

Project No.30

**ระบบควบคุมการเคลื่อนไหวผ่านคลื่นไฟฟ้าสมอง
Robot Movement Controlling System with Motor Imagery(MI)
based Brain-Computer Interface(BCI)**

จัดทำโดย

- | | |
|------------------------|-------------|
| 1. ปกรณ์ อุยสุข | 58070501039 |
| 2. ระวี เลิศกุลวานิชย์ | 58070501055 |
| 3. วิศวัชิต หนูมาศ | 58070501067 |

ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ ดร. สุชาติพย์ มณีวงศ์วัฒนา

“เข้าใจง่ายรับว่าได้อ่านเนื้อหาภายในรายงานฉบับนี้แล้ว”

.....
(ผศ.ดร. สุชาติพย์ มณีวงศ์วัฒนา)

อาจารย์ที่ปรึกษา



ระบบควบคุมการเคลื่อนไหวผ่านคลื่นไฟฟ้าสมอง
Robot Movement Controlling System with Motor Imagery(MI)
based Brain-Computer Interface(BCI)

ปกรณ์ อุยสุข
ระวี เดิศกุลวานิชย์
วิศวชิต หนูมาศ

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ปีการศึกษา 2561

ระบบควบคุมการเคลื่อนไหวผ่านคลื่นไฟฟ้าสมอง

ปกรณ์ อุยสุข
ระวี เลิศกุลวานิชช์
วิศวชิต หนุมานศ

โครงการนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
ปีการศึกษา 2561
ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี

คณะกรรมการสอบโครงการ

..... อาจารย์ที่ปรึกษา
(ดร.สุชาติพย์ มณีวงศ์วัฒนา)

..... กรรมการ
(ดร.ชาตรนต์ หาญสมบูรณ์)

..... กรรมการ
(ดร.อัญชลิสา แต้ตระกูล)

..... กรรมการ
(รศ.ดร.พีรพล ศิริพงศ์วุฒิกร)

**Robot Movement Controlling System with Motor Imagery(MI)
based Brain-Computer Interface(BCI)**

Mr. Pakorn Ouisui
Mr. Rawee Lertkulwanich
Mr. Witsavachit Hanumas

**A Project Submitted in Partial Fulfillment of the Requirements
for the Degree of Bachelor of Engineering
Department of Computer Engineering, Faculty of Engineering
King Mongkut's University of Technology Thonburi
Academic Year 2018**

Project Committee

..... Advisor
(Dr. Suthathip Maneewongvatana, Ph.D.)

..... Chairman
(Dr. Jaturon Hansomburana, Ph.D.)

..... Committee
(Dr. Unchalisra Taetragool, Ph.D.)

..... Committee
(Asst. Prof. Peerapon Siripongwutikorn, Ph.D.)

หัวข้อโครงการ	ระบบควบคุมการเคลื่อนไหวผ่านคลื่นไฟฟ้าสมอง
หน่วยกิต	3 หน่วย
สมาชิก	นายปกรณ์ อุยสุข นายระวี เลิศกุลวานิชย์ นายวิศวชิต หนุมานศ
ที่ปรึกษา	พศ.ดร.สุชาติพย์ มณีวงศ์วัฒนา
ระดับการศึกษา	ปริญญาตรี
ภาควิชา	วิศวกรรมคอมพิวเตอร์
ปีการศึกษา	2561

บทคัดย่อ

โครงการนี้มีจุดประสงค์เพื่อนำเทคโนโลยี Brain-Computer Interface (BCI) มาประยุกต์ใช้ในการควบคุมการเคลื่อนไหว โดยจะทำการตรวจจับคลื่นไฟฟ้าสมอง หรือ Electroencephalogram(EEG) จากการจินตนาการเคลื่อนไหวร่างกาย (Motor Imagery) ได้แก่การจินตนาการว่ายกมือซ้าย มือขวา ขาขวาและขาซ้าย (4 Classes) และจำแนกรูปแบบของคลื่นสมองที่แตกต่างกันนี้เพื่อควบคุมการเคลื่อนไหวของหุ่นยนต์ให้สอดคล้องกับการจินตนาการ ในการเลือกคุณสมบัติ (Feature Extraction) ของข้อมูลจะใช้วิธีการที่มีชื่อว่า Common Spatial Pattern (CSP) และใช้ Linear Discriminant Analysis (LDA) เป็น Classifier ในการจำแนกประเภทของข้อมูล

คำสำคัญ: Brain-Computer Interface(BCI), Electroencephalogram(EEG), Motor Imagery, Common Spatial Pattern(CSP)

Project Title	Robot movement Controlling System with Motor Imagery(MI) based Brain-Computer Interface(BCI)
Project Credit	3 credits
Project Participant	Mr. Pakorn Ouisui Mr.Rawee Lertkulwanich Mr.Witsavachit Hanumas
Advisor	Asst. Prof. Suthathip Maneewongvatana, Ph.D.
Degree of Study	Bachelor's Degree
Department	Computer Engineering
Academic Year	2018

Abstract

This project intent is to use Brain-Computer Interface (BCI) technology to control the movement of robot by detecting the brain wave or Electroencephalogram (EEG) from the imagination of body movement (Motor Imagery). It will detect brain waves or Electroencephalogram (EEG) from the motor movement imagery (Motor imagery) which are imagine lifting left arm, right arm, right leg and left leg (4 classes), then adjust the different brain patterns to control the movement of the robot. Feature extraction of data will be performed by the Common Spatial Pattern (CSP) method and linear discriminant analysis (LDA) is used as a classification in the classification of data.

Keywords: Brain-Computer Interface(BCI), Electroencephalogram(EEG), Motor Imagery, Common Spatial Pattern(CSP)

กิตติกรรมประกาศ

การทำโครงการครั้งนี้ได้รับความช่วยเหลืออย่างดีเยี่ยมจาก พศ.ดร.สุชาทพย์ มณีวงศ์วัฒนา หรือ อาจารย์ฝ่าย ที่ปรึกษาโครงการ ที่กรุณาสละเวลาอันมีค่าให้กำปรึกษา รวมถึงข้อมูล แนวคิด และข้อเสนอแนะที่เป็นประโยชน์อย่างมากต่อการทำโครงการในครั้งนี้ ตลอดจนคอยติดตามดูแลเอาใจใส่ต่อการทำโครงการฉบับนี้จนสำเร็จลุล่วงได้ด้วยดี ผู้จัดทำโครงการจึงขอกราบขอบพระคุณเป็นอย่างสูง ณ ที่นี่ด้วย

ขอกราบขอบพระคุณ คุณพ่อ คุณแม่ และพี่ๆ ที่ให้กำลังใจ ให้ความช่วยเหลือ และให้คำปรึกษาตลอดมา ขอขอบคุณเพื่อนๆ พี่ๆ และน้องๆ ในภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคนที่ให้กำลังใจ คอยดูแล และอยู่เคียงข้างกันเสมอมา

สารบัญ

	หน้า
บท	
บทที่ 1 บทนำ	7
1.1 ที่มาและความสำคัญของปัญหา	7
1.2 วัตถุประสงค์	7
1.3 ขอบเขตโครงการ	8
1.4 ประโยชน์ที่คาดว่าจะได้รับ	8
1.5 แผนการดำเนินงาน	9
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	10
2.1 Literature review	10
2.2 Related Theory	11
2.2.1 Brain Part and Function	11
2.2.2 Brain Signal	13
คลื่นสมองระดับเบต้า Beta Brainwave (ความถี่ระหว่าง 14 – 30 Hz)	13
คลื่นสมองระดับอัลฟ่า Alpha Brainwave (ความถี่ระหว่าง 8 – 13.9 Hz)	13
คลื่นสมองระดับธे�ต้า Theta Brainwave (ความถี่ระหว่าง 4 – 7.9 Hz)	13
คลื่นสมองระดับเดลต้า Delta Brainwave (ความถี่ระหว่าง 0.1 – 3.9 Hz)	14
2.2.3 Brain-Computer Interface(BCI)	14
INVASIVE BCI	15
NON INVASIVE BCI	16
2.2.4 Motor Imagery	16
2.2.5 Electroencephalography (EEG)	17
2.2.6 Data Pre-Processing	19
Channel Location	19
Frequency Filter	20
Re-Referencing	22
Add Event	23

Extract Epoch	24
Run ICA	26
Eye blinking	26
Eyeball movement	27
Muscle artifact	28
2.2.7 Feature Extraction: Common Spatial Pattern (CSP)	29
2.2.8 Feature Extraction: Filter Bank Common Spatial Pattern (FBCSP)	30
2.2.9 Machine Learning: Linear Discriminant Analysis (LDA)	32
2.2.10 Model Evaluation	33
True Positive (TP)	33
False Positive (FP)	33
True Negative (TN)	34
False Negative (FN)	34
Accuracy	34
Precision	34
Recall	34
2.3 Programming Language and Software used	34
2.3.1 MATLAB R2018a	34
2.3.2 Lab Streaming Layer (LSL)	35
2.4 Hardware part and library used	36
2.4.1 OpenBCI	36
Hardware: Cython + Daisy board	36
2.4.2 Headware: Ultracortex Mark IV	37
2.4.3 i-Duino UNO	40
บทที่ 3 การออกแบบและระบบบีบีวิชิวัลจี้	42
3.1 อุปกรณ์ที่ใช้	42
3.2 Software ที่ใช้	42
3.3 ลำดับการปฏิบัติสำหรับผู้เข้าร่วมการทดลอง	43

3.4 Preprocessing	43
3.5 การวิเคราะห์ข้อมูล	44
3.6 Coding and Algorithm	45
3.6.1 Obtained data	45
3.6.2 Extract Epoch	48
3.6.3 Preprocessing	49
3.6.4 Feature Extraction การเลือกคุณสมบัติ	49
3.6.5 Real time feedback	50
3.6.6 Classifier	52
บทที่ 4 ผลการทดลอง	54
4.1 ผลการเก็บข้อมูล	54
4.1.1 ผู้ทดลอง	54
4.1.2 ข้อมูลติบ	54
4.1.3 ข้อมูลหลังจาก preprocessing	55
4.2 ผลการวิเคราะห์ข้อมูล	56
4.2.1 ผลการทดลองเมื่อนำข้อมูล Training/Testing set มาประเมินผล	56
4.2.2 ผลการทดสอบแบบ Real-time และส่งค่าไปควบคุมหุ่นหนวดแบบ	57
บทที่ 5 บทสรุป	62
5.1 ปัญหาที่พบ	62
5.1.1 ปัญหาของโปรแกรม	62
5.1.2 ปัญหาในการ coding	62
5.1.3 ปัญหาในการเก็บข้อมูล	62
5.1.4 ปัญหาในการวิเคราะห์ข้อมูล	63
5.1.5 ปัญหาในการ preprocessing	63
5.2 ข้อเสนอแนะ	63
บรรณานุกรม	65

สารบัญรูปภาพ

รูปภาพ	หน้า
รูปที่ 2.1 Brain	11
รูปที่ 2.2 Beta Wave	13
รูปที่ 2.3 Alpha Wave	13
รูปที่ 2.4 Theta Wave	14
รูปที่ 2.5 DeltaWave	14
รูปที่ 2.6 Invasive BCI	15
รูปที่ 2.7 Non Invasive BCI	16
รูปที่ 2.8 บริเวณที่มีกิจกรรมทางสมองจากการเคลื่อนไหวร่างกายส่วนต่างๆ	17
รูปที่ 2.9 Electroencephalography (EEG)	17
รูปที่ 2.10 แสดงข้อมูลค่าของสัญญาณ EEG ตาม Time domain	19
รูปที่ 2.11 Channel location	20
รูปที่ 2.12 Bandpass filter ในช่วงความถี่ 8-30Hz	21
รูปที่ 2.13 คลื่น EEG และบนกระโหลกก่อนทำการ filter	21
รูปที่ 2.14 คลื่น EEG และบนกระโหลกหลังทำการ filter	22
รูปที่ 2.15 พลังงานสมองหลังทำการ Re-reference	22
รูปที่ 2.16 การเพิ่ม Event ลงในชุดข้อมูล	23
รูปที่ 2.17 การตัด Epoch ของคลาสที่สนใจ	24
รูปที่ 2.18 แสดงค่า ERP เฉลี่ยจาก event ประเภทเดียวกัน	25
รูปที่ 2.19 Eye blinking artifact	26
รูปที่ 2.20 Eyeball movement artifact	27
รูปที่ 2.21 Muscle artifact	28
รูปที่ 2.22 การเลือก Accept หรือ Reject ข้อมูลICA	29
รูปที่ 2.23 LDA Projection	31
รูปที่ 2.24 โปรแกรม MATLAB เวอร์ชัน R2018a	33
รูปที่ 2.25 Lab Streaming Layer (LSL)	34
รูปที่ 2.26 USB Dongle, Cython + Daisy board, 4xBattery AA Socket and wires	35
รูปที่ 2.27 Ultracortex 3D-Printable model	35
รูปที่ 2.28 electrode แบบหนาม	36
รูปที่ 2.29 electrode แบบเรียบ	36
รูปที่ 2.30 electrode แบบ support	37
รูปที่ 2.31 ที่หนีบหูเพื่อเป็นจุดเทียบ ground	37

รูปที่ 2.32 Ultracortex Mark IV เมื่อประกอบเสร็จสมบูรณ์	38
รูปที่ 2.33 i-Duino UNO	38
รูปที่ 3.1 ตำแหน่งอิเล็ก trode ที่ headset ใช้เทียบกับ International 10–20 system	40
รูปที่ 3.2 แผนผังการเก็บข้อมูลสำหรับการ train	41
รูปที่ 3.3 แผนภาพแสดงการทำงานของโครงงาน	43
รูปที่ 3.4 Flow Chart การสุ่มค่า trial	44
รูปที่ 3.5 flow chart ของ loop ที่ใช้เก็บ streaming ข้อมูลสำหรับใช้ train model	45
รูปที่ 3.6 ภาพแสดงช่วงที่ระบบเลือกไปใช้ทำ epoch	46
รูปที่ 3.7 แผนผังการทำ epoch	46
รูปที่ 3.8 ตัวอย่างพารามิเตอร์สำหรับฟังก์ชัน pop_importdata()	47
รูปที่ 3.9 ตัวอย่างพารามิเตอร์สำหรับฟังก์ชัน pop_eegfilt()	47
รูปที่ 3.10 แผนภาพคำนวณการทำงานของโปรแกรมเพื่อหา feature สำหรับแยกเมื่อกับเท้า	48
รูปที่ 3.11 ตัวอย่างหน้าจอแสดงผล feed back	49
รูปที่ 3.12 แผนภาพแสดงข้อมูลชุดแรกที่ถูกเก็บสำหรับ real time feed back	49
รูปที่ 3.13 แผนภาพแสดงขั้นตอนการเก็บข้อมูลชุดต่อๆ ไป	50
รูปที่ 3.14 แผนภาพแสดงคำนวณการทำงานของฟังก์ชัน classifier	52
รูปที่ 4.1 ขั้นตอนการบันทึกสัญญาณจากอุปกรณ์ OpenBCI	53
รูปที่ 4.2 ข้อมูลสัญญาณ EEG ที่ยังไม่ได้ผ่านการ Filter	54
รูปที่ 4.3 ข้อมูลดิบสัญญาณ EEG ที่ผ่านการ Band-pass Filter 8-30 Hz	54
รูปที่ 4.4 ผลการจำแนกประเภทการจินตนาการ	55
รูปที่ 4.5 วงจร Prototype สำหรับสั่งการทำงานผ่านคลื่นสัญญาณประสาท	56

สารบัญตาราง

ตาราง	หน้า
ตารางที่ 1.1 แผนการดำเนินงาน	9
ตารางที่ 4.1 ตารางแสดงความแม่นยำในการแยกประเภทของ LDA Classifier ด้วย CSP	55
ตารางที่ 4.2 Confusion Matrix	57
ตารางที่ 4.3 Accuracy	57
ตารางที่ 4.4 Recall	57
ตารางที่ 4.5 Precision	57

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของปัญหา

โรคหลอดเลือดสมอง เป็นสาเหตุหลักที่ทำให้ผู้ป่วยจำนวนหลายล้านคนทั่วโลกเป็นอัมพาตหรือพิการ ผู้ป่วยที่เป็นอัมพาตนั้นจะมีอาการกล้ามเนื้อของอวัยวะส่วนต่างๆ ในร่างกายไม่สามารถขยับเคลื่อนไหวได้ เนื่องจากภาวะเจ็บป่วยที่ทำให้ระบบสั่งการของสมองสำหรับควบคุมการเคลื่อนไหวของร่างกายเกิดความผิดพลาด หรือการประஸบอับดิเหตุ การเจ็บป่วยที่กระแทกกระเทือนกับสมอง ไขสันหลัง กระดูกคอและเส้นประสาทด่างๆ ที่ควบคุมการเคลื่อนไหวของร่างกาย เหล่านี้ล้วนเป็นสาเหตุที่อาจทำให้เกิดโรคหลอดเลือดสมอง ซึ่งทำให้เส้นทางการส่งกระแสประสาทรหัสว่างสมองและกล้ามเนื้อขาดช่วง และเนื่องจากจำนวนประชากรโลกมีอายุมากขึ้น จำนวนผู้ที่ได้รับผลกระทบจากโรคหลอดเลือดสมองมีแนวโน้มจะเพิ่มขึ้นอย่างมากในอีกไม่ถึงปีข้างหน้านี้ และไม่เพียงแต่อวัยวะร่างกายเท่านั้นที่จะได้รับผลกระทบจากโรคหลอดเลือดสมอง ยังรวมไปถึงระบบการรับรู้ความเข้าใจทางอารมณ์ก็ยังได้รับผลกระทบอย่างมากอีกด้วย

ในช่วง 10 ปีที่ผ่านมา มีงานวิจัยมากมายที่สนใจสนับสนุนการพัฒนาผู้ป่วยที่เป็นอัมพาตด้วยเทคโนโลยี BCI (Brain-Computer Interface) ด้วยวิธีการที่เรียกว่า Motor imagery(MI) กล่าวโดยสรุปคือ อวัยวะต่างๆ ในร่างกายถึงแม้ว่าจะได้รับความเสียหายอย่างหนักจนไม่สามารถควบคุมสั่งการได้ (Offline) แต่สมองนั้นก็ยังสามารถถูกกระตุ้น (Activated) ได้จากการจินตนาการ (Motor imagery) หรือสังเกตการเคลื่อนไหวจากผู้อื่น (Observing movement) [1]

โครงงานเรื่อง ระบบควบคุมการเคลื่อนไหวผ่านคลื่นไฟฟ้าสมอง Robot Movement Controlling System with Motor Imagery(MI) based Brain-Computer Interface(BCI) นี้จะพัฒนาระบบที่สามารถแยกประเภทคลื่นไฟฟ้าสมองที่ได้รับผ่านอุปกรณ์ตรวจจับคลื่นสมอง OpenBCI ว่าเป็นคลื่นไฟฟ้าสมองที่เกิดจากการจินตนาการการเคลื่อนไหวในรูปแบบใดจาก 4 รูปแบบคือมือซ้าย มือขวา ขาซ้ายและขาขวาเพื่อนำไปควบคุมการเคลื่อนไหวของหุ่นยนต์เบื้องต้นนั่นเอง

1.2 วัตถุประสงค์

- เพื่อศึกษา Dataset จาก BCI Competition IV 2a [2] ซึ่งเป็น Dataset คลื่นสมองที่ได้รับจากวิธี Motor Imagery
- เพื่อศึกษาการ Preprocessing ข้อมูลสัญญาณสมอง และทำการความเข้าใจพื้นฐานเกี่ยวกับคลื่นไฟฟ้าสมอง
- เพื่อให้ได้รับสัญญาณคลื่นไฟฟ้าสมอง(EEG) โดยใช้อุปกรณ์ OpenBCI รุ่น Ultracortex

- เพื่อพัฒนาระบบที่สามารถแยกประเภทของคลื่นสมอง ได้ที่รับจากวิธีการ Motor imagery(MI) ออก เป็น 4 รูปแบบคือ มือซ้าย มือขวา ขาซ้ายและขาขวา ได้อย่างถูกต้องแม่นยำ
- เพื่อนำระบบไปประยุกต์ใช้ในการควบคุมการเคลื่อนไหวของหุ่นยนต์เบื้องต้น

1.3 ขอบเขตโครงการ

- โครงการเรื่องนี้มุ่งเน้นไปที่การพัฒนาและศึกษาในองค์ความรู้ที่เกี่ยวกับคลื่นสัญญาณสมอง แล้วจึง นำมาประยุกต์เพื่อควบคุมอุปกรณ์ช่วยในการเคลื่อนไหวต้นแบบ
- ใช้อุปกรณ์ตรวจจับสัญญาณ EEG OpenBCI 8 channels รุ่น Ultracortex
- พัฒนาระบบที่สามารถแบ่งแยกประเภทจากการจินตนาการ 4 รูปแบบคือมือซ้าย มือขวา ขาซ้ายและ ขาขวา

1.4 ประโยชน์ที่คาดว่าจะได้รับ

- สามารถเข้าใจและอธิบายรายละเอียดของ Motor imagery Dataset BCI Competition IV 2a ได้
- สามารถทำการ Preprocess และวิเคราะห์หาคุณลักษณะที่สำคัญของสัญญาณ EEG เพื่อบุคคลรูป แบบการเคลื่อนไหวแต่ละประเภท
- สามารถควบคุมการเคลื่อนไหวของอุปกรณ์ผ่านคลื่นไฟฟ้าสมอง ได้

1.5 แผนการดำเนินงาน

ขั้นตอนการดำเนินการ	ก.ย. 61				ต.ค. 61				พ.ย. 61				ธ.ค. 61				ม.ค. 62				ก.พ. 62				พ.ค. 62						
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4			
1. เสิร์ฟเวอร์ที่ใช้โค้ดภาษา C																															
2. ตั้งค่า Dataset																															
3. ตั้งค่าการเรียนรู้ใน EEGLAB																															
4. ตั้งค่าการ Preprocess ชั้นหนึ่ง																															
5. ตั้งค่า Naive Bayes หรือ Feature Extraction (CSP)																															
6. ตั้งค่าการเรียนรู้ OpenBCI Ultracortex																															
7. ทดลองระบบการคลื่อนไหว																															
8. ทำ Data Streaming ด้วย MATLAB																															
9. พัฒนาระบบการเรียนรู้ CSP และ LDA Classification ด้วย MATLAB																															
10. ออกแบบ GUI สำหรับการทำนายชุดทดสอบ																															
11. เรียนรู้ตัวบ่งชี้ของแต่ละชุดทดสอบ คำนวณผลลัพธ์ทางทำนาย คิดเห็นและอภิปราย																															
12. Test User																															
13. ประเมินผล																															

ตารางที่ 1.1 แผนการดำเนินงาน

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 Literature review

Brain-Computer Interface(BCI) [3] เป็นระบบที่ช่วยให้สมองและอุปกรณ์ภายนอกสามารถสื่อสาร และควบคุมกันได้ ระบบนี้มีเป้าหมายที่จะรับรู้ถึงความตั้งใจของมนุษย์จากกิจกรรมของระบบประสาท ซึ่งโดยทั่วไปได้รับมาจากสัญญาณประสาทแบบ Non-invasive ด้วยชุด Electrode จับสัญญาณ Electroencephalogram (EEG) [3] จำนวนหลายๆ Channel การนำเทคโนโลยี BCI มาใช้งานจริงนั้นสามารถประยุกต์ใช้ได้ในหลากหลายรูปแบบ วิธีหนึ่งนั้นมีชื่อว่า “Motor Imagery” (MI) [4] ซึ่งเป็นกระบวนการทางสมองของการคิดการเคลื่อนไหว เช่น เคลื่อนไหวมือซ้ายหรือมือขวาโดยไม่ต้องดำเนินการจริงๆ ระบบ MI-BCI ได้รับการออกแบบเพื่อค้นหารูปแบบในสัญญาณ EEG และจับคู่สัญญาณกับการเคลื่อนไหวที่คิดหรือจินตนาการอยู่ ข้อมูลที่ได้ดังกล่าวเป็นประโยชน์อย่างมากสำหรับการให้ผู้ป่วยอัมพาตอย่างรุนแรง หรือผู้ป่วยที่ต้องเคลื่อนไหวด้วยอวัยวะเทียมที่จะทำให้พวกเขารู้สึกได้ทางกายภาพ สามารถสื่อสารความต้องการของพวกเขากลางๆ

อย่างไรก็ตามการจัดจำแนกแบบเหล่านี้ยังคงไว้ต่อข้อพิจพลด และความแปรปรวนระหว่างเรื่องของสัญญาณ EEG ที่ต้องการข้อมูลสัญญาณที่มีการ Label จำนวนมาก การขาดข้อมูลที่มี Label อย่างเพียงพอทำให้การ Train ข้อมูลด้วย Classifier ที่มีความซับซ้อนในการปรับค่า Parameter เป็นเรื่องที่ยาก

