

Bachelor's Thesis

Development of Neural-based Control System for 6-DoF Robotic Arm

B.Eng. Electrical Engineering and Information Technology

Author

Tuan Minh Le, 1519561

Supervisors

Prof. Dr. Peter Nauth
Sudeep Sharan

02.09.2025

Abstract

This thesis presents the development of a Brain-Computer Interface (BCI) for controlling a 6 Degrees of Freedom (DoF) robotic arm in real time using neural signals. In this system, an OpenBCI Cyton biosensing board collects Steady-State Visual Evoked Potentials (SSVEPs), one kind of Electroencephalogram (EEG) signal, from a user when they are looking at 1 amongst 9 visual stimulation sources then transmits the collected data to a computer for preprocessing and translation into corresponding desired configurations for the robotic arm with Canonical Correlation Analysis (CCA). Therefore, the computer uses these desired configurations to calculate desired joint angles based on Product of Exponentials (PoE) and Newton-Raphson Method, then actuates the robotic arm with these results. The communication between the biosensing board, the computer and the robotic arm is established based on Robot Operating System (ROS 2) framework. The system was tested carefully, showing promising findings regarding its accuracy, as well as detecting its limitations for future improvements.

Keywords— Electroencephalogram (EEG), Steady-State Visual Evoked Potentials (SSVEPs), Brain-Computer Interface (BCI), Canonical Correlation Analysis (CCA), ROS 2, Product of Exponentials (PoE), Newton-Raphson

Acknowledgement

First and foremost, I would like to express my sincere gratitude to Prof. Dr. Peter Nauth and Mr. Sudeep Sharan for their continuous guidance, expertise, and valuable feedback throughout the development of this thesis. Their support has been instrumental in shaping both the technical and academic aspects of this work. Furthermore, I am grateful to the Autonomous Systems and Intelligent Sensor Technology Lab at Frankfurt University of Applied Sciences for providing access to the robotic arm platform and necessary tools that made this research possible. Lastly, I would like to thank my friends who supported me unconditionally in conducting testing experiments: Gia Khanh Nguyen, Thanh Nguyen, Phuong Bao Nguyen, Kien Giang Dao, The Duong Tran and Gia Bao Luong.

Plagiarism Declaration

With the exception of any statement to the contrary, all the material presented in this report is the result of my own efforts. In addition, no parts of this report are copied from other sources. I understand that any evidence of plagiarism and/or the use of unacknowledged third party materials will be dealt with as a serious matter.

A handwritten signature in black ink, appearing to read "Tuan Minh Le".

Groß-Gerau, 02.09.2025, Tuan Minh Le

Contents

List of Figures	6
List of Tables	7
Abbreviations	8
1 Introduction	9
1.1 Background and Motivation	9
1.2 Objectives	10
1.3 Thesis Structure	10
2 Methodology Overview	11
2.1 Introduction of ROS 2	11
2.2 System Architecture and Workflow	12
3 SSVEP Acquisition-Translation	14
3.1 Concept Overview	14
3.2 Hardware Specifications	15
3.2.1 OpenBCI Cyton Board	15
3.2.2 Gold Cup Electrodes	16
3.2.3 Signa Gel	16
3.3 Stimulation Design	17
3.4 Electrode Placement	17
3.5 Signal Preprocessing	18
3.6 Signal Translation	20
4 Inverse Kinematics of Robotic Arm System	23
4.1 Concept Overview	23
4.2 Hardware Specifications	25
4.2.1 Dynamixel Servo Motors	25
4.2.2 U2D2 and U2D2 Power Hub	26
4.3 Kinematic Modelling	26
4.3.1 Denavit-Hartenberg Parameters	26
4.3.2 Home Configuration Matrix	28
4.3.3 Space Screw Axes	28
4.4 Newton-Raphson Method's Algorithm	29
5 Testings and Discussions	34
5.1 Testing on Steady-State Visual Evoked Potential (SSVEP) Acquisition-Translation Process	34
5.1.1 Introduction of Self-Made Dataset	34
5.1.2 Introduction of 12JFPM Dataset	36
5.1.3 Testing Results	36
5.1.4 Discussion	39
5.2 Testing on Inverse Kinematics Process	40
5.2.1 Introduction of Testing Region	41
5.2.2 Testing Results	41
5.2.3 Discussion	43
6 Conclusion and Future Work	45

List of Figures

1	The 6-Degrees of Freedom (DoF) robotic arm at FRA-UAS.	9
2	Workflow of the system.	13
3	Flow chart of SSVEP Acquisition-Translation Process.	14
4	OpenBCI Cyton biosensing board.	15
5	Gold cup electrodes.	16
6	Signa gel.	16
7	Stimuli with different frequencies created by Quick SSVEP.	17
8	Reference and ground electrodes at earlobe and forehead.	17
9	Placement of 8 channel electrodes in 10-20 system. [14]	18
10	Power Spectral Density (PSD) and Signal-to-Noise Ratio (SNR) plots of SSVEPs before and after Common Average Reference (CAR).	18
11	Example of data preprocessed with CAR.	19
12	PSD and SNR plots of SSVEPs before and after bandpass filtering.	20
13	Analysis concept of Canonical Correlation Analysis (CCA). [15]	20
14	Flow Chart of Inverse Kinematics.	24
15	MX-28, MX-64 and MX-106 motors.	26
16	U2D2 (left) and U2D2 power hub (right).	26
17	Coordinate Frame System of The Robotic Arm.	27
18	OpenBCI Cyton setup for the experiment.	34
19	Connection between electrodes and the volunteer's scalp.	35
20	Alienware AW3225DM curved monitor.	35
21	Visual stimulation sources used in Nakanishi's experiment. [21]	36
22	PSD and SNR plots of 7 used visual stimuli.	37
23	Plots of translation accuracy and Information Transfer Rate (ITR) versus epoch duration.	38
24	BrainWave EEG cap of DUOMED.	40
25	Specifications of the testing region.	41
26	The end-effector is commanded to move to index (8, 8), it receives joint angles from the COMPUTATION_NODE and moves to the correct position.	41
27	The arm is tilted when reaching index (8, 8), which is far away from origin.	44

List of Tables

1	Software and hardware components of the system.	11
2	OpenBCI Cyton Board Specifications and Descriptions.	15
3	The robotic arm's rotating constraints.	23
4	Specifications of servo motors of 6 joints.	25
5	Motor Register Addresses and Their Descriptions	25
6	DH Parameter Table.	27
7	Values of translation accuracy, ITR and number of epochs versus epoch duration. .	38
8	Average correlation of target and largest non-target within correct translations. .	39
9	Target and measured positions within testing region.	43

Abbreviations

BCI Brain-Computer Interface

SSVEP Steady-State Visual Evoked Potential

EEG Electroencephalogram

ITR Information Transfer Rate

ROS 2 Robot Operating System 2

DDS Data Distribution Service

DoF Degrees of Freedom

DH Denavit–Hartenberg

PoE Product of Exponentials

NR Newton-Raphson

CCA Canonical Correlation Analysis

SNR Signal-to-Noise Ratio

CAR Common Average Reference

PSD Power Spectral Density

1 Introduction

1.1 Background and Motivation

Brain-Computer Interface (BCI) technology has recently attracted the attention of the scientific community with its potential to assist the elderly or people with motor impairments. BCI is an advanced communication system that uses neural signals to operate external devices, enabling users to interact with their surroundings through non-physical means. BCI is classified into 2 types: non-invasive systems, which utilize signals that could be acquired from the scalp surface by an external device, and invasive systems, which utilize signals inside the scalp [1]. Although non-invasive BCIs show lower accuracy compared with the invasive ones, they are still currently more preferred and widely used because of their safety, ease of use, and lower cost, since they do not require implantation of electronic devices through human scalp [2].

Given the context where assistive devices are lacking¹ and the worldwide elderly population will reach approximately 1.5 billion by 2050 [3], continuous researches on SSVEP-based BCI technologies (Steady-State Visual Evoked Potential) and robotics are particularly necessary to provide people with better living standards. Some notable researches on these fields have been conducted in recent years, such as a soft robotic glove for post-stroke hand function rehabilitation [4], a robotic wheelchair [5] and a 6-DoF robotic arm [6]. In order to contribute to the growth of these technologies, this project develops a SSVEP-based BCI that control a 6-DoF robotic arm in real time, through that obtain valuable findings on the use of neural signals in assistive robotic applications for future work of Frankfurt University of Applied Sciences (FRA-UAS).

Among all neural signals that could be acquired non-invasively, Electroencephalogram (EEG) signals are widely utilized in a number of robotics researches [7][8] since they are easy to acquire, comparatively low-cost, and have decent signal precision. EEG signals are recorded via electrodes placed on the scalp, measuring voltage potentials that reflect electrical brain activity [9]. These signals convey user's intent and could be translated into commands to operate external devices. One of the most commonly used EEG signal is SSVEP, which has showed the feasibility of practical applications [10][11]. SSVEPs are periodic neural signals produced while a person is looking at a repetitive visual stimulation. These signals always match the frequency or harmonics of that stimulation, enabling computers to analyze and translate them into associated commands [12, p. 241].

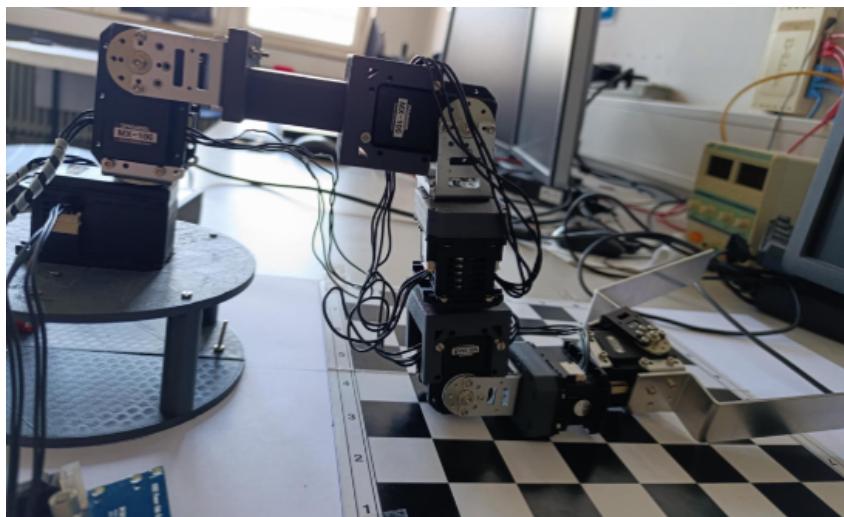


Figure 1: The 6-DoF robotic arm at FRA-UAS.

¹<https://news.un.org/en/story/2022/05/1118212>

1.2 Objectives

The objectives of this project include:

- Developing a SSVEP acquisition-translation process for recording and translating SSVEP signals into desired configurations for the robotic arm.
- Developing a Inverse Kinematics process for solving suitable joint angles that bring the robotic arm's end-effector to a desired configuration.
- Integrating the 2 processes based on Robot Operating System 2 (ROS 2) framework to operate the system.
- Testing the operation of the neural-based control system on the robotic arm of FRA-UAS in practice.

1.3 Thesis Structure

Based on aforementioned objective, this report is constructed as follows.

- **Section 2** describes the methodology of the system, including its node architecture in ROS 2 framework and the entire workflow from collecting SSVEP signals to moving the end-effector to a desired configuration.
- **Section 3** presents the SSVEP acquisition-translation process. It discusses the hardware specifications of used devices, experiment setup for signal acquisition, and the CCA algorithm for signal translation in detail.
- **Section 4** focuses on the Inverse Kinematics process. It covers the hardware specifications of the robotic arm, kinematic modelling for the system, and explanation for the usage of Product of Exponentials (PoE) and Newton-Raphson method in solving suitable joint angles.
- **Section 5** provides testing results on the SSVEP acquisition-translation and Inverse Kinematics processes, followed by the discussions on the obtained results.
- **Section 6** concludes the thesis.

2 Methodology Overview

The BCI system in this project is structured around 2 main components: SSVEP acquisition-translation process and inverse kinematics process. ROS 2 Humble² framework is utilized to construct a communication system that connect these 2 processes and the robotic arm together. The source codes for the system are written in Python language. Important software tools/libraries and hardware components used in this project are listed in Table 1.

Software components	ROS 2 Humble, BrainFlow, MNE-Python, NumPy, Scikit-learn, Quick SSVEP.
Hardware components	OpenBCI Cyton, gold cup electrodes, Signa gel, Dynamixel servo motors, U2D2 USB, U2D2 power hub.

Table 1: Software and hardware components of the system.

2.1 Introduction of ROS 2

Robot Operating System 2 (ROS 2) is a framework containing several useful software libraries and tools that support development of robotic systems. ROS 2 provides standardized communication protocols and interfaces, thereby enabling the reuse of existing code programs from their previous projects and ROS 2 open-source contributions to improves development efficiency.

In ROS 2, real devices are represented by nodes. They communicate with each other using the Data Distribution Service (DDS) protocol³, forming a decentralized architecture where nodes can exchange data peer-to-peer. There are three methods of communication in ROS 2.

- **Topic:** Nodes (publishers) transmit data to a specific topic and other nodes which are subscribers of that topic will receive transmitted data. This method is simple and easy to use but only supports one-way communication.
- **Service:** Nodes (service clients) can send data as a request to a specific node (service server) via a service. The service server will perform some predefined tasks then send data as response back to their clients.
- **Action:** This method is similar to service but is more suitable a long running tasks, as the action server updates action clients on progress of on-going tasks. Besides, action clients can also terminate on-going tasks at any time.

Considering the standardized, decentralized architecture and reusability of ROS 2, this project develops a communication system based on ROS 2 framework to connect SSVEP acquisition-translation process, the inverse kinematics process and the robotic arm together in order to operate the system.

²<https://docs.ros.org/en/humble/index.html>

³https://design.ros2.org/articles/ros_on_dds.html

2.2 System Architecture and Workflow

There are 4 ROS 2 nodes used in this system. The first node is **BCI_NODE**, which contains the acquisition-translation process, responsible for converting SSVEP into desired configuration. The second node is **COMPUTATION_NODE**, which contains the Inverse Kinematics process, responsible for calculating the suitable joint angles for the motors to move the end-effector to the desired configuration. The third node is **MOTOR_NODE**, responsible for rotating the motors. The last node is **USER_NODE**, which is the central node of the system, responsible for receiving and transmitting information between 3 aforementioned nodes. The workflow of the system is described as follows, with its illustration (Figure 2) in the next page.

1. First, a monitor is used for displaying 9 flickering boxes at different frequencies as visualization sources. A test subject will fix their gaze at a box corresponding to their desired configuration, then an EEG-headset formed by a OpenBCI Cyton ⁴ biosensing board and electrodes are used to collect and digitalize SSVEPs before transmitting these digitalized SSVEPs to **BCI_NODE** via Bluetooth and BrainFlow ⁵ library.
2. At **BCI_NODE**, supporting tools from MNE-Python ⁶ and scikit-learn ⁷ libraries are utilized to improve the signal. The received SSVEPs then go through a digital bandpass filter to discard signal components that do not fall into the frequency range 13.5-18.5 Hz to minimize noise and interference of the signal. Common average reference CAR technique is also applied to reduce the influence of common noise across all channels. The preprocessed SSVEPs are therefore translated into their associated configuration by CCA technique.
3. To ensure that the desired configuration has been correctly translated, the **BCI_NODE** will publish a configuration to the topic **CONFIG** only after it has detected that configuration 3 times consecutively.
4. The **USER_NODE** subscribes to that topic and receives the published configuration then transmits it to the **COMPUTATION_NODE**. At the **COMPUTATION_NODE**, the Inverse Kinematics process will utilize PoE and Newton-Raphson (NR) method to compute suitable joint angles for Dynamixel servo motors ⁸ of the robotic arm. These angles are then converted into integer in the range 0-4095, corresponding to 0-359.012 degrees, and are transmitted back to the **USER_NODE**.
5. The **USER_NODE** continue transmitting these integer to the **MOTOR_NODE** to rotate the robotic arm with DYNAMIXEL SDK ⁹ package. The final angles of all motors after these rotations will be transmitted back to the **USER_NODE** to inform the user.

