Brain-Computer Interface for Position Control of a Robot Arm for Rehabilitative Applications

Thesis submitted by

Arnab Rakshit

Doctor of Philosophy(Engineering)

Department of Electronics & Tele-communication Engineering
Faculty Council of Engineering and Technology
Jadavpur University
Kolkata-700032, India
2023

Index No: 38/18/E

TITLE OF THE THESIS:
BRAIN-COMPUTER INTERFACE FOR POSITION CONTROL OF A ROBOT ARM FOR REHABILITATIVE APPLICATIONS

Name, Designation and Institution of Supervisor:

Prof. Amit Konar
Professor
Department of Electronics & Tele-communication Engineering
Jadavpur University
Kolkata-700032
India

List of Publications

List of Publications Related to Thesis

Journal(s)

- **Arnab Rakshit**, Amit Konar, and Atulya K. Nagar. "A hybrid brain-computer interface for closed-loop position control of a robot arm." IEEE/CAA Journal of Automatica Sinica 7, no. 5 (2020): 1344-1360.doi: 10.1109/JAS.2020.1003336. **(Impact Factor: 11.8)**

- **Arnab Rakshit**, Shraman Pramanick, Anurag Bagchi, and Saugat Bhattacharyya. "Autonomous grasping of 3-D objects by a vision-actuated robot arm using Brain–Computer Interface." Biomedical Signal Processing and Control 84 (2023): 104765. https://doi.org/10.1016/j.bspc.2023.104765. (**Impact Factor: 5.1**)

Conference Articles

- **Arnab Rakshit**, and Amit Konar. "Brain–Computer Interface for Fuzzy Position Control of a Robot Arm by Mentally Detected Magnitude and Sign of Positional Error." In Intelligent Systems and Human Machine Collaboration: Select Proceedings of ICISHMC 2022, pp. 163-177. Singapore: Springer Nature Singapore, 2023. https://doi.org/10.1007/978-981-19-8477-8_13

- **Arnab Rakshit**, Susenjit Ghosh, Amit Konar, and Monalisa Pal. "A novel hybrid brain-computer interface for robot arm manipulation using visual evoked potential." In Ninth International Conference on Advances in Pattern Recognition (ICAPR), pp. 1-6. IEEE.doi: 10.1109/ICAPR.2017.8593034.

List of Patents

NIL

List of Presentations in National / International / Conferences/ Workshops :

- **Arnab Rakshit**, and Amit Konar. "Brain–Computer Interface for Fuzzy Position Control of a Robot Arm by Mentally Detected Magnitude and Sign of Positional Error." In Intelligent Systems and Human Machine Collaboration: Select Proceedings of ICISHMC 2022, pp. 163-177. Singapore: Springer Nature Singapore, 2023.

Arnab Rakshit
24/07/23

Akonar

PROF. AMIT KONAR
CO-ORDINATOR
INTELLIGENT AUTOMATION
& ROBOTICS
E.T.C.E. DEPTT., J.U., KOL-32

Other Publications

- Shukla, Anjali, Arnab Rakshit, Amit Konar, Lidia Ghosh, and Atulya K. Nagar. "Decoding of mind-generated pattern locks for security checking using type-2 fuzzy classifier." In *2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1976-1981. IEEE, 2018.*

- Tabassum Hossain, Arnab Rakshit, and Amit Konar. "Brain-computer interface based user authentication system for personal device security." In *2020 International Conference on Computer, Electrical & Communication Engineering (ICCECE)*, pp. 1-6. IEEE, 2020.

- Arnab Rakshit, Rimita Lahiri, Sayan Ghosal, Abhirup Sarkar, Sanchita Ghosh, and Amit Konar. "Robotic link position control using brain computer interface." *In 2016 International Conference on Microelectronics, Computing, and Communications (MicroCom)*, pp. 1-6. IEEE.

- Ankan Dutta, and Arnab Rakshit. "Geometry perspective of estimating learning capability of neural networks." arXiv preprint arXiv:2011.04588 (2020).

- Arnab Rakshit, Anwesha Banerjee, Ankita Mazumder, Poulami Ghosh, Anilesh Dey, and D. N. Tibarewala. "Fractal analysis of EEG signals for studying the effect of cognitive stress on brain." *International Journal of Biomedical Engineering and Technology* 25, no. 2-4 (2017): 336-369.

- Arnab Rakshit, Anwesha Khasnobish, and D. N. Tibarewala. "A Naïve Bayesian approach to lower limb classification from EEG signals." In *2016 2nd International Conference on Control, Instrumentation, Energy & Communication (CIEC)*, pp. 140-144. IEEE, 2016.

- Arnab Rakshit, and Rimita Lahiri. "Discriminating different color from EEG signals using interval-type 2 fuzzy space classifier (a neuro-marketing study on the effect of color to Cognitive State)." In *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), pp. 1-6. IEEE, 2016.*

- Arnab Rakshit, Anwesha Banerjee, and D. N. Tibarewala. "Electro-oculogram based digit recognition to design assitive communication system for speech disabled patients." In 2016 *International Conference on Microelectronics, Computing and Communications (MicroCom)*, pp. 1-5. IEEE, 2016.

- Anwesha Banerjee, Arnab Rakshit, and D. N. Tibarewala. "Application of Electrooculography to estimate word count while reading text." In 2016 *International Conference on Systems in Medicine and Biology (ICSMB), pp. 174-177. IEEE, 2016.*

Arnab Rakshit
24/07/23

PROF. AMIT KONAR
CO-ORDINATOR
INTELLIGENT AUTOMATION
& ROBOTICS
E.T.C.E. DEPTT., J.U., KOL-32

Statement of Originality

I, Arnab Rakshit, registered on 18.06.2018 do hereby declare that this thesis entitled "Brain-Computer Interface for Position Control of a Robot Arm for Rehabilitative Applications" contains literature survey and original research work done by the undersigned candidate as part of Doctoral studies.

All information in this thesis have been obtained and presented in accordance with existing academic rules and ethical conduct. I declare that, as required by these rules and conduct, I have fully cited and referred all materials and results that are not original to this work.

I also declare that I have checked this thesis as per the "Policy on Anti-Plagiarism, Jadavpur University, 2019", and the level of similarity as checked by iThenticate software is 3%.

Arnab Rakshit

Signature of Candidate

Date: 24/07/23

Certified by Supervisor:

PROF. AMIT KONAR
CO-ORDINATOR
INTELLIGENT AUTOMATION
& ROBOTICS
E.T.C.E. DEPTT., J.U., KOL-32

Prof. Amit Konar

Professor

Department of Electronics & Tele-communication Engineering

Jadavpur University

Place: Kolkata

Date: 24/7/23

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the thesis entitled "Brain-Computer Interface for Position Control of a Robot Arm for Rehabilitative Applications" submitted by Arnab Rakshit, who got his name registered on 18.06.2018 for the award of Ph.D.(Engg.) degree of Jadavpur University is absolutely based upon his own work under the supervision of Prof. Amit Konar and that neither his thesis nor any part of the thesis has been submitted for any degree/ diploma or any other academic award anywhere before.

Akon 24/7/23

PROF. AMIT KONAR
CO-ORDINATOR
INTELLIGENT AUTOMATION
& ROBOTICS
E.T.C.E. DEPTT., J.U., KOL-32

Prof.Amit Konar
Professor
Department of Electronics & Tele-communication Engneering.
Jadavpur University
Kolkata-700032, India

Acknowledgements

I would like to seize this moment to extend my appreciation to all those who provided me with unwavering support during my doctoral studies. I am deeply grateful for their invaluable mentorship, constructive feedback, and kind counsel throughout this journey.

I would like to express my deepest gratitude to my supervisor, Prof. Amit Konar, for his unwavering support, encouragement, and profound expertise. Throughout the journey of my thesis, he has been a constant source of inspiration, guiding me to overcome obstacles and navigate the challenges not only in my research but also in my personal life. I consider myself extremely fortunate to have had him as a mentor, friend, and guiding light.

I am deeply honored to extend my heartfelt gratitude to Prof. Manotosh Biswas, the Head of the Department of Electronics & Telecommunication Engineering at Jadavpur University. I would also like to express my sincere respect to Prof. Mrinal Kanti Naskar, Prof. P. Venkateswaran, and Prof. Piyali Basak, the Director of the School of BioScience and Engineering. Throughout the duration of my research program, their continuous support and constant encouragement have been invaluable.

I am also thankful to the Council of Scientific and Industrial Research (CSIR), Govt. of India for providing me the financial aid for three years for pursuing my Ph.D. studies.

I would like to convey my deepest gratitude to Prof. D. N.Tibarewala, who is no longer with us. I am forever grateful for the invaluable lessons, insights, and encouragement I received from Prof. Tibarewala. His guidance has shaped my perspective, deepened my understanding, and empowered me to strive for greatness in my academic and professional endeavors.

I am immensely grateful for the wholehearted support my fellow batch mates and senior researchers, without whom my research work would not have been possible. I extend my deepest thanks to Dr. Saugat Bhattacharyya, Dr. Arup Sadhu, Ankan Dutta, and Uddhav Sen for their valuable contributions and assistance throughout the course of my Ph.D. study. In particular, I would like to express my heartfelt appreciation to my friend and junior fellow researcher, Mr. Susenjit Ghosh. From the very beginning of my Ph.D. journey, he has provided me with unconditional help, support, and encouragement.

Finally, I would like to express my utmost respect and gratitude to my beloved parents, Mr. Prodip Rakshit and Mrs. Suniti Rakshit, as well as my younger brother Ayan. Their unwavering love, affection, care, and trust have been a constant source of strength and inspiration throughout my journey. Additionally, I would like to pay my respects to my late maternal grandfather, Late Hari Pada Sain. Though he is no longer with us, his guidance and blessings have always remained in my heart, serving as a guiding light in my endeavors. I am truly blessed to have such a loving and supportive family.

Arnab Rakshit

To my beloved mother Mrs. Suniti Rakshit, whose unwavering love, support, and encouragement have been the guiding light throughout my academic journey, this thesis is dedicated to you.

Preface

Brain-Computer Interface (BCI) refers to direct interfacing of a human/animal brain with a computer to acquire, understand and decode the brain states by the computer. BCI aims at achieving 2 fundamental motivations. First, it provides a non-muscular channel of communication between the brain and the computer to understand brain functioning. Second, it offers one modality to control the brain states/activity by suitably chosen stimuli. BCI has gained immense popularity over the last 2 decades for its widespread applications and versatility in neural rehabilitation (of patients suffering from neuro-motor disorders), mind-controlled external interface for gaming, auto-prompting cognitive failures in driving, brain-response based collaborative decision-making for defense and other related applications, and many others. One of the common problems in neural rehabilitation by BCI means is concerned with position control of the end-effecter associated with a robot arm. This has interesting and useful applications in the design of artificial prosthetic limbs for neuro-motor disabled patients.

The thesis deals with the design and development of new control strategies for BCI-based position control of prosthetic robotic limbs, to rehabilitate a person with provision for controlling its motion in desired direction and orientation. The study undertaken in the thesis begins with single-link position control. Once a link of the robotic arm is correctly positioned with high positional accuracy, the BCI system is expected to generate a control command for the next selected link. The process of link selection of the robot arm thus is continued by the subject's own choice with the available signaling support by the BCI system, until the target position of the end-effecter is reached

Controlling motor movement (planning and execution) in the desired manner is an important issue for motion control of a BCI-based robotic arm. For example, turning a robotic link by the desired angle requires (mentally imagined) motor switch to activate an electromechanical motor for turning for a given duration at constant speed (until the targeted position of the link is reached), and deactivate the motor for a pause (no turning). Naturally, such motion corresponds to an on-off control strategy in traditional control theory. Such kind of on-off control is needed for position control of individual links. When a number of links are to be turned in the desired order, the task becomes relatively complex. Hence, the link selection is to be carried out first by utilizing one evoked potential, called Steady-state visual evoked potential (SSVEP). After the link is selected, the position of the link is displaced by the desired angle by controlled duration motor activation followed by no motor-activation.