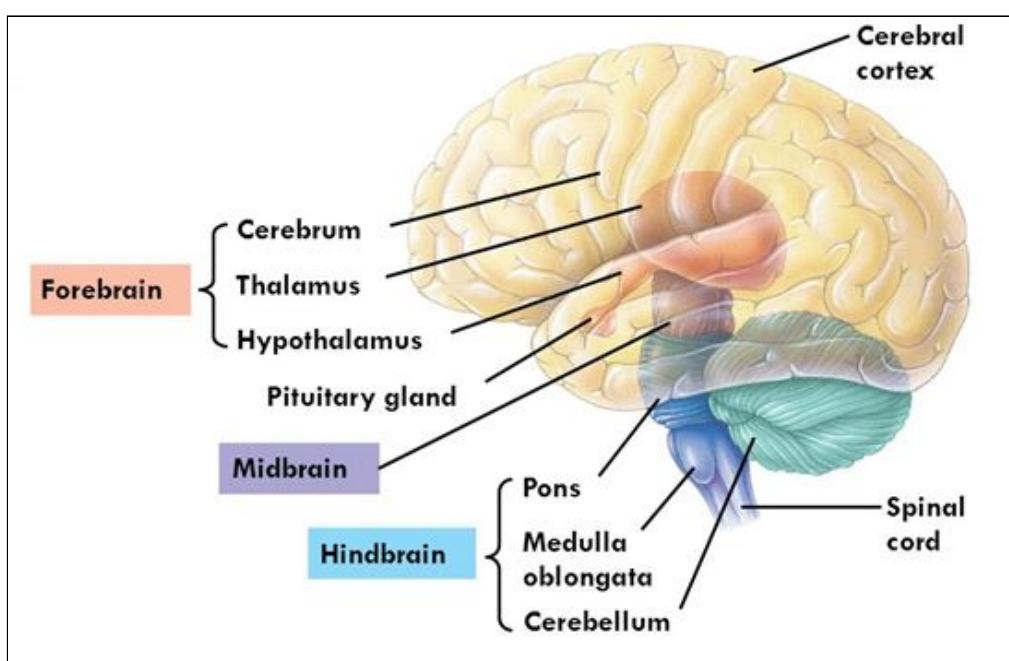
อย่างไรก็ตามความรู้เกี่ยวกับโครงสร้างของสัญญาณ EEG ช่วยให้เราสามารถออกแบบตัวแยกคุณลักษณะ (Feature Extractor) ที่ครอบคลุมข้อมูลในล่วงหน้าที่ต้องการ ได้อย่างถูกต้องและเหมาะสมมากยิ่งขึ้น ซึ่งคุณลักษณะที่ได้นี้สามารถรวมกับ Classifier ประเภทที่ง่ายๆ อย่าง Support Vector Machine (SVM) [5] หรือ Linear Discriminant Analysis (LDA) [6] ได้อย่างดี สำหรับการแยกคุณสมบัติที่จะสามารถจำแนกความแตกต่างของแต่ละประเภทข้อมูลมากที่สุดนั้น ได้มีการแนะนำวิธีการต่างๆ เช่น อัลกอริズึม Common Spatial Pattern (CSP) [7] ที่เป็นที่รู้จักกันดีเรียนรู้เกี่ยวกับการหา spatial filters [8] ซึ่งเพิ่มการจำแนกได้สูงสุดระหว่างชุดข้อมูลสองชุด ได้อย่างดี และนิยมใช้กันทั่วไปในงานศึกษาเรื่องสัญญาณประสาทที่เกิดจาก การจินตนาการ

2.2 Related Theory

ในการทำโครงการนี้ ผู้จัดทำได้ทำการศึกษาเกี่ยวกับเทคโนโลยี และความรู้ที่เกี่ยวข้อง เพื่อนำมาพัฒนาโครงการให้เป็นไปตามที่ตั้งไว้มีหัวข้อดังต่อไปนี้

2.2.1 Brain Part and Function

สมอง [9] เป็นอวัยวะที่สำคัญ และมีความซับซ้อนที่สุดของร่างกายมนุษย์ มันเป็นศูนย์กลางของระบบประสาท มีหน้าที่ควบคุมอวัยวะต่างๆ ในร่างกาย สมองประกอบด้วย 3 ส่วน ได้แก่ สมองส่วนหน้า (Forebrain), สมองส่วนกลาง (Midbrain) และสมองส่วนหลัง (Hindbrain) ดังรูปที่ 2.1



รูปที่ 2.1 Brain

[ที่มา: <https://www.askiitians.com/blog/books-didnt-tell-brain/>]

- สมองส่วนหน้า (Forebrain)
 - ซีรีบรัม (Cerebrum)

เป็นส่วนที่มีขนาดใหญ่ที่สุด เป็นศูนย์ควบคุมความคิดและการกระทำต่างๆ แบ่งออกเป็นสองซีก แต่ละซีกเรียกว่า Cerebral hemisphere และแต่ละซีกจะแบ่งได้เป็น 4 พู ดังนี้

 - Frontal lobe ทำหน้าที่ควบคุม การให้เหตุผล การวางแผน อารมณ์ ส่วนของความสามารถในการพูดและการแก้ปัญหา

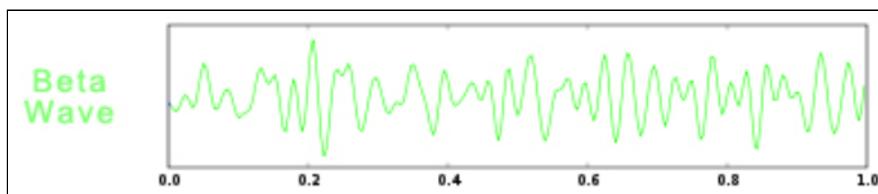
- Parietal Lobe เกี่ยวข้องกับความรู้สึกจากการสัมผัส ความเจ็บปวด ความรู้สึกจากอุณหภูมิ ช่วยแยกแยะระหว่างรูปร่าง ขนาด และสี ช่วยควบคุมการรับรู้เชิงพื้นที่และภาพ
- Occipital lobe การประมวลผลภาพทั้งหมดถูกควบคุมโดยส่วนนี้
- Temporal lobe เกี่ยวข้องกับการพูด หน่วยความจำ การได้ยิน การจัดลำดับ และการจัดระเบียบความคิด
- ทาลามัส (Thalamus)
 - เป็นศูนย์รวมกระแสไฟฟ้าของเข้าออกและแยกกระแสไปยัง Cerebrum
- ไฮปोทาลามัส (Hypothalamus)
 - ทำหน้าที่เป็นศูนย์กลางของระบบประสาಥ้อต โน้มติควบคุมปริมาณน้ำ และอุณหภูมิในร่างกาย รวมถึงวงจรการตื่นและหลับ
- สมองส่วนกลาง (Midbrain)
 - เป็นสถานีรับส่งประสาทระหว่างสมองส่วนหน้ากับส่วนหลังท้ายและส่วนหน้ากับนัยน์ตาซึ่งทำหน้าที่ควบคุมการเคลื่อนไหวของลูกตาและม่านตา
- สมองส่วนหลัง (Hindbrain)
 - พอนส์ (Pons)
 - เกี่ยวข้องกับการกลืน การเคลื่อนไหวของลูกตา และการยกใบหน้า
 - ซีรีเบลลัม (Cerebellum)
 - เกี่ยวข้องกับการประสานงานของอวัยวะ เพื่อเคลื่อนไหวและสมดุล
 - เมดูลา ออบลองกาตา (Medulla oblongata)
 - เกี่ยวข้องกับอวัยวะที่ทำงานอัตโนมัติ เช่น การหายใจ อัตราการเต้นของหัวใจ

2.2.2 Brain Signal

สมองมีการส่งข้อมูลเป็นสัญญาณไฟฟ้า [10] เพื่อควบคุมอวัยวะภายในร่างกาย โดยเราสามารถแบ่งคลื่นสมองตามความถี่ได้เป็น 4 กลุ่ม ดังนี้

- คลื่นสมองระดับเบต้า Beta Brainwave (ความถี่ระหว่าง 14 – 30 Hz)

เป็นคลื่นสมองที่เร็วที่สุด เป็นช่วงที่สมองเปิดรับข้อมูล มีการตื่นตัวของระบบประสาททุกด้าน เช่น ความรู้สึกหงุดหงิด สับสน ทำให้ประสิทธิภาพในการคิดตัดสินใจไม่ดี ดังรูปที่ 2.2

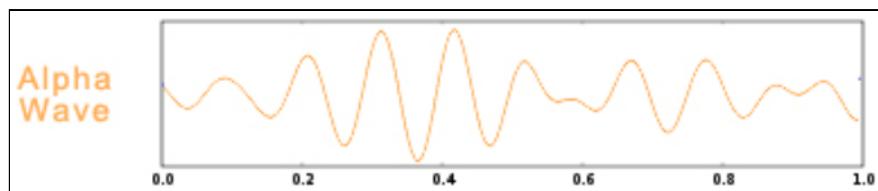


รูปที่ 2.2 Beta Wave

[ที่มา: <https://itsusync.com/different-types-of-brain-waves-delta-theta-alpha-beta-gamma>]

- คลื่นสมองระดับอัลฟ่า Alpha Brainwave (ความถี่ระหว่าง 8 – 13.9 Hz)

คลื่นสมองช่วงนี้จะเกิดเมื่อเรามีความสงบ แต่อยู่ในภาวะตื่นตัว ภาระนี้ทำให้รับข้อมูลได้ที่สุด เรียกว่าความจำได้ง่าย สมองสามารถเปิดรับข้อมูลได้อย่างเต็มที่และเรียนรู้ได้อย่างรวดเร็ว มีความคิดสร้างสรรค์ เป็นสภาวะที่จิตมีประสิทธิภาพสูง ดังรูปที่ 2.3

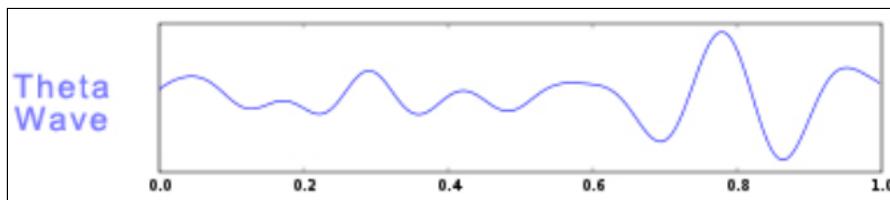


รูปที่ 2.3 Alpha Wave

[ที่มา: <https://itsusync.com/different-types-of-brain-waves-delta-theta-alpha-beta-gamma>]

- คลื่นสมองระดับธีต้า Theta Brainwave (ความถี่ระหว่าง 4 – 7.9 Hz)

คลื่นสมองช่วงนี้จะเกิดเมื่อเรามีการผ่อนคลายระดับลึก เกิดขึ้นเมื่อนอนหลับฝัน มีการดึงข้อมูลจากจิตให้สำนึกได้ เกิดการแก้ไขปัญหาโดยไม่รู้ตัว ดังรูปที่ 2.4

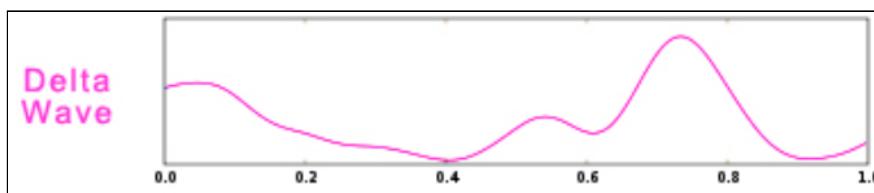


รูปที่ 2.4 Theta Wave

[ที่มา: <https://itsusync.com/different-types-of-brain-waves-delta-theta-alpha-beta-gamma>]

- คลื่นสมองระดับเดลต้า Delta Brainwave (ความถี่ระหว่าง 0.1 – 3.9 Hz)

เป็นคลื่นสมองที่ช้าที่สุด เกิดขึ้นในขณะนอนหลับ สมองทำงานตามความจำเป็น เท่านั้น แต่กระบวนการของจิตใต้สำนึกจะจัดและเก็บข้อมูลอย่างต่อเนื่อง เป็นช่วงที่ร่างกาย กำลังพักผ่อนอย่างเต็มที่ หลับลึกโดยไม่มีความฝัน จะรู้สึกสดชื่นเป็นพิเศษเมื่อ yan ตื่น ดัง รูปที่ 2.5



รูปที่ 2.5 DeltaWave

[ที่มา: <https://itsusync.com/different-types-of-brain-waves-delta-theta-alpha-beta-gamma>]

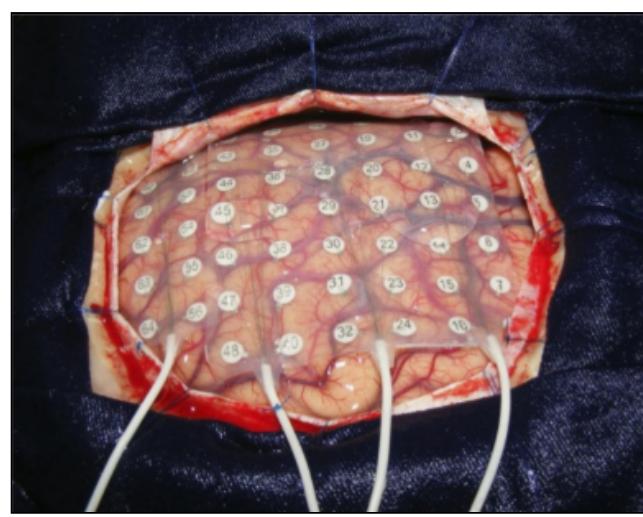
2.2.3 Brain-Computer Interface(BCI)

Brain Computer Interface(BCI) [11] บางทีก็เรียกว่า Mind-Machine Interface (MMI), Direct Neural Interface(DNI), Synthetic Telepathy Interface(STI) หรือ Brain Machine Interface(BMI) ซึ่งทั้งหมดนี้คือวิธีการที่จะสามารถทำให้สมองกับอุปกรณ์ภายนอกสามารถสื่อสารกันได้โดยตรง เทคโนโลยีBCI ถูกนำมาใช้อย่างแพร่หลายในการช่วยเหลือผู้ป่วยที่มีความพิດ พลາดของระบบประสาทสัมผัสและการเคลื่อนไหว สมองของมนุษย์ประกอบด้วยเซลล์ประสาท มากมาย แต่ละเซลล์ประสาทได้ถูกเชื่อมต่อกันโดยประสาทน้ำเข้า(Dendrite) และเกนประสาทน้ำออก(Axon) ทุกๆการกระทำการของมนุษย์ เช่นการคิด การเคลื่อนไหว ความรู้สึก หรือการระลึกถึงบางสิ่งบางอย่าง ล้วนทำให้เกิดการทำงานของเซลล์ประสาท การทำงานดังกล่าวจะส่งคลื่นไฟฟ้าจากเซลล์ประสาทหนึ่งไปยังอีกเซลล์ประสาทหนึ่งด้วยความเร็วถึง250 ไมล์ต่อชั่วโมง สัญญาณดังกล่าว เกิดขึ้นจากการที่เยื่อหุ้มเซลล์ของแต่ละเซลล์ประสาทมีศักย์ไฟฟ้าที่ต่างกัน สัญญาณที่ตรวจจับได้นั้นสามารถตีความได้ว่ามีความหมายว่าอย่างไร และเราสามารถใช้มันเพื่อสร้างอุปกรณ์ไว้แก้ปัญหาบางอย่างได้อย่างไร ดังนั้น BCI คือระบบที่ไวติดต่อสื่อสาร โดยตรงระหว่างสมองของมนุษย์ กับคอมพิวเตอร์ เพื่อที่จะพัฒนาระบบBCIที่มีประสิทธิภาพ เราควรศึกษาเทคนิคที่มีความเป็นไป

ได้และเหมาะสมกับปัญหาที่เราจะแก้ไข เมื่อแบ่งเทคโนโลยีการทำแบบกว้างๆแล้ว BCI จะถูกแบ่งออกเป็น 2 เทคนิคคือ Invasive BCI และ Non Invasive BCI

● INVASIVE BCI

Invasive BCI [12] คือ วิธีการเชื่อมต่อระหว่างเซลล์ประสาทกับคอมพิวเตอร์ซึ่งมีข้อไฟฟ้าเป็นแพงสำหรับผู้อยู่ภายใต้ในสมองระหว่างการผ่าตัดและจะถูกผ่าไว้ที่นั่นแบบถาวร ระบบของอุปกรณ์ที่พัฒนาจากเทคนิคแบบ Invasive BCI จะให้คุณภาพและความแม่นยำที่สูงมากเมื่อเทียบกับเทคนิคแบบ Non invasive BCI เพราะสัญญาณที่ได้นั้นมาจากสมองเนื้อเทา(Grey Matter)โดยตรง ที่จะประกอบด้วยเซลล์ประสาท(Neuron), Dendrite และ Axon แต่น่าเสียดายที่ Invasive BCI นั้นมีค่าใช้จ่ายในการดำเนินการและความซับซ้อนในการผ่าตัดที่สูงมาก เช่นเดียวกับ Electrocorticography (ECoG) ก็อหันน์ใน Invasive BCI มันคืออุปกรณ์ที่ฝังลงไปในกระโหลกศีรษะแค่บางส่วนเท่านั้น แต่ส่วนที่เหลือจะอยู่ภายนอกสมองไม่ได้เชื่อมต่อกับสมองเนื้อเทา ECoG เป็นวิธีที่มีแนวโน้มที่สูงมากที่จะเป็นที่นิยมอย่างแพร่หลายในเทคนิคแบบ Invasive BCI เพราะมีรายละเอียดเชิงพื้นที่ที่สูงมาก, สัญญาณรบกวนต่ำ, ให้ความถี่ในช่วงที่กว้างกว่า, มีความเสี่ยงทางการแพทย์น้อยกว่า, มีความซับซ้อนในการดำเนินการที่น้อยกว่า อีกทั้งยังมีความเป็นไปได้ที่จะมีความมั่นคงและเสถียรภาพในระยะยาวที่สูงกว่าอีกด้วย คุณสมบัติที่กล่าวมาทั้งหมดนี้เราจะเห็นได้ว่า Invasive BCI จะสามารถช่วยเหลือผู้ที่มีร่างกายพิการได้ แต่ก็ยังน่าเสียดายที่ ECoG ก็มีค่าใช้จ่ายที่ยังจำเป็นต้องสูงอยู่ดีในปัจจุบัน ดังรูปที่ 2.6



รูปที่ 2.6 Invasive BCI
[ที่มา: <https://mnepilepsy.org/>]

● NON INVASIVE BCI

Non invasive BCI [13] เป็นเทคนิคที่ได้ความนิยมสูงที่สุด ที่ Electrode จำเป็นที่จะต้องถูกวางอยู่บนหนังศีรษะ คือออกกระโหลกนั่นเอง วิธีการทำแบบ Non invasive BCI มีข้อจำกัดคือมักจะมีสัญญาณรบกวนสูง, มีความละเอียดของสัญญาณที่ต่ำ โดยขึ้นอยู่กับระยะทางระหว่าง Electrode กับสมอง นั่นทำให้มีปัญหาในการบันทึกผลการทำงานด้านในของสมอง อย่างไรก็ตาม Non invasive BCI มีข้อดีที่ต้นทุนด้านค่าใช้จ่ายต่ำกว่า Invasive BCI มาก และอุปกรณ์ยังสามารถพกพาได้สะดวก อีกทั้งยังไม่จำเป็นต้องมีการผ่าตัดใดๆเกิดขึ้น เทคนิค Non invasive BCI ส่วนใหญ่จะใช้คลื่นไฟฟ้าสมองหรือ EEG ซึ่งทำให้เราสามารถเห็นภาพกิจกรรมของสมองได้ ดังรูปที่ 2.7



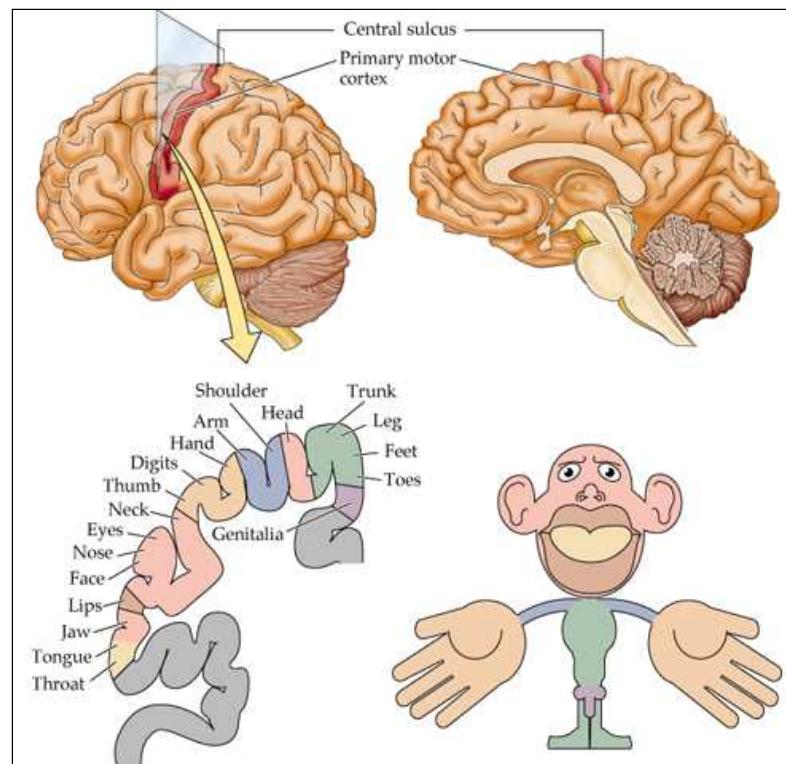
รูปที่ 2.7 Non Invasive BCI

[ที่มา: <https://hearing4all.eu>]

2.2.4 Motor Imagery

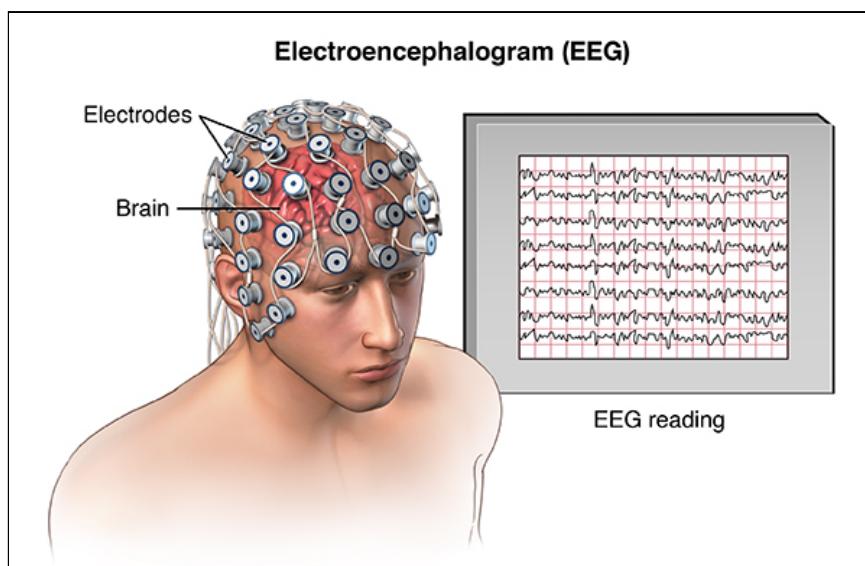
Motor Imagery คือ Mental Imagey [14] คือ สมรรถภาพในการจินตนาการวัตถุหรือเหตุการณ์ที่ไม่ได้เกิดขึ้นหรืออยู่ ๆ ณ ที่ตรงนั้น เป็นอีกหนึ่งความสามารถในการรับรู้ (cognitive) ของมนุษย์ เราสามารถจินตนาการว่าเราจะกำลังขึ้นเครื่องบินอยู่ เราเป็นนักฟุตบอลมืออาชีพ หรือเราเป็นศิลปินร็อก เราสามารถจินตนาการว่าวัตถุอีกด้านจะหน้าตาเป็นอย่างไรขณะที่เรามองจากอีกด้าน เราสามารถจินตนาการความรู้สึกเมื่อเรากำลังหยับวัตถุขึ้นหนึ่งได้ เราสามารถอธิบายถึงรูปร่างและคุณลักษณะของมันได้แม่ไม่ต้องสัมผัสมันก่อน ลิ่งที่กล่าวมาเหล่านี้ทั้งหมดคือ Motor Imagery ในแบบต่างๆ สามารถนิยามได้ว่าเป็นกระบวนการรับรู้แอบแฝงของการจินตนาการว่าเคลื่อนไหว อย่างไร โดยไม่ได้ขับอวัยวะนั้นจริงกระบวนการฟิกซ์้อมหรือจำลองการกระทำอย่างหนึ่ง ซึ่งเกิด