⁴<https://docs.openbci.com/Cyton/CytonSpecs/>

⁵<https://brainflow.readthedocs.io/en/stable/>

⁶<https://mne.tools/stable/index.html>

⁷<https://scikit-learn.org/stable/>

⁸<https://emanual.robotis.com/docs/en/dxl/>

⁹https://emanual.robotis.com/docs/en/software/dynamixel/dynamixel_sdk/overview/

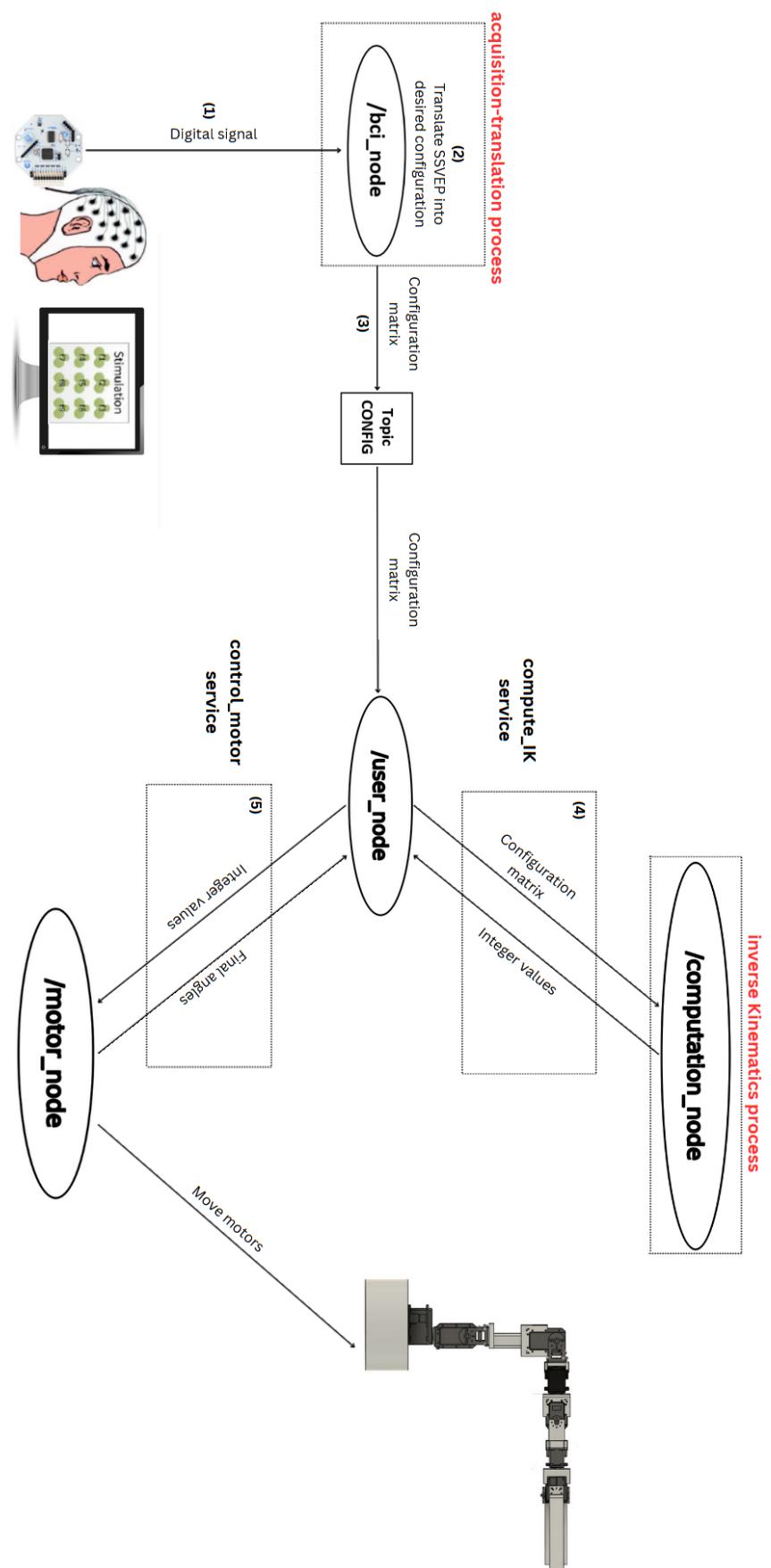


Figure 2: Workflow of the system.

3 SSVEP Acquisition-Translation

3.1 Concept Overview

The SSVEP acquisition-translation is the core process of the BCI system developed in this project. Its primary function is to acquire Steady-State Visual Evoked Potentials (SSVEPs) from the user's scalp and translate them into desired configurations for the robotic arm. The process comprises several sequential stages: signal acquisition, signal preprocessing, and signal translation.

The process begins with the presentation of repetitive visual stimuli in the form of flickering boxes on a screen, each assigned a unique frequency and corresponding to a specific desired configuration. When the user focuses on looking at a specific box, their brain generates periodic neural responses that match the flickering frequency of that stimulus. These SSVEP signals are recorded through multiple electrodes positioned on the occipital region of the scalp using the OpenBCI Cyton biosensing board.

The recorded SSVEP signals are then transmitted wirelessly to a computer via Bluetooth. Upon reception, the signals are digitally processed to remove noise and extract frequency-domain features. Bandpass filtering and spatial filtering techniques such as Common Average Reference (CAR) are employed to enhance the signal-to-noise ratio. Subsequently, Canonical Correlation Analysis (CCA) is applied to compare the preprocessed signals against reference sine-cosine waveforms corresponding to the stimulus frequencies. The frequency yielding the highest correlation is selected as the target frequency, then the desired configuration is detected.

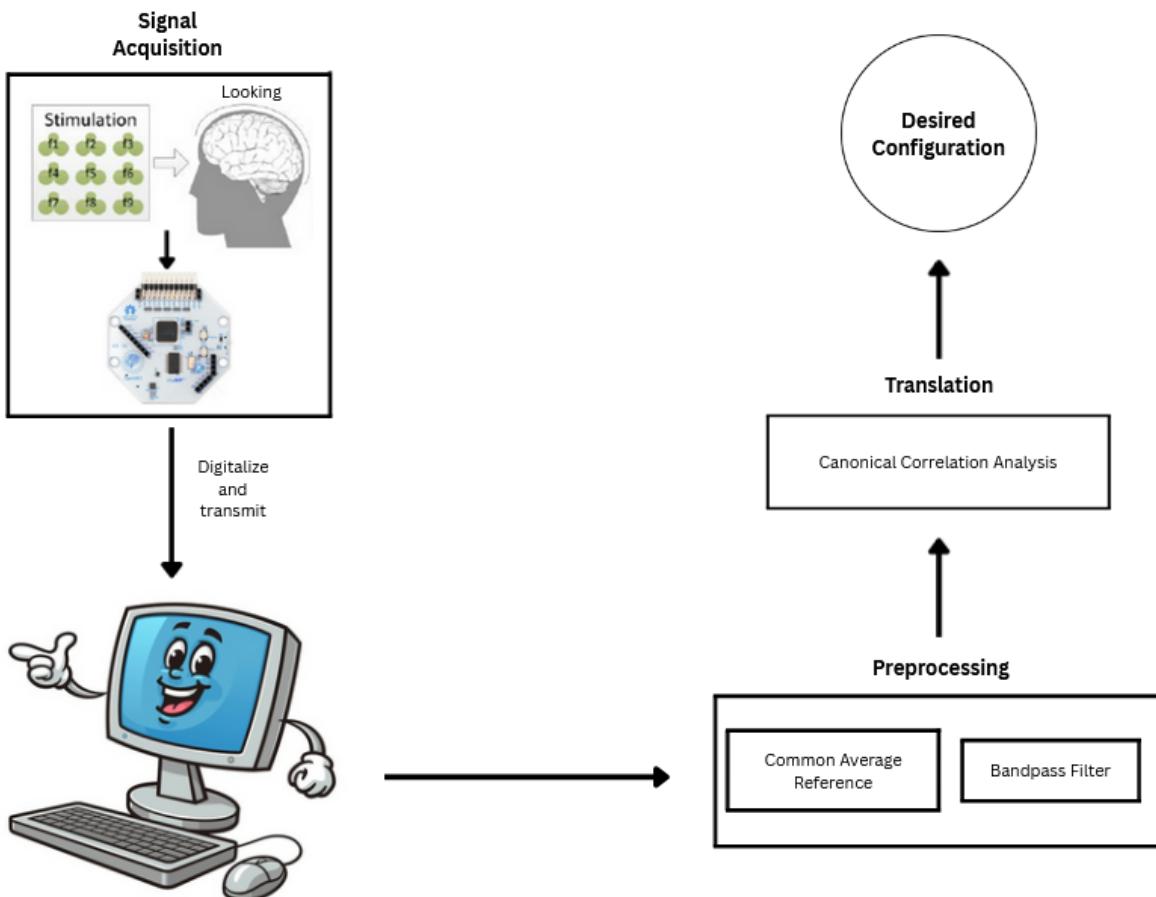


Figure 3: Flow chart of SSVEP Acquisition-Translation Process.

3.2 Hardware Specifications

3.2.1 OpenBCI Cyton Board

OpenBCI Cyton is a compact and cost-effective biosensing board that supports acquisition of bio-signal such as EEG. Analog signals could be acquired from up to 8 channels and transmitted as digital signals to other devices wirelessly via Bluetooth. Specifications of the Cyton board are described in the Table 2.

Specification	Description
Power Supply	Operates within 3–6 V, DC battery only.
Microcontroller	PIC32MX250F128B with chipKIT UDB32-MX2-DIP bootloader.
Measurement Voltage Range	Capable of measuring signals within +2.5 V to -2.5 V.
Analog-Digital Converter	ADS1299 – a 24-bit ADC designed for biopotential measurements and noise filtering.
Sampling Rate	250 Hz
Voltage Resolution	0.298 μ V/bit – smallest detectable change based on the formula $5/2^{24}$ (Voltage Level / 2^{bits}).
Wireless Communication	RFduino RFD22301 module for Bluetooth low-energy transmission.
Programmable Gain	1, 2, 4, 6, 8, 12, 24
Signal-to-Noise Ratio	121 dB – indication of board's signal quality, which is near to devices tailored to medical applications.

Table 2: OpenBCI Cyton Board Specifications and Descriptions.

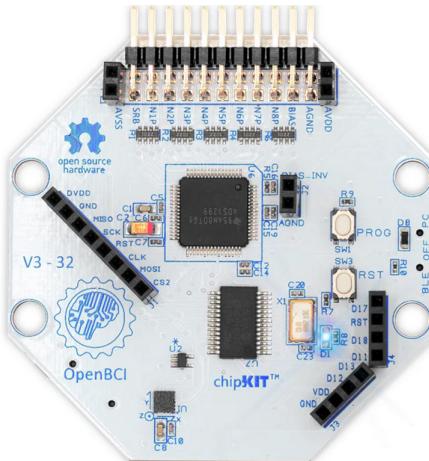


Figure 4: OpenBCI Cyton biosensing board.¹⁰

¹⁰<https://shop.openbci.com/products/cyton-biosensing-board-8-channel>

In the context of this project, analog SSVEP signals recorded from 8 channels are converted into digital format by the ADS1299 ADC and then transmitted as packages to a computer by the RFduino RFD22301 module via Bluetooth. Each transmitted package has 31 bytes, including a header (2 bytes), signal values recorded from 8 channels at one time point (24 bytes in total - 3 bytes for each channel), auxiliary data (6 bytes) and a footer (1 byte). A USB Dongle is usually plugged in the computer to enable this Bluetooth communication achieve the highest data rates.

3.2.2 Gold Cup Electrodes

Gold cup electrodes are widely used in experiments and projects that require bio-signal acquisition due to their low cost, excellent signal quality and reliable skin contact. The electrodes consist of a 24-karat gold-plated silver disc with an approximate diameter of 10 mm.



Figure 5: Gold cup electrodes.¹¹

3.2.3 Signa Gel

Signa Gel is a medical gel used to improve the conductivity of electrodes, optimizing signal transmission from a scalp to a computer. It is bacteriostatic, water-soluble and easy to clean, making it safe and hygienic for experiments on the scalp.



Figure 6: Signa gel.¹²

¹¹<https://www.ternimed.de/Gold-Cup-electrode-with-Teflon-or-Silicone-cable>

¹²<https://www.ternimed.de/Signa-gel-Elektrodengel>

3.3 Stimulation Design

It has been proved that experiments with SSVEPs provide best results when analysing frequencies within the range 12 to 18 Hz [13]. Therefore, 9 stimulation frequencies used in this project fall within the range of 14 to 18 Hz, incremented by 0.5 Hz. The step size of 0.5 Hz is expected to ensure different stimuli are able to be distinguished by the translation algorithm.

Because of the project's time limitation, Quick SSVEP, a web-based application is used to establish stimulation sources. Stimuli are displayed on a monitor screen as a 3x3 matrix, as shown in Figure 7.

14	17.5	15
16.5	18	14.5
16	15.5	17

Figure 7: Stimuli with different frequencies created by Quick SSVEP.

3.4 Electrode Placement

In order to collect SSVEPs, 10 electrodes are used in total: 1 reference electrode, 1 ground electrode, and 8 electrodes for 8 channels. While the reference electrode serves as a baseline voltage level to calculate differential voltages between itself and each channel electrode, the ground electrode does not measure signals but helps reduce common-mode noise. They are respectively placed at the user's right earlobe and forehead.



Figure 8: Reference and ground electrodes at earlobe and forehead.¹³

¹³<https://docs.openbci.com/GettingStarted/Biosensing-Setups/EEGSetup/>

Meanwhile, the remaining 8 channel electrodes are positioned at Oz, O1, O2, POz, PO3, PO4, PO5, PO6 in the 10-20 system, highlighted in Figure 9, to collect the most relevant signals, since these positions are strongly associated with the primary visual cortex, which produces SSVEPs.

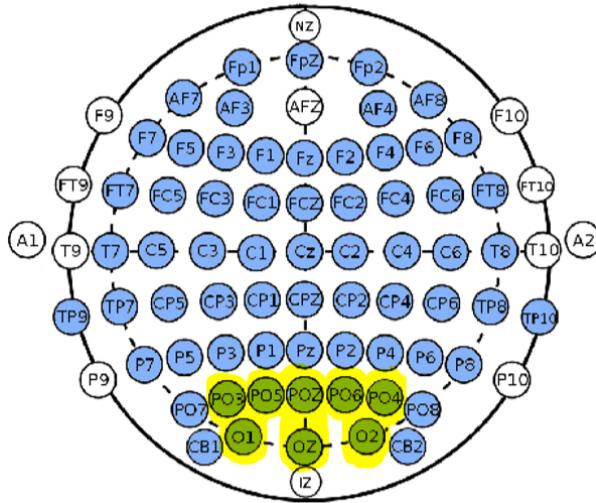


Figure 9: Placement of 8 channel electrodes in 10-20 system. [14]

3.5 Signal Preprocessing

After digitalized SSVEP data are transmitted to the computer, they need to be conditioned before analysing to remove external noise and physiological artifacts as well as enhance signal quality. There are 2 main techniques are used: CAR and frequency-range prefiltering.