The motivation of the present research is to generate control commands for motion of the individual links in a desired manner, so as to position the end-effecter at the targeted

Arnab Rakshit

position. One well-known control strategy that needs special mention in the present context is concerned with Motor Imagery based motor activation and Error-Related Potential (ErrP) based motor deactivation to position a link in the target position. The strategy is easy from the implementation point of view but results in large steady-state error in the position of the end-effecter. Additionally, the said scheme allows only fixed-order link selection because of the limitation of BCI-based communication between the subject and the controller. The thesis aims at handling the above problems by 3 distinctive approaches, presented in Chapters 2-4. In addition, the cognitive load of the subject in the BCI-based position control schemes usually is high. The thesis proposes one solution to reduce the cognitive load of the subject, by utilizing brain signals acquired to understand subject's intension, and generating control and actuation commands by additional automation to perform autonomous reaching and grasping of the target object. The additional automation has provisions for self-sensing, local control and actuation.

The thesis includes 5 Chapters. Chapter 1 provides a thorough review of the existing strategies for position control by brain-computer interface means. It begins with an introduction to Brain-Computer interfaces (BCI), and reviews the scope of closed-loop position control in the context of brain-inspired rehabilitative robotics. Traditional electroencephalography (EEG)-based brain signals used in the existing BCI interface are outlined next. The discussion spans over 4 important brain signals, including Event-Related De-synchronization followed by Event-Related Synchronization (ERD-ERS), Error-Related Potential (ErrP), P300, and Steady-State Visual Evoked Potential (SSVEP). The importance of the above signals in BCI-based position control of a robot arm in 2D and 3D work spaces are explained in brief. Although BCI earned a great success in both laboratory and clinical environments, traditional BCI-based control even today lacks a formal and precise nomenclature and taxonomy. The chapter makes an early attempt to classify the existing BCI- based control strategies into 4 major heads: i) purely open loop control, ii) closed-loop with visual feedback, iii) closed-loop with encoded feedback, iv) Auxiliary control based closed-loop system. All these four types of control strategies have their relative merits and demerits.

However, the last strategy is more important than the rest for its inherent advantages of relieving the cognitive load of the subject participating in the BCI-loop. The auxiliary controller in addition offers high precision control, which usually is difficult to achieve by human agents. A brief overview on the architectural details of the fourth scheme stated above is given next in the section. The architecture of scheme 4 includes a Neural Interface and an Auxiliary Controller in cascade with the human brain in the loop. The modules involved in building the neural interface are discussed next. The functional architecture of the auxiliary control strategy is reviewed with reference to P-I-D control, vision-based shared control, and model predictive control. The state-of-the-art research on BCI-based control schemes is reviewed in detail in the later part of the Chapter. The chapter ends with a discussion on the scope of the thesis.

Arnab Rakshit

Chapter 2, 3, and 4 are original contributions of the present thesis, demonstrating new approaches to controller design for BCI-based rehabilitative applications with an aim to reduce steady-state positional error, settling time and also attempting to improve the relative stability of the proposed control systems by judiciously choosing the parameters of the controller. Chapter 2 is concerned with an important issue of velocity modulation of the individual links of a robot arm, each time the link crosses the desired target position. The velocity modulation includes speed reversal with a gradual declination in the amplitude of speed, each time the link or the end-effecter crosses the target position. The velocity modulation is advantageous in the sense that it has a tighter control over the position of the end-effecter. In the present implementation of velocity modulation, given in Chapter 2, two control loops are utilized, where the outer loop takes care of position control and the inner loop attempts to control the speed of the end-effecter. The 2-loop cascade control naturally has improved performance in time-response, such as settling time and positional overshoot. Besides handling steady-state error, the additional merit of the proposed control strategy includes the design of a brain-actuated controller that produces a control command based on the occurrence of a P300 signal liberated by the subject in response to the zero crossings of the positional error signal. A stability analysis undertaken using root contour plots demonstrates that the proposed brain-actuated control system is stable for wider variations of controller parameters. An analysis of relative stability is undertaken to determine the optimal settings of the controller parameters that jointly improve settling time and DC gain. The analysis of stability further envisages that the choice of the initial speed of the end-effecter is of primary concern to limit the stability of the overall system. A large initial speed may result in instability as it may allow sustained oscillations for prolonged durations, whereas a small initial choice of velocity improves peak-overshoot at the cost of additional settling time. The choice of the initial velocity thus is an important concern for the proposed brain-actuated control system. The experimental evidence reveals that the system achieves an average success rate of 92.1% in a position control task, which is significantly higher than the other state-of-the-art approaches. The settling time of the system also came down 9.92s along with the significant reduction of peak-overshoot and positional steady-state error.

The above chapter also provides a thorough comparison of the controller performance with other hybrid-BCI-based position control schemes. However, it is found that the proposed control strategy outperforms the other approaches in terms of four performance metrics i.e. success rate, steady-state error, peak-overshoot and settling time.

Chapter 3 examines the problem of controller design by means of fuzzy logic. The work has importance over the preceding work presented in Chapter 2 with reference to the extraction of more information about the assessment of positional overshoot/undershoot by the BCI system itself to offer better freedom to the controller to improve positional accuracy in steady-state. It is important to note that the ErrP signal

Arnab Rakshit

released by the experimental subject during the test phase of the control system just refers to the information that the subject just has experienced a zero-error crossing in the positional offset, but it hardly indicates the magnitude of positional error due to the movement of the end-effecter. The proposed fuzzy control policy can take decision about speed setting of the end-effecter or the links based on 2 parameters: magnitude and sign of errors. It is therefore important to mention here how we calibrate the magnitude of positional error. The magnitude of error can be approximately assessed by checking whether the end-effecter or the desired link crosses some user-defined landmark positions, spaced apart by small fragments of distance around the target position. Fixing landmarks undoubtedly creates additional complexity, but helps the BCI-system to determine the approximate magnitude and sign of error. A fuzzy-rule-based system has been designed to set the speed of the selected link or the end-effecter in the desired direction based on the sign and magnitude of error. The control scheme was verified with the participation of twelve volunteers. An analysis of the performance of the proposed fuzzy controller reveals that the peak overshoot, settling time, and steady-state error decreased significantly with respect to the same of state-of-the-art algorithms.

In the above two works, the user is solely responsible for creating the complex trajectory of the robot arm using mental commands to reach and grasp any desired object. Such manual complex trajectory creation imposes a high cognitive load on human subjects, especially for the grasping task which requires highly precise position control to align the end-effecter with the desired object for proper grasping. Besides, the above two works employ a control scheme that controls a single link of the robot at a time, i.e., instead of controlling the end-effector motion directly, the subject controlled the individual links.

The work presented in Chapter 4 eliminates the above drawbacks by employing a vision-based novel shared controller that enables the robot autonomously reach and grasp the desired object with minimal brain commands. The subject in the present scheme is relieved from planning a complex trajectory for the robot link to align it with the desired object. A 3D vision camera is mounted on the moving robot arm to capture the live feed of the surrounding environment while the participating subject observes the feedback of the camera on a digital monitor. The proposed system employs an image processing algorithm to automatically localize the objects present within the field of view (FOV) of the camera and to blink the centroid of the objects on the monitor. The subject performs two simple tasks to virtually select the desired object present in the surrounding environment. First, he/she performs the motor imagery to rotate the camera in 3D space (rotating the 1st and 5th joint of the robot) to bring the desired object into its Field of view. Once the object is visible in the monitor, he/she needs to focus on the target object and he is expected to release P300 at the instant the target object flashes. The control logic is designed in such a way that whenever a P300 is received, the system registers the location of the object and actuates the end-effector of the robot to reach the

Asnab Rakshit

target location using inverse kinematics.

Next, a CNN-based novel robotic grasp detection network named Overlapping Object Grasp Net (OOGNet) is employed to precisely grasp the target object by a parallel plate gripper mounted at the end of the robot arm. The proposed Overlapping Object Grasping Network (OOGNet) generates a grasp rectangle, bounding box, and object class for each object in the image, thus associating each predicted grasp with its object. A custom loss function is proposed for simultaneous object and grasp detection to affiliate each estimated grasp with its corresponding object. The grasp rectangle is converted to the gripper configuration consisting of five parameters (center coordinate of the grasping rectangle, the height of the parallel plates of the gripper, the distance between the two parallel plates of the gripper, and the orientation of the grasping rectangle) for accurate grasping of the object. The network is also capable of grasping the desired object even if the object is partially overlapped by other objects.

As the entire reaching and grasping phase of the robot is made autonomous in the present thesis, the proposed scheme relieves the subject from complex manual trajectory planning, which reduces the overall cognitive load of the participating subject. The scheme also requires very little subject involvement in achieving a task (such as object pick and place) compared to existing state-of-the-art BCI schemes. Such minimal involvement of the human subject reduces the error arising from BCI decoding performance, hence increasing the overall task achievement accuracy. In addition, the system achieves a zero-steady state error within a negligible settling time. The control scheme was verified with the voluntary participation of 10 subjects. The task completion accuracy of 93.4% is achieved on average using the proposed control strategy. Most importantly the positional steady-state error came down to a negligible magnitude while the peak overshoot became almost zero.

The proposed grasping network was also tested separately in the presence of both a single object and multiple objects. In the single-object layout, the system was able to correctly grasp the object with 97.8% accuracy while in the multi-object layout, the network achieved 96.4% accuracy to precisely grasp any object.

The chapter also provides a comparison of the cognitive workload developed in the subjects while operating a robot using their mental commands following three distinct BCI strategies (including the proposed strategy). NASA-TLX questionnaire survey developed by NASA Ames Research Center is used to assess the workload of the subjects. The workload assessment in this study employs a comprehensive rating system comprising six distinct dimensions: Mental Demands, Physical Demands, Temporal Demands, Performance, Effort, and Frustration. Participants are asked to assign ratings to each sub-scale based on their assigned tasks. Once the rating process is done, they participate in pairwise comparisons to determine the most influential sub-scale. The overall score of the test is calculated by assigning weights to the sub-scales based on the fre-

Aronab Rakshit

quency of selection during the comparison task. The result shows that the proposed task imposes the least cognitive load on the participating subjects compared to the other manual trajectory planning-based state-of-the-art BCI robot manipulation techniques.

Chapter 5 provides a self-review of the thesis and also examines the scope of possible extension of the thesis in both theory and practice.

Arnab Rakshit
Department of Electronics & Tele-communication Engineering.
Jadavpur University
Kolkata-700032, India

Contents

List of Figures

List of Tables

Chapter 1

Introduction to BCI-Based Motor Control

This chapter provides an introduction to the subject of motion control using Brain-Computer Interface (BCI). It begins with a classification of BCI-based controllers, and gradually explores various control policies used in standard BCI systems. The different modules of a closed-loop BCI are discussed in brief with their current state-of-the-art approaches. Next, the chapter provides a thorough review of recent approaches to BCI-based robotic device manipulation within the context of closed-loop BCI. The chapter also provides an outline of the electroencephalography (EEG)-based brain signals commonly used in closed-loop BCI-based robot control tasks. The chapter ends with a discussion of the scope of the present thesis.

1.1 Introduction

Human brain comprises several trillions of neurons connected in tandem and parallel to serve as the basic infrastructure of knowledge/information processing. Although the complete picture of brain functionality remained a mystery to this date, there remains evidence of local processing of signals pertaining to perception, memory and learning, and motor control and coordination by specific regions/lobes of the brain. It is noted that neurons engaged in processing different cognitive tasks had evolutions over several decades to specialize themselves in handling the selected tasks. For instance, neurons responsible for processing olfactory signals usually have long axonal lengths to carry messages to the olfactory bulbs from the receptor end [1]. In contrast, the neurons participating in memory tasks are relatively small in length to carry messages for storing in the Long-Term Memory (LTM) from the Working Memory (WM) [2]. The motor cortex in the human brain takes the responsibility to generate the necessary commands for voluntary actuation and/or control of the motor actions intended by the subject. The Parietal lobe also has an important role in planning the intended motor action. In fact, there exists coordination between the Parietal lobe and the motor cortex to plan and execute a motor task [3]. In the present research, we emphasize greatly the decoding of motor actions. Such decoding is needed to execute the intended motor task by an artificial robotic limb. Several strategies of BCI-based motor control are available in the literature. We would discuss them in detail in a subsequent section. Capturing the motor planning and motor execution requires brain-signal acquisition equipment. Different modalities of brain signal/image acquisition techniques are available in the literature. A few of these that deserve mentioning include electroencephalography (EEG), functional Magnetic Resonance Imaging (f-MRI), Positron Emission Tomography (PET), functional Near Infrared Spectroscopy (f-NIRS) and Electrocorticography (ECOG). EEG is preferred to other brain signal/image acquisition devices for its excellent temporal response, portability, and low cost, and above all its non-invasive characteristics. In this thesis, we would restrict ourselves to examining the response of BCI-driven motor activations using the EEG modality.