ขึ้นในความคิด ซึ่งมีการนำกระบวนการนี้ไปใช้อย่างกว้างขวางในวงการกีฬา, การฟื้นฟูระบบประสาท และการค้นคว้าวิจัยทางด้าน ประสาทวิทยาและจิตวิทยาที่เกี่ยวกับการรับรู้



รูปที่ 2.8 บริเวณที่มีกิจกรรมทางสมองจากการเคลื่อนไหวร่างกายส่วนต่างๆ
[ที่มา: https://operativeneurosurgery.com/doku.php?id=primary_motor_area]

2.2.5 Electroencephalography (EEG)



รูปที่ 2.9 Electroencephalography (EEG)
[ที่มา: <https://hvmn.com/biohacker-guide/cognition/eeg-measures-of-cognition>]

EEG [14] คือการบันทึกคลื่นไฟฟ้าสมอง เป็นคลื่นไฟฟ้าสมองแบบแรกที่ใช้เทคนิคแบบ Non invasive จะทำให้เห็นภาพกิจกรรมทางไฟฟ้าของสมองได้ สัญญาณ EEG ถูกตรวจจับได้จากกะโหลกศีรษะ ซึ่งจะมีสัญญาณรบกวนหรือNoise ซึ่งเป็นผลมาจากการถูกแทรกแซงทางไฟฟ้าและการขับของ Electrode การใช้ Electrode จำนวนหลาย Channel อาจจะทำให้เกิดNoiseและเกิดความช้าช้อนของสัญญาณได้ ซึ่งอาจจะทำให้ประสาทวิภาคในการทำงานของ BCI ลดลงด้วย และส่งผลทำให้ต้องใช้เวลาในการเตรียมตัวที่นานขึ้นในการทำBCI ดังนั้นการเลือกจำนวน Channel ที่น้อยที่สุดหรือเหมาะสมที่สุดจะลดผลกระทบที่สูดเข่นกันทั้งด้านของประสาทวิภาคและความสะดวกสบายในการดำเนินการ เนื่องจากความง่ายในการใช้งาน, ต้นทุนที่ต่ำและให้ความละเอียดเชิงเวลาที่สูง EEG จึงเป็นวิธีที่เป็นนิยมอย่างแพร่หลายของ BCI ในปัจจุบัน จุดเด่นของ EEG มีดังนี้

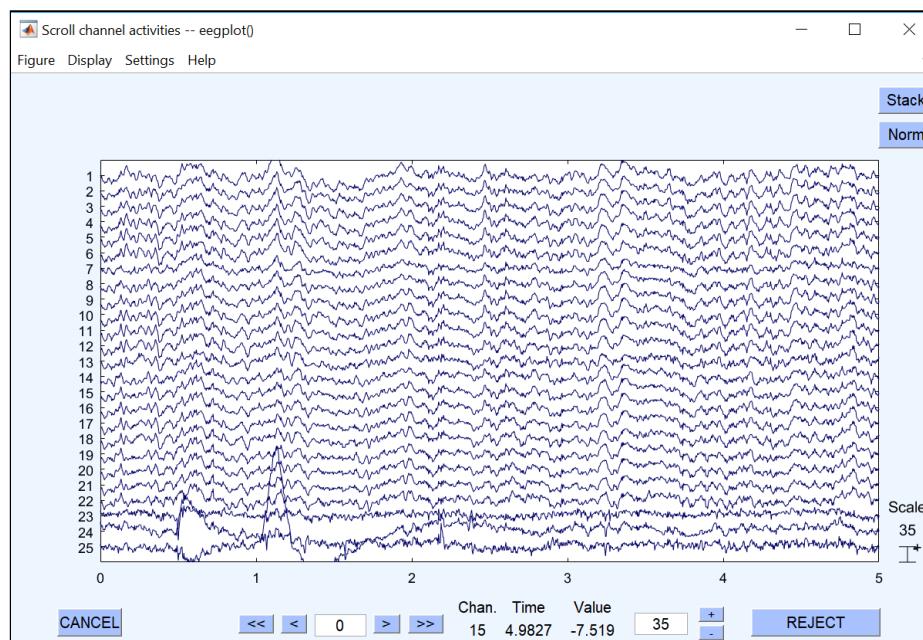
- ค่าใช้จ่ายด้านハードแวร์ต่ำกว่าเทคนิคอื่น ๆ อย่างมาก
- EEG Sensor มีข้อจำกัดในการใช้งานน้อยกว่าวิธีการแบบ fMRI, SPECT, PET, MRS, หรือ MEG เป็นต้น ยกตัวอย่างเช่น MEG ต้องใช้ความคู่กับอุปกรณ์ที่ประกอบด้วยเครื่องตรวจจับแม่เหล็กที่สามารถตรวจจับได้ในระดับความแม่นยำ 20,000Hz หรือมากกว่า ถ้าต้องการ ยกตัวอย่าง เช่นวิธีแบบ MEG และ EROS
- EEG ให้ความละเอียดเชิงเวลาที่สูงมาก ในหน่วยมิลลิวินาที EEG โดยทั่วไปจะถูกบันทึกผลที่ Sampling rate ระหว่าง 250-2000 Hz ในการตรวจจับโดยใช้เซ็นเซอร์ที่ติดต่อผิวหนัง แต่ในยุคหลังๆ นี้ EEG สามารถบันทึกได้ที่ Sampling rate ถึง 20,000Hz หรือมากกว่า ถ้าต้องการ ยกตัวอย่าง เช่นวิธีแบบ MEG และ EROS
- EEG ไม่ทำให้เกิดเสียงรบกวน ทำให้การรับรู้และตอบสนองต่อสิ่งเร้าดีขึ้น
- EEG ไม่ทำให้เกิดอาการอืดอัด ซึ่งแตกต่างจาก fMRI, SPECT, PET, MRS และบางครั้ง MEG ก็ยังทำให้รู้สึกอืดอัดได้
- EEG จะไม่ถูกบันทึกจากคลื่นสมานแม่เหล็ก ($> 1\text{ Tesla}$) ต่างจากเทคนิคอื่นๆ โดยเฉพาะ กับ MRI และ MRS ผลกระทบจากสมานแม่เหล็กเหล่านี้อาจก่อให้เกิดปัญหาที่ไม่พึงประสงค์มากmany ยกตัวอย่างเช่น ใช้วิธีเหล่านี้กับกลุ่มทดสอบที่มีการปลูกฝังโลหะในร่างกาย เช่นเครื่องกระตุนศีรษะ ใจที่มีส่วนประกอบเป็นโลหะ เป็นต้น
- Non invasive BCI เป็นเทคนิคซึ่งแตกต่างจาก ECoG โดยสิ้นเชิง เพราะจริงๆแล้ว ECoG นั้นต้องนำ Electrode ไปวางบนพื้นผิวของสมองโดยตรง
- EEG สามารถตรวจจับกระบวนการเรียนรู้ (Covert processing) ของสมองได้ ซึ่งก็คือกระบวนการกระบวนการที่ไม่ต้องการการตอบสนองทางร่างกาย เช่นการคิด, การจินตนาการ
- EEG สามารถใช้ในกลุ่มทดลองที่อ่อนเพลีย ไม่สามารถตอบสนองได้

- EEG เป็นวิธีการที่มีประสิทธิภาพสูงมากในการติดตามความเปลี่ยนแปลงของสมองในช่วงเวลาต่างๆ การวิเคราะห์สัญญาณ EEG ขณะนอนหลับ สามารถบ่งบอกถึงลักษณะสำคัญของการพัฒนาสมองซึ่งขึ้นอยู่กับเวลา รวมถึงการประเมินผลการเจริญเติบโตของสมองในวัยรุ่น
- การทำเข้าใจสัญญาณ EEG สามารถทำได้ดีกว่าเมื่อเทียบกับเทคนิคอื่นๆ ที่ใช้ในงานวิจัยต่างๆ เช่น ค่า BOLD response ในวิธี MRI

2.2.6 Data Pre-Processing

คณะผู้จัดทำได้ทำการดาวน์โหลดข้อมูลสาขาวาระจากอินเทอร์เน็ต [15] ซึ่งข้อมูลดิบเหล่านี้คือข้อมูล EEG ที่เก็บระหว่างผู้เข้าร่วมการทดลองกำลังทำกิจกรรมเพื่อเก็บข้อมูลเกี่ยวกับ motor imagery โดยแยกประเภทข้อมูลเป็น 4 อย่าง ได้แก่ 1. การจินตนาการว่าขับมือซ้าย 2. มือขวา 3. ขาทึ้งสองข้างพร้อมกัน 4. แลบลิ้น

กราฟที่ได้จากการพล็อตข้อมูล EEG จากอิเล็กโทรดแต่ละตัวที่มีการจัดเรียงแบบ 10-20 system โดยที่แกนในแนวนอนคือเวลาและแกนในแนวตั้งคือความต่างศักย์มีลักษณะดังรูป 2.10

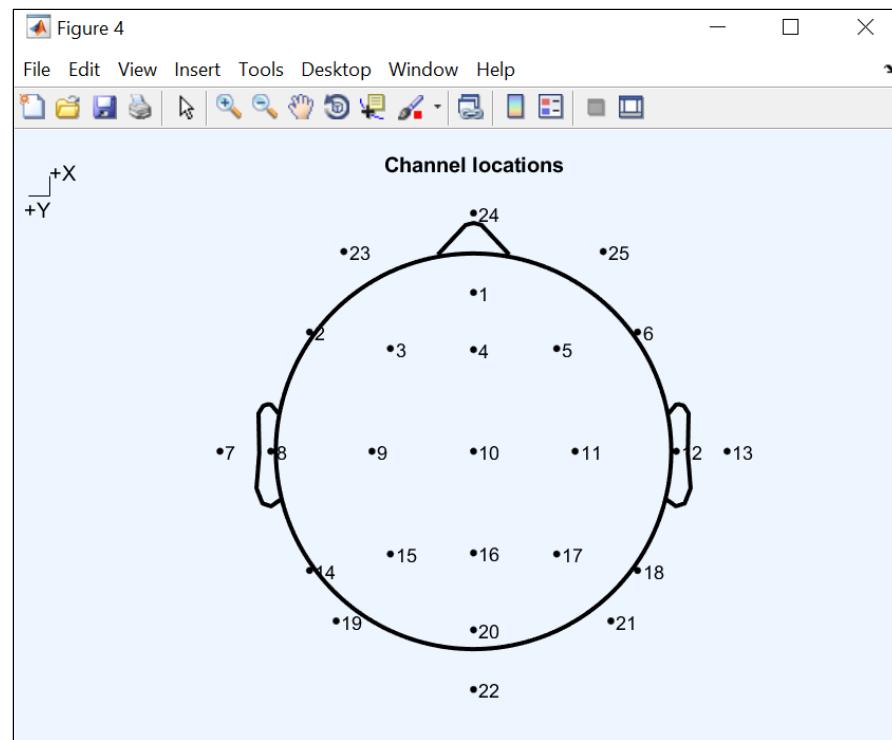


รูปที่ 2.10 แสดงข้อมูลดิบของสัญญาณ EEG ตาม Time domain

● Channel Location

คณะผู้จัดทำต้องการพล็อตตำแหน่งของอิเล็กโทรดลงไปบนภาพจำลองส่วนศีรษะในระบบสองมิติ ให้สัมพันธ์การจัดวางอิเล็กโทรดแบบ 10-20 system ตามที่ผู้เก็บข้อมูล

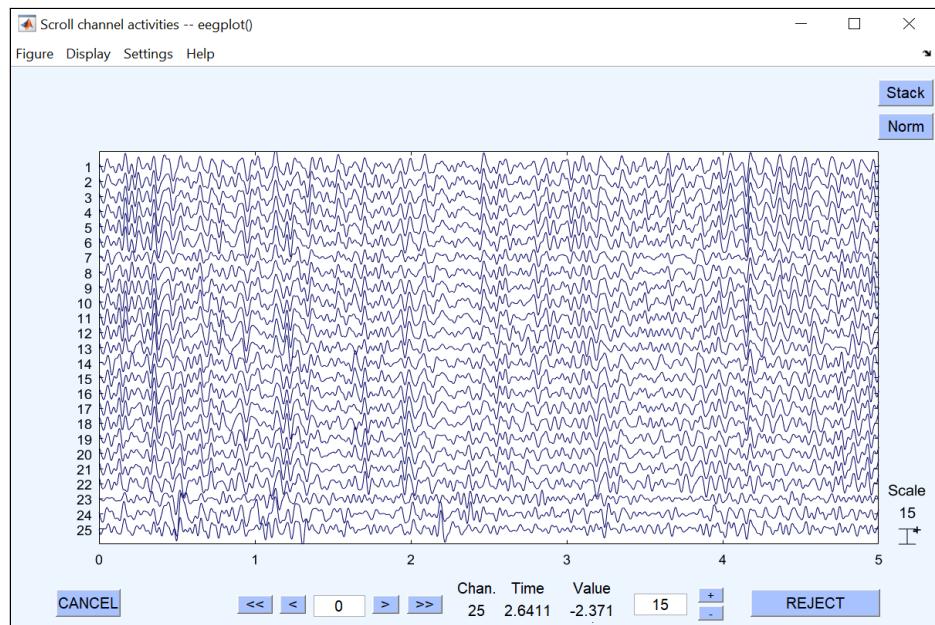
สาหรณะได้ใช้ คณะผู้จัดทำได้ทำการกำหนดตำแหน่งของอิเล็กโทรดต่างๆดังแสดงในรูปที่ 2.11



รูปที่ 2.11 Channel location

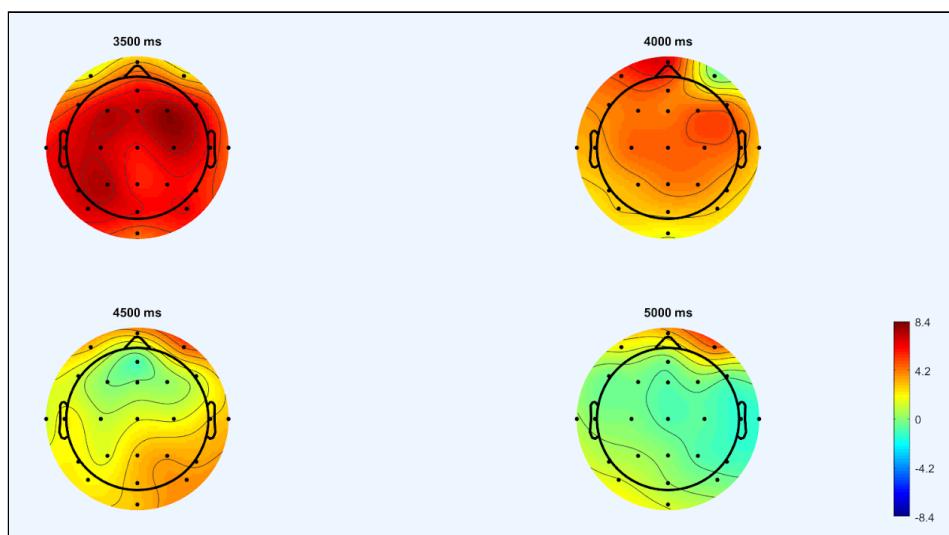
● Frequency Filter

คณะผู้จัดทำได้ทำการกรองความถี่ที่ไม่ต้องการผ่านออกໄไป ซึ่งเนื่องจากว่าช่วง 8 - 30 Hz เป็นช่วงความถี่ของ SMR (Sensorimotor Rhythm) ซึ่งเป็นคลื่นที่เกิดขึ้นระหว่าง motor imagery คณะผู้จัดทำจึงทำการกรองความถี่ต่ำผ่านโดยที่ cut off frequency ของความถี่เป็น 30Hz และ กรองความถี่สูงผ่านโดยที่ cut off frequency มีค่าเท่ากับ 8 Hz ความถี่ที่ถูกตัดออกแสดงในภาพ 2.12

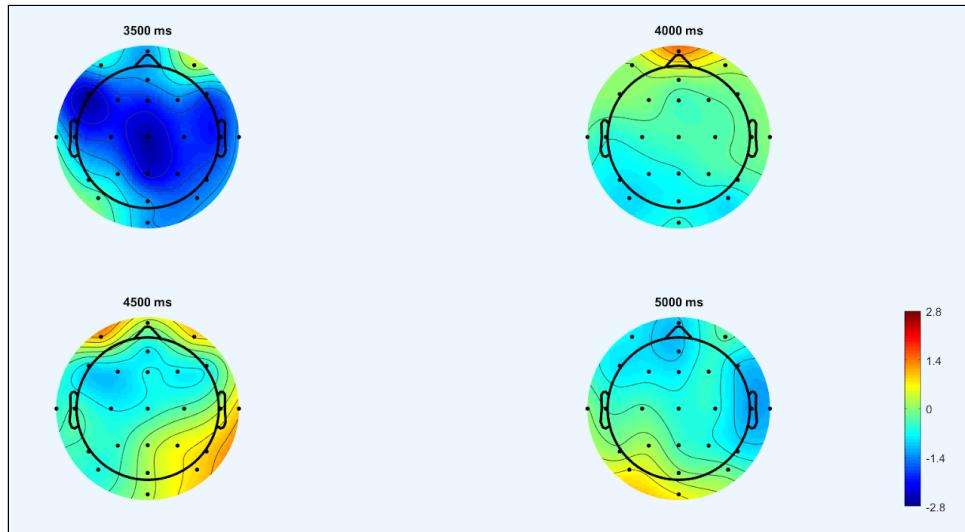


รูปที่ 2.12 Bandpass filter ในช่วงความถี่ 8-30Hz

รูปแสดงผลลัพธ์งานสมองก่อน filter แสดงในรูปที่ 2.13 และหลัง filter แสดงในรูปที่ 2.14



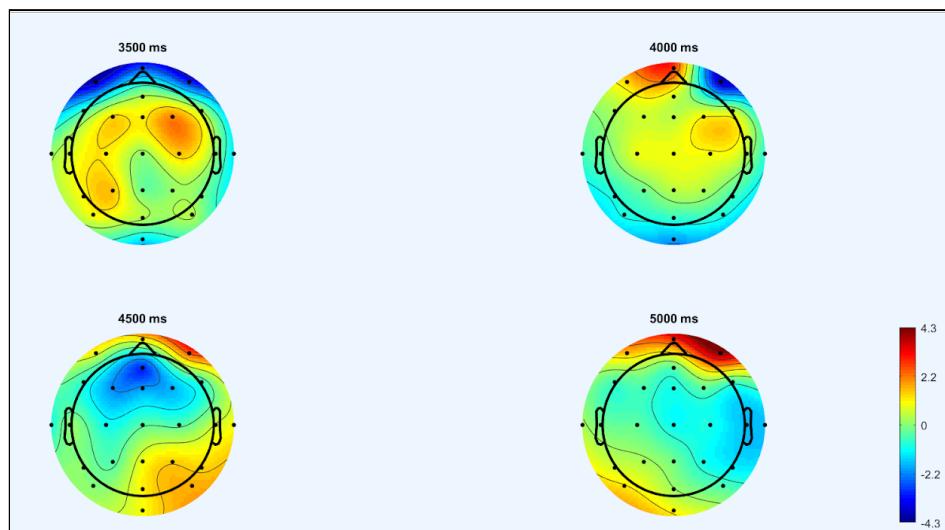
รูปที่ 2.13 คลื่นEEG แสดงบนกะโหลกก่อนทำการ filter



รูปที่ 2.14 คลื่นEEG แสดงบนกะโหลกหลังทำการ filter

● Re-Referencing

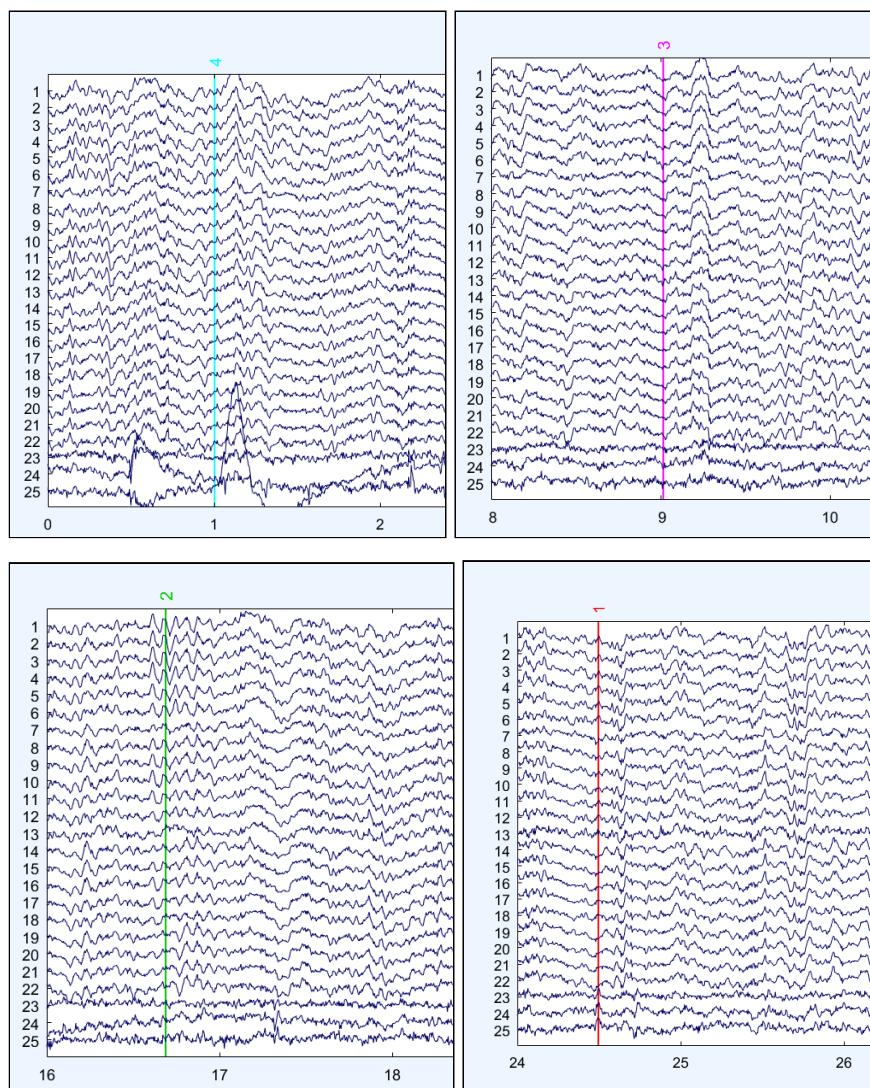
ใน EEG ศักยไฟฟ้าที่ถูกบันทึกในแต่ละอิเล็กโทรดนั้นมีความสัมพันธ์กับศักยไฟฟ้าของอิเล็กโทรดอื่น การเลือกตำแหน่งอ้างอิงจึงเป็นประเด็นสำคัญที่ต้องนำมาพิจารณา ในบริบทนี้ คณะผู้จัดทำได้ใช้ค่าเฉลี่ยเป็นค่าอ้างอิงเป็นเบื้องต้น เนื่องจากเป็นวิธีที่สามารถทำได้ง่าย รูปแสดงผลลัพธ์งานสมองหลัง Re-reference แสดงในรูปที่ 2.15 ในช่วงเวลาเดียวกัน



รูปที่ 2.15 พลังงานสมองหลังทำการ Re-reference

● Add Event

ข้อมูลสาระณะที่คณะผู้จัดทำได้ดาวน์โหลดมาใช้นั้น จะมีรายละเอียดที่บันทึกไว้ในช่วงเวลาหนึ่งๆของข้อมูลนั้น เป็นข้อมูลที่เก็บระหว่างที่ผู้เข้าร่วมการทดลองกำลังจินตนาการว่าขับส่วนใดของร่างกายอยู่ (มือซ้าย, มือขวา, ขาทิ้งสองข้าง, และลิ้น) ซึ่งคณะผู้จัดทำได้ทำเครื่องหมายไว้ในกราฟว่าผู้เข้าร่วมการทดลองได้เริ่มเข้าสู่กิจกรรมนั้นที่เวลาใด โดยแบ่งเป็นเลข 1,2,3,4 แทนการจินตนาการว่าขับมือซ้าย, ขวา, ขาทิ้งสองข้างพร้อมกัน และการแลบลิ้นตามลำดับดังแสดงในรูปภาพ 2.16 เพื่อให้ง่ายแก่การสังเกตุพฤติกรรมของสมองเมื่อจินตนาการตามกิจกรรมต่างๆตามที่ได้กำหนดไว้