First, CAR technique simply computes the global mean of 8 channels at each time point and subtracts that mean from the data value of each channel at that time point. This technique helps reduce the impact of interference which is common across 8 channels. Figure 10 shows the plots of PSD, which represents how strong the frequency power is, and SNR, which represents the difference between the frequency itself with neighboring frequencies, before and after being preprocessed by CAR. It can be observed from the plots that the frequencies of interest (12 and 15 Hz) has become more distinguishable after being preprocessed with CAR, as the SNR of 12 and 15 Hz from about 3 to near 10.

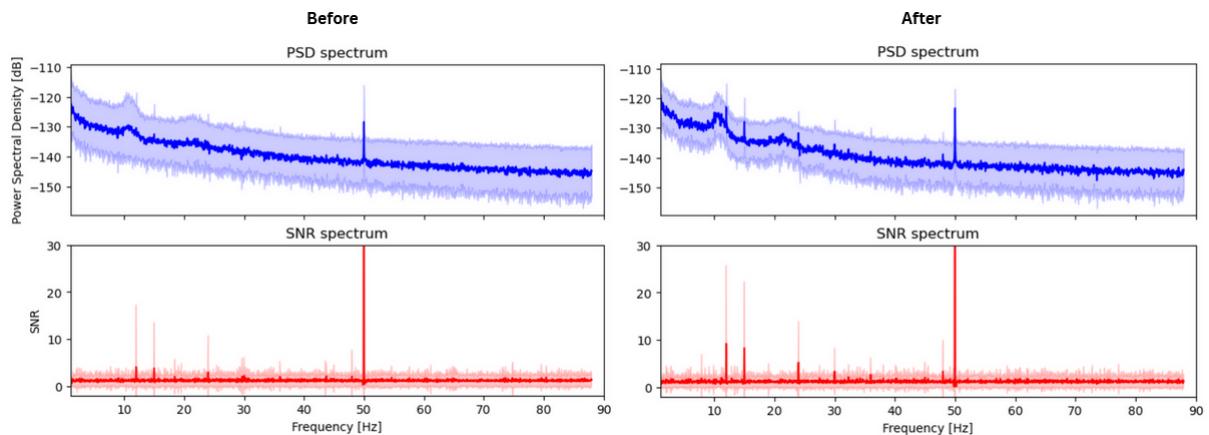


Figure 10: PSD and SNR plots of SSVEPs before and after CAR.

In Figure 11, an example of data preprocessed with CAR is illustrated. As shown in the figure, the average of the first sampling point of all channels (red boxes) is calculated, which is 2. Then, this average value is subtracted from the first sampling point of each channel, creating new values for these points (blue boxes). This process is repeated for all sampling points.

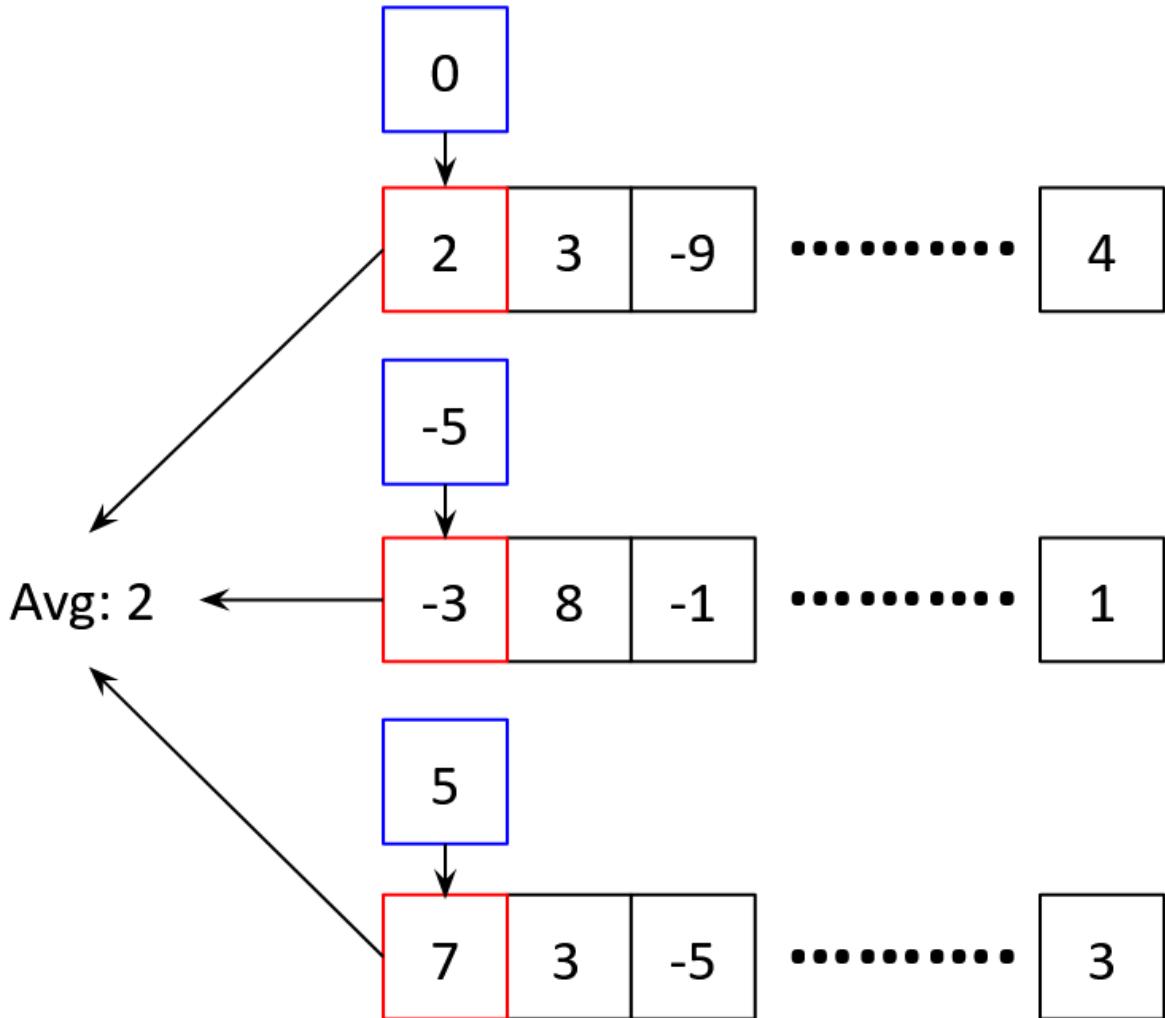


Figure 11: Example of data preprocessed with CAR.

After that, the SSVEP signal will go through a bandpass filter with the range between 13.5 and 18.5 Hz to help the algorithm focus better on 9 used frequencies. This range only covers the fundamental frequencies, not also their harmonics, because no significant improvement was observed when analysing broader bandwidth compared with this range. Figure 12 shows the plots of PSD and SNR of the signal before and after going through a bandpass filter. It can be observed that although the PSD of the focused frequencies decreased a little, they are still significantly higher than the others. As a result, the SNR of the target frequency (15 Hz) increases from about 2 to around 5, promising better accuracy of the translation algorithm which will be presented later.

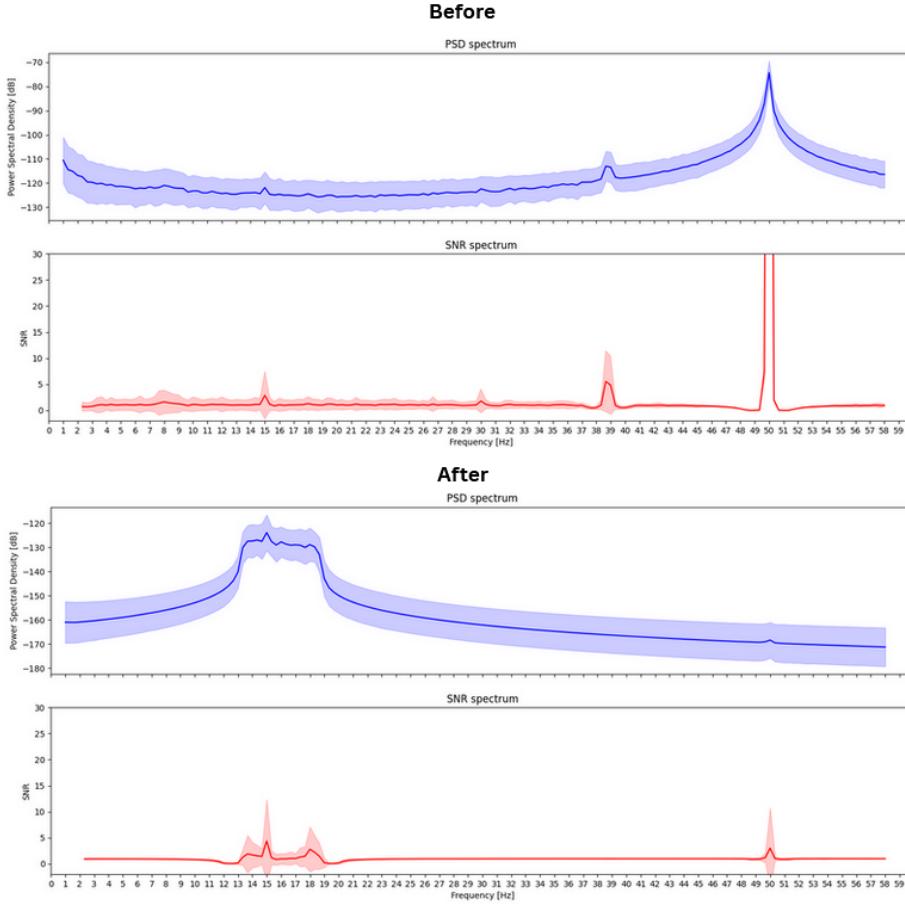


Figure 12: PSD and SNR plots of SSVEPs before and after bandpass filtering.

3.6 Signal Translation

Canonical Correlation Analysis (CCA) is the technique used in this project to directly translate SSVEP into desired configuration for the robotic arm (the position and orientation of the end-effector that we want it to achieve), without any training required. The main concept of CCA technique is illustrated in Figure 13. This technique find 2 set of weights $w_x \in \mathbb{R}^{8 \times 1}$ and $w_y \in \mathbb{R}^{2 \times 1}$ for these 2 sets of variables that maximize the correlation between the 2 overall SSVEP $x = X^T w_x$ and $y = Y^T w_y$ with respect to w_x and w_y . By applying this technique on each frequency of visual stimulus, target frequency could be detected by comparing correlations and choosing the largest amongst them.

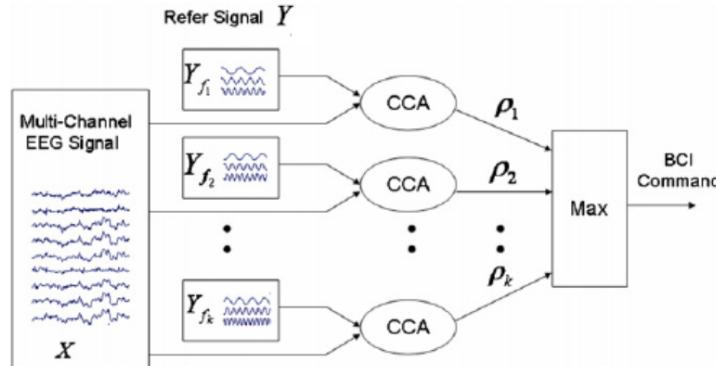


Figure 13: Analysis concept of CCA. [15]

As mentioned, CCA works on 2 sets of variables. The first set $x(t)$ is the data collected by certain channels that forming the overall SSVEP which is going to be analyzed, while the second set $y(t)$ is reference sine and cosine functions that are hypothesized as components of the aforementioned overall SSVEP (Fourier series). In the context of this project, 8 channels and 2 reference functions (1 harmonic) are used, as described in the Equation (1) and (2).

$$X = \begin{bmatrix} X_1[n] \\ X_2[n] \\ X_3[n] \\ X_4[n] \\ X_5[n] \\ X_6[n] \\ X_7[n] \\ X_8[n] \end{bmatrix} = \begin{bmatrix} X_1[1] & X_1[2] & \dots & X_1[N] \\ X_2[1] & X_2[2] & \dots & X_2[N] \\ X_3[1] & X_3[2] & \dots & X_3[N] \\ X_4[1] & X_4[2] & \dots & X_4[N] \\ X_5[1] & X_5[2] & \dots & X_5[N] \\ X_6[1] & X_6[2] & \dots & X_6[N] \\ X_7[1] & X_7[2] & \dots & X_7[N] \\ X_8[1] & X_8[2] & \dots & X_8[N] \end{bmatrix} \in \mathbb{R}^{8 \times N} \quad (1)$$

$$Y = \begin{bmatrix} Y_1[n] \\ Y_2[n] \end{bmatrix} = \begin{bmatrix} \sin(2\pi f \frac{1}{S}) & \sin(2\pi f \frac{2}{S}) & \dots & \sin(2\pi f \frac{N}{S}) \\ \cos(2\pi f \frac{1}{S}) & \cos(2\pi f \frac{2}{S}) & \dots & \cos(2\pi f \frac{N}{S}) \end{bmatrix} \in \mathbb{R}^{2 \times N} \quad (2)$$

where

- n is the sample index.
- N is the total sample of 1 epoch.
- f is the fundamental frequency.
- S is the sampling rate.

In traditional approach, optimal w_x and w_y for each frequency can be solved with the eigenvalue equations in Equation (3), where ρ^2 (square of the correlation between x and y) is eigenvalue and w_x , w_y are eigenvectors [16]. Because we only need the sets of weights that produce the highest correlation, we only have to solve w_x and w_y using the largest eigenvalue ρ^2 .

$$\begin{cases} C_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{yx} w_x = \rho^2 w_x \\ C_{xy} w_y = \rho \lambda_x C_{xx} w_x \end{cases} \quad (3)$$

where

$$\lambda_x = \sqrt{\frac{w_y^T C_{yy} w_y}{w_x^T C_{xx} w_x}} \quad (4)$$

Another approach is to iteratively optimize w_x and w_y to maximize the correlation. This iterative approach provides faster computation, costs less memory and offers better numerical stability as it avoiding matrix inversion. This project utilizes the iterative approach to solve the CCA problem as follows.

1. We firstly create $X \in \mathbb{R}^{8 \times 750}$ based on data collected from 8 channels and $Y \in \mathbb{R}^{2 \times 750}$ according to Equation (2). Then we assume the first row of Y to be $y_1 \in \mathbb{R}^{750 \times 1}$, and a large number to be $w_{x0} \in \mathbb{R}^{8 \times 1}$ for the initial iteration.