1.2 Closed Loop BCI

Brain-Computer Interfacing (BCI) refers to interfacing the human/animal brain with a computer, bypassing the muscular pathways of communication, to understand and decode brain activity by the computer. The primary contribution of this technology lies in the field of rehabilitation and mainly focuses on controlling robotic prostheses with mental commands. However, there are other valuable contributions also, such as the development of a BCI-based virtual keyboard[4][5], wheelchair control[6][7], BCI-based gaming[8][9], etc. The use of BCI in rehabilitation was designed to enhance the quality of life for individuals experiencing various conditions, including complete or partial paralysis, motor difficulties associated with locked-in syndrome, cerebral palsy Amyotrophic Lateral Sclerosis (ALS), amputations, and cranium trauma.

BCI systems are of 2 common types: Open-loop systems and Closed-loop systems. In an open-loop BCI, the subject either by being stimulated externally or by mentally imagined

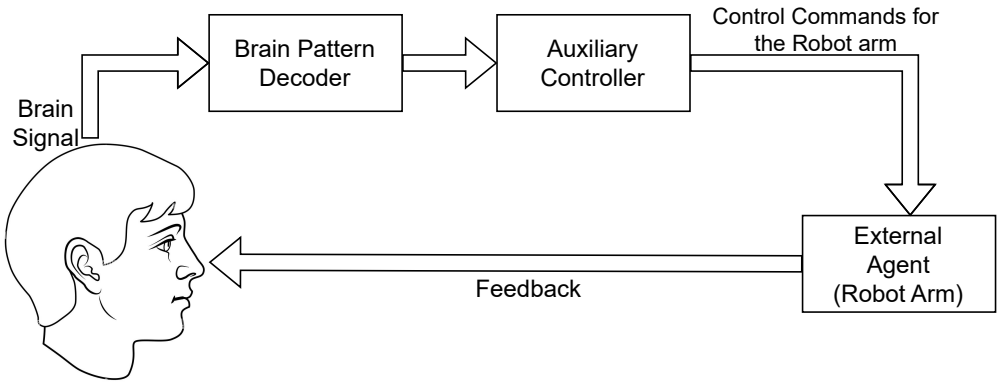

Figure 1.1: Schematic Diagram of a Closed-loop BCI system

cognitive tasks releases certain brain signals, which are used up by the associated system as open-loop control signals/commands. These open-loop control signals/commands may activate a relay or a switch to perform the desired task. In the open-loop system, the external state of the controlled agent does not influence the current brain state of the subject.

In a closed-loop BCI, a feedback measure of the targeted action is generated by suitable sensors to offer necessary feedback to the subject to assist him in moderating his decisions to release necessary brain-actuated commands in the next time slot. Hence the control action of the human subject is influenced by the present state of the external device. As an example, a subject controlling a robotic arm receives visual feedback about the position of the end-effector in a closed-loop system. Such feedback establishes a both-way communication between the human and the external agent, which is proven to be highly useful to treat severe neurological disorders like depression, epilepsy, Parkinson's disease, and obsessive-compulsive disorder (OCD) by stimulating a part of the brain based on feedback.

Apart from medical use, the neuro-feedback signal has also got widespread usage in robotic prosthesis control as it enables the human subject to determine how well he/she can control the robot and reach the desired target and participate in the task interactively. It also enables them to take corrective action in case an error occurs. The additional information in terms of the feedback signal has the capability to alter the brain state of the human subject to achieve the desired performance. A schematic overview of a general closed-loop system is given in Fig. 1.1.

Literature shows that various types of feedback have been used to provide additional information about the external agent to the operating subject. Such feedback includes visual feedback, vibrotactile feedback, electrical or magnetic stimulation, and optogenetic and sonogenetic feedback. Among them, visual feedback is the simplest yet powerful form of feedback, hence the current thesis considers visual feedback for designing the closed-loop Brain-machine Interface. The visual feedback is incorporated into the system through different modalities like P300, SSVEP, and ErrP brain patterns, which are utilized here to design the brain-actuated controller to actuate the robot arm with the user intent.

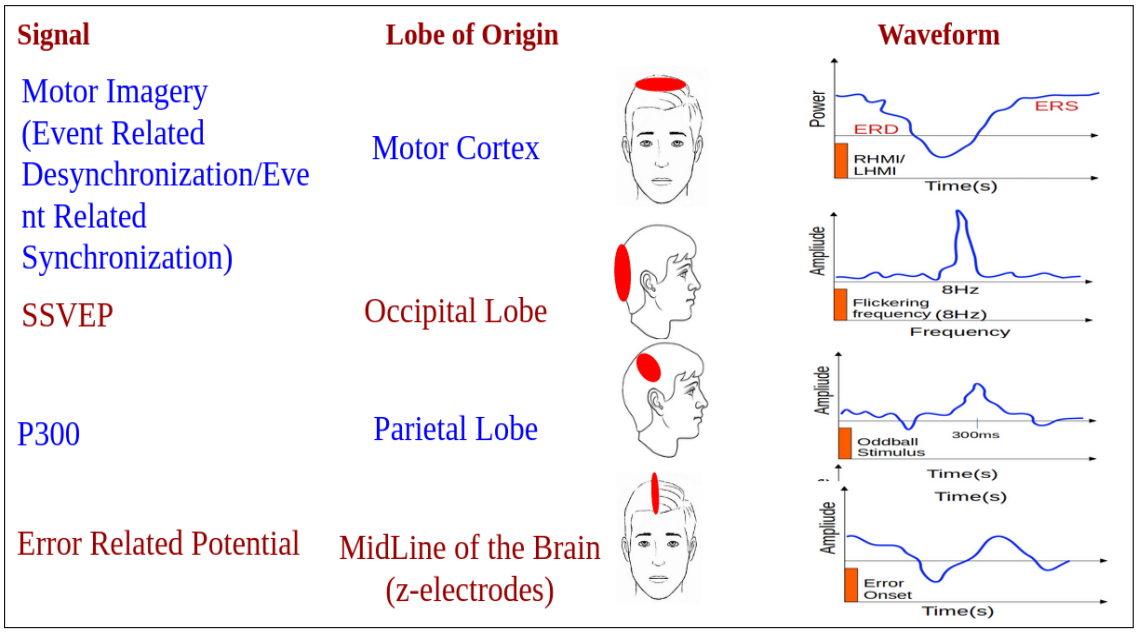

Figure 1.2: Different brain signals used in the BCI along with their waveform shape and origin.

1.3 Commonly used Signals in EEG-BCI System

There exist quite a few brain signals which are used to design different BCI-based systems, among them Motor Imagery (MI) signal, P300, ErrP, and Steady-state visual Evoked potentials are widely used in EEG-based BCI systems for their high detection accuracy through EEG. Fig.1.2 provides a ready reference of the commonly used BCI signals and their origin and shape. A details description of the signals is given below.

1.3.1 Motor Imagery Signal

When a person undertakes imagination about the movement of his voluntarily controllable body parts/organs, such as upper and lower limbs, jaws, etc., the process is referred to as motor imagination, and the imagined item is often called motor imagery (MI). On the other hand, when a person executes the movement of his voluntarily controllable organs, the process is referred to as motor execution (ME). It is apparent from the BCI literature [10] that during both motor imagination and execution phases, the human brain releases a specialized Event-Related Potential (ERP), which primarily has two main temporal segments. The first component, called Event-Related De-Synchronization (ERD), which lies in the Rolandic Mu band (8-12 Hz) and beta band (13-19 Hz) appears on-screen as a waveform of gradually diminishing amplitude. The ERD originates in the Parietal lobe during motor imagination with a peak at the Pz electrode and also from the Motor Cortex region during the motor execution phase with a peak at Cz electrode. The second segment, called Event Related Synchronization (ERS) signal, which appears in the beta band (13-18 HZ), originates from the motor cortex (Parietal lobe) and is released after motor execution (imagination) is stopped. The ERS signal exhibits a gradually increasing power until the previous power level that appeared before the occurrence of ERD is restored. The ERD-ERS

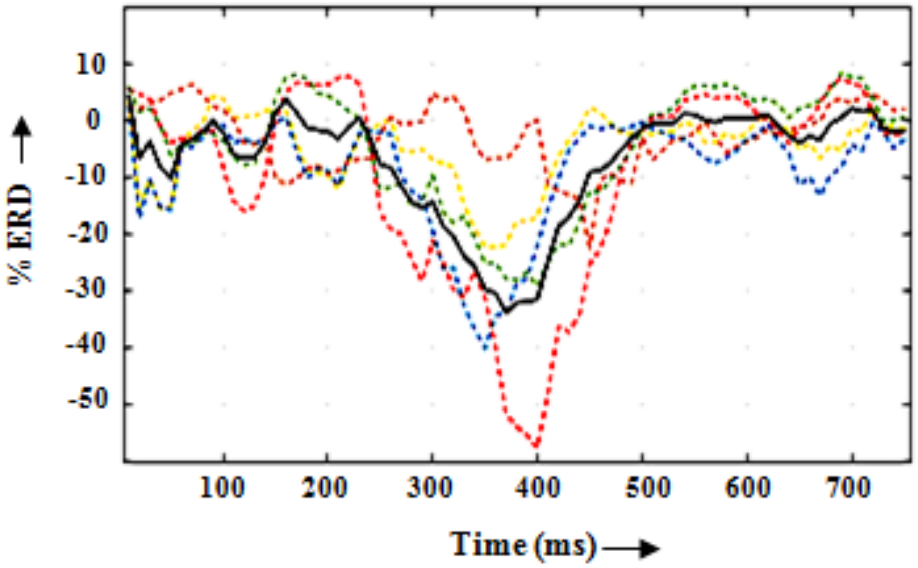

Figure 1.3: Figure showing multiple traces of motor-imagery brain signal with the dotted lines. The grand average is marked with a solid line.

together looks like a v-type of signal with a peak at the left motor cortex for the Right-Hand Motor Imagery or Execution (RHMI or RHME) and a peak at the Right Motor Cortex for the Left-Hand Motor Imagery or Execution (LHME). This is due to the inherent contra-lateral connection between our brain and the body. In most of the BCI literature that employs ERD/ERS signals, MI/ME are undertaken for open-loop control of an external device, such as a robotic car [11], [12][13], artificial robotic link [14], and the like. In recent times, researchers have shown keen interest to develop closed-loop position control applications for a robot arm by employing one additional signal along with ERD/ERS.

1.3.2 ErrP Signal

In a BCI-based robotic device control application, the additional signal used in conjunction with ERD/ERS is Error-Related Potential (ErrP) and P300. The ErrP signal is used to detect the occurrence of a positional overshoot around a given target position. The subjects are expected to liberate ErrP signal when they observe any person or a robotic device committing any error or they themselves commit any error. The signal can be thought of as the response of the brain to an erroneous action.

The ErrP brain signal is characterized by the occurrence of Error Related negativity followed by Error Related Positivity (Pe). ERN[15] is a sharp negative going peak that is generated around 50-100ms after the subject observes/commits any error. On the other hand, Error Related Positivity is a positive going peak that appears right after the ERN[16].

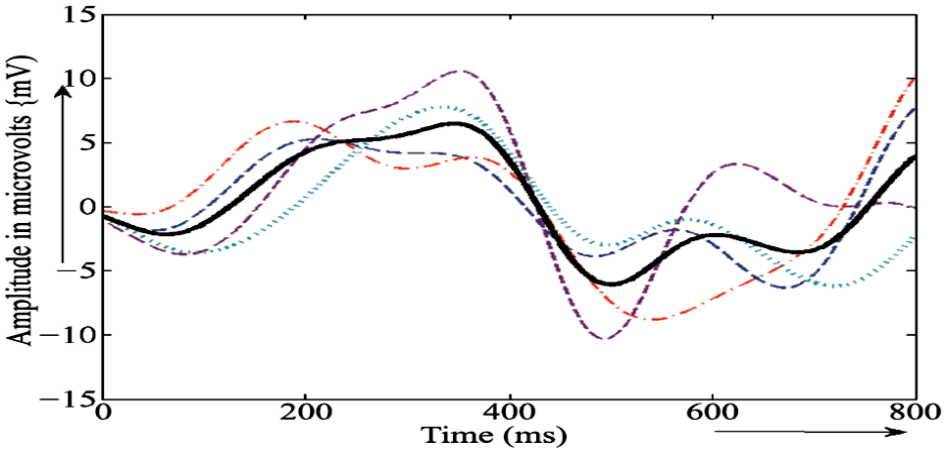

Figure 1.4: Figure showing multiple trials of Error Related Potential (ErrP) brain patterns in dotted lines. The grand average of the trials is shown with the solid line.

