of the IEEE International Conference on Automatic and Logistics, Jinan, China, pp.1353-1357, Aug. 18-21, 2007.
- [17] DC Motor Control Design. (2018). Retrieved from [https://www.maplesoft.com/support/help/maple/view.aspx?path=applications%2FDC Motor](https://www.maplesoft.com/support/help/maple/view.aspx?path=applications%2FDC%20Motor).
- [18] M. Dooner, J. Wang and A. Mouzakitidis, “Dynamic modelling and experimental validation of an automotive windshield wiper system for hardware in the loop simulation”, Systems Science & Control Engineering: An Open Access Journal, Vol. 3, 230–239, 2015.
- [19] EMG dataset in Lower Limb Data Set (2018). Retrieved from <http://archive.ics.uci.edu/ml/datasets/emg+dataset+in+lower+limb>.

ภาคผนวก ก.
คำสั่งในการเขียนโปรแกรม MATLAB

โค้ด MATLAB สำหรับการสอนทำทางเดิน โดยใช้วิธีการ Fitnet

```

WalkFX_1 = xlsread('ไฟล์ข้อมูล EMG.xlsx');
inputs = 1:5700; % ในที่นี่มีจำนวนข้อมูล 5700 ค่า
targets = WalkFX_1;
P = inputs; T = targets;

%Step 1: % Create a Fitting Network
hiddenLayerSize = 20;
net = fitnet(hiddenLayerSize);

% Set up Division of Data for Training, Validation, Testing
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
net.trainParam.lr = 0.01; % Learning rate used in some gradient schemes
net.trainParam.epochs = 3000; % Max number of iterations
net.trainParam.goal = 1e-6; % Error tolerance; stopping criterion

% Train the Network
[net,tr] = trainlm(net,inputs,targets);

% Test the Network
outputs = net(inputs);
errors = gsubtract(outputs,targets);
performance = perform(net,targets,outputs)

% View the Network
view(net)

% Plot result and compare
figure;
grid;

```

ภาคผนวก ข.
ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

- [1] โอบนิธิ วรรณรัตน์, ยุทธนา คัดใจเดียว, “การออกแบบสร้างชุด Lower Limb Exoskeleton เพื่อช่วยเดินโดยใช้การควบคุมแบบ PID และกลไกแบบ Four Bar Linkages”, การประชุมวิชาการงานวิจัยและพัฒนาเชิงประยุกต์ ครั้งที่ 10 (Conference proceeding 10th ECTI-CARD-2018), หน้า. 69 -72, มิถุนายน 2018.
- [2] Obnithi Karantararat, Yuttana Kitjaidure, “The Walking Assistance System using The Lower Limb Exoskeleton Suit Commanded by Backpropagation Neural Network”, 2018 11th Biomedical Engineering International Conference (BMEiCON), Chiang Mai, Thailand, Date Added to IEEE *Xplore*: 14 January 2019.

รายงานสืบเนื่องจากการประชุมวิชาการ
Conference Proceedings

ECTI-CARD 2018

การประชุมวิชาการ งานวิจัย และพัฒนาเชิงประยุกต์ ครั้งที่ 10

ณ ศูนย์วัฒนธรรมภาคเหนือตอนล่าง วังจันทน์ ริเวอร์วิว จ.พิษณุโลก

26 - 29 มิถุนายน 2561



ECTI
Association



บทความวิจัย

การประชุมวิชาการ งานวิจัยและพัฒนาเชิงประยุกต์ ครั้งที่ 10

10th ECTI-CARD 2018, Phitsanulok Thailand

การออกแบบสร้างชุด Lower Limb Exoskeleton เพื่อช่วยเดินโดยใช้การควบคุมแบบ PID และกลไกแบบ Four Bar

Linkages

The Design of a Lower Limb Exoskeleton Suit for Walking Assistance Using PID Controller and Mechanism of Four Bar Linkages

โอบนิธิ ภิรมย์รัตน์, ยุทธนา ลิดใจเดียว

ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขที่ 1 ซอยฉลองกรุง 1 เขตลาดกระบัง กรุงเทพมหานคร 10520 โทรศัพท์: 02-329-8344 Email: o.karantarat@gmail.com

บทคัดย่อ

ในงานวิจัยนี้ได้ออกแบบชุด Lower Limb Exoskeleton สำหรับช่วยออกแรงทดแทนส่วนที่ร่างกายไม่สามารถเคลื่อนไหวได้ เช่น ผู้ป่วยที่มีปัญหาด้านการเดิน ในการออกแบบชุด Exoskeleton จะใช้ PID controller มาควบคุมมอเตอร์กระแสตรง และกลไกแบบ Four Bar Linkages เพื่อเพิ่มเสถียรภาพในช่วง Stance phase และนอกจากนี้กลไกที่ออกแบบยังลดจำนวนมอเตอร์ที่ต้องใช้ลง ทำให้สามารถลดต้นทุนในการสร้างชุด Exoskeleton ได้อีกด้วย หลังจากที่ได้ออกแบบและจำลองผลแล้ว จึงนำแบบจำลองไปประยุกต์สร้างเป็นชุด Exoskeleton ของจริง ซึ่งสามารถควบคุมการเดินได้ตามที่ออกแบบไว้

คำสำคัญ ชุดช่วยเดิน, กลไกชิ้นต่อโยงสี่ชิ้น, การควบคุมแบบพีไอดี

Abstract

In this study, the Lower Limb Exoskeleton was designed to support non-moving body parts such as patients with walking problems. The Exoskeleton design uses PID controller to control a DC motor and mechanism of Four Bar Linkages to increase stability during the stance phase. In addition, the designed mechanism reduces the number of motors required. The cost of creating the Exoskeleton can also be reduced. This model is applied to create a real Exoskeleton, which can control gait as designed.

Keyword: Lower Limb Exoskeleton, Four bar linkages, PID control

1. บทนำ

ชุด Exoskeleton เป็นชุดที่ช่วยในการเคลื่อนไหวและช่วยในการออกแรงส่วนต่างๆของร่างกาย สำหรับชุด Exoskeleton ในงานวิจัยนี้จะเน้นไปที่ทางการแพทย์ ซึ่งใช้กับผู้ป่วยที่มีปัญหาด้านการเดินจากโรคต่างๆทางระบบประสาทและกล้ามเนื้อ เช่น โรคกล้ามเนื้ออ่อนแรง, โรคโปลิโอ และโรคอัมพาตครึ่งซีก เป็นต้น จำเป็นต้องใช้อุปกรณ์เพื่อช่วยในการฟื้นฟู บำบัด หรือช่วยออกแรงทดแทนส่วนที่ร่างกายไม่สามารถเคลื่อนไหวได้ ซึ่งชุด Exoskeleton เป็นอีกวิธีการหนึ่งที่จะนำมาใช้เพื่อช่วยเหลือผู้ป่วยเหล่านี้