Example:

$$X = \begin{bmatrix} 1.811 & 2.552 & 2.396 & \dots & -1.527 & -0.866 & 0.254 \\ 1.197 & 1.794 & 1.641 & \dots & -0.910 & -0.518 & 0.297 \\ 1.367 & 1.874 & 1.656 & \dots & -1.164 & -0.639 & 0.330 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0.795 & 1.375 & 1.316 & \dots & -0.535 & -0.667 & -0.026 \\ 0.638 & 1.045 & 0.881 & \dots & -0.692 & -0.322 & 0.096 \\ 0.637 & 1.126 & 0.993 & \dots & -0.658 & -0.357 & -0.063 \end{bmatrix} \quad (5)$$

$$\begin{aligned}
 Y &= \begin{bmatrix} \sin(2\pi f \frac{1}{S}) & \sin(2\pi f \frac{2}{S}) & \dots & \sin(2\pi f \frac{N}{S}) \\ \cos(2\pi f \frac{1}{S}) & \cos(2\pi f \frac{2}{S}) & \dots & \cos(2\pi f \frac{N}{S}) \end{bmatrix} \\
 &= \begin{bmatrix} \sin(2\pi 15 \frac{1}{250}) & \sin(2\pi 15 \frac{2}{250}) & \dots & \sin(2\pi 15 \frac{750}{250}) \\ \cos(2\pi 15 \frac{1}{250}) & \cos(2\pi 15 \frac{2}{250}) & \dots & \cos(2\pi 15 \frac{750}{250}) \end{bmatrix} \\
 &= \begin{bmatrix} 0.368 & 0.6845 & \dots & 0 \\ 0.9298 & 0.729 & \dots & 1 \end{bmatrix}
 \end{aligned} \tag{6}$$

$$y_1 = [0.368 \ 0.6845 \ \dots \ 0]^T \tag{7}$$

$$w_{x0} = [99 \ 99 \ 99 \ 99 \ 99 \ 99 \ 99 \ 99]^T \tag{8}$$

2. The objective is to optimize w_x and w_y in order to make x and y as similar as possible. Therefore, we assume $X^T w_{x1} = y_1$ to find a better set of weights w_{x1} . This least square problem can be solved by multiplying both sides with the Moore-Penrose pseudoinverse of X^T [17]. The Moore-Penrose pseudoinverse matrix can be calculated by the formulas in Equation (9). The obtained w_{x1} is then divided by its norm to ensure unit length.

$$J^\dagger = \begin{cases} J^T (JJ^T)^{-1}, & \text{if } J \text{ has more columns than rows (right inverse: } JJ^\dagger = I) \\ (J^T J)^{-1} J^T, & \text{if } J \text{ has more rows than columns (left inverse: } J^\dagger J = I) \end{cases} \tag{9}$$

Example:

$$\begin{aligned}
 (X^T)^\dagger X^T w_{x1} &= (X^T)^\dagger y_1 \\
 w_{x1} &= (X^T)^\dagger y_1 \\
 w_{x1} &= (XX^T)^{-1} X y_1 \quad \text{then} \quad w_{x1} = \frac{w_{x1}}{\|w_{x1}\|}
 \end{aligned} \tag{10}$$

$$w_{x1} = [0.666 \ 0.433 \ -0.141 \ 0.239 \ -0.256 \ -0.029 \ -0.166 \ -0.445]^T$$

3. After obtaining normalized w_{x1} , we use it to calculate x_1 . We again assume $Y^T w_{y1} = x_1$ to find calculate w_{y1} and normalize it like in Step 2.

Example:

$$x_1 = X^T w_{x1} = [1.172 \ 1.562 \ \dots \ 0.304]^T \tag{11}$$

$$w_{y1} = (YY^T)^{-1} Y x_1 \quad \text{then} \quad w_{y1} = \frac{w_{y1}}{\|w_{y1}\|} = [0.921 \ 0.39]^T \tag{12}$$

4. At the end of each iteration, we evaluate the squared difference $\Delta w_x^2 = (w_{x1} - w_{x0})^2$ to decide whether to continue the iteration or not. If $\Delta w_x^2 < \varepsilon$ (e.g., $\varepsilon = 10^{-6}$), it means w_x as well as w_y have reached their convergences and the iteration will be stopped since we have obtained the optimal w_x and w_y . Otherwise, we uses w_{y1} to calculate y_2 and repeat Step 2 to find w_{x2} .

Example:

$$\Delta w_x^2 = (w_{x1} - w_{x0})^2 = 78349.4 > \varepsilon \quad (= 10^{-6}) \implies \text{Continue the iteration} \tag{13}$$

$$y_2 = Y^T w_{y1} = [0.702 \ 0.915 \ \dots \ 0.39]^T \quad , \text{repeat Step 2 to find } w_{x2} \tag{14}$$

This process of optimizing w_x and w_y is applied to every frequency of the stimulation sources. After obtaining optimized w_x and w_y , the correlation between x and y can be computed and compared with each other to detect the target frequency.

4 Inverse Kinematics of Robotic Arm System

4.1 Concept Overview

Inverse Kinematics (IK) is the process of finding suitable joint angles that can move the end-effector of the system to the desired configuration. The main idea of the inverse kinematics process of this system is to apply Newton-Raphson [18], a numerical method, to iteratively adjust an initial guess θ^0 of joint angles to gradually achieve the desired configuration T_{sd} obtained from the SSVEP acquisition-translation process. The initial guess θ^0 is randomized within the range of physical constraints specified in Table 3, which is obtained by careful experiments and observation.

Joint	Operating range (rads)
1	$[-\pi, \pi]$
2	$[-\pi/2, \pi/2]$
3	$[0, \pi]$
4	$[-\pi, \pi]$
5	$[-\pi/2, \pi/2]$
6	$[-\pi, \pi]$

Table 3: The robotic arm's rotating constraints.

Each iteration involves several steps to evaluate the difference between current solution and the desired configuration, and to adjust joint angles. First, configuration matrix at each iteration T_{sb} is calculated using Product of Exponentials PoE formula and corresponding joint angles (Forward Kinematics). After that, body twist error \mathcal{V}_b is computed for evaluation of the solution as a cost function. There are 3 angular velocity components and 3 linear velocity components contained in the body twist error \mathcal{V}_b .

The system in this project only try to reach the desired position and do not care about the orientation of the end-effector. Therefore, if any linear velocity components of \mathcal{V}_b are below a predefined threshold ε , the current angles will be the final solution and the process stops. Otherwise, the body Jacobian matrix J_b of this configuration will be computed and then used to adjust the joint angles for the next iteration.

Furthermore, it is noticeable that the Newton-Raphson method updates joint angles without considering physical constraints, which means it may return a solution that exceed the maximum angle the joints can rotate [19]. To address this problem, additional conditions are used to verify if the final solution falls within the possible range of joint angles. If the final solution exceeds physical constraints, the initial guess will be randomized again and a new Inverse Kinematics process will start in order to find the valid final solution, as different initial guess can lead to different final solutions.

After obtaining the valid final solution for 6 joint angles, those angles will be converted into corresponding integer ranging from 0-4095. These integer will then be transmitted to 6 Dynamixel servo motors to control the robotic arm via U2D2 USB cable. The workflow of this inverse kinematics process is visualized in Figure 14.

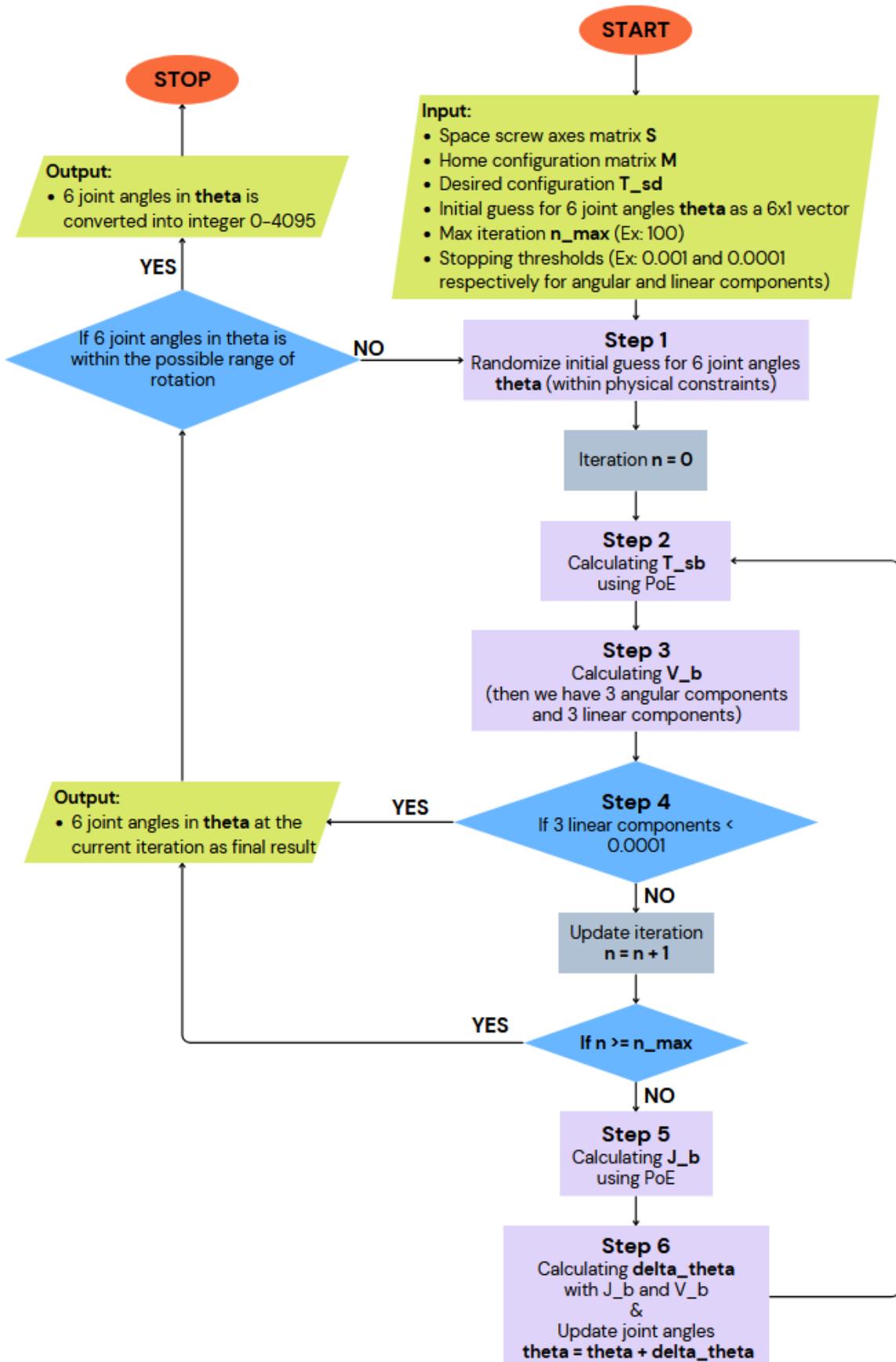


Figure 14: Flow Chart of Inverse Kinematics.

4.2 Hardware Specifications

4.2.1 Dynamixel Servo Motors

The robotic arm is built from 6 Dynamixel servo motors. The type of motor used for each joint and their corresponding specifications are shown in Table 4.

Joint	Series	Rated Torque	Rated Input	Weight	Communication
1	MX-106	8.4 N.m	12 V - 5.2 A	153 g	RS-485
2	MX-106	8.4 N.m	12 V - 5.2 A	153 g	RS-485
3	MX-106	8.4 N.m	12 V - 5.2 A	153 g	RS-485
4	MX-64	6.0 N.m	12 V - 4.1 A	135 g	RS-485
5	MX-28	2.5 N.m	12 V - 1.4 A	77 g	RS-485
6	MX-28	2.5 N.m	12 V - 1.4 A	77 g	RS-485

Table 4: Specifications of servo motors of 6 joints.

As observed from the table, the MX-106 motors are the strongest ones with the rated torque 8.4 N.m, and also the heaviest ones. Therefore, they are used for the first 3 joints to make the system more stable. All motors use RS-485 port for communication. In order to control the rotation angles of the motors and their speed, following control parameters are used, as shown in Table 5.

Address	Name	Description
8	Baud Rate	Serial communication speed between motors and computer. In this project, the default baud rate of 1,000,000 bps is used for all motors to maximize communication speed and minimize transmission errors.
11	Operating Mode	This defines how the motor is controlled. In this project, the default Position Control Mode is used. This allows an integer in the range 0–4095 to control the rotation angle, corresponding to 0–359.012 degrees.
112	Profile Velocity	Controls the motor's rotation speed. A value of 10 is used in this project, which is slow enough to ensure safety.
116	Goal Position	Value in the range 0–4095, corresponding to a rotation angle of 0–359.012 degrees.
64	Torque Enable	Determines whether the motor is locked (1) or not (0). Profile Velocity and Goal Position can only be changed when Torque Enable is 0 and 1, respectively.

Table 5: Motor Register Addresses and Their Descriptions



Figure 15: MX-28, MX-64 and MX-106 motors.¹⁴

4.2.2 U2D2 and U2D2 Power Hub

The U2D2 module is used to connect the Dynamixel servo motors with a computer. U2D2 is a small USB communication converter that can transmit information from the computer to Dynamixel motors and vice versa via RS-485 communication port. In addition, the U2D2 powerhub is used to supply power for all the motors at rated voltage 12 V.



Figure 16: U2D2 (left) and U2D2 power hub (right).¹⁵

4.3 Kinematic Modelling

4.3.1 Denavit-Hartenberg Parameters

Denavit–Hartenberg (DH) convention is used to assign frames for the robotic arm, which is shown in Figure 17. With the assigned frame coordinate, we can not only better understand

¹⁴https://www.servomagazine.com/magazine/article/april2012_Ferguson

¹⁵<https://www.robotis.us/>

the construction of the arm, but also have the ability to use DH parameters to double check the result of PoE in solving Forward Kinematics.

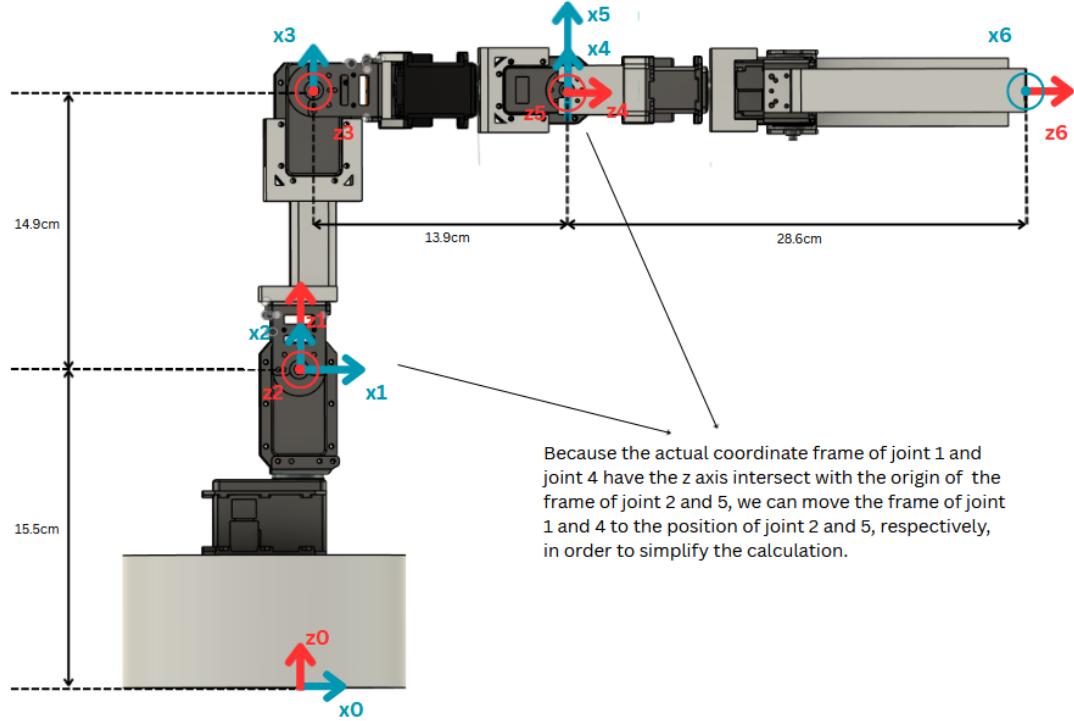


Figure 17: Coordinate Frame System of The Robotic Arm.