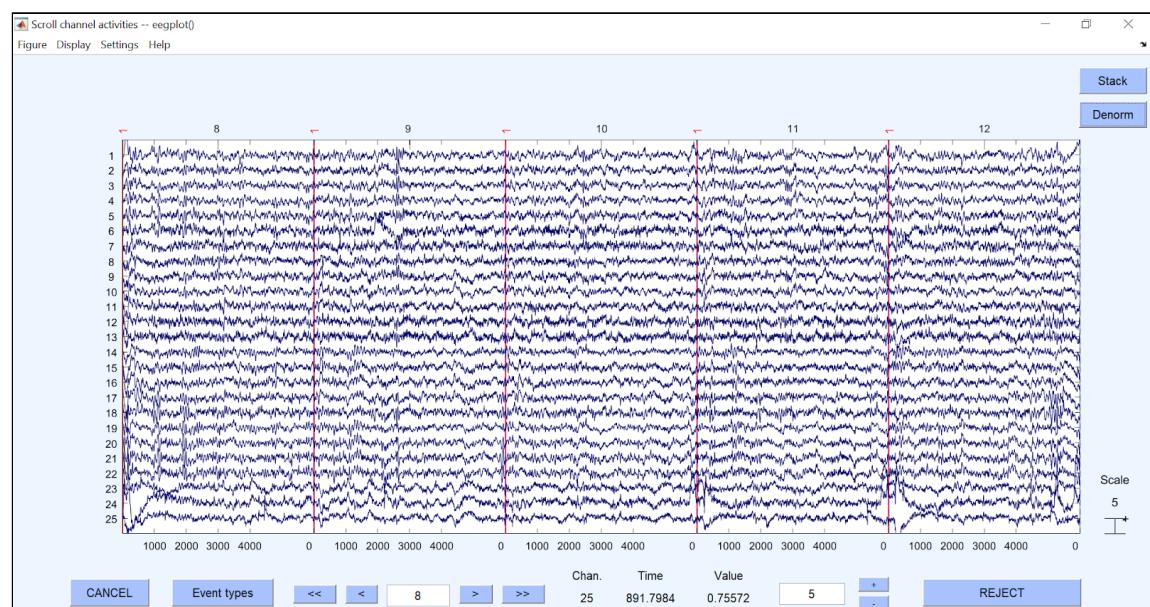


รูปที่ 2.16 การเพิ่ม Event ลงในชุดข้อมูล

● Extract Epoch

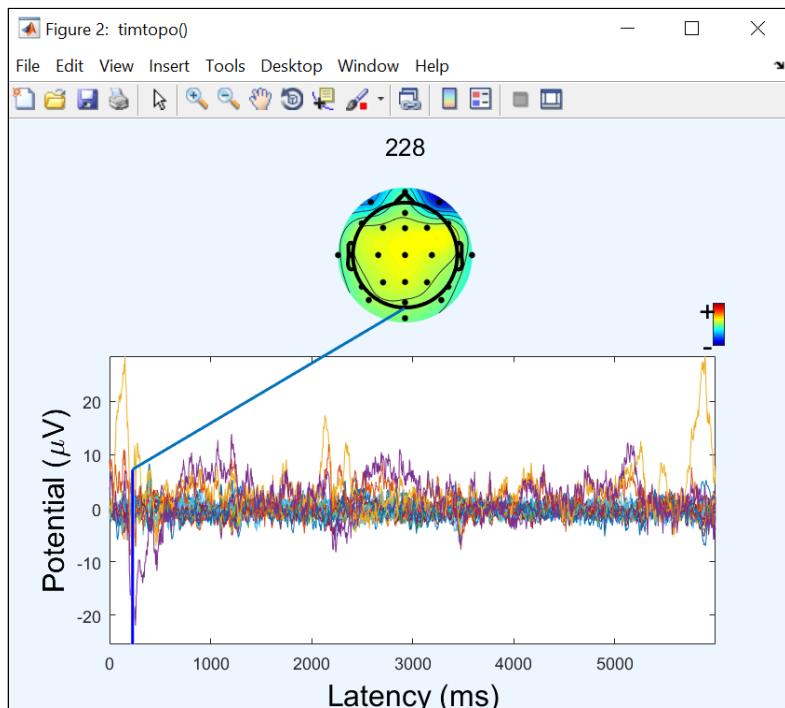
การทำ Epoch ในบริบทของการประมวลผลสัญญาณ คือ กระบวนการหนึ่งซึ่งจะดึงเอาสัญญาณออกมาจากสัญญาณแบบต่อเนื่อง (continuous signal) ในระยะเวลาที่เจาะจง ออกมานเป็นส่วนๆ เปรียบเสมือนว่านำข้อมูลมาแยกส่วนเก็บไว้เป็นกล่องๆ ซึ่ง epoch ก็เปรียบเสมือนกล่อง กล่องหนึ่งนั้นเอง โดยที่การดึงเอาสัญญาณออกมานี้จะสัมพันธ์กับเหตุการณ์ (event) ของข้อมูล

ในทำนองเดียวกันนี้ เราได้ดึงเอาสัญญาณออกมารอโดยที่เหตุการณ์ที่สัมพันธ์คือประเภทของการจินตนาการทั้งสี่ที่ได้กล่าวไว้ข้างต้น ซึ่งระยะเวลาที่จะดึงออกมามีผู้จัดทำได้กำหนดให้เริ่มจากวินาทีที่ 3.5 หลังจากที่เหตุการณ์เริ่ม จนถึงวินาทีที่ 5 หลังจากเหตุการณ์เริ่ม สืบเนื่องมาจากกระบวนการทดลองเพื่อจัดเก็บข้อมูลของเข้าของชุดข้อมูล สาธารณณะได้กำหนดให้ผู้เข้าร่วมการทดลอง เริ่มจินตนาการตั้งแต่วินาทีที่ 3 ถึงวินาทีที่ 6 หลังจากเหตุการณ์เริ่ม ตัวอย่างการทำ epoch ที่สัมพันธ์กับการจินตนาการว่าขับมือซ้ายแสดงในภาพ 2.17



รูปที่ 2.17 การตัด Epoch ของคลาสที่สนใจ

หลังจากที่ได้ดึงเอาสัญญาณต่อเนื่องมาแบ่งไว้ใน epoch แล้ว คณบัญชีจัดทำได้นำ epoch ที่อ้างถึงเหตุการณ์ประเภทเดียวกันมาหาค่าเฉลี่ยแล้วแสดงออกมานเป็น epoch เดียว (averages ERP) ซึ่งเป็นวิธีการทางคณิตศาสตร์ที่ใช้ในการสังเกตพฤติกรรมของสัญญาณ ตัวอย่างหลังงานสมองที่จำลองบนเครื่องแสดงดังภาพ 2.18

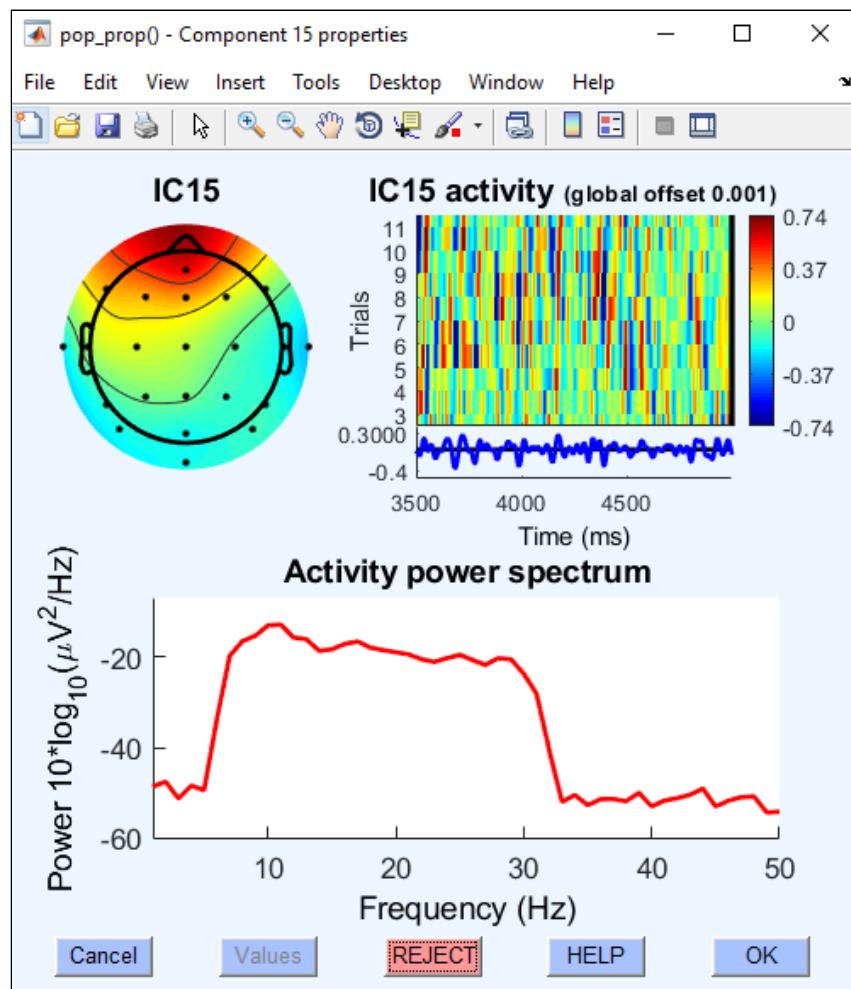


รูปที่ 2.18 แสดงค่า ERP เนลลี่จาก event ประเภทเดียวกัน

● Run ICA

Independent Component Analysis เป็นหนึ่งในวิธีการประมวลผลสัญญาณเพื่อแยกแยะแหล่งที่มาของสัญญาณโดยการวิเคราะห์สัญญาณซึ่งทับซ้อนผสมกันอยู่ที่ตัวรับสัญญาณ ในที่นี้คณะผู้จัดทำต้องการหาแหล่งที่มาของ artifact ต่างๆที่ซ่อนอยู่ในข้อมูลแล้วทำการ reject ทั้ง แสดงดังรูปที่ 2.19 และ 2.20

○ Eye blinking



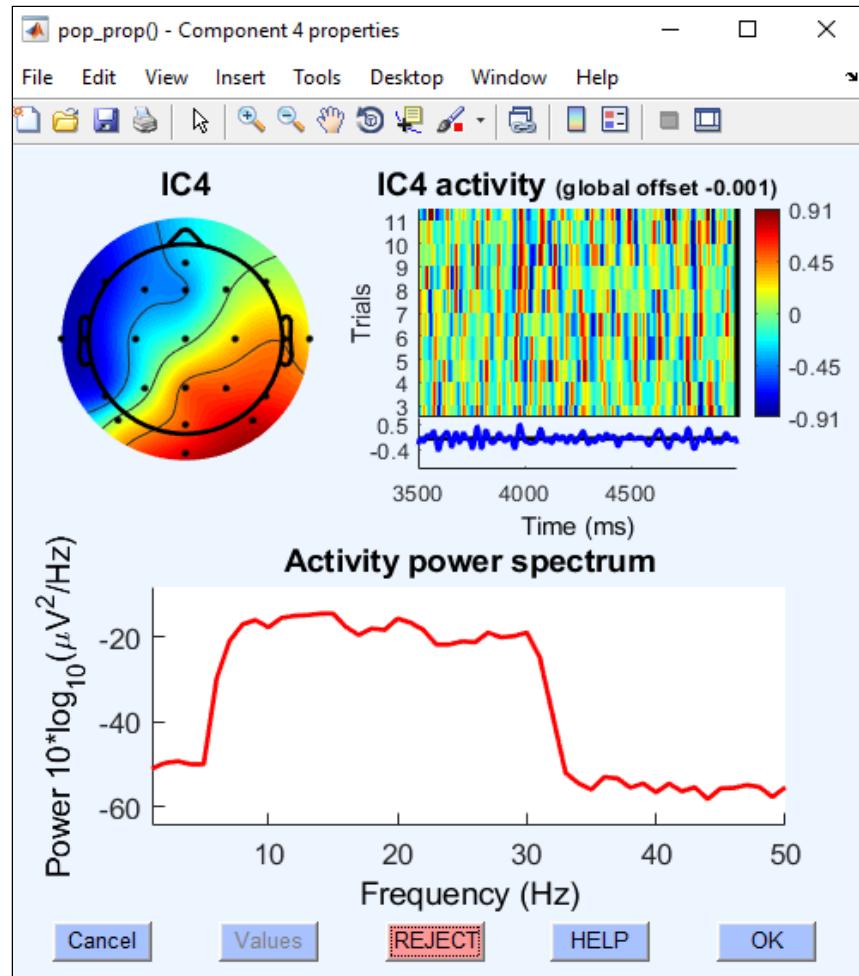
รูปที่ 2.19 Eye blinking artifact

ลักษณะเช่นนี้ถือว่าเป็น artifact ของการกระพริบตาด้วยเหตุผลหลัก 2 ข้อ

คือ

1. การค่อยๆลดลงของ EEG Spectrum จากบนลงล่าง
2. ความเข้มข้นของ EEG Spectrum บริเวณหน้าผาก

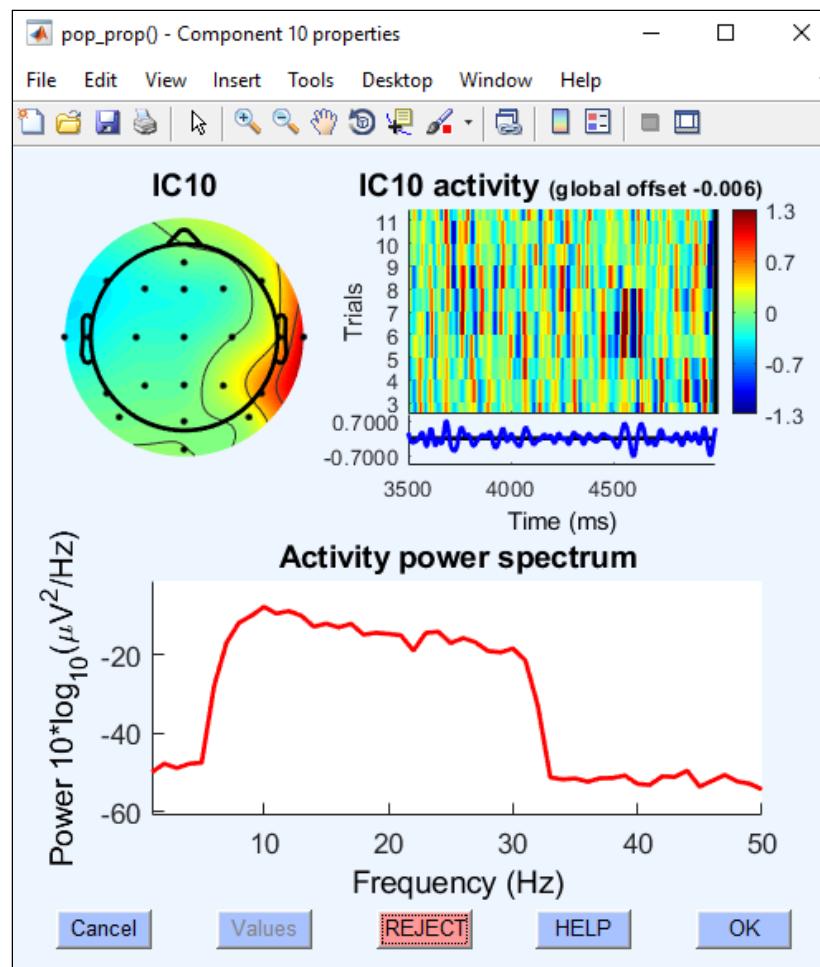
- Eyeball movement



รูปที่ 2.20 Eyeball movement artifact

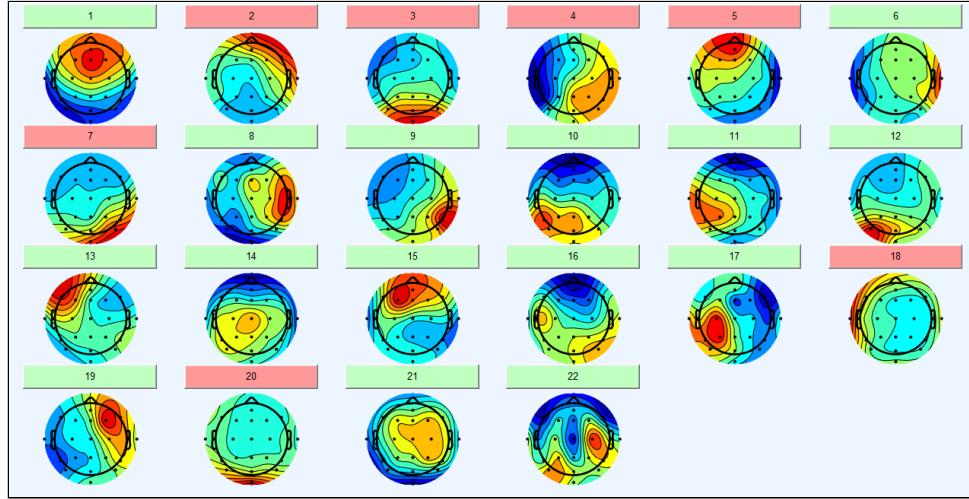
ลักษณะของ artifact สำหรับการกรอกลูกตามีคุณสมบัติสำคัญคือ EEG Spectrum จะแบ่งแยกความเข้มข้น(สีแดงและสีน้ำเงิน)ชัดเจน ออกเป็น 2 ฝั่งที่ตรงข้ามกัน สีบานี่องมาจากการที่ธรรมชาติของลูกตาทั้ง 2 ข้างจะกลอกไปในทิศทางเดียวกัน สมองก็จะทำงานสลับข้างกันนั่นเอง ดังรูปที่ 2.21

- Muscle artifact



รูปที่ 2.21 Muscle artifact

การเคลื่อนไหวกล้ามเนื้อจะแสดง EEG Spectrum ที่มีความเข้มข้นสูงเป็นแค่พื้นที่เล็กๆ ที่หนึ่งบริเวณเด็นของกะโหลก ตัวอย่างการเลือก Accept หรือ Reject ในการ Run ICA เป็นดังรูปที่ 2.22



รูปที่ 2.22 การเลือก Accept หรือ Reject ข้อมูลICA

2.2.7 Feature Extraction: Common Spatial Pattern (CSP)

แนวคิดพื้นฐานของ common spatial pattern คือการรวมกันของ spatial channel ซึ่งได้มีการหาค่าเฉลี่ยความแปรปรวนของ 2 class ที่มีค่าสูงที่สุดไว้อธิบายได้ดังนี้

ให้สัญญาณ $X \in \mathbb{R}^{N_c \times N_s}$ สามารถคำนวณ covariance matrix ของสัญญาณได้จาก

$$C = \frac{1}{N_s - 1} (X X^T) \quad (2.1)$$

โดยที่ N_c หมายถึงจำนวน channels

N_s หมายถึงค่า sampling rate

X^T หมายถึง transpose ของ matrix X

ต่อไปเป็นสมการการคำนวณหาค่า Average covariance Matrix ของ Class j ที่เกิดขึ้นใน trial

$$\bar{C}_j = \frac{1}{N_j} \sum_{k=1}^{N_j} C_j^{(k)} \quad (2.2)$$

โดยที่ $C_j^{(k)}$ หมายถึง Covariance matrix เมื่อ Class j ปรากฏ

N_j หมายถึงจำนวนครั้งที่ Class j ปรากฏขึ้น

เป้าหมายของการทำ CSP คือการหาค่า Spatial filter $\omega \in \mathbb{R}^{N_c}$ ซึ่งคือค่า Rayleigh quotient สูงสุด

$$J(\mathbf{w}) = \frac{\mathbf{w}^T \bar{\mathbf{C}}_1 \mathbf{w}}{\mathbf{w}^T \bar{\mathbf{C}}_2 \mathbf{w}}. \quad (2.3)$$

สมการนี้ได้มาจากการแก้ปัญหา Eigenvalue decomposition (GEVD)

$$\bar{\mathbf{C}}_1 \mathbf{U} = \Sigma \bar{\mathbf{C}}_2 \mathbf{U} \quad (2.4)$$

โดยที่ Σ คือ diagonal matrix ที่ประกอบด้วยค่า eigenvalue เรียงจากมากไปน้อย

\mathbf{U} ประกอบด้วย column ที่ระบุถึง eigenvectors

การจับคู่ค่า eigenvalues ที่มีค่าสูงที่สุด และต่ำที่สุด จะสามารถทำให้หาค่า eigenvectors ได้ซึ่งจะทำให้ได้เซทของ spatial filter ออกรมา

สุดท้าย feature f_1 สามารถคำนวณได้จากการหา Logarithm ของ Spatial filter และการ normalize variance ดังสมการ

$$f_l = \log \left(\frac{\mathbf{w}_l^T \mathbf{X} \mathbf{X}^T \mathbf{w}_l}{\sum_k \mathbf{w}_k^T \mathbf{X} \mathbf{X}^T \mathbf{w}_k} \right) \quad (2.5)$$

วิธีการที่จะต้องมี CSP เพื่อแก้ปัญหาที่เป็น multiclass นั้นสามารถทำได้ด้วยการดำเนินการกับทั้ง 2 คลาส เป็นทุกๆ combination ที่เป็นไปได้ ดังนั้นในโครงการนี้ซึ่งมี 4 class ผลลัพธ์ของการ combination จะเป็นดังนี้ $\{(1,2),(1,3),(1,4),(2,3),(2,4),(3,4)\}$ [3]

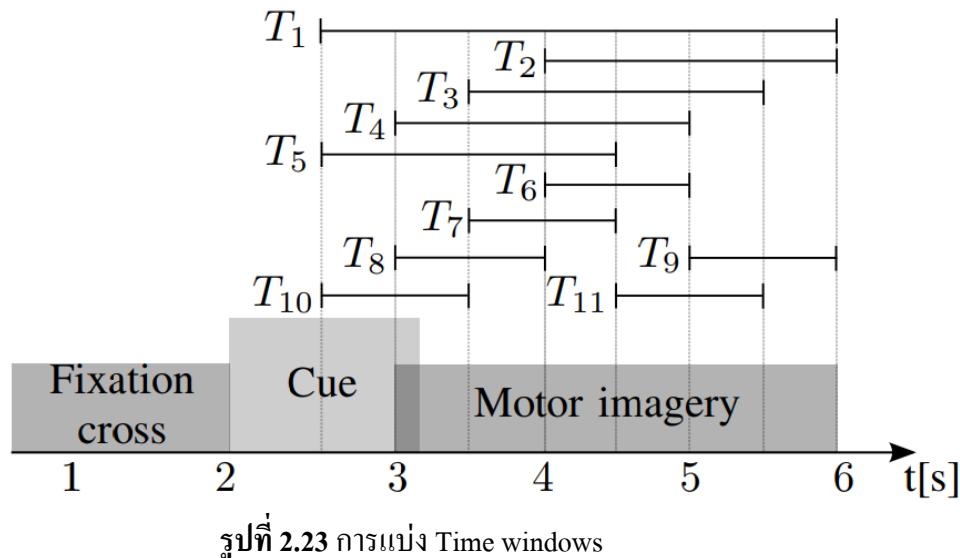
2.2.8 Feature Extraction: Filter Bank Common Spatial Pattern (FBCSP)

หลักการทำงานของ Filter Bank Common Spatial Pattern (FBCSP) คือเป็นการเพิ่มความละเอียดของคุณสมบัติด้วยการใช้ช่วงเวลาหลายๆช่วง และนำช่วงความถี่ของสัญญาณหลายๆช่วงมาใช้ในการหาคุณสมบัติเพิ่มเติม อย่างไรได้ดังนี้

2.2.8.1 Preprocessing data: Extract time window

เนื่องจาก Dataset ที่เราใช้นั้นถูกเก็บข้อมูลมาจาก Subject ที่มีความหลากหลายทั้งประสบการณ์ในการทำการทดลองลักษณะนี้ ทั้งอายุ และเพศ ดังนั้นเราจะไม่สามารถสรุปได้เลยว่าในภาระรูปแบบของการทดลองเก็บข้อมูล EEG นั้น Subject แต่ละคนได้ทำการอินนาการ ณ เวลา