งานวิจัยนี้จึงได้พัฒนาชุด Exoskeleton ในส่วนล่าง หรือ Lower Limb Exoskeleton ที่มีชื่อว่า Electro Armor 6 ใช้ในการผ่อนแรงในการเดิน แบบ ต้นทุนต่ำ แต่สามารถใช้งานได้ โดยประยุกต์ใช้หลักการของ Four Bar linkages เพื่อลดจำนวนของมอเตอร์ลง โดยสามารถลดจำนวนมอเตอร์ลงได้ข้างละ 1 ตัวในส่วนของเขา ซึ่งกลไกที่ออกแบบนี้ ยังเพิ่มเสถียรภาพในช่วง Stance phase อีกด้วย และใช้การควบคุมแบบ PID เพื่อให้ชุดเคลื่อนไหวได้อย่างนุ่มนวล ไม่กระตุก



รูปที่ 1 ชุด Exoskeleton ต้นแบบ: Electro Armor 6

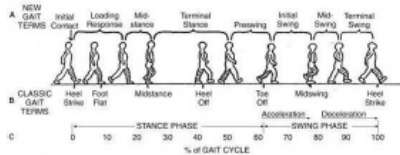
บทความวิจัย

การประชุมวิชาการ งานวิจัยและพัฒนาเชิงประยุกต์ ครั้งที่ 10
10th ECTI-CARD 2018, Phitsanulok Thailand

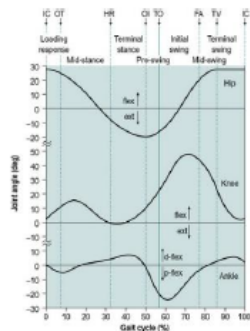
2. ทฤษฎี และการออกแบบ

2.1 การเดินของมนุษย์

การเดินของมนุษย์นั้น สามารถแบ่งได้เป็น 2 ช่วง นั่นคือ ช่วงที่เท้าสัมผัสพื้น (Stance phase) คิดเป็นร้อยละ 60 ของ 1 รอบการเดิน และช่วงที่แกว่งเท้า (Swing phase) คิดเป็นร้อยละ 40 ของ 1 รอบการเดิน ดังรูปที่ 2 [1] ส่วนมุมในการเดินจะอ้างอิงจาก [2] ในรูปที่ 3



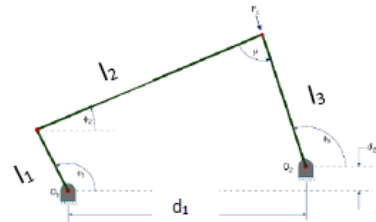
รูปที่ 2 Gait Analysis [1]



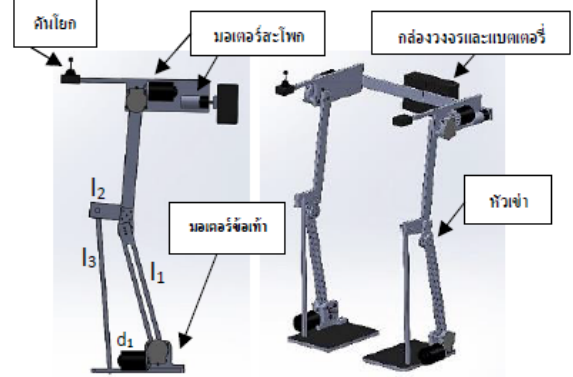
รูปที่ 3 องศาการเคลื่อนที่ของสะโพก, เข่า และข้อเท้า [2] มุมอิงจากขาที่แรงเงาในรูปที่ 2

2.2 Four Bar Linkages

กลไกแบบ Four Bar Linkages ใช้การส่งกำลังของคาน โดยจะมีคานส่งกำลัง l_1 , d_1 เป็น Frame, l_1 คือ ตัวขับ และ l_3 คือ ตัวถูกขับ ดังรูปที่ 4 เมื่อตัวขับเชิงมุม θ_1 หมุน จะทำให้คาน l_2 ขยับด้วย เราจะสามารถนำกลไกแบบ Four Bar linkages ไปออกแบบให้สามารถลดจำนวนมอเตอร์ที่ใช้ในการเดินได้ โดยนำไปใช้บริเวณข้อเท้ากับหัวเข่า เพราะในเวลาเดิน มุมองศาการเคลื่อนที่ของข้อเท้าและหัวเข่าจะมีความสัมพันธ์กัน โดยให้ l_1 เป็นส่วนน่อง, d_1 เป็นส่วนเท้า, O_1 เป็นส่วนข้อเท้า และสร้างส่วนที่ติดงากับ l_1 เป็นส่วนต้นขา ดังรูปที่ 5 ในงานวิจัยนี้จะใช้มุมมองในการเดินจากเอกสารอ้างอิง [3] และออกแบบให้มอเตอร์ขับเคลื่อนจะอยู่ตรงข้อเท้าเพื่อให้น้ำหนักถูกกระจายลงบนฝ่าเท้า การออกแบบลักษณะนี้จะทำให้ CG (Center of Gravity) ของชุดที่วางมอเตอร์บริเวณข้อเท้าจะอยู่ต่ำลงมากกว่า CG ของชุดที่วางมอเตอร์ไว้ที่บริเวณหัวเข่า ทำให้สามารถเพิ่มเสถียรภาพในช่วง Stance Phase ได้

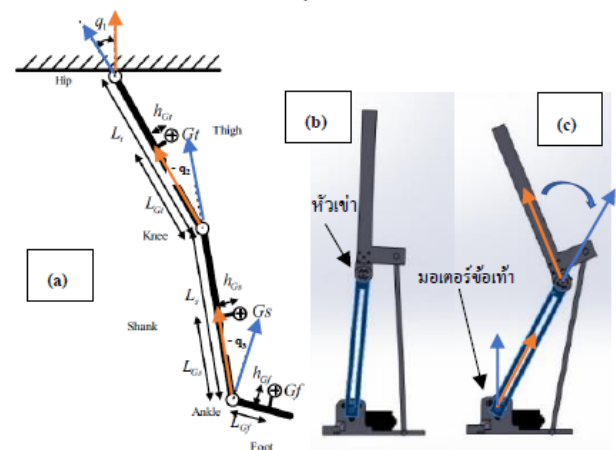


รูปที่ 4 กลไกแบบ Four-bar linkages [4]



รูปที่ 5 การออกแบบส่วนกลไก Four Bar linkages ของชุด Exoskeleton

ในการออกแบบจำเป็นต้องคำนึงถึงสรีระร่างกายของมนุษย์ด้วย จึงออกแบบให้มุมมองขาที่จะกวาดได้มีองศาจำกัด ดังตารางที่ 1 เพื่อไม่ให้มีการบิดเจ็บจากการที่ชุด Exoskeleton ขยับเกินกว่าองศาที่ร่างกายมนุษย์ขยับได้ แต่ยังทำให้ชุด Exoskeleton สามารถเคลื่อนไหวองศาได้ครอบคลุมช่วงการเดิน มุมอ้างอิงจะเป็นช่วงที่ยืนตรง เท้าจะขนานกับพื้น ส่วนเข่าและต้นขาจะตั้งฉากกับพื้น ความยาวของชิ้นส่วนของต้นขา และความยาวของเข่าคำนวณจากการวัดสัดส่วนร่างกายมนุษย์ (anthropometry) จาก [4]