The Pe can be further categorized into two components: frontocentral and centroparietal. The frontocentral Pe, which is associated with the P3a, emerges immediately following the ERN (error-related negativity). On the other hand, the late Pe occurs in the centroparietal region and has a latency of 200-400 ms after the error. This late Pe component appears to be linked to the P3b[17]. Fig.1.4 shows the multiple traces of the ErrP signal and their grand average.

1.3.3 P300 Signal

Another important signal used in BCI is the oddball signal which is commonly known as P300. Such brain signal is generated as a response to odd, rare, or infrequent stimuli [18]. When a subject recognizes any rare or odd stimulus, a positive going peak after approximately 300ms of onset of the stimulus is generated over the parietal, central, and frontal regions of the brain and mainly distributed over the "z" lines of electrodes,i.e., $Pz, Cz, and Fz$.

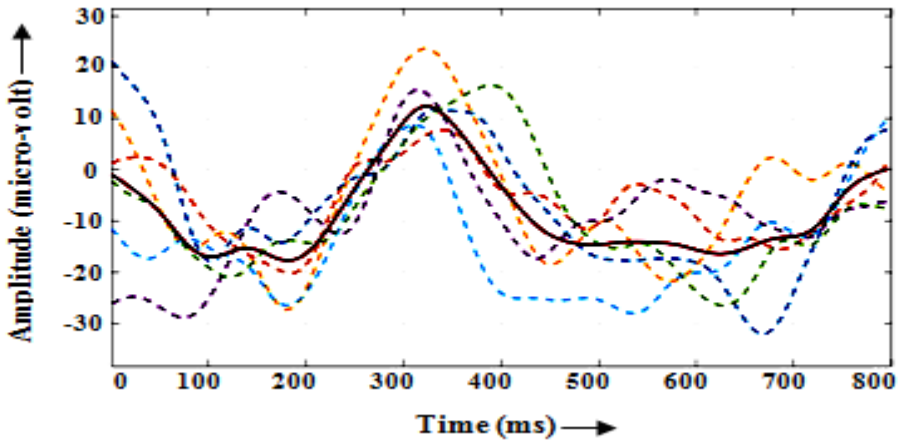

Figure 1.5: Figure showing P300 brain patterns in the dotted lines. The grand average of the trials is marked with a solid line.

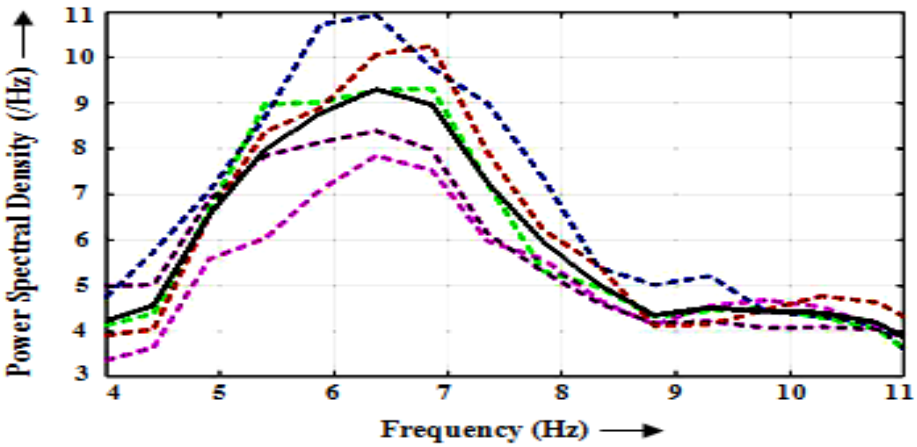

Figure 1.6: Figure showing SSVEP brain patterns in the dotted lines. The grand average of the trials is marked with a solid line.

However, the signal amplitude is highest in the *Pz* electrode and gradually attenuates with the distance over the scalp. The latency of the P300 signal usually varies between 250-750ms. The P300 signal, which usually responds to an oddball stimulus, here has been utilized to check the possible overshoot of the end-effector of a robot arm around a given target position. A typical P300 waveform is presented in Fig.1.5.

1.3.4 Steady-State Visual Evoked Potential

The third important class of signal is Steady-state visual evoked potential. Such a signal is generated as a response of the brain to repetitive visual stimuli. It is observed that a subject when perceives repetitive visual stimuli or gazes at flickering sources (such as flickering LED), a particular brain pattern modulated by the fundamental source frequency is generated in the primary visual cortex of the brain, which is popularly known as SSVEP. SSVEP along with the source frequency also contains the harmonics of the source frequency, however, their amplitude is relatively low. The frequency of the visual source stimulus ranges between 3.5Hz - 75Hz. The SSVEP signal has been extensively used in BCI for its high recognition rate and high information transfer rate. Another major advantage of using SSVEP is the minimum subject preparation time compared to the other modalities such as MI, ErrP, etc. SSVEP signal (in response to 6.5Hz flickering frequency) of multiple subjects is presented in Fig.1.6. It is evident from the figure that the signal has the highest amplitude near 6.5Hz frequency which is the frequency of the source stimulus.

1.4 Closed-Loop Position Control Using MI signals

In this section, we present a simple scheme for BCI-based position control in one dimension. Consider Fig. 1.7, where the position control of a single-link robotic arm is considered. Given a start position and a target position on the circular track. A motor-controlled robotic link needs

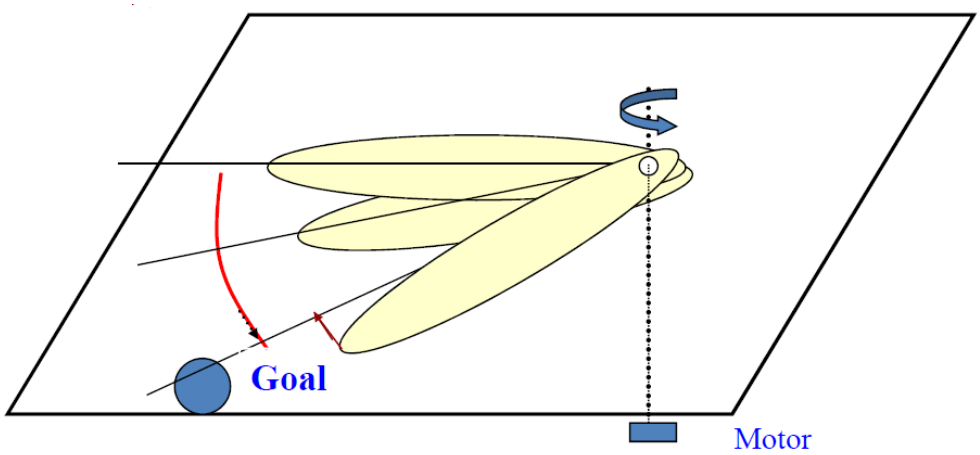

Figure 1.7: BCI-based position control scheme in 1-Dimension

to be moved from the start point to the target point by a human subject. The subject here employs MI signal to start moving the robotic link over the circular rim and releases the ErrP (or P300) signal when the link crosses the desired target position. The BCI system here needs to extract certain features from the MI signals and feeds them to a trained classifier to classify the signal into Right-hand MI (RHMI) or no motor imagery. If it is RHMI, the robotic link is commanded to turn right (clockwise). If no, no control command for link movement is issued to the robot actuator. After the robotic link crosses the target position, the subject releases an ErrP or P300. The task of the BCI system next is to extract certain EEG features and feed those features to a pre-trained classifier to determine the class: Target crossed/ not crossed. In case the target is crossed, the BCI system commands the robotic link to stop movement. In a more sophisticated system, the robotic link is commanded to turn back by a pre-defined/pre-calculated) offset angle.[19].

Stopping the robotic link at a position close enough to the target is an interesting open problem. The current literature on BCI employs Error-Related Potential (ErrP) and or P300 to address the above problem. The ErrP signal is released by the subject usually from the midline electrodes of his brain when he recognizes the commitment of movement-related errors and/or finds a second person or even a machine to commit an error. In the present context, the positional error is committed by a movable robot arm, but this too helps the participating subject to release the ErrP signal. It is noteworthy that subjects occasionally fail to release ErrP, although the movable robotic arm commits a positional error. P300 would be the right choice for those subjects as P300 hardly fails to liberate, when an oddball stimulus, such as the commitment of error is noticed by the subject.

The position control scheme introduced above has been extended in [19] for a 2-dimensional and 3-dimensional position control scheme. The fundamental difference between one-dimension and multi-dimensional position control lies in fixing the target point. For instance, in one-dimensional position control, the target is a point on the trajectory of motion of the end-effector, as indicated in Fig.1.8. In 2-dimensional position control, the target is a line. In other words, the last link carrying the end-effecter has to be aligned with the target line. Again, for 3-dimensional

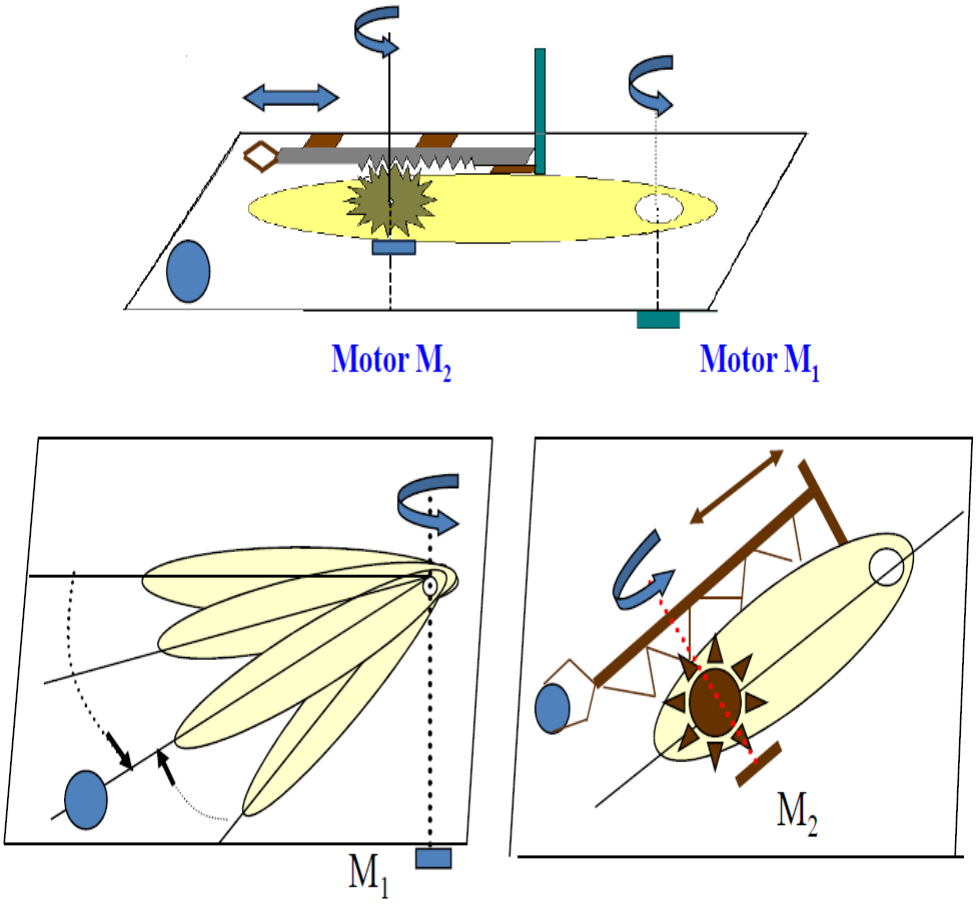

Figure 1.8: BCI-based position control scheme in 2-Dimension

position control, the target is a plane. In [19], the authors attempted to move the entire robot arm from one starting plane to the desired plane (Fig.1.10). Here, the target is a 3D plane. The complete architecture of the closed-loop BCI system employed for position control of a robot arm in 2-D and 3-D space is given in Fig.1.9 and Fig.1.11 respectively. Online selection of the target, however, is not easy, as it adds cognitive load to the subject during the planning phase. This is taken care of in subsequent research [20] by controlling the target position of the end-effector only rather than the last link carrying the end-effecter. This is realized by employing Inverse kinematics of the robot arm.