To construct a DH parameter table from the assigned frame coordinate, we trace the movements that transform frame $\{i - 1\}$ to frame $\{i\}$ following the sequence below to acquire 4 parameters α_{i-1} , a_{i-1} , d_i and ϕ_i [20, p. 519].

- A rotation of frame $\{i - 1\}$ about its \hat{x} -axis by an angle α_{i-1} .
- A translation of this new frame along its \hat{x} -axis by a distance a_{i-1} .
- A translation of the new frame formed by (b) along its \hat{z} -axis by a distance d_i .
- A rotation of the new frame formed by (c) about its \hat{z} -axis by an angle ϕ_i .

As a result, corresponding DH parameters are presented in Table 6:

$T(\{i-1\} \rightarrow \{i\})$	α_{i-1} (rad)	a_{i-1} (cm)	d_i (cm)	ϕ_i (rad)
$0 \rightarrow 1$	0	0	15.5	θ_1
$1 \rightarrow 2$	$\pi/2$	0	0	$\pi/2 + \theta_2$
$2 \rightarrow 3$	0	14.9	0	θ_3
$3 \rightarrow 4$	$\pi/2$	0	13.9	θ_4
$4 \rightarrow 5$	$-\pi/2$	0	0	θ_5
$5 \rightarrow 6$	$-\pi/2$	0	28.6	$\pi/2 + \theta_6$

Table 6: DH Parameter Table.

To calculate Forward Kinematics using the DH parameters, we simply multiply all the transformation matrices together according to Equation 15 and 16.

$$T_{sb} = T_{06} = T_{01} T_{12} T_{23} T_{34} T_{45} T_{56} \in \mathbb{R}^{4 \times 4} \quad (15)$$

$$\begin{aligned} T_{i-1,i} &= \text{Rot}(\hat{x}, \alpha_{i-1}) \text{Trans}(\hat{x}, a_{i-1}) \text{Trans}(\hat{z}, d_i) \text{Rot}(\hat{z}, \phi_i) \\ &= \begin{bmatrix} \cos \phi_i & -\sin \phi_i & 0 & a_{i-1} \\ \sin \phi_i \cos \alpha_{i-1} & \cos \phi_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -d_i \sin \alpha_{i-1} \\ \sin \phi_i \sin \alpha_{i-1} & \cos \phi_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & d_i \cos \alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \quad (16) \end{aligned}$$

4.3.2 Home Configuration Matrix

Home configuration matrix M is the configuration of the frame 6 attached to the end-effector in Figure 17 when all the joint angles are 0. Applying the Table 6, Equation 15 and 16, we obtain M as below.

$$M = \begin{bmatrix} 0 & 0 & 1 & 42.5 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 30.4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4} \quad (17)$$

4.3.3 Space Screw Axes

Space screw axes are the twists that represent the motion of joints with respect to the space frame 0 in Figure 17. Knowing space screw axes S_i and their skew-symmetric matrix $[S_i]$ of all joints is necessary for applying PoE to Forward Kinematics. The construction of S_i and $[S_i]$ is provided in (18) and (19). In case the angular velocity is 0 (prismatic joint), then v is chosen as $-\hat{s}$.

$$S_i = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} \omega \\ v \end{bmatrix} = \begin{bmatrix} \hat{s} \\ -\hat{s} \times q + h\hat{s} \end{bmatrix} \in \mathbb{R}^{6 \times 1} \quad (18)$$

$$[S_i] = \begin{bmatrix} [\omega] & v \\ 0 & 0 \end{bmatrix}, \quad [\omega] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \in \mathbb{R}^{3 \times 3} \quad (19)$$

where

- q is any point on the axis.
- \hat{s} is a unit vector in the direction of the axis.
- h is the screw pitch, which defines the ratio of the linear velocity along the screw axis to the angular velocity about the screw axis.
- $\omega \in \mathbb{R}^{3 \times 1}$ is the angular velocity vector of a specific joint, where ω_1 , ω_2 , and ω_3 are, respectively, the angular velocity about Ox , Oy and Oz axis; $[\omega]$ is the skew-symmetric matrix of ω .
- $v \in \mathbb{R}^{3 \times 1}$ is the linear velocity vector of a specific joint, where v_1 , v_2 , and v_3 are, respectively, the linear velocity along Ox , Oy and Oz axis.

The space screw axis matrix S of the robotic arm used in this project is obtained as below.

$$S = [S_1 \ S_2 \ S_3 \ S_4 \ S_5 \ S_6] \\ = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & -1 & -1 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 15.5 & 30.4 & 0 & 30.4 & 0 \\ 0 & 0 & 0 & 30.4 & 0 & 30.4 \\ 0 & 0 & 0 & 0 & -13.9 & 0 \end{bmatrix} \in \mathbb{R}^{6 \times 6} \quad (20)$$

4.4 Newton-Raphson Method's Algorithm

Newton-Raphson method is a numerical approach to iteratively adjust the joint angles in order to approximately reach the desired configuration. Although it is not as accurate as the analytical approach that finding solution based on geometric equations, Newton-Raphson method can operate well with different complex robotic systems with many degrees of freedom, enabling us to upgrade the system without extensively changing the algorithm in the future.

This project finds the valid final solution for joint angles using Newton-Raphson method as follows.

1. The process begins with the desired configuration being received and the initial guess θ^0 being randomized within the valid range of rotation specified in Table 3.

Example:

$$T_{sd} = \begin{bmatrix} 0 & -1 & 0 & 39.364 \\ -1 & 0 & 0 & 11.2 \\ 0 & 0 & -1 & 1.8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (21)$$

$$\theta^0 = (0, 0, 0, 0, 0, 0) \quad (22)$$

2. Next, the configuration matrix T_{sb} formed by the initial guess θ^0 is computed using PoE formula in Equation 23.

$$T_{sb} = T_{06} = e^{[S_1]\theta_1} e^{[S_2]\theta_2} e^{[S_3]\theta_3} e^{[S_4]\theta_4} e^{[S_5]\theta_5} e^{[S_6]\theta_6} M \quad (23)$$

The matrix exponential $e^{[S]\theta} \in \mathbb{R}^{4 \times 4}$ can be approximated using its Maclaurin series expansion. As a result, the general form of the result matrix is derived, as shown in Equation 24 and 25.

$$e^{[S]\theta} = I + [S]\theta + \frac{[S]^2\theta^2}{2!} + \frac{[S]^3\theta^3}{3!} + \dots \\ = \begin{bmatrix} e^{[\omega]\theta} & (I\theta + (1 - \cos\theta)[\omega] + (\theta - \sin\theta)[\omega]^2)v \\ 0 & 1 \end{bmatrix} \quad (24)$$

If $\omega = 0$ and $\|v\| = 1$, then

$$e^{[S]\theta} = \begin{bmatrix} I & v\theta \\ 0 & 1 \end{bmatrix} \quad (25)$$

Similarly, the matrix exponential $e^{[\omega]\theta} \in \mathbb{R}^{3 \times 3}$ can also be approximated using its Maclaurin series expansion, as shown in Equation 26.

$$e^{[\omega]\theta} = I + [\omega]\theta + \frac{[\omega]^2\theta^2}{2!} + \frac{[\omega]^3\theta^3}{3!} + \dots \\ = I + \left(\theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \dots \right) [\omega] + \left(\frac{\theta^2}{2!} - \frac{\theta^4}{4!} + \frac{\theta^6}{6!} - \dots \right) [\omega]^2 \\ = I + \sin\theta [\omega] + (1 - \cos\theta) [\omega]^2 \quad (26)$$

Example:

$$S_1 = [0 \ 0 \ 1 \ 0 \ 0 \ 0]^T \quad (27)$$

$$\Rightarrow [\omega_1] = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad v_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (\text{according to Equation 18 and 19}) \quad (28)$$

$$\Rightarrow e^{[\omega_1]\theta_1} = I + \sin \theta_1 [\omega_1] + (1 - \cos \theta_1) [\omega_1]^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (29)$$

$$\Rightarrow (I\theta_1 + (1 - \cos \theta_1)[\omega_1] + (\theta_1 - \sin \theta_1)[\omega_1]^2) v_1 = [0 \ 0 \ 0]^T \quad (30)$$

Combine (29) and (30), we have:

$$e^{[S_1]\theta_1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (31)$$

Continue the calculation with the remaining 5 joints, we obtain:

$$T_{sb} = \begin{bmatrix} 0 & 0 & 1 & 42.5 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 30.4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (32)$$

3. After having the current configuration matrix T_{sb} , we compute the body twist error \mathcal{V}_b and evaluate the difference between T_{sb} and the desired configuration matrix T_{sd} . The body twist error \mathcal{V}_b is obtained by taking logarithm of T_{bd} , as shown in Equation 33 and 34.

$$T_{bd} = T_{sb}^{-1} T_{sd} = T_{bs} T_{sd} \quad (33)$$

$$\log(T_{bd}) = \begin{bmatrix} [\omega_b]\phi & v_b\phi \\ 0 & 0 \end{bmatrix}, \quad [\omega_b] = \begin{bmatrix} 0 & -\omega_{b3} & \omega_{b2} \\ \omega_{b3} & 0 & -\omega_{b1} \\ -\omega_{b2} & \omega_{b1} & 0 \end{bmatrix}, \quad \mathcal{V}_b = \begin{bmatrix} \omega_b\phi \\ v_b\phi \end{bmatrix} \quad (34)$$

The variable ϕ can be extracted from T_{bd} as shown in Equation 35. Once we have obtained ϕ , we can then determine ω_b . This process depends on the value of $\cos \phi$ [20, p. 87]. Note that although ϕ has 2 possible solutions, $[\omega_b]\phi$ and $v_b\phi$ will always have 1 solution regardless of the ϕ we choose. Therefore, the positive solution of ϕ is chosen to use in this project.

$$\text{trace}(R) = 1 + 2 \cos \phi, \quad \text{with} \quad T_{bd} = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \quad (35)$$

- **Case 1:** $-1 < \cos \phi < 1$

This is the general case. The skew-symmetric matrix of ω_b can be computed as:

$$[\omega_b] = \frac{1}{2 \sin \phi} (R - R^\top) \quad (36)$$

- **Case 2:** $\cos \phi = -1$

This corresponds to a rotation of $\pi + k2\pi$, leading (36) to be ambiguous. Therefore, ω_b can be computed as (37), (38) or (39). Actually, we can use any among these options, but it is recommended to choose based on which diagonal element

is the largest in order to improve numerical stability. Selecting the expression with the largest denominator minimizes the risk of division by a small number, thereby reducing numerical error and avoiding instability when the denominator approaches zero.

$$\omega_b = \frac{1}{\sqrt{2(1+r_{33})}} \begin{bmatrix} r_{13} \\ r_{23} \\ 1+r_{33} \end{bmatrix} \quad \text{when } r_{33} \text{ is largest.} \quad (37)$$

$$\omega_b = \frac{1}{\sqrt{2(1+r_{22})}} \begin{bmatrix} r_{12} \\ 1+r_{22} \\ r_{32} \end{bmatrix} \quad \text{when } r_{22} \text{ is largest.} \quad (38)$$

$$\omega_b = \frac{1}{\sqrt{2(1+r_{11})}} \begin{bmatrix} 1+r_{11} \\ r_{21} \\ r_{31} \end{bmatrix} \quad \text{when } r_{11} \text{ is largest.} \quad (39)$$

- **Case 3:** $\cos \phi = 1$

This corresponds to a rotation of $k2\pi$, leading (36) to be ambiguous. The twist can be treated as purely translational and $[\omega_b]$ is a zero matrix.

$$[\omega_b] = 0 \quad (40)$$

With the computed value of ω_b , v_b can now be determined and then we will have a complete body twist \mathcal{V}_b [20, p. 106].

$$v_b = \left[\frac{1}{\phi} I - \frac{1}{2} [\omega_b] + \left(\frac{1}{\phi} - \frac{1}{2} \cot\left(\frac{\phi}{2}\right) \right) [\omega_b]^2 \right] p \quad (41)$$

Example:

$$T_{bd} = T_{sb}^{-1} T_{sd} = \begin{bmatrix} 1 & 0 & 0 & -11.2 \\ 0 & 0 & 1 & 28.6 \\ 0 & -1 & 0 & -3.136 \\ 0 & 0 & 0 & 1 \end{bmatrix} \implies R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad (42)$$

With R , we can calculate ϕ :

$$\begin{aligned} \text{trace}(R) &= 1 + 2 \cos \phi \\ \Leftrightarrow 1 + 0 + 0 &= 1 + 2 \cos \phi \\ \Leftrightarrow \cos \phi &= 0 \\ \Leftrightarrow \phi &= \pi/2 \end{aligned} \quad (43)$$

Because $-1 < \cos \phi < 1$, Equation 36 is used:

$$[\omega_b] = \frac{1}{2 \sin \phi} (R - R^T) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & \pi/2 \\ 0 & -\pi/2 & 0 \end{bmatrix} \quad (44)$$

$$\implies v_b = [-11.2 \ 24.9254 \ 19.9994]^T \quad (45)$$

$$\implies v_b \phi = [-17.5929 \ 39.1527 \ 31.4149]^T \quad (46)$$

4. In this project, the system only try to reach the desired position and do not care about the orientation of the end-effector. Therefore, only the absolute values of 3 linear velocity components of the body twist error \mathcal{V}_b , which is $v_b \phi$, will be compared with the threshold $\varepsilon = 0.0001$ to decide to stop the iteration or not.