รูปที่ 6 (a) มุมอ้างอิงของขามนุษย์ [5] ให้การกวาดชิ้นส่วน ในทิศทางเข้มนาทิศาเป็นบวก จะได้มุมสะโพก(q_1), เข่า($-q_2$), ข้อเท้า($-q_3$), L_i คือความยาว

บทความวิจัย

การประชุมวิชาการ งานวิจัยและพัฒนาเชิงประยุกต์ ครั้งที่ 10

10th ECTI-CARD 2018, Phitsanulok Thailand

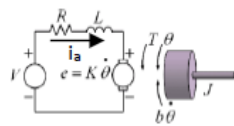
ของต้นขา (Thigh), L_2 คือความยาวของแข้ง (Shank), แกนอ้างอิงคือ ลูกศรสีส้ม และแกนการแกว่งคือ ลูกศรสีน้ำเงิน และ (b) $-q_2 = -3$ องศา, $-q_3 = 5$ องศา และ (c) $-q_2 = -50$ องศา, $-q_3 = 30$ องศา

ตารางที่ 1 เปรียบเทียบช่วงมุมองศา (degree) สูงสุดและต่ำสุดใน 1 รอบการเดินของข้อต่อส่วนต่างๆ ระหว่างมนุษย์กับชุด Exoskeleton ที่ออกแบบ

	Minimum angle		Maximum angle	
	มนุษย์	Exoskeleton	มนุษย์	Exoskeleton
สะโพก (q_1)	-20	-15	30	30
เข่า (q_2)	-2	3	50	50
ข้อเท้า (q_3)	-25	-10	10	10

2.3 แบบจำลองทางคณิตศาสตร์ของมอเตอร์กระแสตรง

วงจรของมอเตอร์กระแสตรงและโหลดทางกลจะเป็นดังรูปที่ 7 ซึ่งสมการอนุพันธ์ที่บรรยายพลวัตของส่วนทางกล ให้การเคลื่อนที่ของกลไกเป็นอุดมคติ อธิบายได้โดยสมการที่ (1) และ สมการความสัมพันธ์ของแรงดันไฟฟ้า ดังสมการที่ (2)



รูปที่ 7 มอเตอร์กระแสตรงที่ต่อกับโหลดทางกล [6]

$$J\ddot{\theta} + b\dot{\theta} = \tau = k_t i_a \tag{1}$$

$$L\dot{i}_a + Ri_a = V - k_e \dot{\theta} \quad ; k_e = k_t = k \tag{2}$$

ทำการแปลงลาปลาซ แล้วจัดรูปสมการ จะได้

$$\frac{\theta(s)}{V(s)} = \frac{k}{s((Js + b)(Ls + R) + k^2)} \tag{3}$$

แบบจำลองของมอเตอร์ไฟฟ้ากระแสตรงจะเป็นดังสมการที่ (3) เลือกใช้ DC Wiper Motor 12V เป็นตัวขับเคลื่อนค่าพารามิเตอร์ที่เป็นคุณสมบัติของ Wiper Motor โดยอ้างอิงจาก [7] แทนค่า $L = 9.6 \times 10^{-4} \text{H}$, $R = 0.6 \Omega$, $J = 0.004 \text{kgm}^2$, $b = 0.0185 \text{Nm/(rad/s)}$ และ $k = 0.065 \text{V/rpm}$ ลงใน (3) จะได้ฟังก์ชันถ่ายโอนของมอเตอร์กระแสตรง คือ

$$\frac{\theta(s)}{V(s)} = \frac{1}{(5.9077 \times 10^{-11})s^3 + (0.0369)s^2 + 0.235s} \tag{4}$$

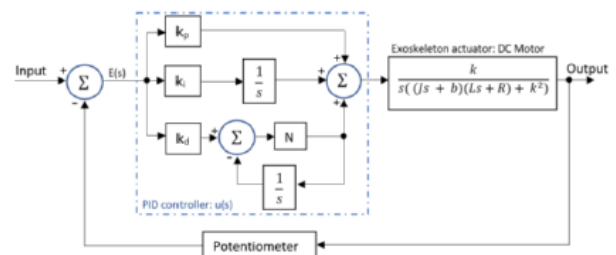
2.4 PID Controller

การควบคุมแบบ PID เป็นระบบควบคุมแบบป้อนกลับที่จะนำค่าผิดพลาด $E(s)$ จากผลต่างของมุมที่ต้องการกับมุมที่ป้อนกลับจากระบบมาที่อินพุต มาผ่านฟังก์ชัน P, ฟังก์ชันปริพันธ์ I และฟังก์ชันอนุพันธ์ D แล้วนำสัญญาณทั้งหมดมารวมกันเพื่อส่งออกไปควบคุมอุปกรณ์ แต่เนื่องจากฟังก์ชันอนุพันธ์มีความไวต่อสัญญาณรบกวนมาก เราจึงต้องใช้ Filter ไปเพื่อขจัดสัญญาณรบกวน ซึ่งเราสามารถกำหนดอัตราการจัดสัญญาณรบกวนโดยปรับค่า Filter coefficients (N) และสามารถปรับความเร็วในการเข้าสู่เป้าหมายได้โดยการปรับค่า K_p, K_i, K_d ในการออกแบบการควบคุม PID จะใช้สมการที่ (5) และ (6) ดังนี้

$$E(s) = \text{setpoint} - \text{feedback} \tag{5}$$

$$u(s) = E(s)(K_p + K_i \frac{1}{s} + K_d \frac{N}{1+N\frac{1}{s}}) \tag{6}$$

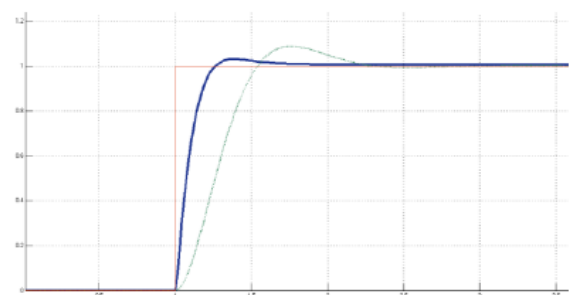
ในการสร้างชุด Exoskeleton จะนำแบบจำลองของมอเตอร์ไฟฟ้ากระแสตรงมาใช้ ดังรูปที่ 8



รูปที่ 8 การควบคุมแบบ PID ของชุด Exoskeleton

3. ผลการทดลอง

นำสมการที่ (4) มาทดลองการควบคุม PID ในรูปที่ 8 โดยใช้โปรแกรม Simulink ใน MATLAB ได้ผลดังรูปที่ 9