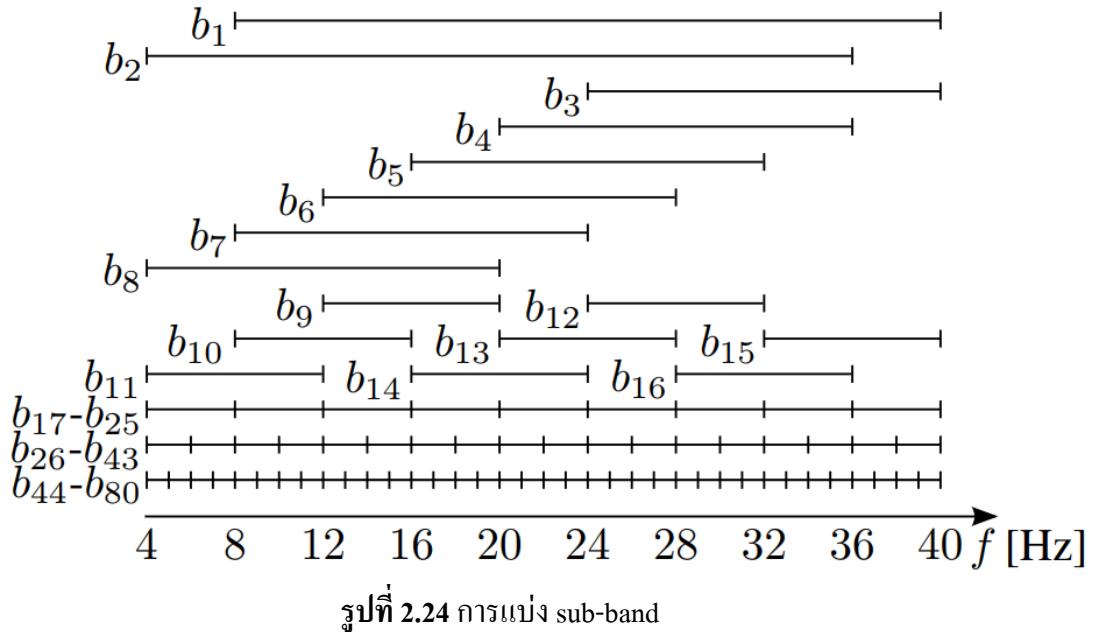
ในหลังจากที่ Cue ปรากฏขึ้นให้เห็น ดังนั้นพิการจึงต้องแบ่งช่วงของ Time window ออกเป็นหลาย ๆ ช่วงในการคำนวณหาคุณลักษณะ โดยพิการเราได้วางขอบเขตของ Time window ไว้ กว้างที่สุดคือช่วง 2.5 - 6.0 วินาที (ช่วงที่ Cue เพิ่งเริ่มปรากฏได้ 0.5 วินาทีไปจนถึงช่วงที่สิ้นสุดการจินตนาการ) และจึงแบ่งย่อยลงไปเรื่อย ๆ ตามลำดับ จากที่ได้แสดงในรูป ตำแหน่งของ Time window จะมีการ Overlap กันของ Data ใน 1 Trial ซึ่งถูกแบ่งออกเป็นหลาย ๆ ช่วงคือ ($T_1 - T_{11}$) แสดงดังรูปที่ 2.23 [3]



2.2.8.2 Preprocessing data: Extract spectral band

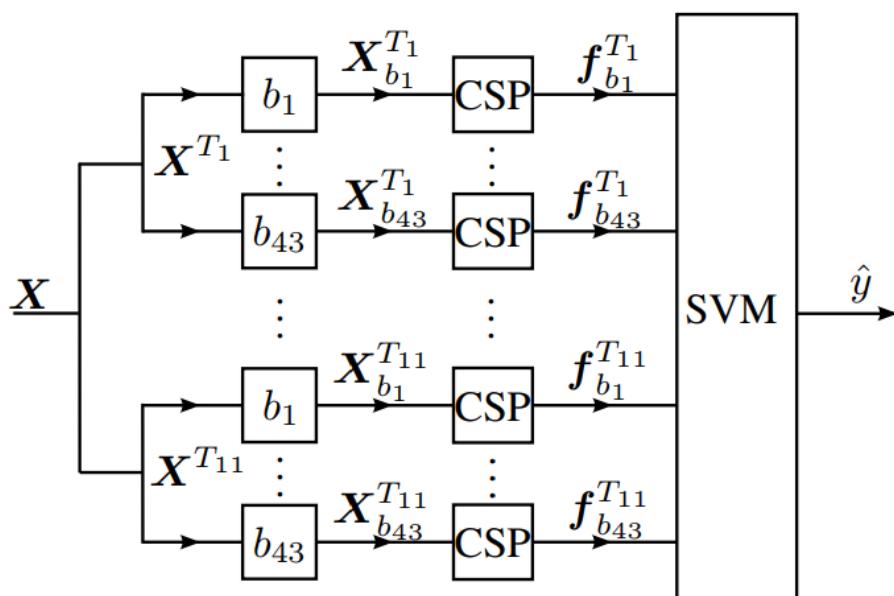
โดยทั่วไปความถี่ของสัญญาณสมองที่เกิดขึ้นจากการจินตนาการนั้น [3] จะเกิดอยู่ที่ช่วงระหว่าง 8 - 30Hz แต่ถึงจะเป็นเรื่อนี้ความถี่ของสัญญาณคลื่นสมองที่จะเกิดขึ้นมักมีความสัมพันธ์กันอย่างมากกับตัว Subject แต่ละคน ดังนั้นการคำนวณหาคุณลักษณะด้วยช่วงความถี่เดียวก็คือ 8 - 30Hz นั้นจะทำให้เกิดการลดความละเอียดของคุณลักษณะลงได้ เราจึงได้ขยายช่วงความถี่ที่จะคำนวณคุณลักษณะ ออกเป็น 4 - 40Hz และได้แบ่งเป็นหลาย ๆ ช่วงความถี่ดังภาพ โดยมี

Bandwidths อยู่ที่ 32 Hz และมีการ Overlapping ของแต่ละช่วง Frequency band $b_1 - b_{43}$ แสดงดังรูปที่ 2.24 [3]



Feature Extraction: CSP Algorithm for each $X_{B_b}^{T_t}$; $1 \leq t \leq 11$, $1 \leq b \leq 43$

ทั้ง Time window และ Frequency Band window ก่อให้เกิดการเลือกช่วงข้อมูลที่มีความทับซ้อน ซึ่งจะช่วยเพิ่มประสิทธิภาพในการทำ Feature extraction ซึ่งเป็นขั้นตอนหนึ่งในขั้นตอนของการทำ CSP



รูปที่ 2.25 แผนภาพแสดงการทำงานของ FBCSP

2.2.9 Machine Learning: Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis [16] เป็นเทคนิคการลดมิติของข้อมูล นิยมใช้เป็นขั้นตอน Pre-processing ของ Machine Learning และเพื่อการจำแนกรูปแบบ (Classification) เป้าหมายหลักของการลดมิติของข้อมูลคือการลบคุณลักษณะที่ซ้ำซ้อนและคุณลักษณะที่สัมพันธ์กัน โดยการเปลี่ยนคุณสมบัติ จากพื้นที่ที่มีมิติคุณสมบัติที่สูงไปเป็นพื้นที่ที่มีมิติคุณสมบัติที่มีขนาดต่ำกว่า Linear Discriminant Analysis เป็นเทคนิคการจำแนกประเภท Supervised learning

เป้าหมายของ Linear Discriminant Analysis คือการแสดงคุณสมบัติ (Feature) ในพื้นที่ที่มีมิติสูงสู่พื้นที่ที่มีมิติที่ต่ำกว่า ลิ้งนี้สามารถทำได้ใน 3 ขั้นตอน ขั้นตอนแรกคือการคำนวณการแบ่งแยกระหว่างคลาสที่แตกต่างกัน (เช่นระยะห่างระหว่างค่าเฉลี่ยของแต่ละคลาส) หรือที่เรียกว่า “Between-class Variance” แสดงดังสมการที่ 2.6

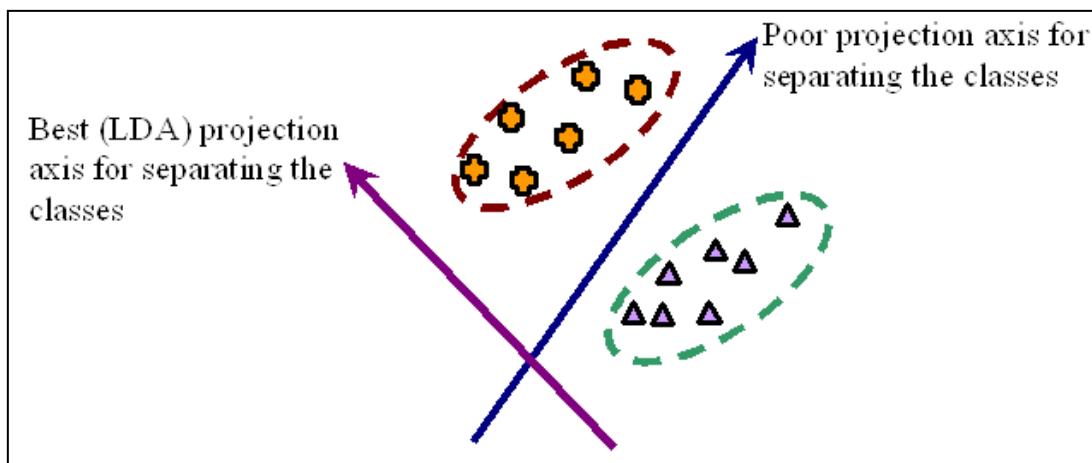
$$S_b = \sum_{i=1}^g N_i (\bar{x}_i - \bar{x})(\bar{x}_i - \bar{x})^T \quad (2.6)$$

ขั้นตอนที่สองคือการคำนวณระยะทางระหว่างค่าเฉลี่ยและตัวอย่างข้อมูลในแต่ละคลาส ซึ่งเรียกว่า ความแปรปรวนภายในคลาส แสดงดังสมการที่ 2.7

$$S_w = \sum_{i=1}^g (N_i - 1) S_i = \sum_{i=1}^g \sum_{j=1}^{N_i} (X_{i,j} - \bar{x}_i)(\bar{x}_{i,j} - \bar{x}_i)^T \quad (2.7)$$

ขั้นตอนที่สาม คือการสร้างพื้นที่มิติที่ต่ำกว่าซึ่งมีความแปรปรวนเพิ่มขึ้นระหว่างคลาส และลดความแปรปรวนภายในคลาสให้น้อยที่สุด ให้ P เป็นการฉายพื้นที่ที่มีมิติต่ำกว่าซึ่งเรียกว่า Fisher's Criterion แสดงดังสมการที่ 2.8

$$P_{lida} = arg_P max \frac{|P^T S_b P|}{|P^T S_w P|} \quad (2.8)$$



รูปที่ 2.26 LDA Projection

[ที่มา: <https://medium.com>]

2.2.10 Model Evaluation

ในการประเมินผลของ Model [17] เราใช้ Confusion matrix สำหรับ Multi-class เกณฑ์เหล่านี้ในการวิเคราะห์ประสิทธิภาพของ Model

- **True Positive (TP)**

ผลการทำนายทายได้ถูกต้องตามค่าที่คาดหวัง (Actual) ที่เป็น Positive

- **False Positive (FP)**

ผลทำนายไม่ถูกต้อง จากความจริงที่ไม่เป็นสิ่งนั้น แต่ผลทำนายว่าเป็นสิ่งนั้น

- **True Negative (TN)**

ผลการทำนายทายได้ถูกต้อง สำหรับความเป็นจริงไม่เป็นสิ่งนั้น ผลการทำนายที่ทำนายว่าไม่เป็นสิ่งนั้น

- **False Negative (FN)**

ผลการทำนายไม่ถูกต้อง ความเป็นจริงเป็นสิ่งนั้น แต่ผลทำนายว่าไม่เป็นสิ่งนั้น

- **Accuracy**

ค่าที่บ่งบอกว่าโปรแกรมสามารถทำนายได้แม่นยำขนาดไหน หาได้จาก $(TP+TN)/(TP+TN+FP+FN)$

● **Precision**

ค่าที่บอกว่าโปรแกรมทำนายว่าจริง ถูกต้องเท่าไร หาได้จาก $TP/(TP+FP)$

● **Recall**

ค่าที่บอกว่าโปรแกรมทำนายได้ว่าจริง เป็นอัตราส่วนเท่าไรของจริงทั้งหมด หาได้จาก $TP/(TP+FN)$

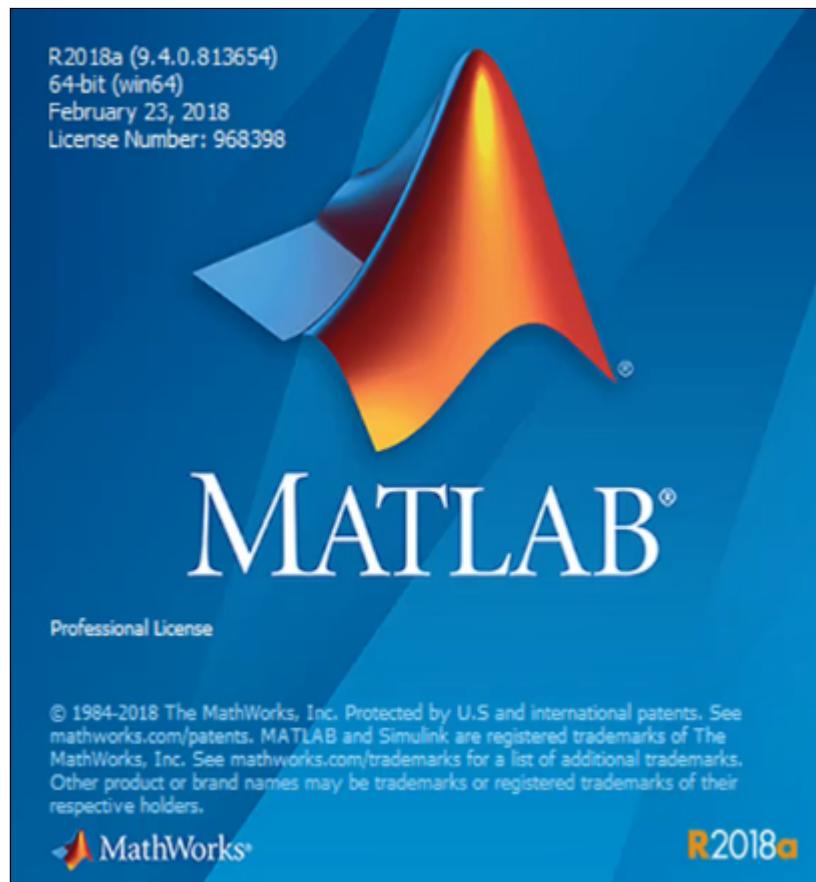
2.3 Programming Language and Software used

2.3.1 MATLAB R2018a

Matlab (แมตแล็บ) [18] เป็นภาษาคอมพิวเตอร์ระดับสูงที่มาพร้อมด้วยสภาพแวดล้อมการทำงานเชิงโต้ตอบซึ่งสามารถคำนวณคณิตศาสตร์ที่ซับซ้อนได้อย่างรวดเร็วมากกว่าภาษาคอมพิวเตอร์สมัยก่อน เช่น ภาษา C, ภาษา C++ หรือ ภาษา Fortran

Matlab เป็นภาษาคอมพิวเตอร์ระดับสูงที่สำหรับคำนวณเชิงตัวเลข Numerical Computing แสดงผลกราฟฟิก และเขียนแอปพลิเคชัน ทำให้เราสามารถคำนวณผลลัพธ์ พัฒนาอัลกอริทึม สร้างแบบจำลอง และแอปพลิเคชันได้จ่ายและรวดเร็วมาก ภายในตัว Matlab ประกอบด้วยภาษาคอมพิวเตอร์ Toolbox กลุ่มฟังก์ชัน สำเร็จรูปในแต่ละสาขาวิชา และฟังก์ชันพื้นฐานจำนวนมาก ทำให้การวิเคราะห์ทำได้หลากหลายวิธี พร้อมกับคำตอบที่รวดเร็ว

เราสามารถนำ Matlab ไปประยุกต์ใช้งานได้หลายสาขางาน การประมวลผลสัญญาณ (Signal Processing) การสื่อสาร (Communication) การประมวลผลภาพและวิดีโอ (Image and Video Processing) ระบบควบคุม (Control System) การวัดและควบคุม (Instruments and Control) การคำนวณทางเศรษฐศาสตร์ (Economic) การคำนวณทางชีววิทยา Biology และอื่นๆ ดังรูปที่ 2.27



รูปที่ 2.27 โปรแกรม MATLAB เวอร์ชัน R2018a

[ที่มา: <http://eduotech.blogspot.com/2018/03/matlab-r2018a-64bit-product-list.html>]

2.3.2 Lab Streaming Layer (LSL)

Lab Streaming Layer (LSL) [19] เป็นโปรแกรมบนภาษา python ซึ่งสามารถ抓รับสัญญาณประสาทได้แบบ Real-time จากอุปกรณ์ OpenBCI และส่งค่าสัญญาณประสาทไปยังโปรแกรม Matlab สำหรับนำสัญญาณประสาทเหล่านั้นมาประมวลผลต่อไป แสดงดังรูปที่ 2.28



รูปที่ 2.28 Lab Streaming Layer (LSL)
[ที่มา: <https://www.abbreviations.com/term/1636908>]

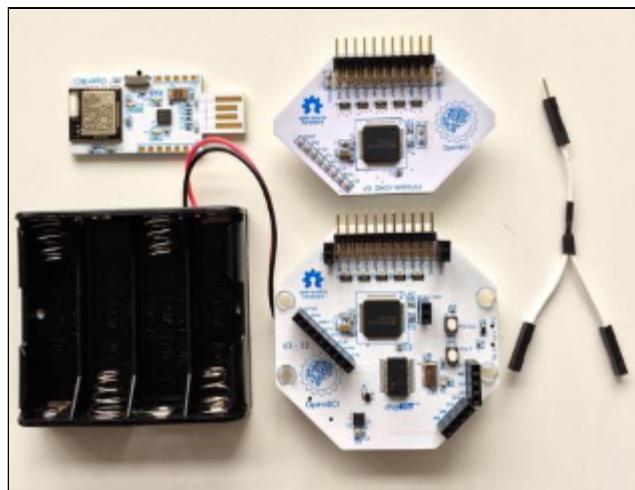
2.4 Hardware part and library used

2.4.1 OpenBCI

- **Hardware: Cython + Daisy board**

Cython OpenBCI Board [20] เป็น Arduino-Compatible จำนวน 8 channels ที่ใช้สำหรับการตรวจจับสัญญาณประสาท มีหน่วยประมวลผล 32-bit Cyton OpenBCI Board ใช้ไมโครคอนโทรลเลอร์ PIC32MX250F128B ให้หน่วยความจำที่มากและความเร็วในการประมวลผลที่รวดเร็ว บอร์ดมาพร้อมกับ bootloader chipKIT™ และเฟิร์มแวร์สำหรับ OpenBCI เวอร์ชันล่าสุด สัญญาณสมองที่ตรวจจับจะมี sampling rate เท่ากับ 250 Hz

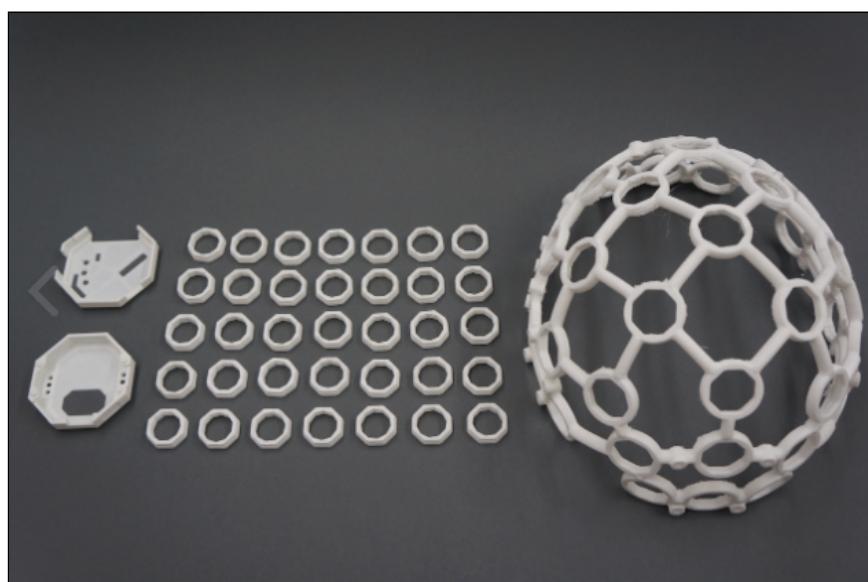
OpenBCI Cyton Board และ OpenBCI Daisy Module (ซึ่งเสียบเข้ากับด้านบนของ OpenBCI Cyton Board) สามารถใช้เพื่อเก็บตัวอย่างกิจกรรมทางสมอง (EEG), กิจกรรมกล้ามเนื้อ (EMG) และกิจกรรมหัวใจ (ECG) ได้มากถึง 16 channels ระบบสื่อสารแบบไร้สายไปยังคอมพิวเตอร์ผ่านdongle OpenBCI USB โดยใช้โมดูลวิทยุ RFDuino นอกจากนี้ยังสามารถสื่อสารแบบไร้สายกับอุปกรณ์มือถือหรือแท็บเล็ตที่รองรับ Bluetooth Low Energy (BLE) แสดงดังรูปที่ 2.29



รูปที่ 2.29 USB Dongle, Cython + Daisy board, 4xBattery AA Socket and wires
[ที่มา: <https://docs.openbci.com>]

2.4.2 Headware: Ultracortex Mark IV

Ultracortex Mark IV [21] เป็นชุดอุปกรณ์สวมศีรษะ open-source 3D-printable ซึ่งผลิตมาเพื่อใช้ทำงานร่วมกับ OpenBCI Board โดยกีตาม สามารถบันทึกกิจกรรมสมอง (EEG) ระดับกิจกรรมกล้ามเนื้อ (EMG) และกิจกรรมหัวใจ (ECG) ตัวของ Ultracortex Mark IV นั้นไม่สามารถรับสัญญาณได้ด้วยตัวเอง แต่จะต้องติดตั้ง electrode เข้าไปโดยสามารถติดตั้ง electrode ได้สูงสุดถึง 35 จุด ในระบบจัดวางแบบ 10-20 แสดงดังรูปที่ 2.30, 2.31 , 2.32 , 2.33 , 2.34 และ 2.35



รูปที่ 2.30 Ultracortex 3D-Printable model

[ที่มา: <https://docs.openbci.com>]



รูปที่ 2.31 electrode แบบหัวนำ
[ที่มา: <https://docs.openbci.com>]



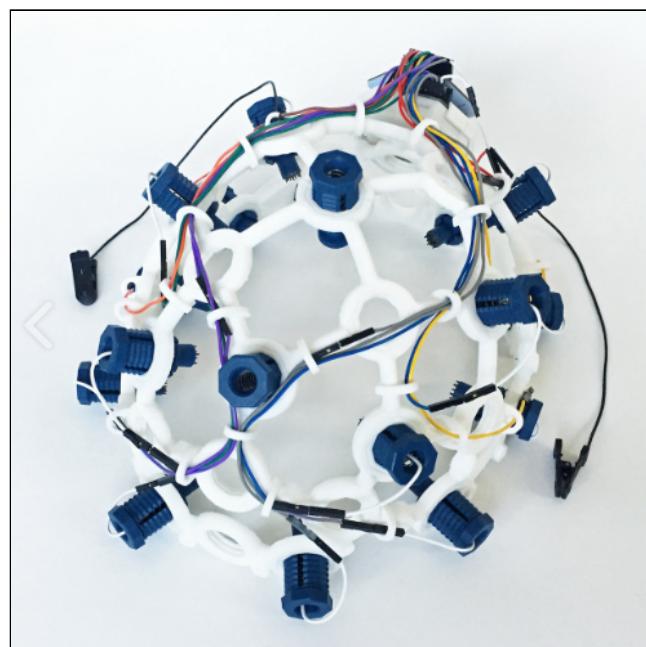
รูปที่ 2.32 electrode แบบเรียบ
[ที่มา: <https://docs.openbci.com>]



รูปที่ 2.33 electrode แบบ support
[ที่มา: <https://docs.openbci.com>]

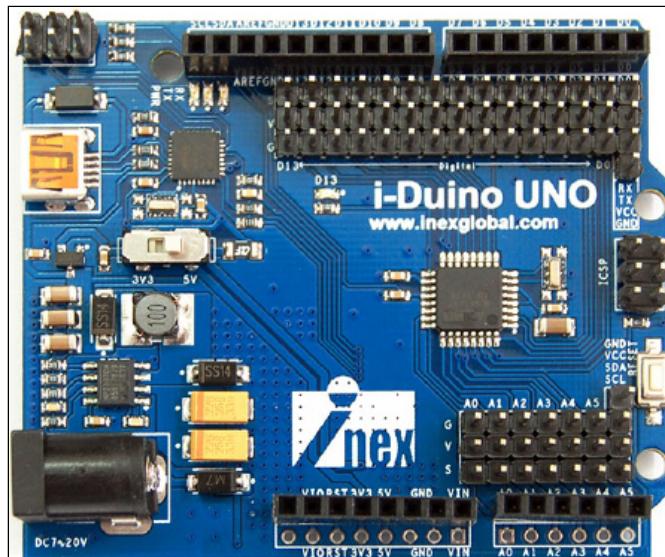


รูปที่ 2.34 ที่หนีบหุ้มเพื่อเป็นจุดเทียบ ground
[ที่มา: <https://docs.openbci.com>]



รูปที่ 2.35 Ultracortex Mark IV เมื่อประกอบเสร็จสมบูรณ์
[ที่มา: <https://docs.openbci.com>]

2.4.3 i-Duino UNO



รูปที่ 2.36 i-Duino UNO

[ที่มา: <http://www.appsofttech.com/store/product/view>]