5. If the iteration have to go further, the body Jacobian matrix J_b will be computed. The body Jacobian matrix J_b is a matrix containing partial derivatives of orientations about Ox, Oy, Oz axes and position x, y, z of the end-effector with respect to six joint angles, expressed in the body frame, as shown in Equation 47.

$$J_b = \begin{bmatrix} \frac{\partial \phi_{Ox}}{\partial \theta_1} & \frac{\partial \phi_{Ox}}{\partial \theta_2} & \frac{\partial \phi_{Ox}}{\partial \theta_3} & \frac{\partial \phi_{Ox}}{\partial \theta_4} & \frac{\partial \phi_{Ox}}{\partial \theta_5} & \frac{\partial \phi_{Ox}}{\partial \theta_6} \\ \frac{\partial \phi_{Oy}}{\partial \theta_1} & \frac{\partial \phi_{Oy}}{\partial \theta_2} & \frac{\partial \phi_{Oy}}{\partial \theta_3} & \frac{\partial \phi_{Oy}}{\partial \theta_4} & \frac{\partial \phi_{Oy}}{\partial \theta_5} & \frac{\partial \phi_{Oy}}{\partial \theta_6} \\ \frac{\partial \phi_{Oz}}{\partial \theta_1} & \frac{\partial \phi_{Oz}}{\partial \theta_2} & \frac{\partial \phi_{Oz}}{\partial \theta_3} & \frac{\partial \phi_{Oz}}{\partial \theta_4} & \frac{\partial \phi_{Oz}}{\partial \theta_5} & \frac{\partial \phi_{Oz}}{\partial \theta_6} \\ \frac{\partial x}{\partial \theta_1} & \frac{\partial x}{\partial \theta_2} & \frac{\partial x}{\partial \theta_3} & \frac{\partial x}{\partial \theta_4} & \frac{\partial x}{\partial \theta_5} & \frac{\partial x}{\partial \theta_6} \\ \frac{\partial y}{\partial \theta_1} & \frac{\partial y}{\partial \theta_2} & \frac{\partial y}{\partial \theta_3} & \frac{\partial y}{\partial \theta_4} & \frac{\partial y}{\partial \theta_5} & \frac{\partial y}{\partial \theta_6} \\ \frac{\partial z}{\partial \theta_1} & \frac{\partial z}{\partial \theta_2} & \frac{\partial z}{\partial \theta_3} & \frac{\partial z}{\partial \theta_4} & \frac{\partial z}{\partial \theta_5} & \frac{\partial z}{\partial \theta_6} \end{bmatrix} \in \mathbb{R}^{6 \times 6} \quad (47)$$

To find J_b , we first calculate the product of exponentials for each column in Equation 48 to obtain their corresponding matrix T . The matrices T are therefore used to calculate their corresponding adjoint matrix as shown in Equation 49. Then, the adjoint matrices and space screw axes are multiplied together to form each column of the space Jacobian matrix J_s (Equation 48).

$$J_s = [S_1 \quad [\text{Ad}_{e^{[S_1]\theta_1}}]S_2 \quad [\text{Ad}_{e^{[S_1]\theta_1}e^{[S_2]\theta_2}}]S_3 \quad \dots \quad [\text{Ad}_{e^{[S_1]\theta_1}\dots e^{[S_5]\theta_5}}]S_6] \quad (48)$$

$$[\text{Ad}_T] = \begin{bmatrix} R & 0 \\ [p]R & R \end{bmatrix}, \quad \in \mathbb{R}^{6 \times 6} \quad \text{with} \quad T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \quad \in \mathbb{R}^{6 \times 6} \quad \text{is the result of PoE.} \quad (49)$$

The space Jacobian matrix J_s is then transformed to the body Jacobian matrix J_b by multiplying the inverse of adjoint matrix of the arm's current configuration in the space frame T_{sb} with the space Jacobian matrix J_s , as shown in Equation 50.

$$\begin{aligned} J_s &= [\text{Ad}_{T_{sb}}] J_b \\ \Leftrightarrow J_b &= [\text{Ad}_{T_{sb}}]^{-1} J_s \end{aligned} \quad (50)$$

Example:

The first column of J_s is the first column of S . By using the results of $e^{[S_1]\theta_1}$ computed in (31), we obtain the second column as below:

$$[\text{Ad}_{e^{[S_1]\theta_1}}] = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (51)$$

$$\Rightarrow [\text{Ad}_{e^{[S_1]\theta_1}}]S_2 = [0 \quad -1 \quad 0 \quad 15.5 \quad 0 \quad 0]^T \quad (52)$$

Continue the process to find the remaining columns, we obtain J_s as below:

$$J_s = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & -1 & -1 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 15.5 & 30.4 & 0 & 30.4 & 0 \\ 0 & 0 & 0 & 30.4 & 0 & 30.4 \\ 0 & 0 & 0 & 0 & -13.9 & 0 \end{bmatrix} \quad (53)$$

We then use $T_s b$ computed in (32) to obtain J_b as below:

$$[\text{Ad}_{T_{sb}}] = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 30.4 & 0 & 0 & 0 & 0 & 1 \\ 0 & 42.5 & 30.4 & -1 & 0 & 0 \\ -42.5 & 0 & 0 & 0 & -1 & 0 \end{bmatrix} \quad (54)$$

$$\Rightarrow J_b = [\text{Ad}_{T_{sb}}]^{-1} J_s = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ -42.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & -42.5 & -42.5 & 0 & -28.6 & 0 \\ 0 & -14.9 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (55)$$

6. The last step before repeating Step 2 is to update new joint angles θ^1 with Equation 56. Since this project only try to reach the desired position, we only take the Moore-Penrose pseudoinverse (Equation 9) of the last 3 rows of J_b and multiply it with the 3 linear velocity components of \mathcal{V}_b to find θ^1 .

$$\theta^1 = \theta^0 + (J'_b)^\dagger \mathcal{V}'_b \quad (56)$$

$$\text{with : } J'_b = J_b[4 : 6, :] \in \mathbb{R}^{3 \times 6} \text{ and } \mathcal{V}'_b = \mathcal{V}_b[4 : 6] \in \mathbb{R}^{3 \times 1}$$

Example:

$$J'_b = \begin{bmatrix} -42.5 & 0 & 0 & 0 & 0 & 0 \\ 0 & -42.5 & -42.5 & 0 & -28.6 & 0 \\ 0 & -14.9 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (57)$$

$$\mathcal{V}'_b = [-17.5929 \ 39.1527 \ 31.4149]^T \quad (58)$$

$$\theta^1 = \theta^0 + (J'_b)^\dagger \mathcal{V}'_b = (0, 0, -0.72757, 0, -0.48961, 0) \quad (59)$$

5 Testings and Discussions

5.1 Testing on SSVEP Acquisition-Translation Process

The offline tests on SSVEP acquisition-translation process were taken separately from the system's operation to examine its performance. Data for testing was collected from a self-made experiment with a volunteer by OpenBCI Ganglion board and from the open source dataset 12JFPM of the paper "A Comparison Study of Canonical Correlation Analysis Based Methods for Detecting Steady-State Visual Evoked Potentials" [21]. The performance of the acquisition-translation process is evaluated based on translation accuracy and ITR, a metric reflects the amount of information transferred per unit time [22], as shown in Equation 60.

$$ITR \left(\frac{bits}{min} \right) = \left[\log_2 N + P \times \log_2 P + (1 - P) \times \log_2 \left(\frac{1 - P}{N - 1} \right) \right] \frac{S}{T} \quad (60)$$

where

- N is the number of stimuli. For example, in the self-made dataset, $N = 9$ because there are 9 flickering boxes.
- P is the translation accuracy. It is actually the ratio of the number of correctly translated trials to the total number of trials.
- S is the total number of translation trials.
- T is the total duration of signal recording.

5.1.1 Introduction of Self-Made Dataset

The self-made SSVEP dataset was collected in my laboratory in Groß-Gerau from a healthy 22-year-old male volunteer by using 10 gold cup electrodes, Signa gel, a 8-channel OpenBCI Cyton biosensing board and a USB Dongle at the sampling rate of 250 Hz. The setup of OpenBCI Cyton board used in experiment is shown in Figure 18.



Figure 18: OpenBCI Cyton setup for the experiment.

The electrodes were placed at volunteer's right earlobe, forehead and 8 positions around the occipital region, including Oz, O1, O2, POz, PO3, PO4, PO5 and PO6, according to the 10-20 system, as mentioned in Section 3.4. The Signa gel was added to the contact surface between electrodes and the volunteer's scalp, then nylon tape was used to tighten the connection between

them. The connection between electrodes and the volunteer's scalp in different views is shown in Figure 19.



Figure 19: Connection between electrodes and the volunteer's scalp.

A 31.6-inch curved monitor Alienware AW3225DM (Dell) with a refresh rate of 180 Hz and a resolution of 2560 x 1440 was used to display 9 flickering stimuli with different frequencies, as shown in Figure 7. The frequencies of stimuli fell within the range 14–18 Hz with intervals of 0.5 Hz.

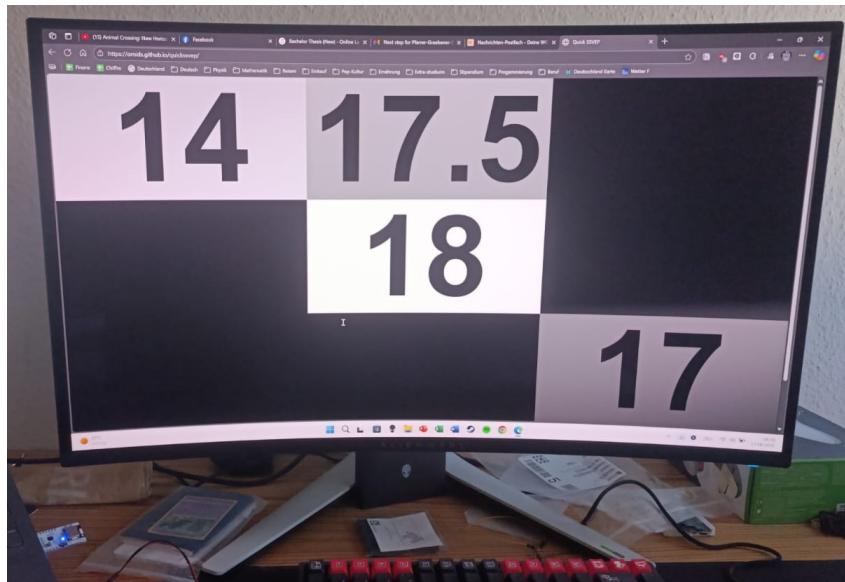


Figure 20: Alienware AW3225DM curved monitor.

The volunteer was positioned around 40 cm from the monitor screen. He was required to look at each box continuously for 3 minutes with 1-minute pause intervals between trials. Additionally, the volunteer was requested to avoid any movement and eye blink in order to minimize interference signals. The recorded SSVEP signals were expected to be stored as 9 .csv files, corresponding to 9 stimuli, by using OpenBCI GUI. However, only 7 stimuli were correctly recorded because of some technical issues with the used computer. The data in each file is actually an

array with the dimension $(N, 8)$, where N is the number of sampling points, whose formula is shown in Equation 61.

$$\text{Num of sampling points} = \text{Total recording duration} \times \text{Sampling rate} \quad (61)$$

5.1.2 Introduction of 12JFPM Dataset

The 12JFPM dataset is an open EEG dataset that Nakanishi et al. collected when evaluating the performance of CCA technique on translating SSVEP in 2015. The dataset was recorded from the occipital region of 10 healthy subjects (9 men and 1 woman with the average age 28 years old) by Ag/AgCl electrodes and a 8-channel BioSemi ActiveTwo EEG system at the sampling rate of 2048 Hz then was down-sampled to 256 Hz. Subjects were presented with 12 flickering boxes at distinct frequency and phase on a 27-inch LCD monitor (ASUS VG278) with a refresh rate of 60Hz and a resolution of 1280×800 pixels. The frequencies of stimuli fell within the range 9.25–14.75 Hz with intervals of 0.5 Hz. The phases of stimuli were defined as 0, 0.5π , 1π and 1.5π radians. Figure 21 shows the stimuli used in this experiment, which was developed using the Psychophysics Toolbox extensions [23]. Subjects were required to focus on each stimulus for approximately 4.3516 seconds, repeating for 15 trials. Because the recorded SSVEP signal was down-sampled to 256 Hz, there are 1114 sampling points in each trial ($4.3516 / 256$).

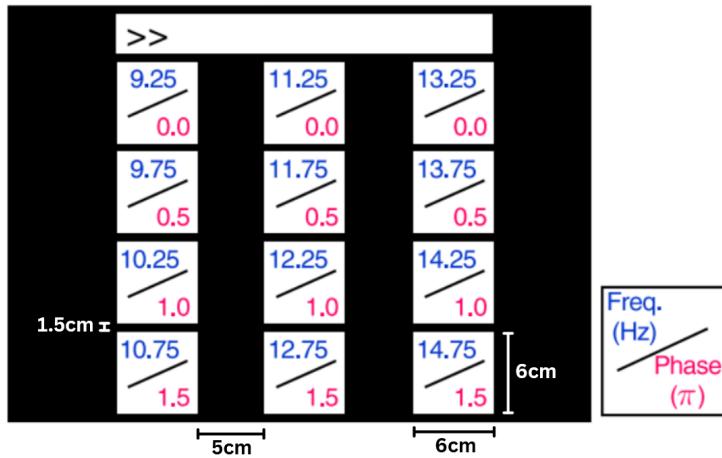


Figure 21: Visual stimulation sources used in Nakanishi's experiment. [21]

The entire dataset is stored as 10 separate .mat files, corresponding to 10 subjects. Recorded SSVEP signal in each file is actually an array with the dimension $(12, 8, 1114, 15)$, corresponding to number of stimuli, number of channels, number of sampling points and number of trials. There was a delay time of 0.15 second before the SSVEP signals were generated, so sampling points before the 39th one were considered irrelevant to the experiment and were omitted when being analysed.

5.1.3 Testing Results

The self-made dataset firstly went through the preprocessing stage described in Section 3.5 to omit signal components outside the range 13.5–18.5 Hz and remove common noise across all channels with CAR technique. The PSD and SNR plots of the 7 frequencies after the preprocessing stage are shown in Figure 22. The self-made dataset was then divided into epochs and these epochs were sequentially analysed by CCA technique at translation stage as described in Section 3.6 for predicting which frequency they belonged to. Different epoch durations were used to examine its impact on translation accuracy and ITR. The obtained plots of translation accuracy

and ITR with respect to epoch duration were shown in Figure 23, with their specific values listed in Table 7. In addition, the average values of the target and the largest non-target frequency correlation (including only correct translation epochs) were presented in Table 8.

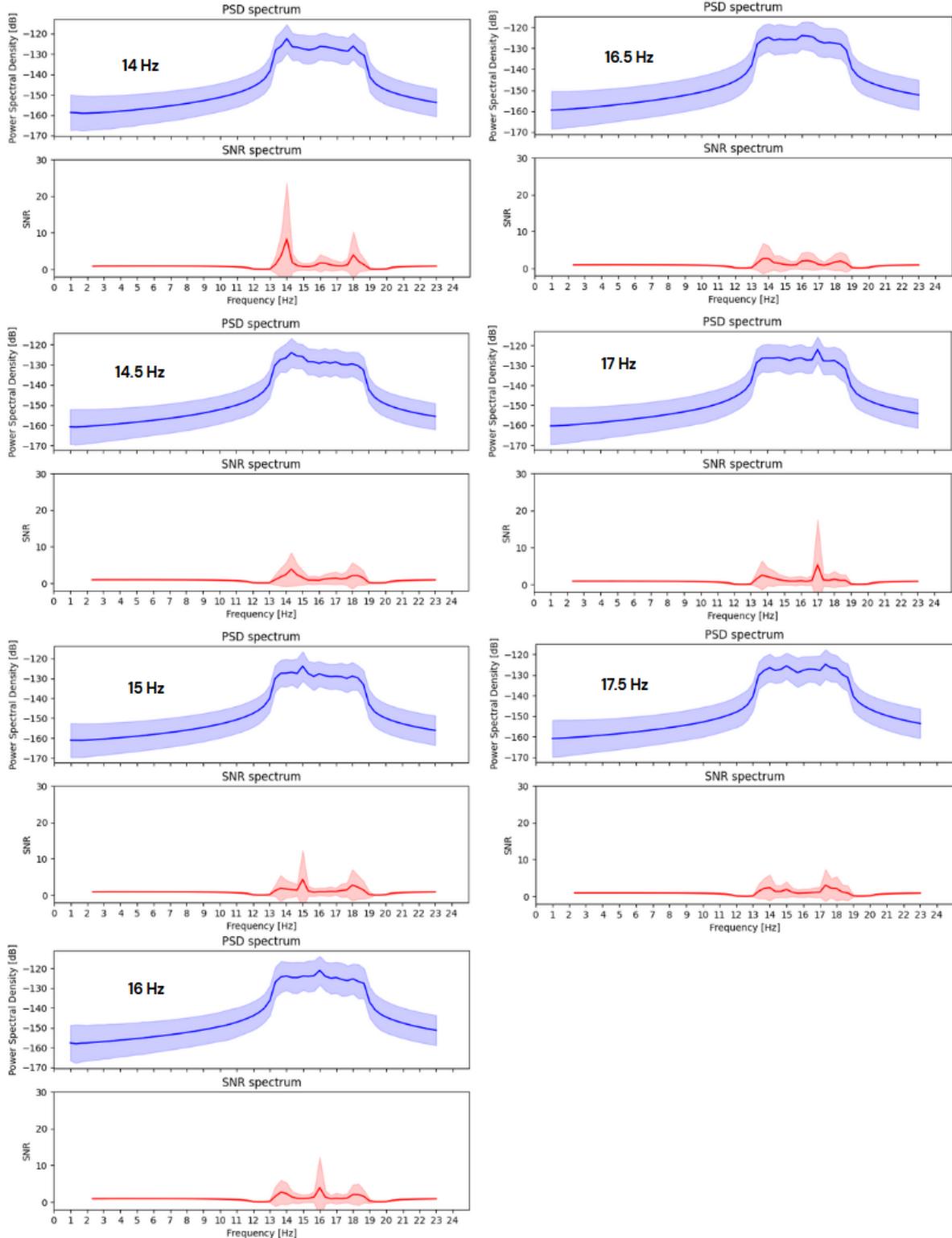


Figure 22: PSD and SNR plots of 7 used visual stimuli.