รูปที่ 9 กราฟเปรียบเทียบระหว่างผลการตอบสนองของแบบจำลองมอเตอร์กระแสตรงที่ไม่ใส่ส่วนควบคุม PID (เส้นประสีเขียว) และกราฟผลการ

บทความวิจัย

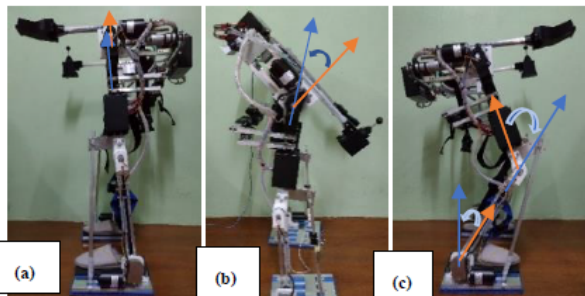
การประชุมวิชาการ งานวิจัยและพัฒนาเชิงประยุกต์ ครั้งที่ 10

10th ECTI-CARD 2018, Phitsanulok Thailand

ตอบสนองของแบบจำลองมอเตอร์กระแสตรงที่ใช้การควบคุม PID (เส้นสีน้ำเงิน) โดยกราฟเส้นสีแดง คือ อินพุต

ซึ่งจะเห็นได้ว่า การควบคุมแบบ PID สามารถตอบสนองได้เร็วกว่า และเกิด Overshoot น้อยกว่า การลดการเกิดโอเวอร์ชูต ก็เพื่อทำให้เมื่อนำไปใช้ในชุด Exoskeleton ไม่เกิดการกระตุกในการทำงานช่วง Swing phase จากการทดลองปรับจูนค่าพารามิเตอร์ในโปรแกรม MATLAB สรุปได้ว่าใช้ค่า $K_p = 2.88$, $K_i = 0.255$ และ $K_d = 0.32$ และค่า Filter coefficients (N) = 106.84

จากนั้นจึงนำหลักการทั้งหมดที่กล่าวมาก่อนหน้า มาประยุกต์สร้างชุด Exoskeleton โดยมีส่วนประกอบต่างๆ คือ 1. Potentiometer ใช้ในการวัดมุมองศาการหมุนของมอเตอร์แต่ละตัว 2. Gyroscope MPU6050 จะทำหน้าที่วัด Zero Moment Point ของชุด Exoskeleton ตรวจสอบว่าชุดมีการเอียงจนเสียสมดุลหรือไม่ ในช่วง Terminal Stance phase ถ้าชุดเริ่มมีการโน้มไปข้างหน้ามากเกินไปจนองศาที่กำหนดไว้ก็จะให้มอเตอร์ในส่วนข้อเท้าและส่วนสะโพกหมุนกลับไปในทางที่กลับมามสมดุลอีกครั้ง 3. ไมโครคอนโทรลเลอร์ Arduino Mega 2560 ใช้ในการประมวลผล และ 4. DC Wiper Motor ขนาด 12V 55rpm 4 ตัว และ DC gear motor จำนวน 2 ตัวในส่วนสะโพก และ 5. ใช้บอร์ดขับมอเตอร์กระแสตรง เบอร์ MD13S



รูปที่ 10 ชุด Exoskeleton ขยับสะโพก (q_1), เข่า (q_2) และข้อเท้า (q_3) ตามที่ได้กำหนดค่าไว้ (a) $q_1 = 0$ องศา, $q_2 = 3$ องศา และ $q_3 = -5$ องศา, (b) $q_1 = -30$ องศา, (c) $q_2 = 50$ องศา และ $q_3 = -30$ องศา

ในการควบคุมการทำงานของชุด Exoskeleton จะมีอยู่ 2 วิธี นั่นคือ 1. ควบคุมด้วยสวิตช์ (ในสภาวะที่ผู้ป่วยเดินไม่ได้ แต่ผู้ป่วยสามารถชี้แขนได้) เมื่อทำการกดสวิตช์ก็จะทำให้ชุด Exoskeleton เริ่มเดินตามท่าทางที่ได้โปรแกรมเอาไว้ 2. ควบคุมโดยการใช้ Force sensor ที่บริเวณฝ่าเท้า (ในกรณีที่ผู้ใช้งานเป็นผู้ที่ไม่สามารถชี้กล้ามเนื้อได้เต็มที่ หรือ ในกรณีของการ

กายภาพบำบัด) ในช่วงที่เริ่มจะก้าวขา เช่น ขาขวา น้ำหนักตัวจะถูกทิ้งไปที่เท้าข้างซ้าย เมื่อตรวจจับแรงกดที่บริเวณ Force Sensor ได้ ไมโครคอนโทรลเลอร์ก็จะสั่งการให้ก้าวขาขวา และเป็นอย่างนี้เช่นเดียวกับการทิ้งน้ำหนักที่เท้าข้างขวา แล้วก้าวขาข้างซ้าย

4. สรุป

ชุด Electro Armor 6 ที่นำเสนอในบทความนี้เป็นชุด Exoskeleton ที่นำกลไกแบบ Four-bar linkages เข้ามาใช้เพื่อลดจำนวนมอเตอร์ที่ใช้ในการเดินและเพิ่มเสถียรภาพในช่วง Stance phase และใช้ PID controller เพื่อมาควบคุมมุมมององศาของมอเตอร์แต่ละตัว จากผลการทดลอง Simulate ได้ค่า $K_p = 2.88$, $K_i = 0.255$ และ $K_d = 0.32$ และค่า Filter coefficients (N) = 106.84 เพื่อลดการเกิดโอเวอร์ชูต ทำให้เมื่อนำไปใช้ในชุด Exoskeleton จะไม่เกิดการกระตุกในการทำงานช่วง Swing phase น้ำหนักโดยรวมของชุดเท่ากับ 17 กิโลกรัม ชุดนี้เหมาะสำหรับคนที่สูงในช่วง 165-170 cm สำหรับในการพัฒนาต่อไปในอนาคต จะมีการปรับปรุงวิธีการควบคุม เพื่อให้ผู้ป่วยสามารถนำไปใช้งานได้ง่ายขึ้น

เอกสารอ้างอิง

- [1] Gait. (2018). Retrieved from <https://www.physio-pedia.com/Gait/>
- [2] Normal gait. (2018). Retrieved from <https://musculoskeletalkey.com/normal-gait/>
- [3] Four bar linkages. (2018). Retrieved from <http://www.writeopinions.com/four-bar-linkage>
- [4] anthropometry. (2018). Retrieved from <https://www.slideshare.net/vimal333/anthropometry-15401067>
- [5] Zhiyong Yang, Lihua Gui, Xiuxia Yang, Wenjin Gu, Yuanshan Zhang, "Simulation Research of Exoskeleton Suit Based on Sensitivity Amplification Control", Proceedings of the IEEE International Conference on Automatic and Logistics, Jinan, China, pp.1353-1357, Aug. 18-21, 2007
- [6] DC Motor Control Design. (2018). Retrieved from <https://www.maplesoft.com/support/help/maple/view.aspx?path=applications%2FDCMotor>
- [7] M. Dooner, J. Wang and A. Mouzakitis, "Dynamic modelling and experimental validation of an automotive windshield wiper system for hardware in the loop simulation", Systems Science & Control Engineering: An Open Access Journal, Vol. 3, 230-239, 2015