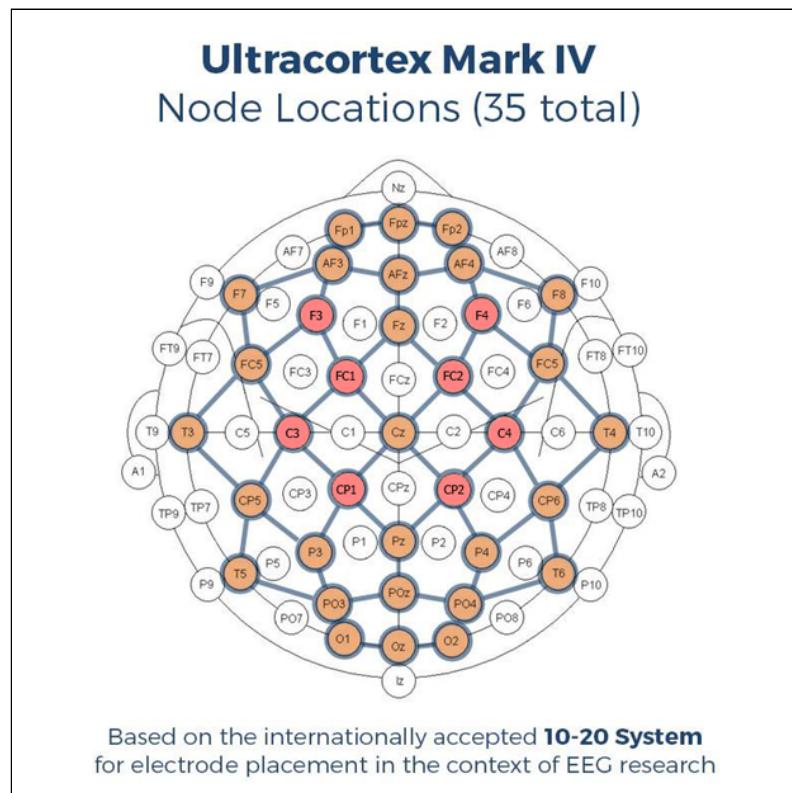
บอร์ด i-Duino UNO [22] มีสวิตช์เพื่อเลือกใช้ไฟเลี้ยงทั้ง +5V และ +3.3V ทำให้บอร์ด i-Duino UNO เป็นบอร์ด Arduino UNO compatible ในไม่กี่รุ่นในโลกที่รองรับไฟเลี้ยงทั้งสองระบบนี้ ด้านพอร์ตสำหรับต่อใช้งาน i-Duino UNO ได้จัดเตรียมจุดต่อพอร์ตที่เข้ากับบอร์ด Arduino UNO ทุกประการ ทำให้รองรับบอร์ด Arduino shield ได้ทุกแบบ มีการจัดเรียงขาในแบบ gravitational-ไฟเลี้ยง-สัญญาณ ให้กับทุกขาพอร์ตตั้งแต่ D0 ถึง D13 และ A0 ถึง A5 (หรือ D14 ถึง D19) และยังเตรียมจุดต่อพอร์ตแบบ 4 ขาสำหรับเชื่อมต่ออุปกรณ์ระบบบัส I2C และ UART ทำให้สะดวกต่อการเชื่อมต่อกับอุปกรณ์ภายนอกอย่างมาก ด้านซิปไมโครคอนโทรลเลอร์หลัก ใช้ ATmega328 เมื่อเทียบกับ Arduino UNO มาตรฐาน บรรจุเฟิร์มแวร์บูตโหลดเดอร์ที่ตรงกับ Arduino UNO ส่วนซิปที่ทำหน้าที่เชื่อมต่อกับพอร์ต USB ใช้ไมโครคอนโทรลเลอร์ ATmega8U2 ที่บรรจุเฟิร์มแวร์เพื่อทำหน้าที่เป็นตัวแปลงสัญญาณพอร์ต USB เป็นสัญญาณข้อมูลอนุกรม UART แสดงดังรูปที่ 2.36

บทที่ 3

การออกแบบและประเมินวิธีวิจัย

3.1 อุปกรณ์ที่ใช้

ใช้อุปกรณ์ตรวจจับคลื่นไฟฟ้าสมอง OpenBCI Ultracortex Mark IV ซึ่งมี sampling rate 250 Hz โดยจะใช้อิเล็กโทรดทั้งหมดจำนวน 8 channels และวางแผนของอิเล็กโทรดตาม International 10–20 system ซึ่งจะแสดงตำแหน่งของอิเล็กโทรดที่ติดต่ออยู่ที่ F3, F4, FC1, FC2, C3, C4, CP1 และ CP2 ดังรูปที่ 3.1



รูปที่ 3.1 ตำแหน่งอิเล็กโทรดที่ headset ใช้เทียบกับ International 10–20 system ตำแหน่งที่เป็นสีชมพูก็คือ ตำแหน่งที่คณะผู้จัดทำใช้วางอิเล็กโทรดในการทดลอง
[ที่มา: <https://docs.openbci.com>]

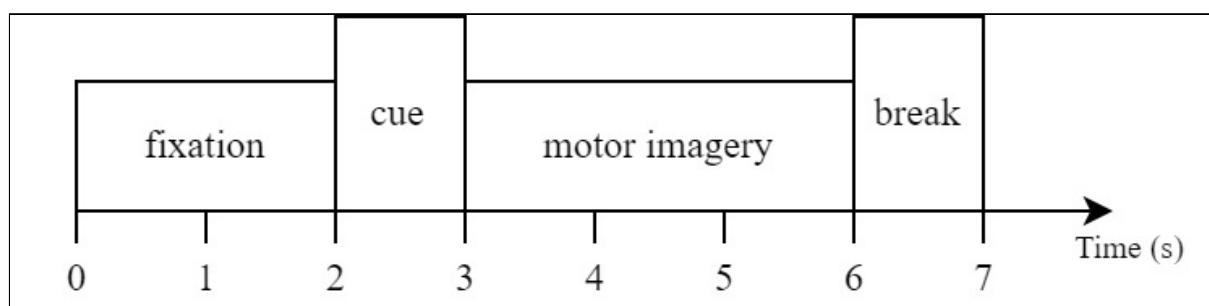
3.2 Software ที่ใช้

- สำหรับ graphical user interface ตอบโต้กับผู้ร่วมการทดลอง จะเขียนขึ้นมาจาก Matlab
- การรับค่าข้อมูลสัญญาณประสาทจาก OpenBCI เราจะใช้งานเครื่องมือที่มีชื่อว่า Lab Streaming Layer (LSL) บนภาษา python ซึ่งสามารถรับสัญญาณประสาทได้แบบ real-time และส่งค่าเข้าโปรแกรม Matlab สำหรับนำสัญญาณประสาทเหล่านั้นมาประมวลผลต่อไป

3. วิเคราะห์ข้อมูลด้วยโปรแกรม Matlab
4. ข้อมูลที่ได้มาจะถูกเก็บเป็น array ไว้ใน workspace ของ Matlab
5. การ preprocessing จะใช้ tool box EEG Lab ของโปรแกรม Matlab

3.3 ลำดับการปฏิบัติสำหรับผู้เข้าร่วมการทดลอง

ในขั้นตอนแรกของการทดลอง ผู้เข้าร่วมการทดลองจะต้องถูกเก็บข้อมูลล้วนๆ สมองขณะที่กำลังจินตนาการการเคลื่อนไหวอวัยวะก่อนเสมอ สำหรับใช้ในการ train model โดยจะเก็บรูปแบบการเคลื่อนไหว (class) ทั้งหมด 4 รูปแบบ ได้แก่ การขยับมือซ้าย-ขวา และการขยับขาซ้าย-ขวา ใน การทดลองนี้ ในการทดลองจะปล่อยให้ผู้เข้าร่วมการทดลองนั่งกับเก้าอี้แบบมีพิงมือ ในการเริ่มต้นการปฏิบัติ (เวลาศูนย์วินาที) จะมี FixationPoint ปรากฏขึ้นบนหน้าจอแสดงผลสีดำ หลังจากนั้นในวินาทีที่สองจะมีลูกศรปรากฏขึ้นเป็นเวลา 1 วินาที ซึ่งลูกศรจะสัมพันธ์กับคลาสทั้งสี่ของการปฏิบัติ ลูกศรที่ปรากฏบนหน้าจอเป็นสีทึบๆ บวกให้ผู้เข้าร่วมการทดลองทำการจินตนาการถึงการขยับอวัยวะที่กำหนดไว้จนกระทั่งลูกศรหายไปในวินาทีที่หก หลังจากนั้นจะมีการเว้นช่วงให้พักสั้นๆ 1 วินาทีซึ่งหน้าจอจะเป็นสีดำ นับเป็น 1 trial โดยผู้ร่วมการทดลองจะต้องจินตนาการการเคลื่อนไหวทั้งหมด 12 trials (คลาสละ 3 trials) สำหรับการนำข้อมูลทั้งหมดนี้มาใช้ในการ train ขั้นตอนการทดลองคั่งกล่าวแสดงดังรูป



รูปที่ 3.2 แผนผังการเก็บข้อมูลสำหรับการ train

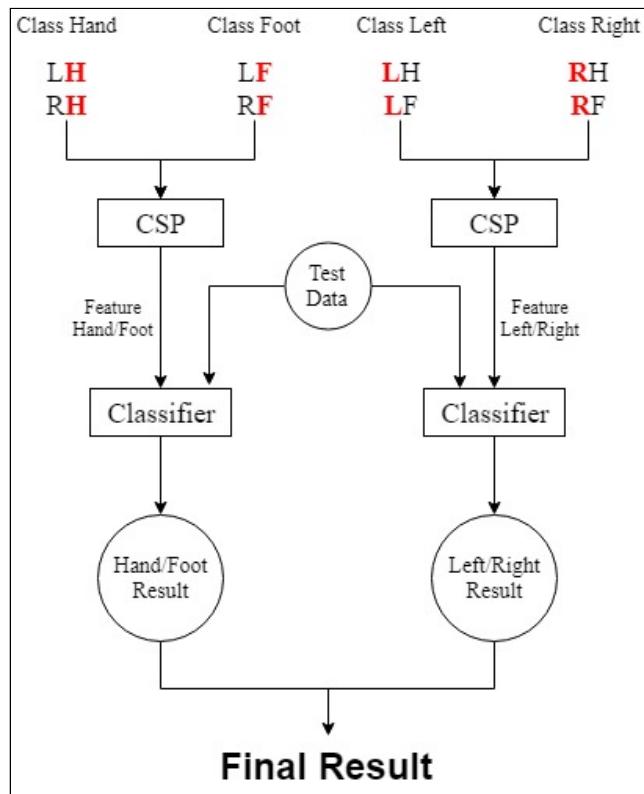
3.4 การเตรียมข้อมูล (Preprocessing)

หลังจากเก็บข้อมูลแล้วระบบจะนำข้อมูลที่ได้ไปกรองความถี่ที่ไม่ต้องการออกก่อน ซึ่งเนื่องจากช่วงความถี่ 8 - 30 Hz เป็นช่วงความถี่ของ SMR (Sensorimotor Rhythm) ที่เป็นคลื่นที่เกิดขึ้นระหว่าง Motor imagery คณะผู้จัดทำจึงทำการกรองความถี่ต่อไปโดยที่ Cut Off Frequency ของความถี่เป็น 30Hz และกรองความถี่สูงกว่าโดยที่ Cut Off Frequency มีค่าเท่ากับ 8 Hz

3.5 การวิเคราะห์ข้อมูล

การวิเคราะห์ข้อมูล จะแบ่งเป็น 2 ส่วน คือส่วน train model และ ส่วนที่ใช้วิเคราะห์ข้อมูลที่ streaming เข้ามา

1. ในส่วนของการเก็บข้อมูล การ train model ระบบจะใช้ Common Spatial Pattern ในการหา Feature ของข้อมูล ระบบจะนำข้อมูลที่เก็บไว้สำหรับใช้ train มาแยกการ train เป็นสองแบบ ได้แก่ การหา feature สำหรับแยกมือกับเท้า และ feature สำหรับแยกซ้ายกับขวา โดยระบบจะนำข้อมูลมือซ้ายกับมือขวารวมกันสมมติให้เป็น class ที่ปั่งบอกว่าเป็น “มือ” และนำข้อมูลขาซ้ายกับขาขวารวมกัน สมมติให้เป็น class ที่ปั่งบอกว่าเป็น “เท้า” แล้วจึงนำข้อมูลไป train เพื่อหา feature ที่แยกระหว่างมือกับเท้า ในทำนองเดียวกัน เพื่อหา feature สำหรับแยกซ้ายกับขวา จะนำข้อมูลมือซ้ายและขาซ้าย มารวมกัน และนำข้อมูลมือขวาและขาขวารวมกัน
2. ขณะที่ทำการ streaming ข้อมูล ระบบจะ classify ข้อมูลโดยใช้ Linear Discriminant Analysis ซึ่งจะให้ผลลูกค้าเป็นตัวแปรตัวหนึ่ง ซึ่ง ผลลัพธ์จากการ classify จะถูกเก็บเป็นตัวแปรสำหรับทุกๆ sample ที่ streaming เข้ามา ในรูปแบบของเลขจำนวนเต็ม ในการแยกระหว่าง มือ กับเท้า เลข 1 จะหมายถึง “มือ” และเลข 2 หมายถึง “เท้า” ทำนองเดียวกัน ในการ classify ซ้าย กับขวา เลข 1 จะหมายถึง “ซ้าย” และเลข 2 หมายถึง “ขวา” โดยที่ input ของ classifier จะแบ่ง เป็น 2 แบบ input ตัวแรกของทั้ง 2 แบบคือข้อมูลที่ streaming จากผู้ทดลอง ส่วน input ตัวที่ 2 จะต่างกันที่รูปแบบของ feature แบบที่ 1 ใช้แยกระหว่างมือกับเท้า แบบที่ 2 ใช้แยกซ้ายกับขวา ดังแสดงในแผนภาพ 3.3 โดยตัวแปรที่เก็บผลลัพธ์การ classify ในแต่ละ sample จะถูกเก็บไว้ เป็น array ซึ่งตัวเลขตำแหน่งของผลลัพธ์ใน array กับลำดับที่ของ sample ที่ตรงกัน จะมีความ สัมพันธ์กันระหว่างผลลัพธ์จากการ classify กับ sample นั้นๆ ระบบจะทำการนับจำนวน class ที่ classify ได้สำหรับช่วง 5 วินาทีที่ streaming ข้อมูล มาหาบอร์เซ็นระหว่าง class 1 และ class 2 ซึ่งหาก class ไหนมีบอร์เซ็นมากกว่าจะเป็นผลลัพธ์ที่ระบบแสดงออกมายังเห็น



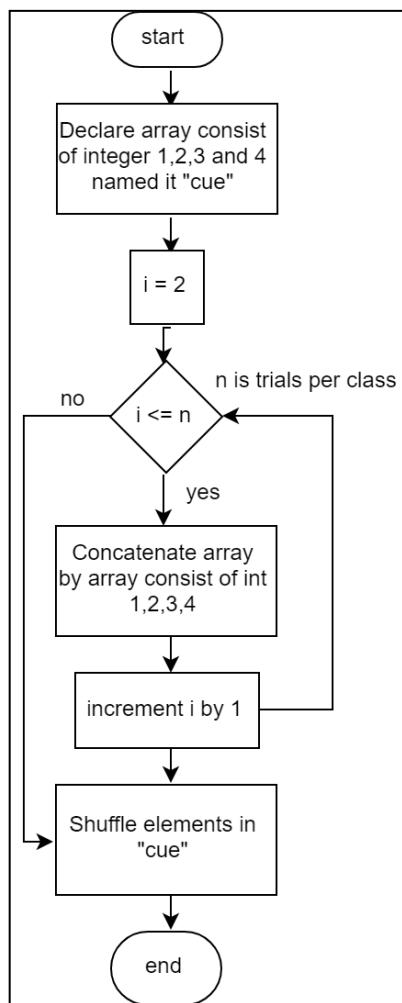
รูปที่ 3.3 แผนภาพแสดงการทำงานของโครงงาน

3.6 Coding and Algorithm

โปรแกรมทั้งหมด จะทำงานบนโปรแกรม matlab ตามลำดับ โดยแบ่งเป็น 6 ส่วน ดังนี้

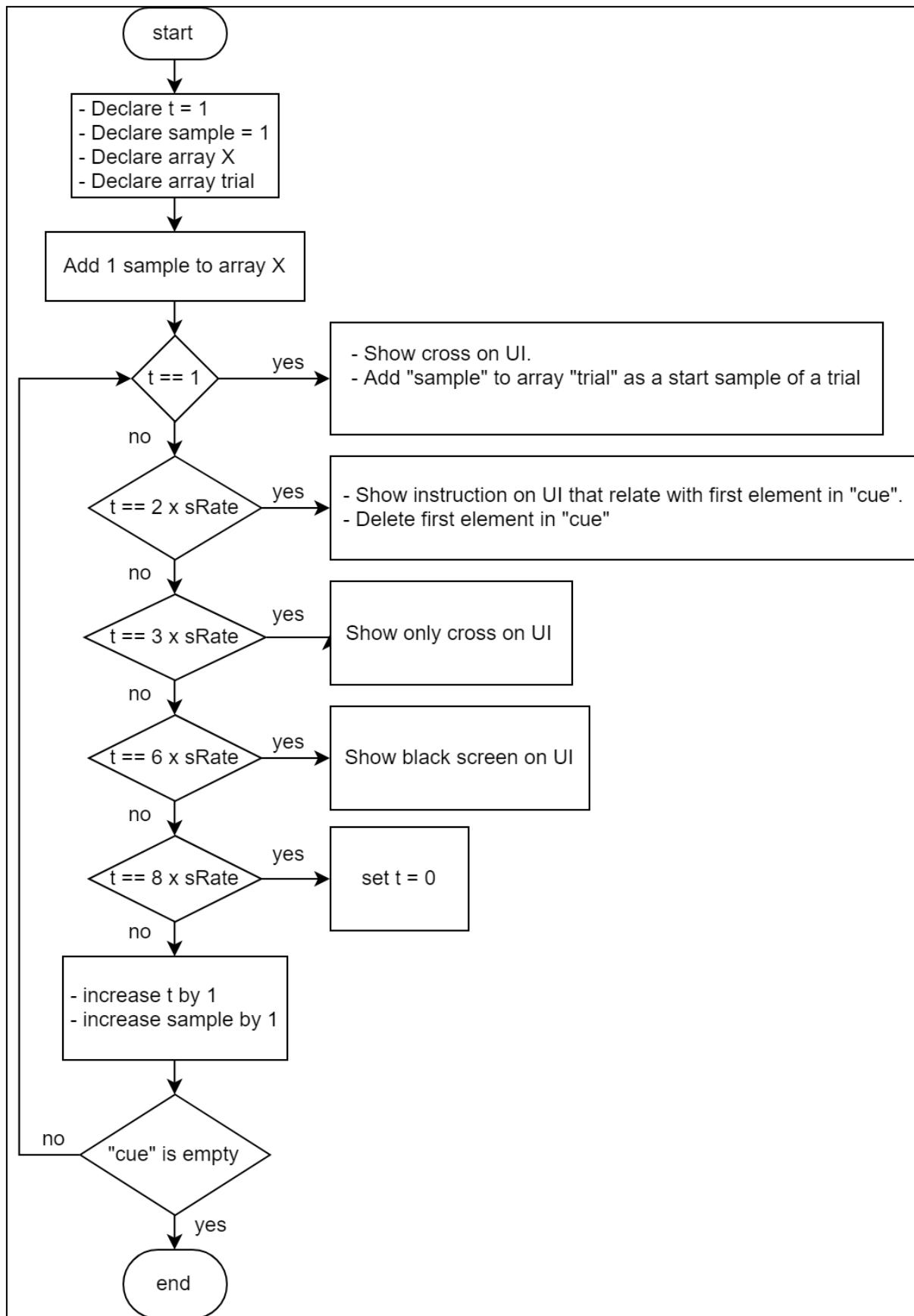
3.6.1 Obtained data

1. เรียก LSL library
2. สรุปลำดับของคำสั่งที่จะเรียกขึ้นบนหน้าจอ (cue) จากรูป 3.4 แสดงลำดับการทำงานของการสุ่ม cue ซึ่งจะถูกเก็บไว้ใน array โดยที่ element ใน array จะเป็นเลขจำนวนเต็มที่มีค่า 1, 2, 3 และ 4 แทน class แต่ละคลาส และแต่ละเลขจะมีจำนวนเท่ากับ n โดยที่ n คือจำนวน trial ต่อ class หนึ่งๆ ซึ่งในการทดลองครั้งนี้ ผู้ศึกษาได้กำหนดให้เก็บ 5 trial ต่อหนึ่ง class ทำให้จำนวน trial ทั้งหมดคือ 20 trial แสดงดังรูปที่ 3.4



รูปที่ 3.4 flow chart การสุ่มค่า trial

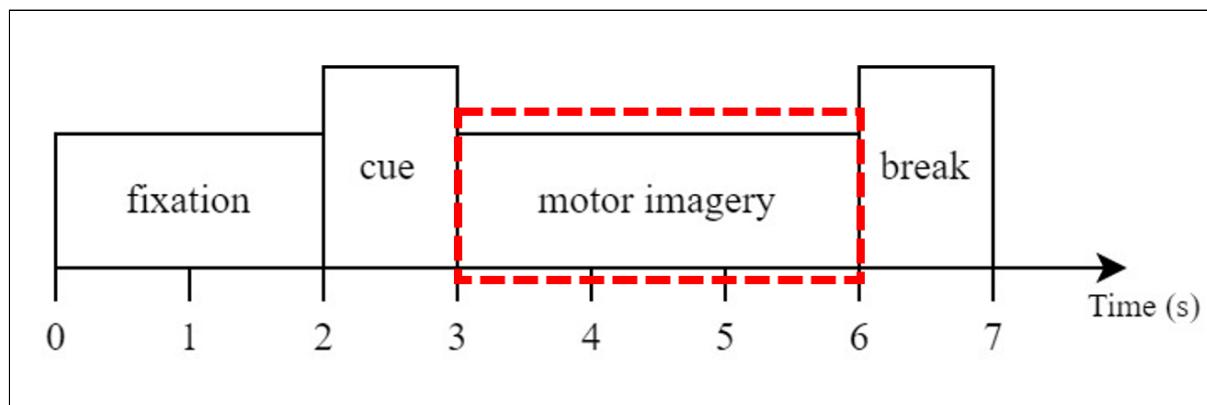
3. เรียกหน้าต่าง Graphic User Interface ขึ้นมา โดยในลำดับนี้ จะเป็นหน้าต่างสีดำ เป็นภาษาไทย
4. เข้าสู่ loop สำหรับเก็บข้อมูล โดยลำดับการทำงานใน loop จะสามารถแสดงได้ดัง รูป 3.5 เมื่อออกจากloop จะต้องรอการคีดข้อมูลจาก usb dongle ซึ่งมี sampling rate ของตัวเอง และลำดับที่ของ sample คือ index ของ loop ผู้ใช้ทำจึง ประมาณค่าเวลาในการเก็บข้อมูลและเปลี่ยนหน้า Interface ในหน่วยวินาทีได้จาก จำนวนเท่าของ sampling rate โดยการกำหนดตัวแปร t แทนเวลาในหน่วยวินาที จากนั้นจึงนำไปคูณกับค่า sampling rate เมื่อต้องการนำไปใช้ในส่วนของโค้ด แสดงดังรูปที่ 3.5



รูปที่ 3.5 flow chart ของ loop ที่ใช้เก็บ streaming ข้อมูลสำหรับใช้ train model

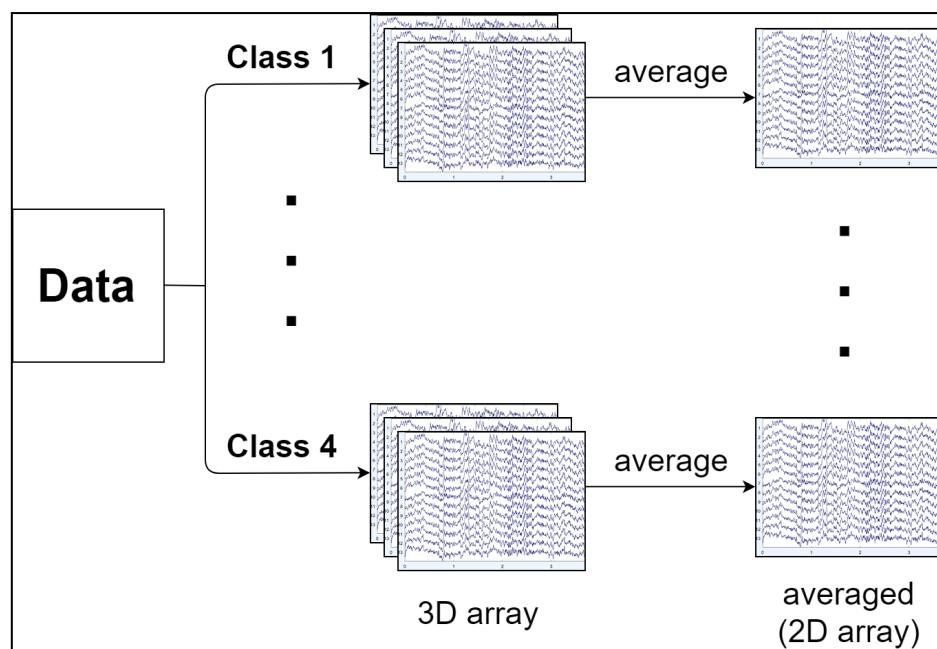
3.6.2 Extract Epoch

ในส่วนนี้ ผู้จัดทำได้ทำการเลือกข้อมูลในช่วงเวลาที่ผู้เข้าร่วมได้รับคำสั่งให้เริ่มจินตนาการ จนถึงช่วงเวลาที่ให้หยุดจินตนาการ (เป็นจำนวนทั้งหมด 375 sample ซึ่งมีค่าเท่ากับ 3 วินาที) ดัง แสดงในรูป 3.6 ที่เป็นการจินตการ class เดียวกันทั้งหมด มาทำการหาค่าเฉลี่ย แล้วเก็บไว้ใน ตัวแปร array เพื่อใช้ในการ train model ต่อไป (ผู้จัดทำเก็บไว้ทั้งหมด 4 ตัวแปร โดยที่ 1 ตัวแปร จะ ใช้สำหรับ 1 class)