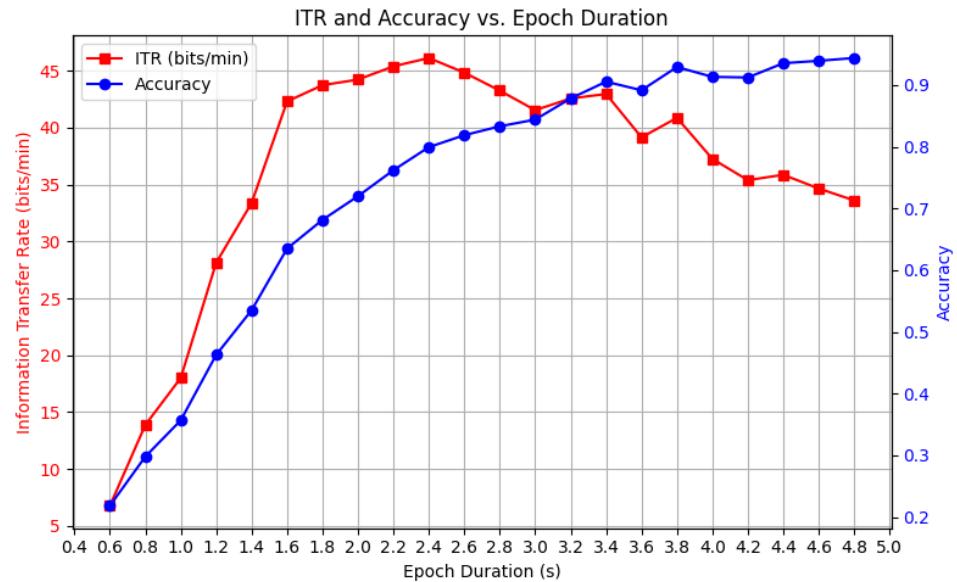


Figure 23: Plots of translation accuracy and ITR versus epoch duration.

Epoch Duration (s)	Accuracy (%)	ITR (bits/min)	Num Epochs
0.6	21.83	6.78	2162
0.8	29.8	13.88	1621
1.0	35.68	18.01	1295
1.2	46.29	28.13	1078
1.4	53.51	33.38	925
1.6	63.49	42.29	808
1.8	68.15	43.72	719
2.0	71.98	44.21	646
2.2	76.19	45.38	588
2.4	79.93	46.11	538
2.6	81.85	44.82	496
2.8	83.3	43.24	461
3.0	84.38	41.53	429
3.2	87.81	42.55	402
3.4	90.53	42.95	380
3.6	89.14	39.14	359
3.8	92.9	40.85	338
4.0	91.3	37.24	322
4.2	91.21	35.38	307
4.4	93.52	35.85	293
4.6	93.91	34.64	279
4.8	94.36	33.6	266

Table 7: Values of translation accuracy, ITR and number of epochs versus epoch duration.

Epoch Duration (s)	Average Correlation of Target	Average Correlation of Largest Non-Target
0.6	0.9954	0.9907
0.8	0.9811	0.9665
1.0	0.9606	0.9312
1.2	0.9355	0.8935
1.4	0.9151	0.8565
1.6	0.8939	0.8225
1.8	0.8775	0.7952
2.0	0.8606	0.7624
2.2	0.8503	0.7418
2.4	0.8360	0.7205
2.6	0.8250	0.7007
2.8	0.8162	0.6825
3.0	0.8073	0.6649
3.2	0.7976	0.6516
3.4	0.7912	0.6399
3.6	0.7857	0.6225
3.8	0.7797	0.6113
4.0	0.7726	0.6036
4.2	0.7702	0.5879
4.4	0.7660	0.5800
4.6	0.7573	0.5736
4.8	0.7543	0.5626

Table 8: Average correlation of target and largest non-target within correct translations.

Meanwhile, the 12JFPM dataset had already been divided into 4.356-second epochs by the authors so the test only examine the performance of the SSVEP acquisition-translation process with that epoch duration. The sampling points before the 39th one were discarded according to the report of the authors to improve signal quality. After that, it also went through the pre-processing and translation stages. the obtained translation accuracy and ITR were 89.52% and 39.13 bits/min, respectively, examined with 1680 epochs in total.

5.1.4 Discussion

As observed from the PSD and SNR plots of the self-made dataset (Figure 22), there is a distinguishable peak at the frequency that matches the box a user was looking at. However, the SNR values of these peaks are not very high, which is only about 5 in general, suggesting the presence of noise in the recorded SSVEP signals or wrong placement of electrodes. According to the accuracy and ITR plots (Figure 23 and Table 7), there is a distinct trend in which translation accuracy increases with longer epoch durations, while ITR initially rises and gradually decreases after reaching a peak of 46.11 bits / min at the 2.4-second epoch duration. These results are as expected, since longer epochs contain more data samples, avoiding the possibility where a non-target frequency accidentally resulting the largest correlation. Additionally, as can be observed from the Table 8, although the correlation of the target frequency tends to decrease with longer epoch durations, the difference between it and that of the largest non-target frequency increases, proving the positive impact of epoch duration on the translation confidence of the acquisition-translation process. This decreasing trend of correlation in general suggests

the presence of external interference signal or the impact of distraction as well as fatigue over prolonged concentration on the visual stimulation sources.

Although the highest ITR of 46.11 bits / min is achieved at the 2.4-second epoch duration, the translation accuracy at this epoch duration is only 79.93 %. In order to ensure the accuracy of the acquisition-translation process when integrating it in the system, it is suggested that 3.4-second epoch duration should be utilized.

The largest problem with the acquisition-translation process is that it is extremely difficult to replicate performance with such high accuracy and ITR using current hardware components. At the moment, electrodes are fixed on the volunteer's scalp by medical tapes, which cannot guarantee stable connection and cause pain for the volunteer when removing the tapes from his hair. Besides, manual measurements of electrode positions are required every time we conduct experiments, which may lead to incorrect electrode placement. In fact, there are many other online experiments has been conducted, which combine both the SSVEP acquisition-translation and Inverse Kinematics processes. However, the accuracy recorded were significantly low, below 30% with all epoch durations. To address this issue, it is suggested that a proper EEG cap should be purchased or designed for future development. One of recommended EEG caps is BrainWave EEG cap of DUOMED.



Figure 24: BrainWave EEG cap of DUOMED.¹⁶

Despite the limitation in reproducibility, the performance of the process's algorithm can still be guaranteed, since it achieved a translation accuracy of 89.52% and an ITR of 39.13 bits/min with a fixed epoch duration of 4.356 seconds when testing with the 12JFPM dataset.

5.2 Testing on Inverse Kinematics Process

The tests on Inverse Kinematics process was taken separately from the system's operation to examine the its capability to move the end-effector to 64 positions within a 25.6 x 25.6 cm region, which also contained the 9 position associated with 9 stimuli used in the SSVEP acquisition-translation process. In this test, the operation system based on ROS 2 described in Section 2.2 was utilized, but the **BCI_NODE** was replaced by another node that enabled a tester input the desired position using keyboard. Besides, the angles of joint 4 and 6 were respectively fixed at 0 and $\pi/2$ to simplify the process.

¹⁶<https://www.duomed.com/en-NL/eeg-ltm/brainwave-eeg-cap>

5.2.1 Introduction of Testing Region

A chess board was used as testing region for the Inverse Kinematics process. It was a 25.6 x 25.6 cm square region, divided into 64 3.2 x 3.2 cm squares arranged as a 8 x 8 matrix. The numeration of these 64 positions and the arrangement of the testing region with respect to the robotic arm are described in Figure 25.

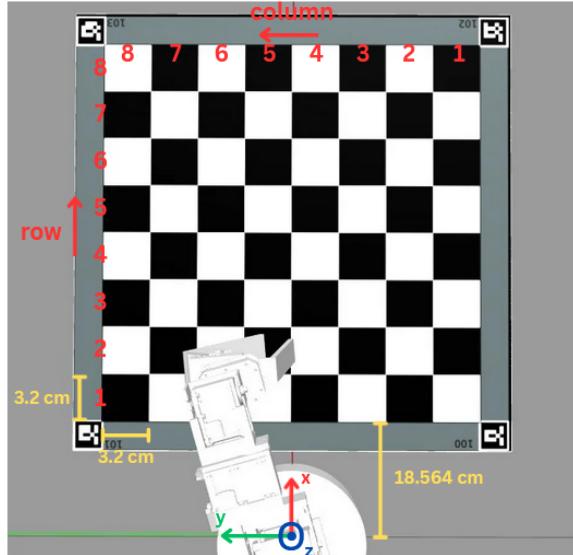


Figure 25: Specifications of the testing region.

5.2.2 Testing Results

The inverse kinematics tests were performed sequentially for each of the 64 target positions. For every position, the tester manually entered the indexes via keyboard, and the end-effector of the robotic arm was expected to reach the positions following Equation 62 regardless of orientation.

$$\begin{cases} x = 13.9 + 3.2 + 3.064 + (row - 1) \times 3.2 \\ y = -11.2 + (column - 1) \times 3.2 \\ z = 6 \end{cases} \quad (62)$$

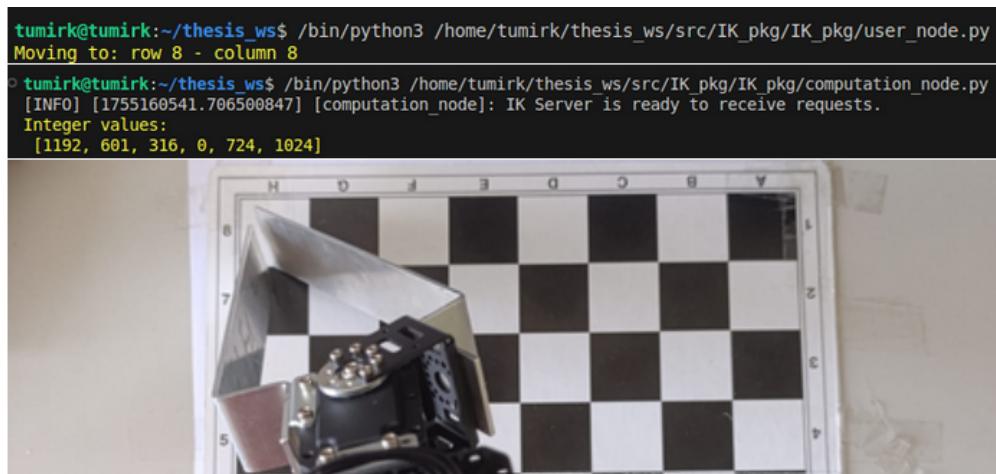


Figure 26: The end-effector is commanded to move to index (8, 8), it receives joint angles from the COMPUTATION_NODE and moves to the correct position.

After being tested carefully, the robotic arm's end-effector successfully reached all 64 positions in general without collision. Positional deviations associated with each index were recorded in Table 9.

Index	Target (x, y, z) [cm]	Measured (x, y, z) [cm]
(1, 1)	(20.164, -11.2, 6.0)	(20.20, -11.2, 5.8)
(1, 2)	(20.164, -8.0, 6.0)	(20.20, -8.0, 5.8)
(1, 3)	(20.164, -4.8, 6.0)	(20.20, -4.85, 5.8)
(1, 4)	(20.164, -1.6, 6.0)	(20.20, -1.55, 5.8)
(1, 5)	(20.164, 1.6, 6.0)	(20.20, 1.65, 5.8)
(1, 6)	(20.164, 4.8, 6.0)	(20.20, 4.75, 5.8)
(1, 7)	(20.164, 8.0, 6.0)	(20.20, 8.05, 5.8)
(1, 8)	(20.164, 11.2, 6.0)	(20.20, 11.25, 5.8)
(2, 1)	(23.364, -11.2, 6.0)	(23.20, -11.25, 5.0)
(2, 2)	(23.364, -8.0, 6.0)	(23.20, -8.05, 5.0)
(2, 3)	(23.364, -4.8, 6.0)	(23.20, -4.85, 5.0)
(2, 4)	(23.364, -1.6, 6.0)	(23.20, -1.55, 5.0)
(2, 5)	(23.364, 1.6, 6.0)	(23.20, 1.65, 5.0)
(2, 6)	(23.364, 4.8, 6.0)	(23.20, 4.75, 5.0)
(2, 7)	(23.364, 8.0, 6.0)	(23.20, 8.05, 5.0)
(2, 8)	(23.364, 11.2, 6.0)	(23.20, 11.25, 5.0)
(3, 1)	(26.564, -11.2, 6.0)	(26.0, -11.15, 3.0)
(3, 2)	(26.564, -8.0, 6.0)	(26.0, -8.05, 3.0)
(3, 3)	(26.564, -4.8, 6.0)	(26.0, -4.85, 3.0)
(3, 4)	(26.564, -1.6, 6.0)	(26.0, -1.55, 3.0)
(3, 5)	(26.564, 1.6, 6.0)	(26.0, 1.65, 3.0)
(3, 6)	(26.564, 4.8, 6.0)	(26.0, 4.75, 3.0)
(3, 7)	(26.564, 8.0, 6.0)	(26.0, 8.05, 3.0)
(3, 8)	(26.564, 11.2, 6.0)	(26.0, 11.25, 3.0)
(4, 1)	(29.764, -11.2, 6.0)	(29.0, -11.25, 1.9)
(4, 2)	(29.764, -8.0, 6.0)	(29.0, -8.05, 1.9)
(4, 3)	(29.764, -4.8, 6.0)	(29.0, -4.85, 1.9)
(4, 4)	(29.764, -1.6, 6.0)	(29.0, -1.55, 1.9)
(4, 5)	(29.764, 1.6, 6.0)	(29.0, 1.65, 1.9)
(4, 6)	(29.764, 4.8, 6.0)	(29.0, 4.75, 1.9)
(4, 7)	(29.764, 8.0, 6.0)	(29.0, 8.05, 1.9)
(4, 8)	(29.764, 11.2, 6.0)	(29.0, 11.25, 1.9)
(5, 1)	(32.964, -11.2, 6.0)	(31.7, -11.15, 1.2)
(5, 2)	(32.964, -8.0, 6.0)	(31.7, -8.05, 1.2)
(5, 3)	(32.964, -4.8, 6.0)	(31.7, -4.85, 1.2)
(5, 4)	(32.964, -1.6, 6.0)	(31.7, -1.55, 1.2)
(5, 5)	(32.964, 1.6, 6.0)	(31.7, 1.65, 1.2)
(5, 6)	(32.964, 4.8, 6.0)	(31.7, 4.75, 1.2)
(5, 7)	(32.964, 8.0, 6.0)	(31.7, 8.05, 1.2)
(5, 8)	(32.964, 11.2, 6.0)	(31.7, 11.25, 1.2)