The 11th Biomedical Engineering International Conference 2018

BMEiCON 2018: Smart System for Smarter Healthcare | Nov 21 – 24, Chiang Mai, THAILAND



The BMEiCON 2018 will be held in a beautiful cultural city, Chiang Mai, Thailand, during November 21-24, 2018. In conjunction with [The 22nd International Computer Science and Engineering Conference \(ICSEC2018\)](#). The organizing committee is pleased to invite all engineers, physicians, scientists, technicians, and technologists to attend and help invent a smarter healthcare system. The topics for regular sessions include, but are not limited to, the followings:

- Biomedical signal processing
- Biomedical imaging and image processing
- Bioinstrumentation
- Bio-robotics and biomechanics
- Biosensors and Biomaterials
- Cardiovascular and respiratory systems engineering
- Cellular and Tissue Engineering
- Healthcare information systems
- Human machine/computer interface
- Medical device design
- Neural and rehabilitation engineering
- Technology commercialization, industry, education, and society
- Telemedicine
- Therapeutic and diagnostics systems
- Recent advancements in biomedical engineering

Submission of Papers

Prospective authors are invited to submit full-length papers, including figures, tables, and references, via our website at <http://www.bmeicon.org/bmeicon2018>

All papers will be peer reviewed and handled electronically. All papers submitted must be previously unpublished and may not be considered for publication elsewhere at any time during the review period.

Any accepted paper included in the final program is expected to have at least one author or qualified proxy attend and present the paper at the conference.

International Steering Committee

Chuchart Pintavirooj, Thailand
Somkiat Wattanasirichaigoon, Thailand
Wibool Piyawattanametha, Thailand
Kosin Chanongthai, Thailand
Kazuhiko Hamamoto, Japan
Shin'ichiro Kanoh, Thailand
Kohji Masuda, Japan
Manfred Glesner, Germany
Keiji Iramina, Japan
Michael Khoo, USA
Sunu Wibirama, Indonesia
Viet Dung Nguyen, Vietnam

Organizing Committee

Honorary Chair
Somkiat Wattanasirichaigoon, SWU
General Chair
Ekkarat Boonchieng, CMU
General Co-Chair
Chuchart Pintavirooj, KMITL
Technical Program Chair Chuchart Pintavirooj, KMITL
Publications Chair Adisorn Leelasanthithum, MU, and
Publicity Chair Sumet Um-Chit, KMUTNB
Financial Chair Sarinporn Visitsattapongse, KMITL
Local Arrangement Chair Varin Chaovatut, CMU
General Secretary Khanita Duangchaemkarn, Univ. of Phayao

Important Dates

Regular paper submission deadline:
August 31, 2018

Notification of acceptance:
September 15, 2018

Camera-ready submission deadline:
October 31, 2018

Early Bird registration deadline:
October 31, 2018

Conference date:
November 21-24, 2018

Contact

Ekkarat Boonchieng, CMU
ekkarat@ieee.org

Chuchart Pintavirooj, KMITL
chuchartpintavirooj@gmail.com

For additional information and submission guidelines, please visit our website at <http://www.bmeicon.org/bmeicon2018/>



The Walking Assistance System using the Lower Limb Exoskeleton Suit Commanded by Backpropagation Neural Network

Obnithi Karantarat, Yuttana Kitjaidure
 Faculty of Engineering
 King Mongkut's Institute of Technology Ladkrabang
 Bangkok, 10520, Thailand
 o.karantarat@gmail.com

Abstract— Currently there are many elderly people who have walking problems. This paper aims to develop and solve these problems by introducing walking assistance system which can recognize 3 types of gestures, include walking, sitting and standing. Our system is divided into 3 main parts including Feature extraction which consists of Time domain and Frequency domain, Classification and Exoskeleton suit system. Conjugate Gradient Backpropagation Neural Network is used to classify sEMG signal of lower limb posture after extracted the features. Then the output of classification is used to command the Exoskeleton suit to perform the gesture according to the results of the recognition. In addition, our paper uses PID controller to control DC motor of Four Bar Linkages Mechanisms of Lower Limb Exoskeleton suit in order to reduce the number of motors and increase stability during the Stance Phase. The results from the experiment have concluded that all feature in time domain has the most recognition rate which up to 99.39%.

Keywords—Lower Limb Exoskeleton, Feature Extraction, Conjugate Gradient Backpropagation Neural Network, PID control, Four bar linkages

I. INTRODUCTION

Many countries around the world are entering into an aging society. Currently there are many elderly people who have walking problems. One way to solve that problem is lower limb Exoskeleton. The Lower Limb Exoskeleton suit is used for walking assist. In this paper we design walking Assistance System for walking aid of aged people.

This paper is developed from [1]. We use the dataset of sEMG for training scaled conjugate gradient backpropagation neural network. The concept is that when the suit recognizes the signal from the lower limb, the motor of the suit will be moved by the motion program. The advantages of design four bar linkages mechanisms are reducing the number of motors in the exoskeleton suit.

II. METHODOLOGY

A. Gait analysis

The human walking in one gait cycle can be divided into 2 phases including stance phase (60% of 1 gait cycle) and swing phase (40% of 1 gait cycle). Stance phase means the phase when the foot contacts with the ground and Swing phase means the phase when the foot does not contact with the ground as shown in fig. 1. Fig. 2 shows the joint angles of the lower limb as shading in fig. 1. from one gait cycle including hip, knee and ankle.

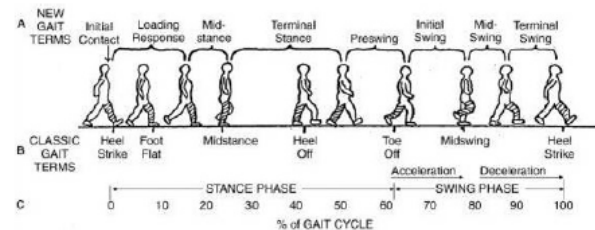


Fig. 1. Gait Analysis [2]

B. Lower Limb Exoskeleton

Four bar linkages mechanism is the simplest movable closed chain linkage, which consists of l_1 , l_2 , l_3 and d_1 . The given d_1 is a frame, l_1 is an actuator, l_2 is a transmission link and l_3 is load as shown in fig. 3. When angle actuator l_1 is working, it will move l_3 because l_1 send the power via l_3 through l_2 . We design Four Bar linkages Mechanisms to reduce the number of motors in the exoskeleton suit. We use the suit with a knee and ankle because when ones start to walk, the movement angle of the ankle and knee will relate to each other. l_1 is the shank, d_1 is the foot, l_2 is the thigh and O_1 is the ankle angle. In this paper, the angle of gait from fig. 2 is used to design the motor at the ankle. This design will make Center of Gravity (CG) lower than the conventional suit. Therefore, it can increase stability during the Stance Phase.