รูปที่ 3.6 ภาพแสดงช่วงที่ระบบเลือกไปใช้ทำ epoch

ในส่วนการทำงานของโปรแกรม ผู้จัดทำได้สร้างตัวแปร 3 มิติ 1 ตัวสำหรับเก็บข้อมูล 1 class โดยมิติที่ 1 คือ จำนวน channel ที่เราใช้ (8 channel สำหรับการทดลองนี้) มิติที่ 2 คือจำนวน sample ในช่วงที่เลือก (375 sample จากที่กล่าวมาแล้วข้างต้น) และมิติที่ 3 คือจำนวน trial ต่อ class (มีค่าเท่ากับ 5 สำหรับการทดลองนี้) โดยมีลำดับการทำงานดังรูป 3.7



รูปที่ 3.7 แผนผังการทำ epoch

3.6.3 Preprocessing

สำหรับการทดลองนี้ ผู้จัดทำได้นำข้อมูลแต่ละ class ที่ถูกเฉลี่ยมากรองความถี่ผ่านช่วง 8 ถึง 30 Hz โดยการเขียน EEGLAB Scripts ซึ่งคำสั่งที่ใช้มีอยู่ด้วยกัน 2 พังก์ชันคือ pop_importdata() สำหรับเก็บข้อมูลให้อยู่ในรูปแบบที่ EEGLAB ใช้งานได้ และพังก์ชัน pop_eegfilt() สำหรับกรองความถี่ผ่าน

สำหรับพังก์ชัน pop_importdata() ผู้จัดทำได้ใส่พารามิเตอร์ดังรูป 3.8 และทำการใส่ค่า trials เป็น 1 เพื่อใช้สำหรับพังก์ชันกรองความถี่ผ่านต่อไป

สำหรับพังก์ชัน pop_eegfilt() ผู้จัดทำได้ใส่พารามิเตอร์ดังรูป 3.9 ซึ่งผลลัพธ์จากการทำ preprocessing จะได้เป็นตัวแปร array 4 ตัว แต่ละตัวคือข้อมูลของแต่ละ class ที่ถูกเฉลี่ยและกรองความถี่ผ่านแล้ว

```
LH = pop_importdata('data',avgClass1,'srate',125,'setname','LeftHand','nbchan',13,'pnts',375,'xmin',0);
LH.trials = 1;
```

รูปที่ 3.8 ตัวอย่างพารามิเตอร์สำหรับพังก์ชัน pop_importdata()

```
filtered_LH = pop_eegfilt( LH, 8, 30, 3*fix(srate/8));
```

รูปที่ 3.9 ตัวอย่างพารามิเตอร์สำหรับพังก์ชัน pop_eegfilt()

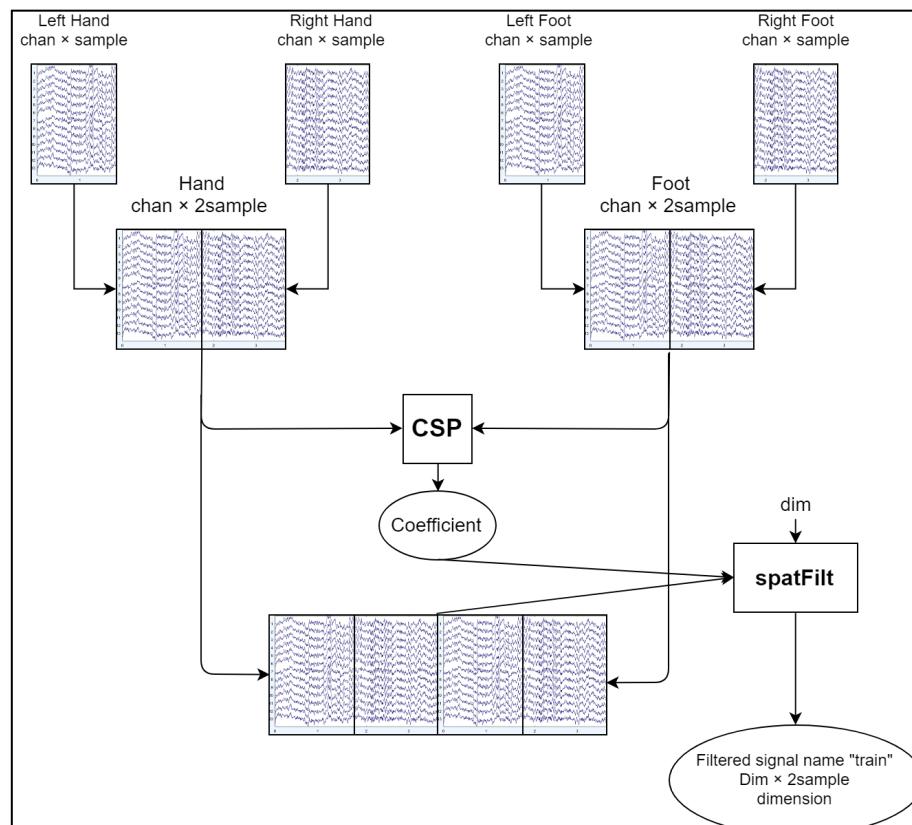
3.6.4 Feature Extraction การเลือกคุณสมบัติ

สำหรับการทำ feature จากข้อมูล จะทำการหา feature 2 ตัว ได้แก่ feature สำหรับใช้แยกการจินตนาการการขับมือ กับการขับเท้า และ feature สำหรับใช้แยกการการจินตนาการการขับฟื้นซ้ายกับขวา โดยโปรแกรมจะมีคำอธิบายดังนี้

1. นำข้อมูลมือซ้ายกับมือขวามาต่อกัน(อธิบายได้ดังรูป 3.10 หน้าถัดไป)เป็นข้อมูลชุดแรก และต่อข้อมูลเท้าซ้ายกับเท้าขวาเป็นข้อมูลชุดที่สอง
2. ใช้ข้อมูลชุดที่หนึ่งและสองเป็น input กับพังก์ชัน CSP() ซึ่งจะได้ output เป็น matrix จตุรัสที่มีขนาดเท่ากับจำนวน channel (ในการทดลองนี้จะได้ matrix ขนาด 8×8) โดยที่ matrix นี้คือ spatial filter coefficients สำหรับแยกมือกับเท้า
3. นำข้อมูลมือและเท้าต่อ กันเป็นชุดเดียว แล้วใช้ข้อมูลชุดนี้, spatial filter coefficients และ dimension (dimension ต้องมีค่าไม่เกินจำนวนแrewของ spatial filter coefficients ซึ่งมีค่าเท่ากับจำนวน channel) เป็น input ให้กับพังก์ชัน spatFilt ซึ่งจะให้ output เป็นลัญญาณที่ถูก filter แล้วออกมารูป matrix ที่มีจำนวนแrew

เท่ากับ dimension และจำนวนคอลัมน์เท่ากับจำนวน sample ของข้อมูลชุดแรก (หรือชุดที่สอง) ทั้งชุด ซึ่งในโปรแกรม จะตั้งชื่อให้ว่า train

4. ในทำงานเดียวกันเพื่อหา feature สำหรับการแยกชั้นและขวา ผู้จัดทำได้นำข้อมูลมาต่อ กัน เช่นเดิมแต่เปลี่ยนเป็นการต่อข้อมูลมือซ้ายกับขาซ้ายเป็นข้อมูลชุดแรก และต่อข้อมูลเท้าซ้ายกับเท้าขวาเป็นข้อมูลชุดที่สองแทน

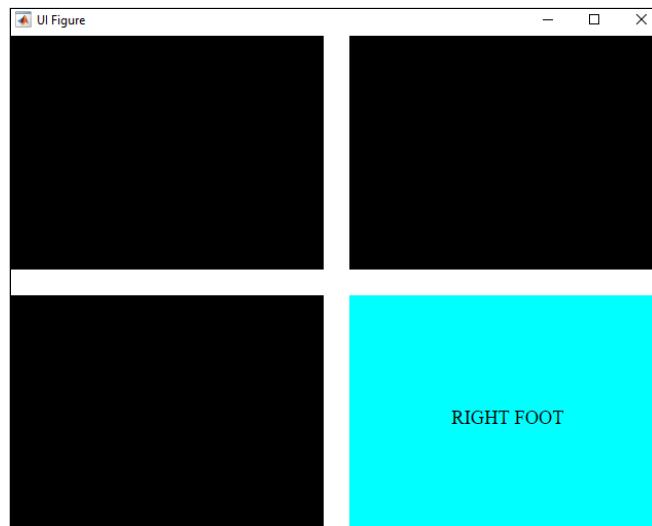


รูปที่ 3.10 แผนภาพลำดับการทำงานของโปรแกรมเพื่อหา feature สำหรับแยกมือกับเท้า

3.6.5 Real time feedback

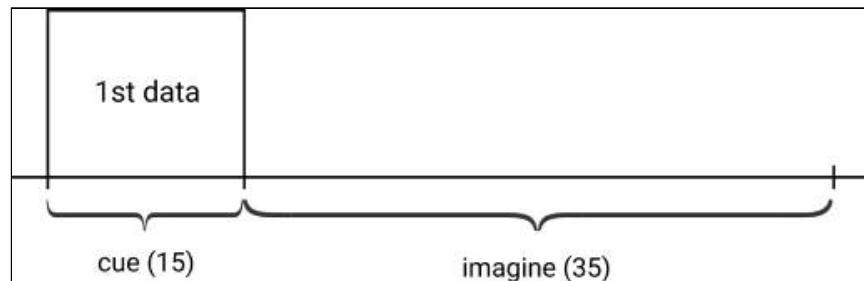
ในส่วนนี้ โปรแกรมจะทำงานทำงานเดียวกันกับตอนที่ streaming ข้อมูลเพื่อเก็บไป train model แต่จะมีสองส่วนที่ต่างกัน คือ

1. ลักษณะของ user interface - ในการ streaming ข้อมูลมาเก็บเพื่อนำไป train model จะให้หน้าต่างแสดงแค่ fixation cross แต่ในส่วนนี้จะแสดงผลเป็น feed back สีฟ้า ตรงช่องที่ระบบทำนายผลไว้แทน ดังตัวอย่างในรูปที่ 3.11

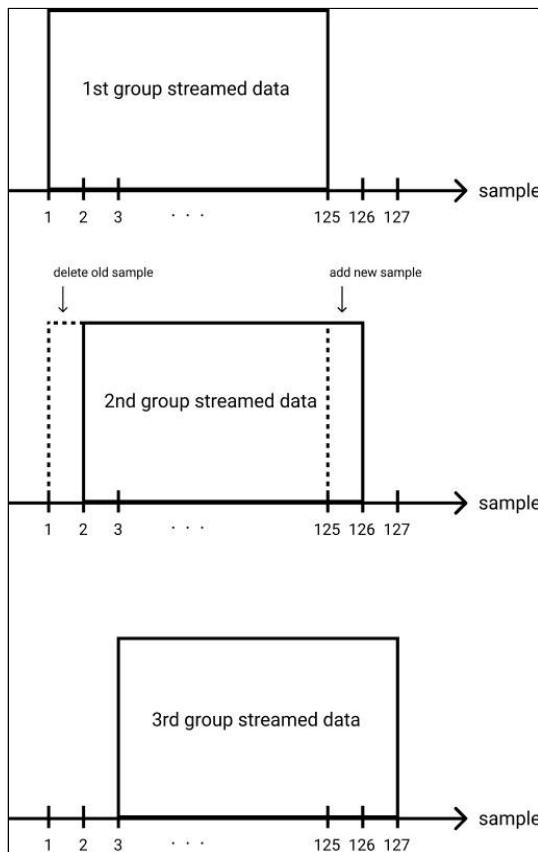


รูปที่ 3.11 ตัวอย่างหน้าจอแสดงผล feedback

2. การเก็บข้อมูลไปใช้จะเก็บไปทีละ 1 วินาที (125 sample) แทน โดยระบบจะเริ่มเก็บข้อมูลชุดแรกตั้งแต่ที่ cue ปรากฏขึ้นบนหน้าจอดังแสดงในรูป 3.12 จนกว่าระบบจะค่อยๆเก็บ sample ต่อไป ซึ่งในขณะเดียวกัน ก็จะลบ sample แรกทึ่งไปด้วย ดังแสดงในรูป 3.13



รูปที่ 3.12 แผนภาพแสดงข้อมูลชุดแรกที่ถูกเก็บสำหรับ real time feedback



รูปที่ 3.13 แผนภาพแสดงขั้นตอนการเก็บข้อมูลชุดต่อๆ ไป

ระบบจะนำข้อมูลชุดที่เก็บมาข้างต้นไปใช้เป็น input ของฟังก์ชัน classifier ซึ่ง output จากฟังก์ชัน classifier จะเป็นตัวแปรจำนวนเต็มที่บ่งบอกว่า class ที่ classify ได้คือ class อะไร จากนั้น จึงแสดงผลผ่านทางหน้าจอ feedback ที่ได้กล่าวมาข้างต้น

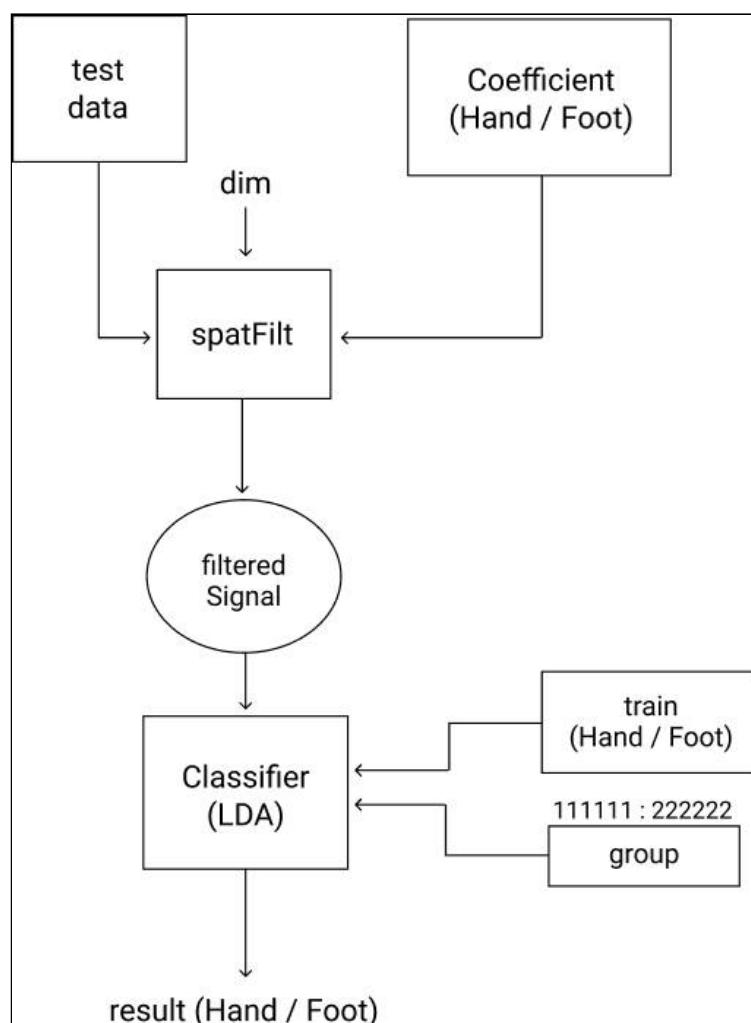
3.6.6 Classifier

ฟังก์ชัน classifier จะรับข้อมูลมาจากส่วน real time feed back เป็นข้อมูล 1 ชุดมา classify ซึ่งจะมีคำนิยามดังนี้

1. นำข้อมูล input และ spatial filter coefficients สำหรับแยกมือกับเท้า ป้อนเข้า ฟังก์ชัน spatFilt และตั้งชื่อให้ตัวแปร ouput ว่า test
2. สร้าง array ชื่อ “group” เป็น label ให้กับ matrix “train” ที่ได้รับมาตอนที่หา feature โดยแบ่งครึ่งแรกของตัวแปร train เป็น label 1 และครึ่งถัดไปเป็น label 2 ขณะนั้น ตัวแปร group จะมีจำนวนคอลัมน์เท่ากับจำนวนคอลัมน์ของตัวแปร train
3. transpose ตัวแปร test, train และ group และใช้เป็น input สำหรับฟังก์ชัน classify() ซึ่งเป็นฟังก์ชันของ matlab โดย output ที่ได้จะมีจำนวนคอลัมน์เท่ากับ ข้อมูลชุดที่รับมาจากส่วน real time feedback ซึ่ง element ใน output จะเป็น label

ที่บ่งบอกว่าระบบเดาว่า sample ที่ตรงกับข้อมูลชุดนี้เป็น class อะไร ในที่นี้ ถ้า label มีค่าเท่ากับ 1 จะหมายถึง “มือ” และถ้า label มีค่าเท่ากับ 2 จะหมายถึง “เท้า”

4. ระบบจะนับจำนวน label ซึ่งถ้า label ไหนมีจำนวนมากกว่า ระบบจะเดาว่าข้อมูลชุดนี้ คือ class ที่ตรงกับ label นั้น
5. ในทำงานองเดียวกัน ระบบจะแยกซ้ายขวาและให้ output ในลักษณะเดียวกัน ในที่นี้ หาก label มีค่าเท่ากับ 1 จะหมายถึง “ซ้าย” และหาก label มีค่าเท่ากับ 2 จะหมายถึง “ขวา”
6. จากนั้นระบบจะนำผลลัพธ์ทั้งหมดรวมกัน แล้วคืนค่ากลับไปที่ส่วน real time feed back เป็นเลขจำนวนเต็มที่บ่งบอกว่า ระบบทำนายว่าข้อมูลชุดนี้คือ class อะไร โดยที่เลข 1 หมายถึง มือซ้าย เลข 2 หมายถึงมือขวา เลข 3 หมายถึงเท้าซ้าย และเลข 4 หมายถึงเท้าขวา ตามลำดับ

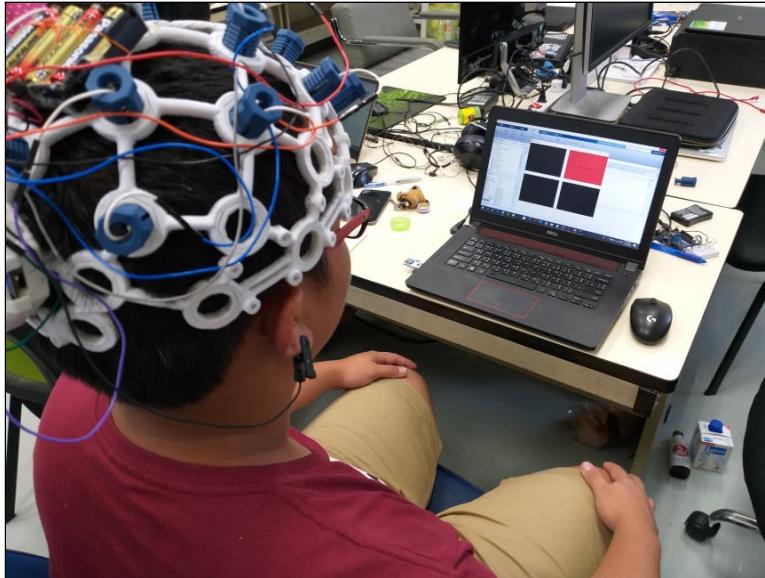


รูปที่ 3.14 แผนภาพแสดงลำดับการทำงานของฟังก์ชัน classifier

บทที่ 4

ผลการทดลอง

4.1 ผลการเก็บข้อมูล



รูปที่ 4.1 ขั้นตอนการบันทึกสัญญาณจากอุปกรณ์ OpenBCI

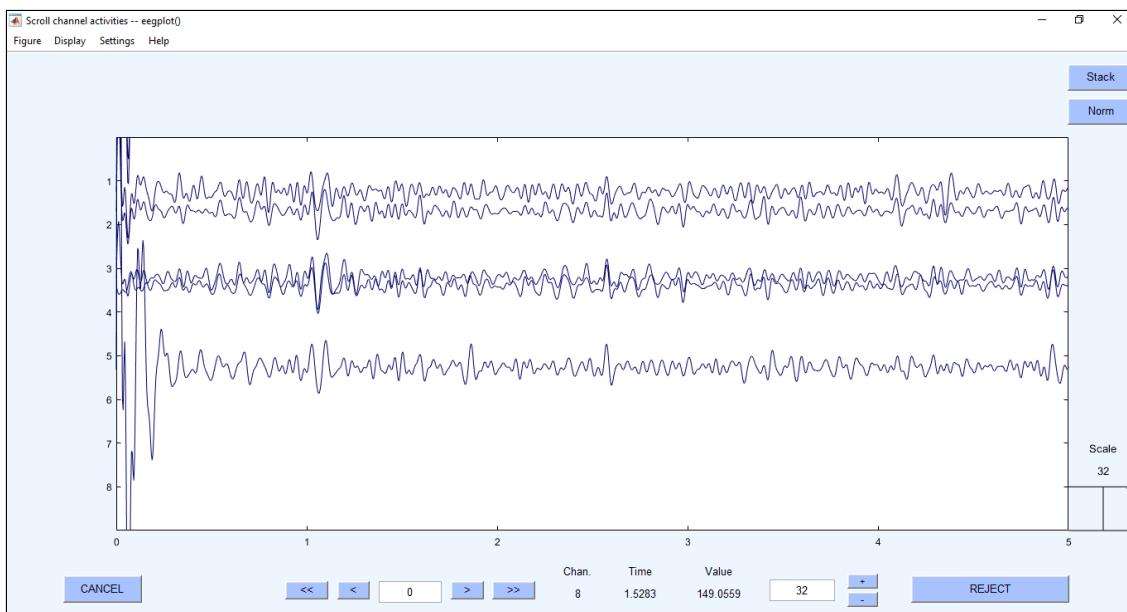
พวกราได้บันทึกข้อมูลดิบสัญญาณประสาท EEG ผ่านอุปกรณ์ OpenBCI จำนวน 8 channel ส่งค่าผ่าน Lab Streaming Layer เข้าสู่โปรแกรม Matlab เพื่อนำสัญญาณดิบเหล่านี้มาทำการ Pre-processing สัญญาณก่อนจะนำไปสู่การคัดเลือกคุณสมบัติด้วย CSP และจำแนกประเภทต่อไป แสดงดังรูปที่ 4.1

4.1.1 ผู้ทดลอง

พวกราได้ทำการบันทึกสัญญาณ EEG โดยมีผู้ทดลองเป็นเพศชาย อายุ 22 ปี จำนวน 1 คน ทำการทดลองในห้องปฏิบัติการภาควิชาวิศวกรรมคอมพิวเตอร์ ภายในห้องปฏิบัติการมีเสียงรบกวนอยู่ที่ประมาณ 39-41 dB