(6, 1)	(36.164, -11.2, 6.0)	(35.0, -11.25, 0.3)
(6, 2)	(36.164, -8.0, 6.0)	(35.0, -8.05, 0.3)
(6, 3)	(36.164, -4.8, 6.0)	(35.0, -4.85, 0.3)
(6, 4)	(36.164, -1.6, 6.0)	(35.0, -1.55, 0.3)
(6, 5)	(36.164, 1.6, 6.0)	(35.0, 1.65, 0.3)
(6, 6)	(36.164, 4.8, 6.0)	(35.0, 4.75, 0.3)
(6, 7)	(36.164, 8.0, 6.0)	(35.0, 8.05, 0.3)
(6, 8)	(36.164, 11.2, 6.0)	(35.0, 11.25, 0.3)
(7, 1)	(39.364, -11.2, 6.0)	(38.0, -11.15, 0.0)
(7, 2)	(39.364, -8.0, 6.0)	(38.0, -8.05, 0.0)
(7, 3)	(39.364, -4.8, 6.0)	(38.0, -4.85, 0.0)
(7, 4)	(39.364, -1.6, 6.0)	(38.0, -1.55, 0.0)
(7, 5)	(39.364, 1.6, 6.0)	(38.0, 1.65, 0.0)
(7, 6)	(39.364, 4.8, 6.0)	(38.0, 4.75, 0.0)
(7, 7)	(39.364, 8.0, 6.0)	(38.0, 8.05, 0.0)
(7, 8)	(39.364, 11.2, 6.0)	(38.0, 11.25, 0.0)
(8, 1)	(42.564, -11.2, 6.0)	(41.1, -11.25, 0.0)
(8, 2)	(42.564, -8.0, 6.0)	(41.1, -8.05, 0.0)
(8, 3)	(42.564, -4.8, 6.0)	(41.1, -4.85, 0.0)
(8, 4)	(42.564, -1.6, 6.0)	(41.1, -1.55, 0.0)
(8, 5)	(42.564, 1.6, 6.0)	(41.1, 1.65, 0.0)
(8, 6)	(42.564, 4.8, 6.0)	(41.1, 4.75, 0.0)
(8, 7)	(42.564, 8.0, 6.0)	(41.1, 8.05, 0.0)
(8, 8)	(42.564, 11.2, 6.0)	(41.1, 11.25, 0.0)

Table 9: Target and measured positions within testing region.

5.2.3 Discussion

The results shown in Table 9 confirm that the inverse kinematics process is capable of accurately controlling the robotic arm within the defined workspace. Positional deviations, especially along the z axis, are actually due to the heavy end-effector and loose connections between links and the base of the arm, as they had been damaged a lot after several experiments for years. Therefore, the positional error tends to increase when the system tries to move the end-effector to positions further away from the origin, and finally even touching the floor when moving the 7th row. The solution for this problem is simply to tighten the screws that fix connections between links and the base, and to adjust the mass distribution in order to concentrate the majority of total mass near the origin by wisely replacing the current motors with lighter or heavier ones as needed.

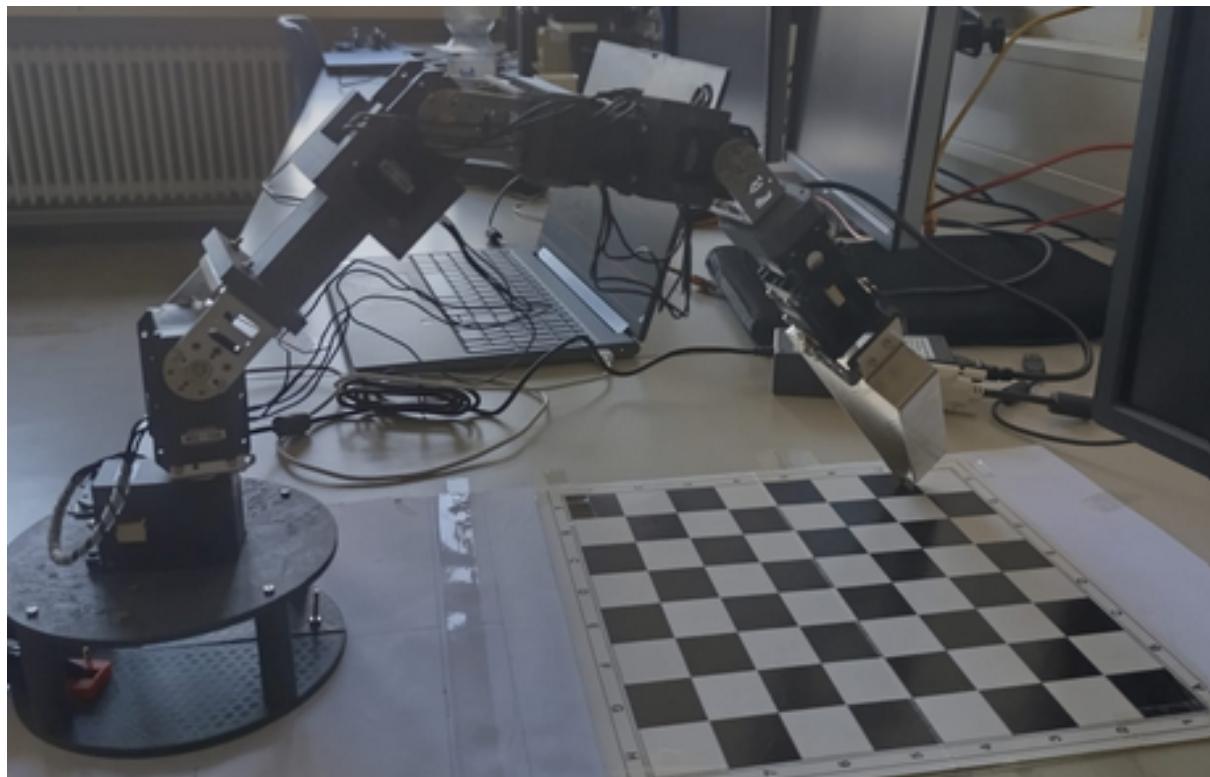


Figure 27: The arm is tilted when reaching index (8, 8), which is far away from origin.

6 Conclusion and Future Work

This thesis presented the development of a neural-based control system for a 6-Dof robotic arm using steady-state visual evoked potentials (SSVEPs) as the primary control signal. The proposed system integrates the SSVEP acquisition-translation process and the Inverse Kinematics process based on ROS 2 framework in order to allow a user to control the robotic arm without physical contact.

The SSVEP acquisition-translation process is implemented using an OpenBCI Cyton biosensing board, gold cup electrodes and Signa gel for signal acquisition, followed by preprocessing with digital bandpass filtering and common average reference (CAR) technique, and classification using canonical correlation analysis (CCA). Meanwhile, the Inverse Kinematics process is established using product of exponentials (PoE) and Newton-Raphson method. When being integrated to the system, the Inverse Kinematics process calculate suitable rotation angles for 6 joints to help the end-effector achieve the desired configuration obtained from the SSVEP acquisition-translation process. The user is only required to look at the visual stimulation box corresponding to their desired configuration.

The 2 aforementioned processes of the system were validated separately. Regarding the SSVEP acquisition-translation process, the testing results suggested that longer epoch durations could improve translation accuracy and the optimal trade-off for translation accuracy and information transfer rate (ITR) might be the 3.4-second epoch duration. Many other online experiments which combine both processes have also been conducted but all of them failed due to the lack of a proper EEG headcap. Besides, strategies to improve the signal acquisition such as purchasing or designing a proper EEG cap should be consider in order to enhance the system's accuracy as well as reusability. Regarding the Inverse Kinematics process, suitable rotation angles for 6 joints that help the end-effector achieve a desired configuration could be accurately solved but there were still some positional deviations caused by loose connections between robot's links and base.

Future work will focus on designing a flexible EEG cap and develop self-made visual stimulation source using the Psychophysics Toolbox library. This self-made visual stimulation source could enable a user to adjust the distance between stimuli, utilize the monitor's refresh rate of 160 Hz and control the stimuli better. Besides, some advanced translation algorithms, such as task-related component analysis (TRCA) or filter bank CCA (FBCCA), could be implemented to improve the translation accuracy and ITR, as well as increasing number of stimuli for more available configurations.

References

- [1] D. Steyrl, R. Kobler, and G. Müller-Putz, “On Similarities and Differences of Invasive and Non-Invasive Electrical Brain Signals in Brain-Computer Interfacing,” *Journal of Biomedical Science and Engineering*, vol. 9, pp. 393–398, 2016. doi: [10.4236/jbise.2016.9803](https://doi.org/10.4236/jbise.2016.9803).
- [2] V. Natarajan and A. Natarajan, “A review of non-invasive BCI devices,” *International Journal of Biomedical Engineering and Technology*, vol. 34, no. 3, pp. 205–233, 2020.
- [3] K. Navaneetham and D. Arunachalam, “Global population aging, 1950–2050,” in *Handbook of Aging, Health and Public Policy: Perspectives from Asia*. Singapore: Springer Nature Singapore, 2022, pp. 1–18, ISBN: 978-981-16-1914-4. doi: [10.1007/978-981-16-1914-4_154-1](https://doi.org/10.1007/978-981-16-1914-4_154-1).
- [4] N. Guo et al., “SSVEP-Based Brain Computer Interface Controlled Soft Robotic Glove for Post-Stroke Hand Function Rehabilitation,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 30, pp. 1737–1744, 2022. doi: [10.1109/TNSRE.2022.3185262](https://doi.org/10.1109/TNSRE.2022.3185262).
- [5] T. Bastos-Filho, A. Floriano, E. Couto, and R. J. Godinez-Tello, “Chapter 15 - Towards a system to command a robotic wheelchair based on independent SSVEP-BCI,” in *Smart Wheelchairs and Brain-Computer Interfaces*, P. Diez, Ed., Academic Press, 2018, pp. 369–379, ISBN: 978-0-12-812892-3. doi: <https://doi.org/10.1016/B978-0-12-812892-3.00015-7>.
- [6] M. Albán-Escobar, P. Navarrete-Arroyo, D. R. De la Cruz-Guevara, and J. Tobar-Quevedo, “Assistance Device Based on SSVEP-BCI Online to Control a 6-DOF Robotic Arm,” *Sensors*, vol. 24, no. 6, 2024, ISSN: 1424-8220. doi: [10.3390/s24061922](https://doi.org/10.3390/s24061922).
- [7] K. Värbu, N. Muhammad, and Y. Muhammad, “Past, Present, and Future of EEG-Based BCI Applications,” *Sensors*, vol. 22, no. 9, 2022, ISSN: 1424-8220. doi: [10.3390/s22093331](https://doi.org/10.3390/s22093331).
- [8] M. Aljalal, S. Ibrahim, R. Djemal, M. Alsulaiman, and G. Muhammad, “Comprehensive Review on Brain-Controlled Mobile Robots and Robotic Arms Based on Electroencephalography Signals,” *Intelligent Service Robotics*, vol. 13, pp. 539–563, 2020. doi: [10.1007/s11370-020-00328-5](https://doi.org/10.1007/s11370-020-00328-5).
- [9] A. Biasiucci, B. Franceschiello, and M. M. Murray, “Electroencephalography,” *Current Biology*, vol. 29, no. 3, R80–R85, 2019, ISSN: 0960-9822. doi: <https://doi.org/10.1016/j.cub.2018.11.052>.
- [10] J. Tang, T. Sun, and S. Wang, *Neural Signal Operated Intelligent Robot: Human-guided Robot Maze Navigation through SSVEP*, 2024. doi: <https://doi.org/10.48550/arXiv.2410.11867>. arXiv: [2410.11867 \[cs.HC\]](https://arxiv.org/abs/2410.11867).
- [11] Y. Ke, P. Liu, X. An, X. Song, and D. Ming, “An online SSVEP-BCI system in an optical see-through augmented reality environment,” *Journal of Neural Engineering*, vol. 17, no. 1, p. 016066, Feb. 2020. doi: [10.1088/1741-2552/ab4dc6](https://doi.org/10.1088/1741-2552/ab4dc6).
- [12] J. Wolpaw and E. W. Wolpaw, *Brain-Computer Interfaces: Principles and Practice*. Oxford University Press, Jan. 2012, p. 241, ISBN: 9780195388855. doi: [10.1093/acprof:oso/9780195388855.001.0001](https://doi.org/10.1093/acprof:oso/9780195388855.001.0001).

- [13] R. Kuš et al., “On the Quantification of SSVEP Frequency Responses in Human EEG in Realistic BCI Conditions,” *PLOS ONE*, vol. 8, no. 10, pp. 1–9, Oct. 2013. doi: [10.1371/journal.pone.0077536](https://doi.org/10.1371/journal.pone.0077536).
- [14] D. Pulver, P. Angkan, P. C. Hungler, and A. Etemad, “EEG-Based Cognitive Load Classification using Feature Masked Autoencoding and Emotion Transfer Learning,” in *Proceedings of the 25th International Conference on Multimodal Interaction (ICMI '23)*, Association for Computing Machinery (ACM), 2023. doi: [10.1145/3577190.3614113](https://doi.org/10.1145/3577190.3614113).
- [15] G. Bin, X. Gao, Z. Yan, B. Hong, and S. Gao, “An online multi-channel ssvep-based brain-computer interface using a canonical correlation analysis method,” *Journal of neural engineering*, vol. 6, p. 046 002, Jul. 2009. doi: [10.1088/1741-2560/6/4/046002](https://doi.org/10.1088/1741-2560/6/4/046002).
- [16] M. Borga, “Canonical correlation: A tutorial,” vol. 4, no. 5, 2001. [Online]. Available: https://www.cs.cmu.edu/~tom/10701_sp11/slides/CCA_tutorial.pdf.
- [17] R. MacAusland, “The moore-penrose inverse and least squares,” *Math 420: Advanced Topics in Linear Algebra*, pp. 1–10, 2014. [Online]. Available: <http://buzzard.ups.edu/courses/2014spring/420projects/math420-UPS-spring-2014-macausland-pseudo-inverse.pdf>.
- [18] S. Akram and Q. U. Ann, “Newton raphson method,” *International Journal of Scientific & Engineering Research*, vol. 6, no. 7, pp. 1748–1752, 2015.
- [19] P. Flores and H. Lankarani, *Contact Force Models for Multibody Dynamics*. Springer Cham, Jan. 2016, p. 130. doi: <https://doi.org/10.1007/978-3-319-30897-5>.
- [20] K. M. Lynch and F. C. Park, *Modern Robotics*. Cambridge University Press, 2017, ISBN: 9781316661239. doi: <https://doi.org/10.1017/9781316661239>.
- [21] M. Nakanishi, Y. Wang, Y.-T. Wang, and T.-P. Jung, “A Comparison Study of Canonical Correlation Analysis Based Methods for Detecting Steady-State Visual Evoked Potentials,” *PLOS ONE*, vol. 10, no. 10, pp. 1–18, Oct. 2015. doi: [10.1371/journal.pone.0140703](https://doi.org/10.1371/journal.pone.0140703).
- [22] T. Schreiber, “Measuring Information Transfer,” *Phys. Rev. Lett.*, vol. 85, pp. 461–464, 2 Jul. 2000. doi: [10.1103/PhysRevLett.85.461](https://doi.org/10.1103/PhysRevLett.85.461).
- [23] D. H. Brainard, “The Psychophysics Toolbox,” *Spatial Vision*, vol. 10, no. 4, pp. 433–436, 1997. doi: [10.1163/156856897X00357](https://doi.org/10.1163/156856897X00357).