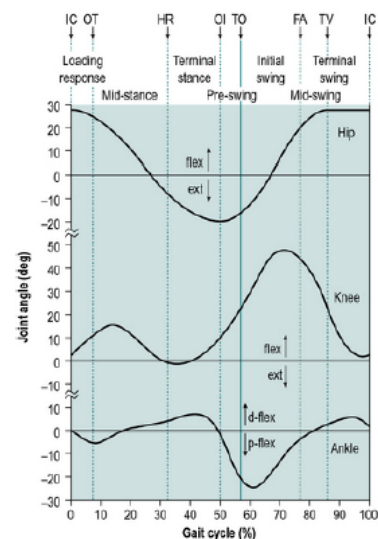


Fig. 2. Joint angle of hip, knee and ankle [3]

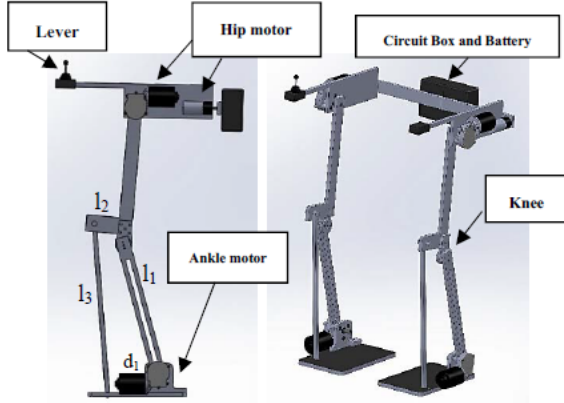


Fig. 3. Design Four Bar Linkages Mechanisms of Lower Limb Exoskeleton suit

TABLE I THE DEGREE RANGES OF JOINT BETWEEN HUMAN AND EXOSKELETON

Joint	Minimum angle		Maximum angle	
	Human	Exoskeleton	Human	Exoskeleton
Hip (q_1)	-20	-15	30	30
Knee (q_2)	-2	3	50	50
Ankle (q_3)	-25	-10	10	10

This suit is designed for the human body, which all the angle ranges of the suit are shown in table I and fig.4. We use Stand phase to be the reference angle, where the foot is parallel to the ground while thigh and shank is perpendicular to the ground. The lengths of the thigh and shank are calculated by anthropometric data from [4]. The angle ranges of the suit can avoid injury from moving angles larger than the human ability. Furthermore, the Exoskeleton suit can cover all the gestures.

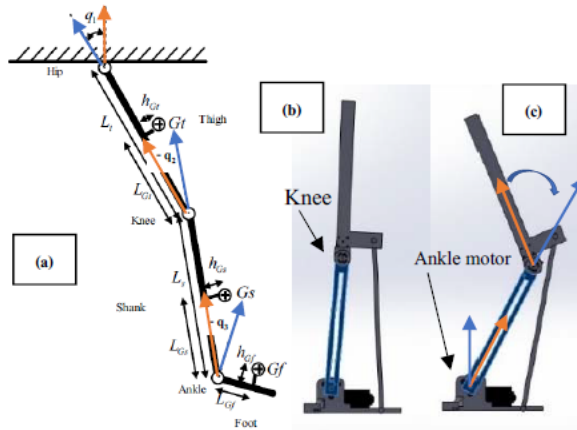


Fig. 4. (a) Reference Angle of Human leg [5]. Define Counterclockwise to positive, Hip angle(q_1), Knee angle($-q_2$), ankle angle($-q_3$). L_t is the length of the thigh, L_s is the length of the shank, reference axis is orange arrow and moving axis is blue arrow and (b) $-q_2 = -3$ degree, $-q_3 = 5$ degree and (c) $-q_2 = -50$ degree, $-q_3 = 30$ degree

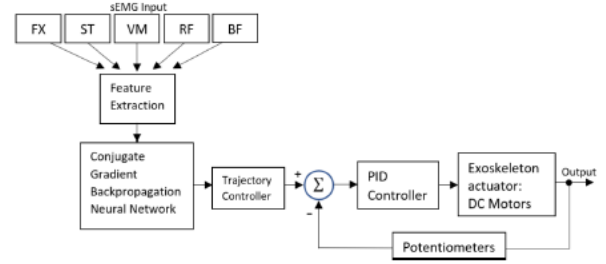


Fig. 5. Exoskeleton System diagram

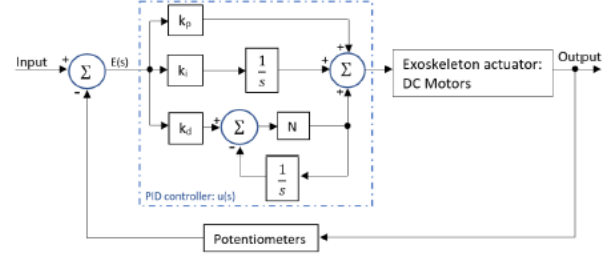


Fig. 6. PID Controller of Exoskeleton actuator diagram

The building block of the walking assistance system consists of input signal (sEMG), Feature Extraction, Backpropagation Neural Networks, Trajectory controllers, PID controllers, DC Motors and potentiometers as shown in fig. 5.

C. PID Controller

PID controller as shown in fig. 6 is a feedback controller that uses error $E(s)$ as the input. The error is the difference between the desire angle and angle of the suit. The PID controller consists of three terms, a proportional term (P), an integration term (I) and a derivative term (D). However, the controller is sensitive to noise. It requires a filter to reduce noise signal. We can adjust the filter coefficients (N) to reduce noise. We use the (1) and (2) to design PID controller.

$$E(s) = \text{setpoint} - \text{feedback} \quad (1)$$

$$u(s) = E(s) \left(K_p + K_i \frac{1}{s} + K_d \frac{N}{1+N\frac{1}{s}} \right) \quad (2)$$

D. Surface Electromyography (sEMG)

Electromyography as shown in fig.7 is the signal from muscle based on detection of the muscle's electrical activity [6]. A surface EMG sensor is an electromyography sensor. We use the dataset of sEMG for training scaled conjugate gradient backpropagation Neural Network to classify the lower limb posture. The sEMG signals including muscle of the anterior thigh, muscle of the posterior thigh and angle of flexion at the knee are measured from Myo Gesture control Armband sensors. The anterior thigh muscles used in hip bending and knees stretching contain Rectus Femoris (RF) and Vastus Medialis (VM). The posterior thigh muscles used in stretches hip and bent knees consist of Biceps Femoris (BF) and Semitendinosus (ST).

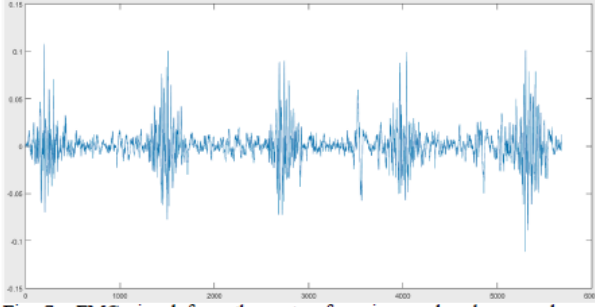


Fig. 7. sEMG signal from the rectus femoris muscle when people are walking.