4.1.2 ข้อมูลดิบ

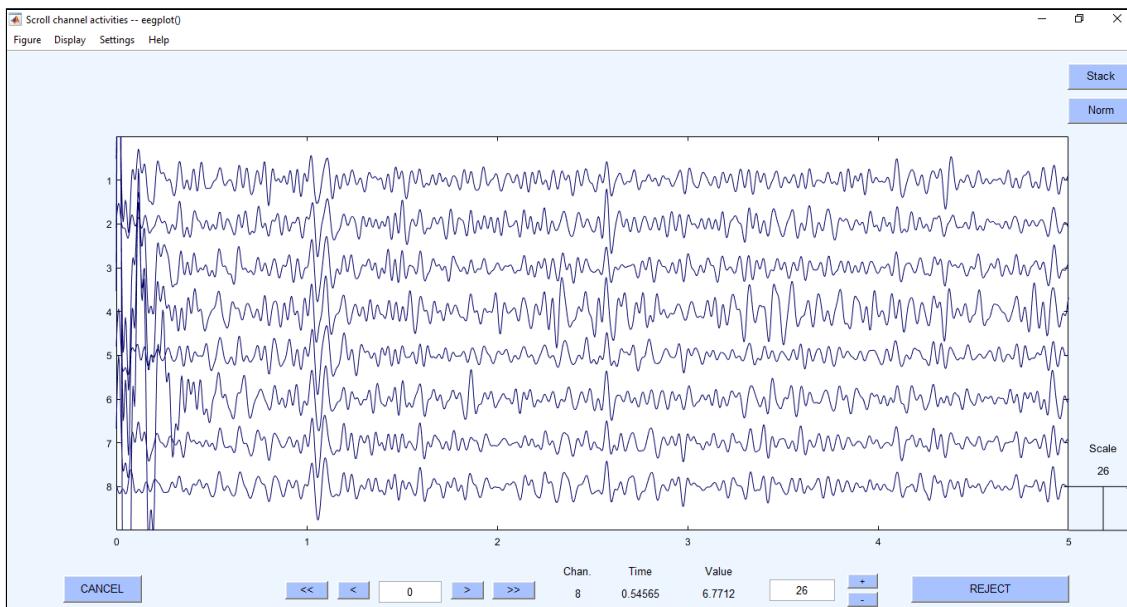
จากการเก็บข้อมูลแต่ละ class เป็นจำนวน class ละ 5 trials เป็นเวลา trial ละ 3 วินาที สามารถแสดงผลข้อมูลที่เก็บได้ ในรูปแบบของกราฟ โดยที่แกนแนวตั้งคือค่าความต่างศักย์ และแกนแนวนอนคือเวลา ออกมากทั้ง 8 channel ได้ดังตัวอย่างในรูป 4.2



รูปที่ 4.2 ข้อมูลสัญญาณ EEG ที่ยังไม่ได้ผ่านการ Filter

4.1.3 ข้อมูลหลังจาก preprocessing

สามารถแสดงผลข้อมูลที่ผ่านการกรองความถี่ผ่านในช่วง 8-30 Hz ในทำนองเดียวกันกับหัวข้อ 4.1.1 ได้ดังตัวอย่างในรูป 4.3



รูปที่ 4.3 ข้อมูลคิบสัญญาณ EEG ที่ผ่านการ Band-pass Filter 8-30 Hz

การ filter ข้อมูลให้เหลือเพียงช่วงความถี่ 8-30 Hz นั้น ทำให้ข้อมูลสัญญาณคิบของคลื่นสมองนั้นสามารถมองเห็นรายละเอียดได้ชัดเจนมากยิ่งขึ้นด้วยตาเปล่า

4.2 ผลการวิเคราะห์ข้อมูล

ตัวอย่างข้อมูลใน array ที่เก็บผลการ classify แสดงในรูป 4.4

	1	2	3	4	5	6	7	8	9	10	11	
1		2	1	1	1	1	1	2	2	2	2	
2												
3												
4												
5												
6												
7												
8												
9												
10												

รูปที่ 4.4 ผลการจำแนกประเภทการจินตนาการ

ผลการ Classify จะแยกประเภทข้อมูลจากทุกๆ Sample ในช่วงเวลาของการจินตนาการ จนนั้นจะทำการเลือกข้อมูลที่มีการถูกแยกประเภทเป็นคลาสนั่นๆ มากที่สุดอ กมาเป็นผลลัพธ์สุดท้ายของการทำนายประเภทข้อมูลนั้น

4.2.1 ผลการทดลองเมื่อนำข้อมูล Training/Testing set มาประเมินผล

ผลลัพธ์จากการทดสอบการจินตนาการการเคลื่อนไหวทั้ง 4 รูปแบบ ทั้งหมด 20 trial (class ละ 5 trial) ได้ผลดังตารางที่ 4.1

ตารางที่ 4.1 ตารางแสดงความแม่นยำในการแยกประเภทของ LDA Classifier ด้วย CSP

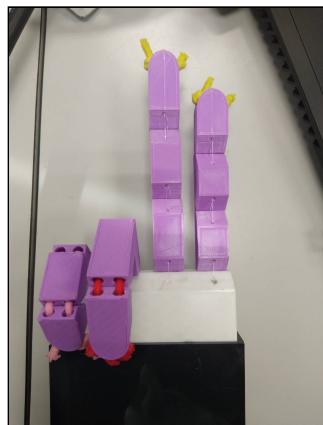
Test Data	Left	Right	Hand	Foot	Result		
					L/R	H/F	True/False
LH	0.36	0.64	0.98	0.02	X	✓	False
RH	0.42	0.58	1.00	0.00	✓	✓	True
LF	0.30	0.70	0.01	0.99	X	✓	False
RF	0.24	0.76	0.08	0.92	✓	✓	True

จากตารางจะเห็นว่าในการจำแนกสัญญาณประสาทที่เกิดจากการจินตนาการ โดยการเลือกคลาสระหว่างมือหรือขา และขาหรือซ้ายแล้วนำผลลัพธ์มาร่วมกันเป็นคำตอบสุดท้ายนั้น Model สามารถจำแนกประเภทของความเป็นขาหรือมือได้ดีกว่า ความเป็นทางซ้ายหรือทางขวา ทั้งนี้ทั้งนั้นอาจจะมีสาเหตุมาจากหลายปัจจัย เช่น

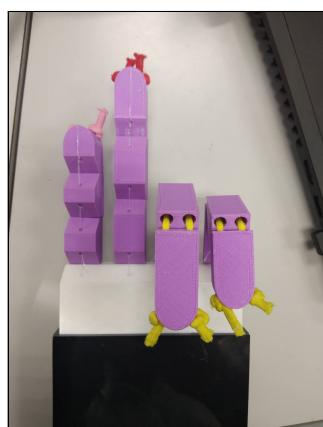
- การจินตนาการการเคลื่อนไหวระหว่างการทดลองนั้นไม่ได้คุณภาพ อาจจะมีการเสียสมາชิกปัจจัยต่างๆ ในห้องทดลอง เช่น เสียงรบกวน
- การติดตั้งอุปกรณ์ตรวจจับสัญญาณ EEG เกิดการหลวมระหว่างการทดลอง เกิดสัญญาณรบกวน หรือได้รับสัญญาณอันไม่เกี่ยวข้องกับการจินตนาการ
- เกิด Artifact จากตัวผู้ทดลอง เช่น การกลืนน้ำลาย การกระพริบตา หรือการขับศีรษะ เป็นต้น

4.2.2 ผลการทดสอบแบบ Real-time และส่งค่าไปควบคุมหุ่นหนวดแบบกล

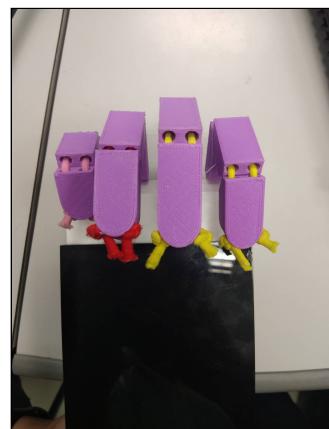
แสดงดังรูปที่ 4.5, 4.6, 4.7 และ 4.8



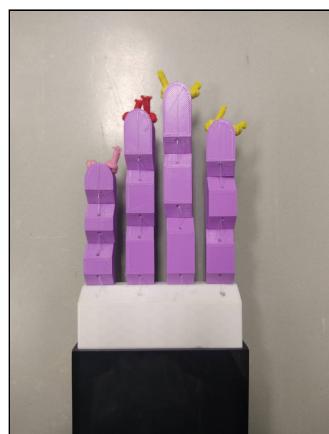
รูปที่ 4.5 รูปแบบนี้ของแขนกลที่ตอบสนองต่อการทำนายว่าผู้ทดลองกำลังจินตนาการมือซ้าย



รูปที่ 4.6 รูปแบบนี้ของแขนกลที่ตอบสนองต่อการทำนายว่าผู้ทดลองกำลังจินตนาการมือขวา



รูปที่ 4.7 รูปแบบนี้ของแขนกลที่ตอบสนองต่อการทำนายว่าผู้ทดสอบกำลังจินตนาการขาซ้าย



รูปที่ 4.8 รูปแบบนี้ของแขนกลที่ตอบสนองต่อการทำนายว่าผู้ทดสอบกำลังจินตนาการขาขวา
จากหลักการทำงานเพื่อทดสอบการทำงานแบบ Real-time ทั้งหมด 20 trial และส่งค่าไป
ควบคุมผลการแสดงได้ผลสรุปดังนี้

ตารางที่ 4.2 Confusion Matrix ของผลการทดสอบด้วยการใช้ Simple CSP

		Prediction			
		Left Hand	Right Hand	Left Foot	Right Foot
Actual	Left Hand	2	7	5	16
	Right Hand	5	9	4	12
	Left Foot	2	4	5	19
	Right Foot	3	8	4	15

จากตารางที่ 4.2 จะเห็นได้ว่า จากจำนวน Trial ที่ไม่เดลได้ทำนายออกมากทั้งหมด 120 Trials (คลาสละ 30 Trials) มีความเอนเอียง (Bias) ไปที่คลาสเท้าขวามากกว่าคลาสอื่นๆ

ตารางที่ 4.3 Accuracy ของผลการทดสอบด้วยการใช้ Simple CSP

Accuracy	0.2583
----------	--------

จากตารางที่ 4.3 ไม่เดลมีความแม่นยำเฉลี่ย 25.83% คำนวณหาได้จาก จำนวน Trial ทั้งหมดที่ทำนายได้ถูกต้อง หารด้วยจำนวน Trial ทั้งหมดที่ได้ทำนาย คุณด้วย 100% ดังสมการ

$$\text{classification accuracy} = \left(\frac{N_{\text{correct}}}{N_{\text{total}}} \right) \times 100\% \quad (4.1)$$

ตารางที่ 4.4 Recall ของผลการทดสอบด้วยการใช้ Simple CSP

Recall			
Left Hand	Right Hand	Left Foot	Right Foot
0.1667	0.3214	0.2778	0.2419

จากตารางที่ 4.4 การวัดค่า Recall ในคลาสแต่ละคลาสจะหาได้จากการนำ Trial ทั้งหมดที่ไม่เดลทำนายถูกในแต่ละคลาส หารด้วยจำนวน Trial ที่ไม่เดลทำนายว่าเป็นคลาสนั้นแต่ความจริงแล้วไม่ใช่

ตารางที่ 4.5 Precision ของผลการทดลองด้วยการใช้ Simple CSP

Precision			
Left Hand	Right Hand	Left Foot	Right Foot
0.1040	0.1147	0.2853	0.848

จากตารางที่ 4.5 การวัดค่า Precision ในคลาสแต่ละคลาสจะหาได้จากการนำ Trial ทั้งหมดที่ไม่เกิดทำนายถูกในแต่ละคลาส หารด้วยจำนวน Trial ทั้งหมดของคลาสนั้น

จากผลการทดลองเมื่อทดสอบการแยกประเภทข้อมูลแบบ real-time พบร่วมกันในตอนนี้นั้น Model มีความ Bias ไปทาง Right Foot เยอะมากจนอยู่ในขั้นที่ไม่สามารถยอมรับได้สำหรับที่คาดว่าจะเกิดขึ้นเป็นครั้งนี้

- Algorithm กระบวนการคำนวณแบบ Real-Time ยังไม่ดีมากพอ
- อุปกรณ์มีความคลาดเคลื่อนในการตรวจวัด
- ความไม่พร้อม หรือขาดประสบการณ์ในการตรวจวัดคลื่นสมองของผู้ทดลอง
- ตำแหน่งในการติดตั้ง Electrode ยังไม่เหมาะสม

เนื่องจากผลการทดลองในการใช้ Feature Extraction ด้วยวิธี CSP แบบชั้นด้วยยังไม่ได้ให้ผลลัพธ์ที่ดีมากพอ พากเราจึงได้นำวิธี FBCSP มาใช้กับข้อมูลชุดเดิม ได้ผลการทดลองดังนี้

ตารางที่ 4.6 Confusion Matrix ของผลการทดลองด้วยการใช้ FBCSP

		Prediction				Precision
		Left Hand	Right Hand	Left Foot	Right Foot	
Actual	Left Hand	15	4	0	11	0.5000
	Right Hand	6	18	1	5	0.6000
	Left Foot	2	4	12	12	0.4000
	Right Foot	3	2	7	18	0.6000
Recall		0.5769	0.6429	0.6000	0.3913	
Overall Accuracy		0.5250				

จากตารางที่ 4.6 ผลการทดลองจากการนำวิธี Feature Extraction ด้วยวิธี Filter Bank Common Spatial Pattern (FBCSP) มาใช้กับข้อมูลชุดเดิมนั้นพบว่าค่า Precision ของคลาสการจินตนาการมีอัตราเฉลี่ย 0.50, ค่า Precision ของคลาสการจินตนาการมีอัตราเฉลี่ย 0.60, ค่า Precision ของคลาสการจินตนาการมีอัตราเฉลี่ย 0.40 และค่า Precision ของคลาสการจินตนาการมีอัตราเฉลี่ย 0.60 และทำให้โมเดลใหม่นั้นมี Accuracy เฉลี่ยเท่ากับ 0.5250 หรือ 52.50%

บทที่ 5

บทสรุป

5.1 ปัญหาที่พบ

5.1.1 ปัญหาของโปรแกรม

ในการ streaming ข้อมูล หากเพิ่งเปิดโปรแกรมใหม่ๆ ค่าที่ได้จะมีความความคลาดเคลื่อน เ酵อะ ในการเก็บข้อมูลสำหรับ train model ผู้จัดทำจึงตัดข้อมูลใน run แรกทิ้ง แล้วใช้ข้อมูลใน run ต่อๆไปแทน

5.1.2 ปัญหาในการ coding

- ผู้จัดทำ EEGLab script สำหรับ import data ไม่สามารถทำให้ฟังก์ชันกำหนดค่าตัวแปร trials เองได้ ซึ่งเป็นค่าที่จำเป็นจะต้องใช้กับฟังก์ชัน filter ต่อ ผู้จัดทำจึงต้องกำหนดค่าตัวแปรด้วยตัวเอง
- ฟังก์ชัน pop_eegfiltnew ซึ่งเป็นฟังก์ชันกรองความถี่ที่ใหม่กว่าและคุณภาพดีกว่า ฟังก์ชัน pop_eegfilt ไม่สามารถทำงานผ่าน EEGLab script ได้ จึงอาจทำให้การกรองความถี่ ไม่ได้คุณภาพเท่าที่ควรนัก
- การส่งค่าไปที่ arduino กินเวลานาน จึงทำให้การ stream ข้อมูลมีความหน่วงจากเดิม

5.1.3 ปัญหาในการเก็บข้อมูล

- สัญญาณสมองของผู้เข้าร่วมการทดลองที่มีผมสัน จะมีสัญญาณรบกวนน้อยกว่าคนที่ผม ยา มาก แต่การทำผู้เข้าร่วมการทดลองที่มีผมสันมาก หรือหัวล้านนั้น ทำได้ยาก ทำให้ทำผู้เข้าร่วมการทดลองได้น้อย
- ข้อมูลที่เก็บจากผู้เข้าร่วมการทดลองในช่วงเวลาที่ต่างกันมีความต่างกัน อาจมาจากปัจจัย ความเห็นอ่อนล้า หรือ ปัจจัยอื่นๆรวมกัน ซึ่งทำให้การใช้ feature ที่ได้จากข้อมูลของผู้เข้าร่วมการทดลองคนหนึ่งมาใช้ classify ข้อมูลของผู้เข้าร่วมการทดลองคนเดิม ในเวลาที่ต่างกัน มีความคลาดเคลื่อนที่เยอะกว่าการใช้ข้อมูลในช่วงเวลาเดียวกันมาก

5.1.4 ปัญหาในการวิเคราะห์ข้อมูล

ถึงแม้ว่าจะได้มีการวาง Electrode ที่ตำแหน่ง Cz เพื่อช่วยในการเพิ่มความต่างระหว่างคลื่นสมองที่เกิดขึ้นจากการจินตนาการว่าขับมือ กับคลื่นสมองที่เกินขึ้นจากการจินตนาการว่าขับเท้า แล้ว แต่ผลที่ได้ยังแยกความต่างระหว่างมือกับเท้าได้ไม่ดีเลย แต่เมื่อนำวิธี Filter Bank Common Spatial Pattern (FBCSP) มาปรับใช้กับข้อมูลชุดเดิมนี้ ทำช่วยให้ลดการ Bias ของโมเดลได้ค่อนข้างมาก และเพิ่ม accuracy ให้กับโมเดลได้ประมาณ 2 เท่า สาเหตุเป็นเพราะว่า Feature ที่ได้รับมาจาก FBCSP นั้นมีความละเอียดที่สูงกว่า CSP แบบธรรมดามาก เพราะมีการคำนวณ Feature จากหลายๆ Time windows และหลายๆ Frequency band นั้นเอง

5.1.5 ปัญหาในการ preprocessing

เนื่องจาก plot กิจกรรมทางสมองในรูปแบบของสีสเปกตรัมบนหัว ด้วย EEGLab ไม่สามารถดูอย่างต่อเนื่องได้ (ทำได้เพียงเลือกชุดเวลาขึ้นมาดู) ทำให้ถึงแม้สัญญาณจะนำสัญญาณไปผ่านการกรองความถี่ที่ไม่ต้องการออก, ตัดอาร์ติเฟคด้วย ICA และทำการ rereference แล้ว แต่ก็ยังไม่เห็นภาพกิจกรรมทางสมองที่สามารถแยกอย่างชัดเจนด้วยตาเปล่าได้

5.2 ข้อเสนอแนะ

- หากเปลี่ยนวิธีเก็บข้อมูล เป็นการเก็บค่าพาพิกัดกิจกรรมทางสมองในรูปแบบสเปกตรัมในทุกๆ sample พร้อมกับผ่านการทำ preprocessing แล้วแสดงผลเป็นภาพ animation แบบ realtime เช่นเดียวกับที่ GUI ของ OpenBCI แสดงออกมานา อาจจะสามารถใช้เครื่องมือหา feature หรือ classify ตัวอื่นๆ ได้ในรูปแบบที่มีประสิทธิภาพมากขึ้น
- ในการทดลองครั้งนี้ ผู้จัดทำไม่ได้ลองใช้ classifier หรือ เครื่องมือสำหรับ feature extraction หลายๆตัว จึงอาจมีการลองใช้เครื่องมือหลายๆตัวเทียบดูเพื่อความแม่นยำที่มากขึ้น
- การควบคุมอุปกรณ์ควรจะทำได้ real time มากกว่านี้ กล่าวคือ สามารถป้อนข้อมูลได้ตลอดเวลา เพื่อให้สามารถบังคับอุปกรณ์ได้อย่างต่อเนื่อง ซึ่งในการทดลองครั้งนี้ ยังจำเป็นต้องใช้สิ่งกระตุ้นอยู่เนื่องจากผู้จัดทำเล็งเห็นว่ามันเหมาะสมกับความเห็นอย่างส่วนมากกว่า จึงจำเป็นต้องหาทางพัฒนาระบบท่อไป
- หากสามารถหาจุดสมดุลระหว่างจำนวนครั้งในการเก็บข้อมูลสำหรับการ train กับความแม่นยำและความเห็นอย่างส่วนมากของผู้เข้าร่วมการทดลองได้ อาจจะช่วยลดระยะเวลาการเก็บข้อมูลและพัฒนาระบบที่ควบคุมอุปกรณ์อย่างต่อเนื่องกว่าเดิมได้

- เนื่องจากเยื่อหุ้มสมองด้านนอกส่วนที่เกี่ยวกับการขับขาไปจนถึงเท้านั้นอยู่ลึกกลงไปในระหว่างรอยต่อของสมองทั้งสองซีก และส่วนที่เกี่ยวกับการขับมือก็ไม่ได้ไกลจากตำแหน่งนั้นมาก จึงอาจทำให้สัญญาณระหว่างการจินตนาการว่าขาบ้มือกับขาเท้านั้น ไม่ต่างกันมากนัก ซึ่งในบางงานวิจัยที่ทำการแยกระหว่างสัญญาณสมอง 4 class จะใช้การจินตนาการว่าขาลีน กับการขับขาทั้งสองข้างแทนเนื่องจากตำแหน่งที่เกิดกิจกรรมทางสมอง ค่อนข้างอยู่ระหว่างกัน ทั้งนี้เพื่อพัฒนาระบบ อาจสามารถเปลี่ยนอวัยวะที่ใช้อยู่ปัจจุบัน เป็นอวัยวะส่วนอื่นแทนได้

បររណាណុក្រម

- [1.] Sjoerd de Vries and Theo Mulder(2007), Motor Imagery and Stroke Rehabilitation: A Critical Discussion, J Rehabil Med; 39: 5–13.
- [2.] C. Brunner, R. Leeb, G. R. Muller-Putzl, A. Schlogl and G. Pfurtscheller(2008), BCI Competition 2008 – Graz data set A, Institute for Knowledge Discovery, Graz University of Technology, Graz, pp. 1-6
- [3.] (2018, November 25). A Beginner's Guide to Brain-Computer Interface and Convolutional Neural Networks. Retrieved from
<https://towardsdatascience.com/a-beginners-guide-to-brain-computer-interface-and-convolutional-neural-networks-9f35bd4af948>
- [4.] Motor imagery. (2019, April 04). Retrieved from https://en.wikipedia.org/wiki/Motor_imagery
- [5.] Support-vector machine. (2019, June 13). Retrieved from
https://en.wikipedia.org/wiki/Support-vector_machine
- [6.] Linear discriminant analysis. (2019, June 09). Retrieved from
https://en.wikipedia.org/wiki/Linear_discriminant_analysis
- [7.] Common spatial pattern. (2019, May 15). Retrieved from
https://en.wikipedia.org/wiki/Common_spatial_pattern
- [8.] M. Hersche, T. Rellstab, P. D. Schiavone, L. Cavigelli, L. Benini and A. Rahimi, "Fast and Accurate Multiclass Inference for MI-BCIs Using Large Multiscale Temporal and Spectral Features," in 2018 26th European Signal Processing Conference (EUSIPCO), Roma, Italy, 2018.
- [9.] Your books didn't tell you this about brain! (2016, May 24). Retrieved from
<https://www.askiitians.com/blog/books-didnt-tell-brain/>
- [10.] BrainWave : គណន៍សមង់ និងប្រព័ន្ធលីខ្មែន់ ធូលឱវត្ត. (n.d.). Retrieved from
<http://nuclear.rmutphysics.com/blog-sci7/?p=3878>
- [11.] Brain–computer interface. (2019, May 11). Retrieved from
https://en.wikipedia.org/wiki/Brain–computer_interface
- [12.] Auditory & Language Neuroscience Lab. (n.d.). Retrieved from
<https://sites.uci.edu/alns/projects-2/electrocorticography-ecog/>
- [13.] Task Group 7: Brain Computer Interface for hearing devices. (n.d.). Retrieved from
<https://hearing4all.eu/EN/Research/C/Task-Group-7.php>

- [14.] Electroencephalography. (2019, May 08). Retrieved from
<https://en.wikipedia.org/wiki/Electroencephalography>
- [15.] Swartz Center for Computational Neuroscience (SCCN)(2014), “Chapter 04: Preprocessing Tools”, www.sccn.ucsd.edu, เข้าถึงได้จาก: https://sccn.ucsd.edu/wiki/Chapter_04:_Preprocessing_Tools, [คืนหนาเมื่อ: 2 ธ.ค. 2561]
- [16.] Sawla, S., & Sawla, S. (2018, June 05). Linear Discriminant Analysis. Retrieved from
<https://medium.com/@srishtisawla/linear-discriminant-analysis-d38decf48105>
- [17.] Confusion Matrix. (2014, July 01). Retrieved from
<https://plagad.wordpress.com/2010/08/26/confusion-matrix/>
- [18.] Nopparat. (n.d.). Matlab (แมตแล็บ) คืออะไร เป็นภาษาคอมพิวเตอร์ระดับสูง. Retrieved from
<https://www.mindphp.com/คุณมีอ/73-คืออะไร/4043-what-is-matlab.html>
- [19.] OpenBCI. (2017, January 29). OpenBCI/OpenBCI_MATLAB. Retrieved from
<https://github.com/OpenBCI/OpenBCI>
- [20.] OpenBCI. (n.d.). Retrieved from <https://docs.openbci.com/Hardware/02-Cyton>
- [21.] OpenBCI. (n.d.). Retrieved from
<https://docs.openbci.com/Headware/01-Ultracortex-Mark-IV>
- [22.] http://www.appsofttech.com. บอร์ด i-Duino UNO. Retrieved from http://www.appsofttech.com/store/product/view/บอร์ด_i_Duino_UNO_เป็น_Arduino_UNO_compatible-27361361-th.html