It is important to mention here that forward and inverse kinematics are two well-known principles for robot motion planning. In the forward kinematics, the current position of the end-effecter and angular position changes of the links are provided, and the final position of the end-effector is determined with respect to a given reference frame of the robot. The inverse kinematics model, on the other hand, requires specifying the current and the target position of the end-effector, and the motivation is to determine the angles of turnings of the links or displacements of junctions required. In a recent model, the authors [20] have emphasized the scope of Inverse kinematics to command the end-effecter to move up/down, right/left, and top/bottom around each position of the end-effector's current position. The BCI planner receives the user feedback to command the robot to move accordingly until the target position is reached.

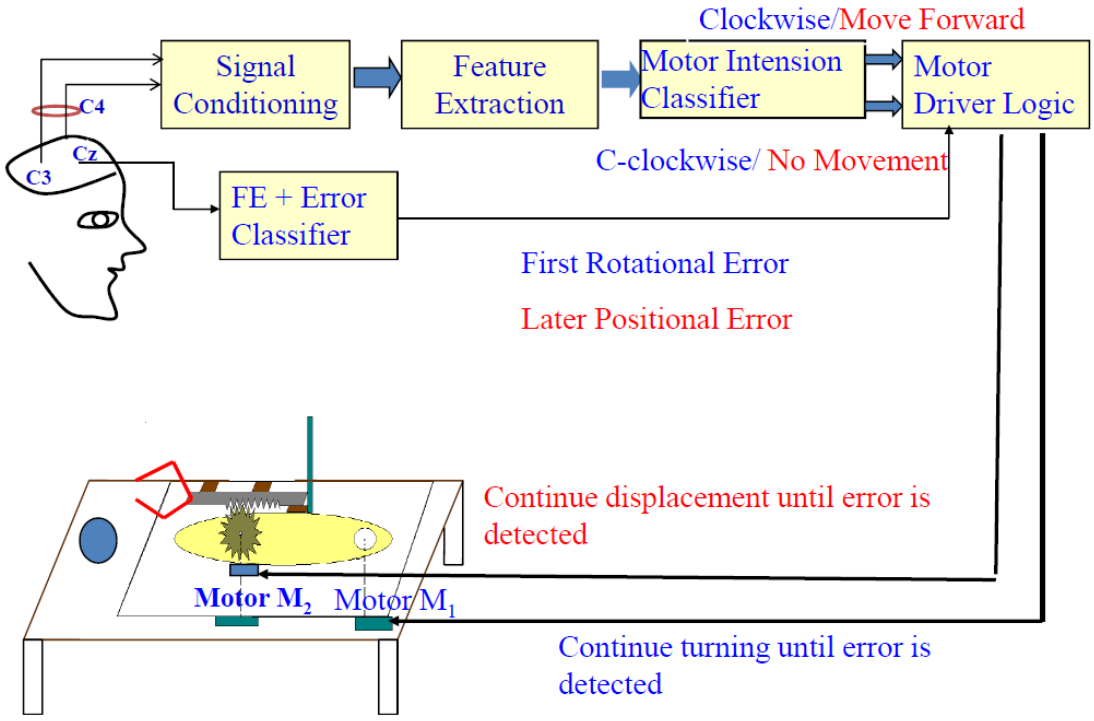

Figure 1.9: BCI-based complete position control scheme in 2D workspace.

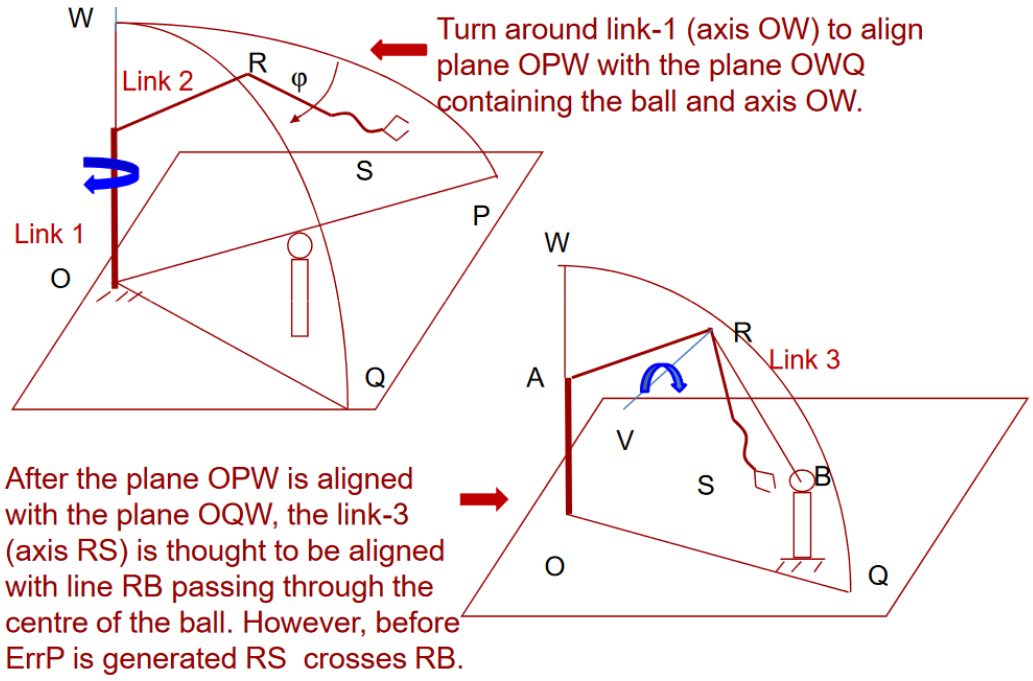

Figure 1.10: BCI-based position control of a robot arm in 3D workspace

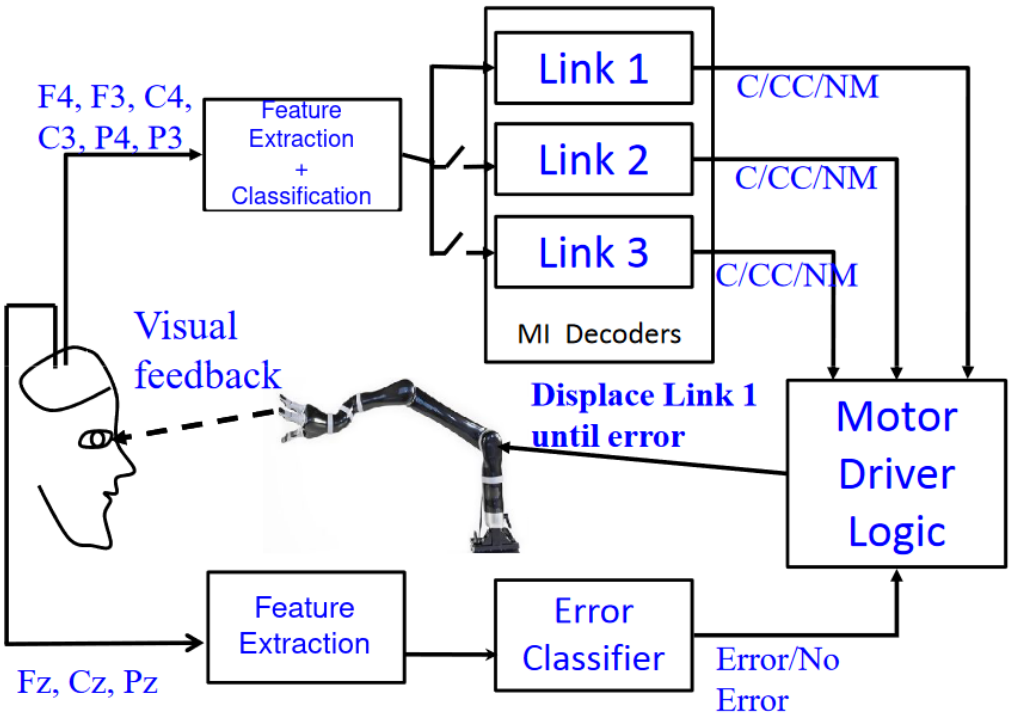

Figure 1.11: Complete closed-loop system of BCI-based position control of a robot arm in 3-D space.

The advent of inverse kinematics lies in the reduction of the cognitive load of the subject, as he/she simply plans to move the end-effecter towards the goal. Naturally, the BCI end-users, who usually are expected to be patients suffering from motor neuron diseases, would have a user-friendly rehabilitative aid that can be controlled with minimum effort.

1.5 Generic Approach of Designing a Closed-loop BCI Framework

Researchers realized that even though the current state of the external agent (ex. position of the end-effector of a robot arm) is used as a feedback signal to the human operator to increase the accuracy and quality of the performance, still the result was not satisfactory. The main reason behind this was the stochastic nature of the human brain which causes the EEG signal to vary significantly over the sessions during the neural activity task and even within the same session. Such variability of EEG signals degrades the system's performance. Another reason is the researcher's inability to perfectly understand the biological underpinnings of any specific neural activity, hence decoding the brain signal with high accuracy is still a challenging task. Among the other reason, researchers have found that it is still not feasible for a human subject to perfectly control the brain rhythms to accomplish a complex task. Even if some of the subjects are able to control his/her brain activity in the desired manner but accomplishing a complex task using a limited number of BCI commands imposes a high cognitive load on them.
Hence, in order to achieve reliable and superior performance, the design of the Brain-Computer Interface (BCI)/ Brain-Machine Interface (BMI) system must contain an auxiliary controller

which is introduced inside the control loop to mitigate the gap between desired performance and the actual performance. When placed in the forward path of the loop, the auxiliary controller takes the cognitive commands of the subject as the input and optimizes it by following a control strategy to produce a suitable control signal. The control signal actuates the external agent (ex. motors of the robot joints) in such a way that the desired performance is achieved. Such auxiliary controllers enable the subject to perform complex tasks relatively easily without causing any mental fatigue. Such controllers are also responsible for ensuring the stability of the system and maintaining certain input-output constraints when necessary, such tasks are not achievable without the auxiliary controller. The auxiliary controller also reduces the dependency of the system exclusively on the BCI decoding performance hence increasing the system accuracy independent of BCI decoding performance.

It is apparent from the above discussion that the auxiliary controller has now become an indispensable part of designing the modern BCI/BMI system. Although the controller is generally placed in the forward loop, there are many examples where the auxiliary controller is placed in the feedback path. When placed in the feedback path, the controller receives the state information from the external agent to be controlled and optimizes it to actuate an encoder that creates the suitable stimulation signal based on the feedback (ex. proprioceptive tactile feedback[21][22]) it receives from the auxiliary controller. The stimulation signal is used to stimulate a part of the human body (Cortical areas of the human brain [23][24], back of the human body[25]) to achieve better performance.

Now, the BCI system can be designed according to any of the generalized frameworks (GFMW) discussed below.

1. *Framework 1*: This framework is an open-loop system where the brain commands are directly applied to the external agent and the operating subject receives no feedback from the agent. This framework includes a neural interface that translates the cognitive commands to the control command of the robot through a series of steps like pre-processing, feature extraction, feature selection, and classification. The framework is represented in Fig.1.12.

2. *Framework 2*: Such types of frameworks include a visual feedback path from the robot to the human subject. The human generates the control command based on the real-time visual feedback he/she receives from the robot or any external agent. As an example, for a position control application, the subject observes the present position of the end-effecter and mentally determines how much it should be moved to reach the target position. The framework is shown in Fig.1.13.

3. *Framework 3*: This framework introduces an encoder in the feedback path to convert the current state information of the robot or an external agent into a vibrotactile or electrical stimulation to stimulate a body part like skin or brain. The electrical stimulation generated by the encoder provides the necessary feedback information to the user to generate a suitable control signal for the agent. The scheme is shown in Fig.1.14.