E. Feature Extraction

Feature Extraction is the process of transforming the input data into a set of features [7] and reducing the amount of data used to train Neural Network. In this paper, time domain and frequency domain approach are used. The time domain consists of Root Mean Square, Mean Absolute Value, Simple Square Integral, Variance of EMG and Integrated Absolute Value. The frequency domain consists of Mean Frequency and Median Frequency.

1) Root Mean Square (RMS)

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (3)$$

2) Mean Absolute Value (MAV)

$$MAV = \frac{1}{N} \sum_{i=1}^N |x_i| \quad (4)$$

3) Simple Square Integral (SSI)

$$SSI = \sum_{i=1}^N |x_i|^2 \quad (5)$$

4) Variance of EMG (VAR)

$$VAR = \frac{1}{N-1} \sum_{i=1}^N x_i^2 \quad (6)$$

5) Integrated Absolute Value (IAV)

$$IAV = \sum_{i=1}^N |x_i| \quad (7)$$

6) Mean Frequency

$$MNF = \frac{\sum_{j=1}^M f_j P_j}{\sum_{j=1}^M P_j} \quad (8)$$

7) Median Frequency

$$\sum_{j=1}^{MDF} P_j = \sum_{j=MDF}^M P_j = \frac{1}{2} \sum_{j=1}^M P_j \quad (9)$$

Where x is the sEMG signal and P is the power spectrum of the sEMG signal.

F. Pattern recognition using the scaled conjugate gradient Backpropagation Neural Network

The Backpropagation neural network is an algorithm for supervised learning by calculating errors from the difference between the output and the target using gradient descent. Pattern recognition networks are feedforward networks that

can be trained to classify inputs according to target classes. The input of the Backpropagation Neural Network is sEMG signal after processed from feature extraction phase. The output of Backpropagation Neural Network is used to control the movement of Exoskeleton Suit.

III. EXPERIMENT AND RESULTS

A. The Exoskeleton model and PID Controller

The model using a 12V DC motor as an actuator is shown in (10) as follows

$$\Theta \left[(Js^2 + bs) \left(\frac{Ls+R}{k_t} \right) + k_e s \right] + \frac{\tau_l(Ls+R)}{rk_t} = V \quad (10)$$

This suit uses two motors. The parameters of the DC motors [8] used in the suit are $L = 9.6 \times 10^{-10}$ H, $R = 0.6 \Omega$, $J = 0.004 \text{ kgm}^2$, $b = 0.0185 \text{ Nm/(rad/s)}$, $r = 7.5$ and $k = 0.065 \text{ V/rpm}$.

From the experiment, we adjust the parameter in MATLAB using $K_p = 1.69$, $K_i = 1.08$, $K_d = 0.66$ and Filter coefficients (N) = 72.72 in PID controller for the hip motor and using $K_p = 4.07$, $K_i = 5.73$, $K_d = 0.67$ and Filter coefficients (N) = 4571.31 for the ankle motor. Fig.8 and Fig. 9 are the angle measured from the hip motor and the ankle motor respectively. The red line is the output angle of motor with PID Controller, the yellow line is the response without PID Controller and the blue line is the reference input.

Fig.10 shows the PID Controller of Exoskeleton actuator diagram which is operated on MATLAB Simulink. All parameters, angle of hip and knee joint are setup as in Fig. 2.

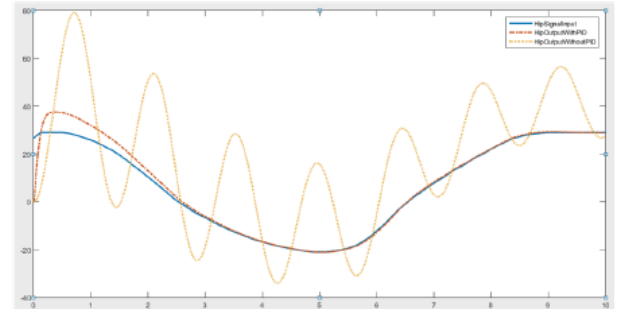


Fig. 8. Result of response time of the DC motor model at hip with PID control.

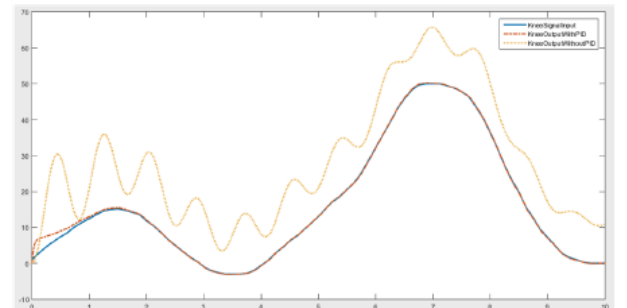


Fig. 9. Result of response time of the DC motor model at ankle with PID control.

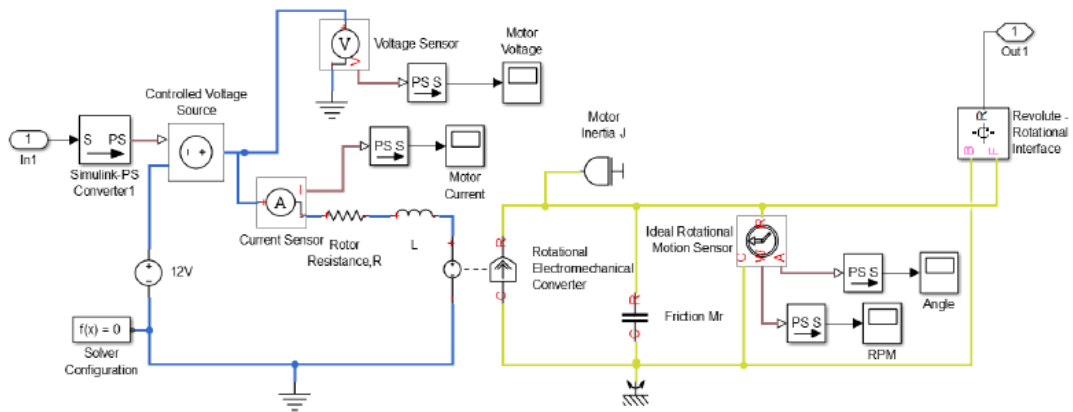


Fig. 10. DC Motor in Simulink

B. Backpropagation Neural Network and Walking Assistance system

In order to test performance of system, we create system simulation in the Simulink as shown in fig. 11. The pattern recognition part is neural network model, providing the controlled command to the exoskeleton suit via trajectory control. In order to train the Neural Network to recognized sEMG signal, we use dataset from [9] including samples from 11 subjects with knee Abnormality previously diagnosed by a professional and 11 normal persons. This dataset has 3 types of gestures, including walking, sitting and standing. These data were collected with electromyography and goniometry equipment MWX8 Data log Biometrics, 14 bits resolution, sampling rate at 1 kHz.