4. *Framework 4*: Such frameworks incorporate an auxiliary controller inside the control loop

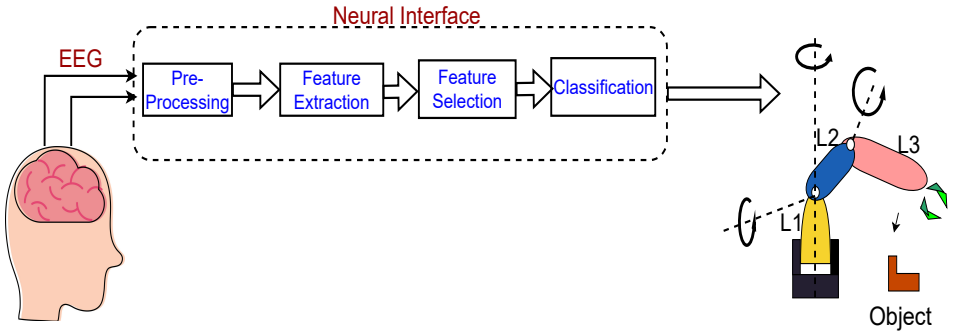

Figure 1.12: Schematic representation of Closed-loop BCI Framework 1

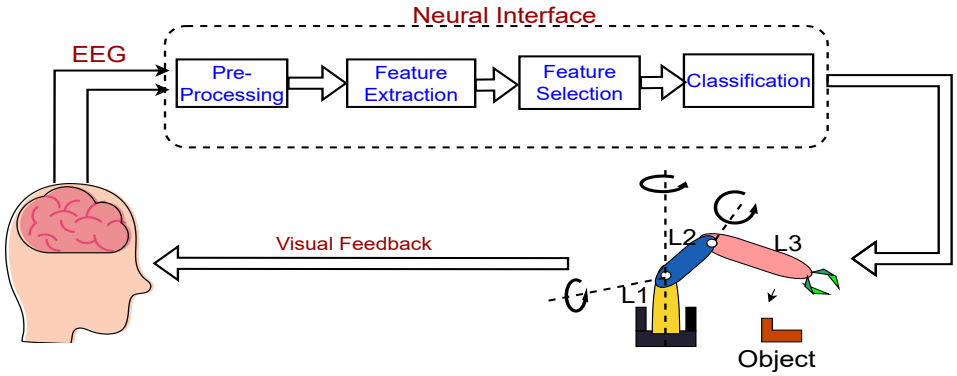

Figure 1.13: Schematic representation of Closed-loop BCI Framework 2

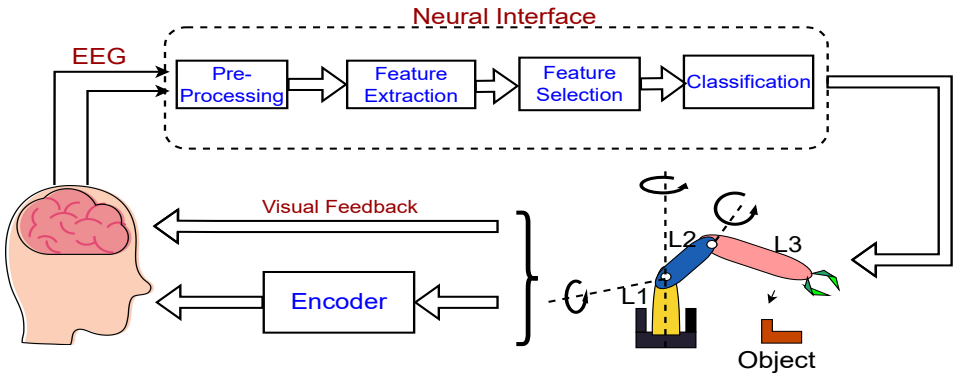

Figure 1.14: Schematic representation of Closed-loop BCI Framework 3

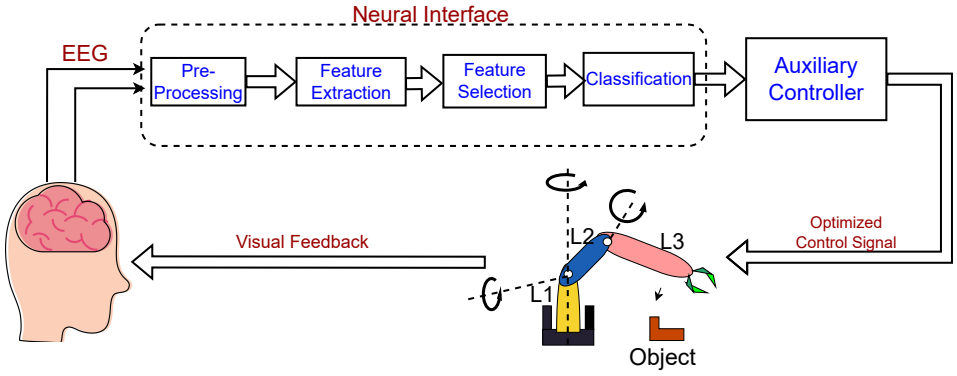

Figure 1.15: Schematic representation of Closed-loop BCI Framework 4

either in the forward path or in the feedback path. The schematic diagram presented in Fig.1.15 considers the auxiliary controller in the forward path. As evident from the figure the auxiliary controller here receives the high-level brain command from the participating subject through the neural interface and modulates them to generate low-level control commands to actuate the robot or any external agent present in the loop. The current thesis adopts this framework for designing the closed-loop system.

1.6 Modules of Closed-Loop Brain-Computer Interface System

The closed-loop BCI framework adopted in this thesis essentially contains two major components, a neural interface, and an auxiliary controller. These two components are cascaded with the human brain to achieve robust and superior performance. Here a brief description of each of them is presented below.

1.6.1 Neural Interface

The neural interface is responsible for decoding high-level human intention into low-level control commands that can be applied to the external device or an auxiliary controller. As discussed earlier, the interface decodes the human intention through a sequence of processing steps i.e. Pre-processing or Artifact Removal, Feature Extraction, Feature Selection, and Classification. Each of the sub-components is described below.

1.6.1.1 Pre-processing

The raw EEG signal acquired from the brain contains noise and artifacts arising due to unwanted muscle movement, eye blinking, power line noise, and other environmental noises. The pre-processing step helps to eliminate the noise using some filtering techniques. Such steps are also important to extract the desired frequency sub-band (which contains the highest amount of information about a cognitive task) from the original EEG signal.
Fast-Fourier transform (FFT) is a widely used method to extract the relevant frequency sub-band from the EEG signal[26]. FFT provides an alternative representation of the time-domain signal in the frequency domain using Discrete Fourier Transform(DFT). It allows us to visualize the separate frequency contents present in the signal. A noisy signal is first converted to a frequency domain signal using DFT and then the inverse DFT(IDFT) is applied to get back the signal in the original domain. While reconstructing, the IDFT eliminates the co-efficient belonging to the undesired frequency components, hence the reconstructed time domain signal is free of noise. Sometimes another variant of FFT popularly known as the Short-time Fourier transform (STFT) is also used to handle the non-stationary nature of EEG signals whose frequency components vary with time[27].

Digital filters are also one of the widely used pre-processing methods in EEG signal processing. The digital filters are designed in such a way that when a signal passes through the filter it retains the desired frequency band of the signal and attenuates the other frequency components.

The allowable frequency band is known as the passband and the attenuated frequency band is known as the stopband. The boundary frequency between the passband and stopband is known as the cut-off frequency. Based on the nature of the passband and stopband the filter can be termed a High-pass, Low-pass, and Band-pass filter[28].
Both the Finite Impulse Response(FIR) and Infinite Impulse response filters are used in the BCI experiment. The FIR filters are non-recursive in nature and always provide a stable response however it's computational efficiency is less than IIR filters which are recursive in nature and requires less amount of memory enabling them to be used in real-time BCI system. Different IIR filters like the Butterworth filter, Chebyshev filter, and Elliptic IIR filter are commonly used in BCI[29].
Blind source separation or BSS is also an effective technique to separate the desired source signal from a mixture of multiple signals[30]. Such a technique is used to eliminate muscle and ocular artifacts from the EEG signal. Another two commonly used techniques are Independent component analysis(ICA) and Principal Component Analysis(PCA). The ICA attempts to solve the BSS problem by representing the set of random variables (EEG observation) as a linear combination of statistically independent components[31]. On the contrary PCA attempts to find the directions (principal components) in the data that capture the maximum variance[32].

Apart from the spectral filtering techniques discussed above, spatial filtering techniques are also very popular in the BCI domain as they help to obtain more localized signals and to reduce the effect of far signal sources. One such spatial filtering technique is Common average referencing (CAR) where the average value of all the electrodes is subtracted from every electrode to reduce the effect of distant signal sources[33]. Laplacian filter on the other hand subtracts the weighted average of the surrounding electrodes from a individual electrode, and the weights are determined based on the distance between the electrodes[33].

1.6.1.2 Feature Extraction

Once the pre-processing phase is over, the brain patterns are subjected to further processing for extracting the hidden features which are not self-revealing in the raw EEG signal. The feature extraction method attempts to find the features that characterize the brain pattern and helps to distinguish between two different classes of brain patterns. There exists a number of feature extraction methods in the BCI literature, among them time domain feature extractors like Hjorth parameters[34], Adaptive Auto-regressive parameters[35], time-frequency domain feature extractors like Wavelet Transform[36], frequency domain feature extractors like Canonical Correlation analysis[37], spatial domain feature extractors like Common-spatial pattern[38], Empirical mode decomposition [39] are mostly used in BCI experiments.
Here, we discuss four major feature extraction methods relevant to four major categories of brain signals (MI, P300, ErrP, and SSVEP).
Common Spatial Pattern:
The Common Spatial Pattern (CSP)[40] technique aims to generate spatial filters that maximize the variance of EEG signals for one class while minimizing it for the other class simultaneously, and vice versa. This approach facilitates the decoding of the user's intended actions in the subse-

quent phase of a Brain-Computer Interface (BCI) system. From a mathematical perspective, this definition involves determining the coefficients of the spatial filter, denoted as 'w,' to optimize the following objective function.

$$O(w) = \frac{w^T E_1^T E_1 w}{w^T E_2^T E_2 w} = \frac{w^T C_1 w}{w^T C_2 w} \tag{1.1}$$

where; For a given class i, E_i represents the trial matrix of $N \times Q$ dimension; N denotes the number of EEG electrodes and Q denotes the time samples corresponding to each electrode. C_i denotes the covariance matrix of class i. It is apparent from the above equation that maximizing the objective function is nothing but the maximization of the signal variance of class 1 and the minimization of the signal variation of class 2. On the other hand, minimizing the objective function leads to the minimization of class 1 variance and maximization of class 2 variance. Hence, the problem has been formulated as an optimization problem to obtain the optimal filter coefficients. The optimization problem is subjected to the following constraints $w^T C_2 w = 1$ while extremizing $w^T C_1 w$. Mathematically, the above expression can be represented in terms of the Lagrange multiplier as below;

$$L(\lambda, w) = w^T C_1 w = \lambda (w^T C_2 w - 1) \tag{1.2}$$

Here, the spatial filters are found to be the eigenvectors of $C_2^{-1} C_1$. However, we consider the eigenvectors associated with both the extreme (highest and lowest) eigenvalues. Once the filter coefficients are obtained, the variance of the filtered signal is calculated, and the logarithm of the variance is taken as the CSP features.

Regularised Common Spatial Pattern In recent years a common trend is to modify the objective function of the CSP by introducing custom terms to achieve a specific solution[38]. One such term is the "penalty term" that penalizes the filter which violates the constraints associated with the formulation of the objective function. The penalty term is generally introduced in the denominator of the objective function. After introducing the penalty term the objective function takes the following form

$$O_{Reg} = \frac{w^T C_1 w}{w^T C_2 w + \beta S(w)} \tag{1.3}$$

here $S(w) = w^T M w$ is quadratic penalty term and β is the parameter of regularization. Like the classical CSP method described above, the optimization of the modified objective function can also be represented in terms of the Lagrange multiplier as below;

$$L(\lambda, w) = w^T C_1 w - \lambda (w^T (C_2 + \beta M) w - 1) \tag{1.4}$$

The spatial filters are obtained as the eigenvectors of the matrix $(C_2 + \beta M) C_1^{-1}$. However, unlike the classical CSP methods, the minimization of the objective function is not desirable here as it leads to the maximization of the penalty term. Hence the objective function is re-formulated here as below;

$$O_{Reg} = \frac{w^T C_2 w}{w^T C_1 w + \beta S(w)} \tag{1.5}$$

Now the filter coefficients can be obtained from the eigenvectors of the matrix $(C_1 + \beta M)C_2^{-1}$.