The Neural Network used in this part has 25 inputs, 10 hidden layers and 3 Output layers. In this paper, we use offline learning neural network. The trajectory controller is the motion program used to control angle of the motor in the suit.

The results of feature extraction trained from the Backpropagation Neural Network experiment are shown in table II. All features in time domain give the highest recognition rate at 99.39% so, we use this type of feature for commanding the lower limb exoskeleton suit.

TABLE II. PERCENT CORRECT CLASSIFICATION

Feature Extraction	Percent Correct Classification
RMS	91.52
MAV	95.25
SSI	94.35
VAR	78.29
IAV	98.39
Mean frequency	81.10
Median frequency	69.37
All feature in time domain	99.39
All feature in frequency domain	85.15
All feature extraction	95.15

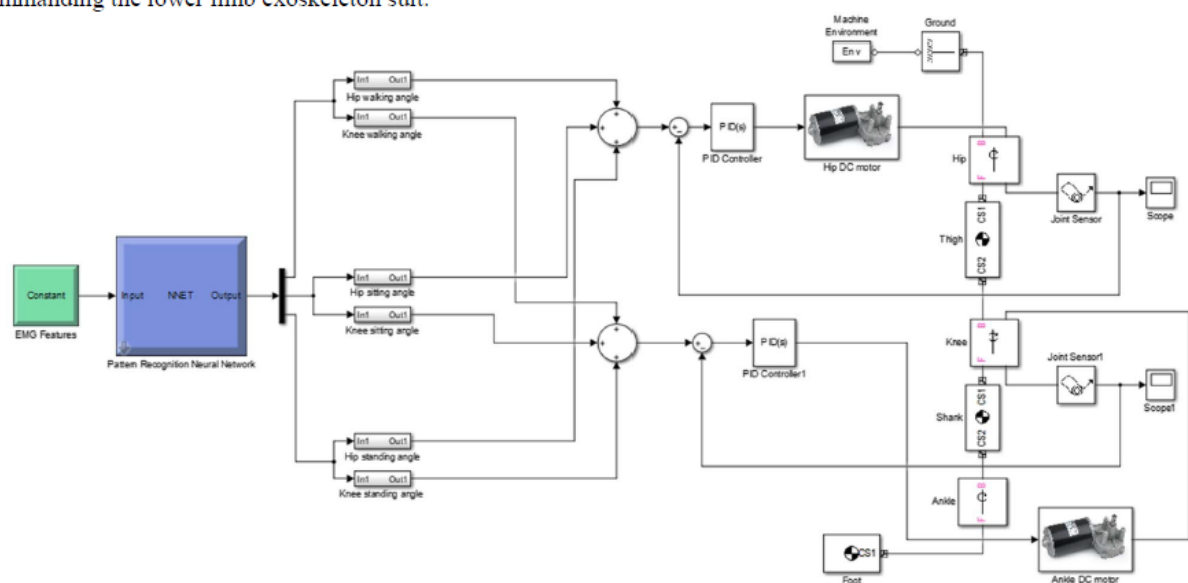


Fig. 11. Exoskeleton and Neural Network simulation in Simulink

IV. CONCLUSION

In this paper, we have designed the walking assistance system for walking aid of aged people. Our system is divided into 3 main parts including Feature extraction which consists of Time domain and Frequency domain, pattern recognition and exoskeleton suit system. We use the sEMG as the input of the system which has 3 types of gestures, including walking, sitting and standing to train the neural network in order to command the lower limb exoskeleton suit. The recognition rate of the sEMG signal is reach 99.39%.

In future work, this system will be use in the real Lower Limb Exoskeleton Prototype as shown in Fig. 12.

REFERENCES

- [1] Obnithi karantararat, Yuttana Kitjaidure, "The Design of a Lower Limb Exoskeleton Suit for Walking Assistance Using PID Controller and Mechanism of Four Bar Linkages", Conference proceeding 10th ECTI-CARD-2018, Phitsamulok, Thailand, pp. 69 -72, june 2018.
- [2] Gait. (2018). Retrieved from www.physio-pedia.com/Gait/
- [3] Normalgait. (2018). Retrieved from <https://musculoskeletalkey.com/normal-gait/>
- [4] anthropometry. (2018). Retrieved from www.slideshare.net/vimal333/anthropometry-15401067
- [5] Zhiyong Yang, Lihua Gui, Xiuxia Yang, Wenjin Gu, Yuanshan Zhang, "Simulation Research of Exoskeleton Suit Based on Sensitivity Amplification Control", Proceedings of the IEEE International Conference on Automatic and Logistics, Jinan, China, pp.1353-1357, Aug. 18-21, 2007.

- [6] Didier Gamet, Odette Fokapu, "Electromyography", Encyclopedia of Biomaterials and Biomedical Engineering, Infoma Healcare USA, Inc., 2008.
- [7] what is Feature Extraction (2018). Retrieved from www.igi-global.com/dictionary/feature-extraction/10960
- [8] M. Dooner, J. Wang and A. Mouzakitis, "Dynamic modelling and experimental validation of an automotive windshield wiper system for hardware in the loop simulation", Systems Science & Control Engineering: An Open Access Journal, Vol. 3, 230-239, 2015.
- [9] EMG dataset in Lower Limb Data Set (2018). Retrieved from <http://archive.ics.uci.edu/ml/datasets/emg+dataset+in+lower+limb>



Fig. 12. Lower Limb Exoskeleton Prototype: Electro Armor 6

ประวัติผู้เขียน

ชื่อ-นามสกุล	นายโอบนิธิ กรีณพรัตน์
วัน เดือน ปีเกิด	28 เมษายน 2538 ที่กรุงเทพมหานคร
ประวัติการศึกษา	2559 วิศวกรรมศาสตรบัณฑิต สาขาวิชาอิเล็กทรอนิกส์ (เกียรตินิยมอันดับสอง) สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ความชำนาญเฉพาะด้าน	1.) ระบบควบคุม 2.) การออกแบบวงจรอิเล็กทรอนิกส์ 3.) การออกแบบหุ่นยนต์
ประสบการณ์และผลงานวิจัย	
พ.ศ.2559	ได้รับถ้วยพระราชทาน สมเด็จพระเทพรัตนราชสุดาฯ สยามบรมราชกุมารี จากการชนะเลิศการประกวดแข่งขันในโครงการ KMITL Innovative Awards 2016 ในระดับอุดมศึกษา
พ.ศ. 2560	ได้รับพระราชทานรางวัลเด็กและเยาวชนดีเด่นแห่งชาติ สาขาสิ่งประดิษฐ์และนวัตกรรม ประจำปี 2560
พ.ศ. 2561	ผลงานวิจัยด้านการออกแบบชุด Exoskeleton สำหรับช่วยในการเดิน