Empirical Mode Decomposition:

EMD (Empirical Mode Decomposition) is particularly well-suited for analyzing signals that exhibit a mixture of frequencies at different scales, such as those containing both slow and fast oscillations[39]. EEG signals, which are non-stationary in nature, pose a challenge for decomposition using fixed basis functions like Fourier Transform or Wavelet Decomposition. In contrast, EMD operates by decomposing the original sequence into Intrinsic Mode Functions (IMFs). IMFs possess the following characteristics:

1. The number of extrema is one more than the number of zero crossings.

2. The local average of an IMF is zero, indicating that the average value of the upper envelope and lower envelope is zero.

The EMD process operates through a series of iterative steps as follows[41]:

a) The local extrema (peaks and valleys) of the signal are identified.

b) The identified extrema are connected using a cubic spline interpolation method to obtain both the upper and lower envelopes.

c) The average of the upper and lower envelopes is computed.

d) The difference between the original signal and the computed mean of the envelope is calculated.

e) The above steps (a-d) are repeated until an Intrinsic Mode Function (IMF) is generated. The iterations continue until a specific criterion is met, which determines the termination of the decomposition process. After applying the sifting process, the EEG signal $x(n)$ can be expressed as the sum of the derived Intrinsic Mode Functions (IMFs) as shown below:

$$x(n) = \sum_{i=1}^{N} \hat{f_i}(n) \tag{1.6}$$

The above equation denotes the IMFs derived from the EEG signal as $f_i(n)$. After applying the EMD algorithm separately to both categories of signals (trials containing the P300 evidence $P_1(n)$ and trials without P300 evidence $P_2(n)$), these signals can be expressed as linear combinations of IMFs, as shown below:

$$P_1(n) = \sum_{i=1}^{N_1} \hat{f_{1,i}}(n) \tag{1.7}$$

$$P_2(n) = \sum_{i=1}^{N_2} \hat{f_{2,i}}(n) \tag{1.8}$$

IMFs obtained for the two categories of signals are N1 and N2. Now each class of the EEG signal is expressed as a linear combination of the IMFs, which is considered a feature vector of the corresponding class.

Canonical Correlation Analysis:

Canonical Correlation Analysis (CCA) is a statistical technique that aims to uncover the underlying covariance or correlation structure between two sets of random variables by expressing

them in terms of linear combinations[42]. CCA is particularly well-suited for the detection of Steady-State Visual Evoked Potentials (SSVEP) because it involves identifying specific stimulus frequencies from recorded EEG signals obtained from multiple channels. By leveraging the correlation information between the recorded EEG signals and the target stimulus frequencies, CCA enables effective detection and analysis of SSVEP responses[37]. For two sets of random variables $X \in R^{m\times n}$ and $Y \in R^{p\times n}$, CCA discovers two such linear combinations, $L_X \in R^m$ and $L_Y \in R^n$ for which the expressions given below,

$$\widetilde{c_1} = L_X^T X \tag{1.9}$$

$$\widetilde{c_2} = L_Y^T Y \tag{1.10}$$

yields maximum correlation subjected to maximization of the objective function given below,

$$\max_{L_X, L_Y} \rho = \frac{E[\widetilde{c_1}\widetilde{c_2}^T]}{\sqrt{E[\widetilde{c_1}\widetilde{c_1}^T][\widetilde{c_2}\widetilde{c_2}^T]}} \tag{1.11}$$

$$= \frac{L_X^T X Y^T L_Y}{L_X^T X X^T L_X L_Y^T Y Y^T L_Y} \tag{1.12}$$

Where ρ denotes the correlation coefficient.
Canonical Correlation Analysis (CCA) is used to discover the strongest connection between two sets of variables: X, representing recorded EEG data with m channels and n sample points, and Y a prepared dataset with dimensions $2HN$. Y contains the target stimulus frequency (f_s) along with other frequencies. Specifically, Y is constructed to include both the sine and cosine components of each frequency f_s(1 to S). By incorporating both components, CCA aims to capture complete information regarding the specific stimulus frequencies. The goal of CCA is to identify the canonical correlations that maximize the relationship between the EEG data X and the target stimulus frequencies in Y, allowing for the detection and examination of relevant EEG responses associated with those specific frequencies.

$$Y = \begin{pmatrix} \sin 2\pi f_s t \\ \cos 2\pi f_s t \\ \vdots \\ \sin 2\pi H f_s t \\ \cos 2\pi H f_s t \end{pmatrix} \quad \text{where} \quad t = \frac{1}{F}, \frac{2}{F}, \ldots,$$

Stepwise Linear Discriminant Analysis:
SWLDA, or Stepwise Linear Discriminant Analysis, applies a modified version of Fisher Linear Discriminant Analysis to classify trials into two categories[43], however, it can be used as a detection tool for wide categories of brain signals. This process involves solving a discriminant function to determine an optimal hyperplane that separates the two categories of the signal. The discriminant function, which is used to compute the distinguishability scores, is expressed as follows:

$$\hat{w}.f(x) - b = 0 \tag{1.13}$$

where b is the bias and $f(x)$ is the transformation function acts on the input vector x. The weight vector w is estimated using the equation below to predict the class label.

$$\hat{w} = (X^T X)^{-1} X^T y \tag{1.14}$$

SWLDA begins by including a single feature in the discriminant function and progressively incorporates additional features based on their statistical significance. When a new feature is added to the feature pool, the algorithm scans the pool in reverse and removes the least statistically significant feature. This iterative process continues until the feature pool reaches a predetermined number of features[44].

1.6.1.3 Feature Selection

It is observed that the feature extraction method sometimes generates complex and high-dimensional features from the original raw EEG data. However, it is not necessary that all the features generated by the feature extractor will be relevant for identifying the targeted mental state. Hence, a subset of features having the highest importance with respect to the targeted mental state must be judiciously selected from the original feature pool. Such a procedure is known as feature selection. The feature selection process has multiple benefits in the BCI application.
First, it helps to extract the most relevant features and discard the redundant features. Relevant features have high inter-class variance and low intra-class variance.
Second, the feature extraction method reduces the dimension of the feature which has a positive effect on the classifier's performance. As the number of features gets reduced the classifier needs to adjust only fewer number of parameters and also reduces the chance of overfitting especially when the number of training samples is small.
Third, fewer number of features speed up the classification process, which is highly required for real-time BCI applications.
There exist quite a few feature selection methods that are extensively used in BCI applications. Among them, Wrapper based methods like linear regressor[45], evolutionary algorithms[46], genetic algorithm[47], filter-based methods like Principal Component Analysis[48][49][32], maximal mutual information[50], embedded methods like SWLDA[51], metaheuristic algorithms like Particle Swarm Optimization[52], Ant colony optimization[53] are very popular. Apart from the above methods, mRMR[54], and Relief[55] are also used as the feature selection tool in EEG-based BCI. Feature selection methods from each of the three major categories are discussed in detail in this thesis. The evolutionary method like Differential Evolution is discussed in Chapter 2, while the embedded method SWLDA is discussed under the feature extraction section (Section 1.6.1.2). Hence, here we will restrict our discussion to PCA, the most popular candidate from the filter methods.

Principal Component Analysis:

PCA aims to reduce the dimensionality of the original dataset while preserving the most im-

portant information captured by the principal components, which are the directions in the data space that maximize the variance.

Let us consider $X \in R^{n \times p}$ to be the EEG feature set of a single trial arising from n EEG channels, and having p number of features. PCA first normalizes the data by subtracting the mean (m) of each feature (mean is computed by averaging the feature values across all channels: $m = (m_1, m_2, ...m_p)$) from each channel. The normalized data matrix is then utilized to compute the covariance matrix. For the normalised matrix $\hat{X}$, the covariance matrix Σ is calculated as $\Sigma = \frac{1}{n}\hat{X}^T\hat{\hat{X}}$. Next, the covariance matrix passes through the eigendecomposition method which reveals the eigenvalues($\lambda_1, \lambda_2, ...\lambda_p$) and eigenvectors($V_1, V_2, ...V_p$) of the matrix. Next, the eigenvectors are sorted in descending order according to their corresponding eigenvalue. This ensures that the principal components are ranked by the amount of variance they explain. Now, we select the d principal components by taking the first d eigenvectors to create a projection matrix S. The original normalized EEG data matrix $\hat{X}$ is now projected onto the S to obtain the matrix $P \in R^{n \times d}$ with reduced dimension, where each row corresponds to a transformed data point and each column represents a principal component score.

1.6.1.4 Classification

Classification is the last component of the neural interface. A properly designed classifier helps to distinguish between different mental states corresponding to the different mental intentions of the subject. Once the feature vector consisting of the discriminative features of an EEG trial is obtained, the classifier categorizes the feature vector and assigns a pre-defined class label to it. The categorized signal coming from the output of a classifier is used as an input to the auxiliary controller. In the case of traditional BCI control where the auxiliary controller is absent, the output of the classifier is directly used to actuate the external device such as motors of robot joints.

The classifiers learn through the instances to categorize the different cognitive states, hence the classifiers are trained with the available data generated during the training phase where the subjects are instructed to perform a given mental task whose class labels are known as apriori. The EEG signal acquired from the scalp of the subject while performing the tasks is processed to obtain the feature vector of that specific class label, which is utilized to train the classifier. Once the training process is over, the trained classifiers are employed in the real-time system for testing their performance. The BCI system employs various classifiers for different applications. For the robot control application, some of the widely used traditional classifiers are given below.

Discriminant analysis is a classification technique that falls under the category of generative modeling methods. In this approach, the goal is to estimate the density of EEG feature vector x within each class c_i based on the given class label. By incorporating the prior probability (unconditioned probability) of classes, we can calculate the posterior probability of c_i using the Bayes formula.

Linear-discriminant Analysis(LDA) classifier is the simplest form of discriminative classifier widely used for classifying motor imagery, P300, ErrP, and other brain signals[56][57]. The main assumption of the LDA is the equal covariance matrix of all classes, hence it presumes

that the classes are linearly separable. It attempts to find the optimal direction in which the classes are best separable[58]. The advantage of the LDA is its low complexity and ability to handle multicollinearity. For a given feature vector x and class labels c_i where $i = 1,2,3...n$ the discriminant function of the LDA is given below;

$$\hat{D}(x) = -\frac{1}{2}(x-\mu_{c_i})^T\Sigma^{-1}(x-\mu_{c_i})^T + log(\pi_{c_i}) \tag{1.15}$$

where Σ is the covariance matrix, μ_k, and π_k is the mean and prior probability of class c_i. The LDA finds the class label c_i (for a given test vector) that maximizes the discriminant function. It should be noted that while formulating the discriminant function the LDA drops the quadratic terms and considers only linear terms.

Another important class of classifier is Quadratic Discriminant Analysis(QDA) which is an extension of previously described LDA[59]. Unlike the LDA, the QDA assumes a different covariance matrix for each class. It attempts to maximize a discriminant function that is quadratic in nature. The discriminant function is very similar to that of LDA but also includes the quadratic terms that were previously discarded in the LDA formulation. The discriminant function is given below;

$$\hat{D}(x) = -\frac{1}{2}(x-\mu_{c_i})^T\Sigma_{c_i}^{-1}(x-\mu_{c_i})^T + log(\pi_{c_i}) \tag{1.16}$$

the notation of the equation remains the same as LDA but with an exception, the covariance matrix Σ_{c_i} is now class dependent. Like the LDA, QDA also tries to find the class label c_i for which the discriminant function gets maximized. In some recent literature, QDA is found to be used as a classifier to classify different cognitive states[60][61].

In LDA and QDA class density is assumed to be Gaussian in nature, but based on Baye's rule, there is another important classifier called Naive Bayes classifier which assumes the input features governing the class densities are independent, hence the class density can be obtained by simply multiplying the marginal densities of each input features[62]. Such an assumption of conditional independence helps to simplify the calculation and increases the classification speed. The Naive Bayes classifier is well suited for real-time EEG classification applications, hence has been widely used in BCI domain[63][64].

Sometimes it is found that due to the complex nature of EEG data, it becomes almost impossible to distinguish between two different classes of data using "Hard" margins, i.e. it is impossible to construct a decision boundary for which all the feature points belong to a class label will be on the one side of the boundary and feature points belong to another class label will be on another side of the boundary. Such situations are tackled by introducing the "soft" margin which allows a small number of feature points to lie on the wrong side of the boundary but with an aim to increase the generalization capability of the classifier. Support Vector Machine (SVM) is one of the very popular classifiers which attempts to maximize this soft margin/hyperplane by minimizing the classification errors[65][66].

Let's say we have EEG feature vectors x_i, where i ranges from 1 to N, representing the training set X. Our objective is to create an ideal hyperplane $g(x) = \omega^T x + \omega_0$ that accurately classifies the training vectors into two linearly separable classes, $\omega_1$and ω_2. In the equation,

ω represents the direction of the hyperplane, and ω_0 represents its position. To achieve this, Support Vector Machines (SVM) aim to find the hyperplane direction that maximizes the margin between both classes, ensuring a clear separation. The distance of a particular point from hyperplane $g(x)$ can be expressed as $z = \frac{|g(x)|}{||\omega||}$. Now, ω and ω_0 can be scaled so that the value of $g(x)$ is +1 in the case of ω_1, and -1 in the case of ω_2. Hence, the task reduces to the problem of computing parameters ω and ω_0 to minimize the objective function

$$J(\omega, \omega_0) = \frac{1}{2}||\omega||^2 \tag{1.17}$$

subject to the constraint $y_i(\omega^T x_i + \omega_0) \geq 1$ for $i = 1, 2, 3...N$.

The above optimization problem is solved using the method of Karush-Kuhn-Tucker(KKT) conditions. The vector parameter ω of the optimal solution comes out to be a linear combination of the number of feature vectors corresponding to non-zero Lagrange multipliers. These vectors are known as Support Vectors. In case the feature points are not linearly separable, a nonlinear mapping function popularly known as the kernel function is used to map the feature points in a higher dimensional space. Once the above mapping is done, the SVM attempts to find an optimal hyperplane that linearly separates the feature points in that higher dimension. Literature shows the widespread usage of this classifier for classifying various brain patterns[67][68][69].

Another important classifier for brain signal classification is k-NN[70]. k-NN is a classifier that maps the feature vector to class labels by storing k-feature points that are nearest to the test feature point under consideration and then it calculates the Euclidian distance of the test feature point from all other feature points. The most probable classes get the nomination from previously selected k-nearest neighbors and the class which gets the most nomination is selected as the class of that particular test feature point. k is a very important parameter of the k-NN classifier as the local density of data is controlled by it and of course, it is much smaller than the training sample size[66].

Apart from the statistical classifiers, Artificial Neural Network is also used as classifier in the BCI task. The ANN is completely data-driven, hence doesn't depend on the probability distribution of the data [71]. The ANN attempts to learn the mapping between feature vectors and class labels by adapting the connection weights between the artificial neurons. It uses EEG feature vectors with known class labels along with a learning rule to update the connection weights[72][73].

However, due to the low signal-to-noise ratio and non-stationarity of the EEG signals, the classification performance is not optimal. Hence, a recent trend is seen where the parameters of the classifier model are updated continuously to tackle the above drawbacks and increase the classification accuracy[74]. Such parameter adaptation also helps to deal with the problem of the non-availability of a high amount of EEG data for classifier training. A very popular approach called transfer learning is also used to deal with the problem of less amount of data, it enables the classifier trained in one BCI domain or session to be used in another BCI domain/session[75][76][77]. There exist many other techniques which are used to increase classification accuracy, merging the feature extraction, feature selection, and classification process

is one of them [74].

Parameter adaptation is done using both supervised and unsupervised learning[78], i.e. with or without having prior knowledge about the class of the data, although most of the adaptive classifiers are designed using supervised learning[79][80][81]. In motor imagery-based BCI, adaptive LDA and QDA were used to classify the MI signals[82][83][78][84][85][86]. An adaptive probabilistic neural network was also used to classify different classes of MI patterns, such models assume a distribution model of feature sets of each class and update it with the new incoming feature vector[87]. Likewise, an altered non-linear Bayesian classifier employed unsupervised or semi-supervised learning methods, where a subset of the incoming trials was labeled. This adaptation incorporated extended Kalman filtering to monitor the fluctuations in the parameters of the class distribution alongside auto-regressive (AR) features[88].

Another important class of classifiers is Convolutional Neural Networks, which is popularly known as CNN. A convolutional neural network is a type of neural network that follows a forward propagation approach, where information moves unidirectionally from the input through the hidden layers to the output. It is characterized by the presence of at least one convolutional layer [89][90]. The convolutional layer employs a convolutional operator that transforms the input signal into an output signal through (Eq.1.18). For a given trial of EEG data x_i with N number of samples, the convolutional filter g_l having size L, transforms the input x_i into output y_i

$$y(n) = \sum_{i=0}^{L-1} g_i x_{n-i} \forall n = 0..1,2,...N-1 \tag{1.18}$$

Although the above equation is given for 1-D signal input, but the idea can be safely extended for multidimensional input. Hence, CNN can be used for multichannel EEG inputs also.

In a general framework of CNN, non-linearities and pooling layers are introduced after the convolutional layer. Such layers combine the relevant local data coming from the convolutional layer output into a solitary value, usually by utilizing an average or maximum operation. Typically, in standard ConvNet architectures, multiple layers consisting of convolution, and non-linearity (and pooling) are sequentially stacked, followed by additional layers, commonly fully connected, which function as a classification layer [91]. It is worth noting that certain architectures employ solely convolutional layers for classification. The model parameters in these architectures consist of the weights assigned to all the convolution filters, as well as the weights associated with the fully connected layers.

Parameters of the network are trained through supervised learning with the aim to minimize the cost function of the following generic form

$$\hat{w} = \underset{w}{argmin} \frac{1}{m} \sum_i J(y_i, D_w(x_i)) + R(w) \tag{1.19}$$

The training data above is represented as $[x_i, y_i]$ for i ranging from 1 to the total number of samples. The prediction function D_w, associated with the ConvNet, is used to predict the labels for the input data x_i. The loss function J is used to measure the discrepancy between the true class of EEG trials x_i and the predicted class. Additionally, $\mathcal{R}$ represents a regularization function that

helps in controlling the parameters of the ConvNet. Stochastic gradient descent and its variants are widely used for optimizing deep ConvNets due to the specific form of global loss, which is the average loss over individual samples. Moreover, the feedforward architecture of D_w enables the computation of gradients at any layer using the chain rule. This efficient computation is accomplished through the back-propagation algorithm [92].

In the BCI domain, CNN is being extensively used to identify and distinguish between different mental commands(brain patterns) due to their superior ability to learn the most discriminating features present in that specific brain pattern [93]. Although there exist quite a few papers that use CNN to detect various brain patterns like Motor Imagery, SSVEP, P300, etc., here we will discuss some of the landmark papers for each category of brain signal.

The initial exploration of CNN for P300-BCI (Brain-Computer Interface) was conducted by Cecotti *et al.* in their pioneering paper[94]. Their CNN architecture consisted of two convolutional layers, where one layer was designed to learn spatial filters and the other layer to learn temporal filters. These layers were followed by a fully connected layer. In addition to this, the researchers also investigated the use of ensembles comprising multiple CNNs with similar architectures. In another instance, Manor R *et al.*[95] used CNN with one spatial convolutional layer along with two other temporal convolutional layers to detect the presence of P300 generated in the context of Rapid Serial Visual Presentation. The network also employed two dense fully connected layers at the end for the classification purpose.

In the case of SSVEP detection, Kwak N S *et al.*[96] proposed the use of CNN consisting of both spatial and temporal convolutional filters. The performance of the CNN was at par with the Canonical Correlation Analysis and multi-layer perception, but the CNN provided a superior performance over the other two classifiers in the case of a noisy signal input coming from a moving subject.

The CNN is also extensively used in deciphering motor imagery signals in BCI. Tabar Y R *et al.*[97] used CNN in conjunction with Deep Belief network (DBN) to classify motor imagery signals. Schirrmeister *et al* [98] investigated different CNN architectures including shallow CNN, deep CNN, hybrid shallow+deep CNN, and residual NN. The shallow CNN consisted of one temporal convolution, one spatial convolution, squaring, mean pooling, and a softmax layer. The deep CNN involved a temporal convolution, a spatial convolution, followed by three layers of standard convolution, and a softmax layer. The hybrid shallow+deep CNN combined the architectures of shallow and deep CNNs by concatenating them. The residual NN consisted of a temporal convolution, a spatial convolution, 34 residual layers, and a softmax layer. Both the deep and shallow CNN architectures outperformed FBCSP significantly in their performance, however, the hybrid and residual CNN failed to do that. Another paper[99] used CNN to classify multiple brain patterns like Motor Imagery (MI), P300, ErrP, etc. They used a compact structure of CNN with few layers and parameters.

1.6.2 Auxiliary Controller

Performance requirements of different BCI applications can not be met only with human efforts, because of the subject's inability to perform complex tasks (such as complex trajectory gener-

ation for the robots, avoiding obstacles during the robot movement, proper grasping of objects using robot gripper, etc) using limited number of BCI commands. It also imposes a high cognitive load on the participating subjects. Hence an additional controller is needed to take care of the low-level tasks. Such controllers are termed auxiliary controllers. The auxiliary controller receives the human command from the neural interface and modulates it to actuate a robot arm or an external device to achieve any specific task. The main role of the auxiliary controller is to provide high-precision control, which usually is difficult to achieve by human agents.
Different control strategies based on the requirements of the BCI application are adopted for designing the auxiliary controller. Here, we will briefly discuss three major strategies used predominantly in the Brain-controlled robot application.

1.6.2.1 PID Conroller

Proportional-Integral-Derivative(PID) controller is a traditional and one of the simple control strategies which have got widespread application in various domains like robotics, process industry, the aerospace industry, etc. Such controllers are also implemented in closed-loop BCI for their easy realization technique [100][101]. The control signal $u(t)$ can be represented by Eq.(1.20).

$$u(t) = K_p e(t) + K_i \int_0^t e(t)\,dt + K_d \frac{de(t)}{dt} \tag{1.20}$$

where $e(t)$ is the error signal that arises due to the mismatch between the actual and desired value. K_p,K_i, and K_d are the proportional constant, derivative constant, and integral constant respectively. It is evident from (1.20) that the control signal $u(t)$ is a function of the error signal, rate of the error, and integral of the error. The objective is to optimize the values of K_p, K_i, and K_d to obtain the desired control signal which can actuate the external agent to get the desired performance. The proportional gain helps to obtain a fast response time, and the integral constant helps to get a zero steady-state error while the derivative constant yields a reduced overshoot by predicting the rate of change of error in advance.
Many studies have used PID controllers in the closed-loop BCI for various applications like robot control, regulating the vibrotactile feedback, generating proprioceptive feedback from the robot, etc [100][101][102][103].

1.6.2.2 Shared Controller

The utilization of a shared controller in a Brain-Computer Interface (BCI)-based robot manipulation task offers inherent advantages. It serves to enhance the accuracy of task completion by the robot while simultaneously reducing cognitive fatigue experienced by the human operator through a reduction of subject involvement in repetitive tasks. The primary objective of the shared controller is to optimize the capabilities of both the human and robot agents within the system by breaking down the task into smaller sub-tasks and assigning them to the most suitable agent (either human or robot) based on their specific strengths. High-level decision-making subtasks are entrusted to the human agent, while lower-level tasks like local navigation and object grasping are autonomously performed by the robot agent using its onboard sensors like RGB(D)

camera, ultrasonic sensors, etc. This approach ensures that human intervention is only required when uncertainties arise in the decision-making process.

Typically, shared control methods employed in BCI applications can be classified into two main types: sequential shared control and all-time shared control[104]. In the case of sequential shared control, the task is shared between the autonomous controller and the human subject in a sequential manner. Initially, the human subject provides high-level commands, such as the desired destination to reach and the autonomous controller then executes the task while ensuring specific conditions are met. In the other type of control method human operator and the shared controller continuously shares the task and the decision is generated at each instant by combining the decisions of the human and the shared controller.

In the present thesis, the first type of shared controller has been designed with the help of a 3D vision camera.

Vision-based Controller: Literature shows that vision-based shared controllers are predominantly used in the closed-loop BCI[105][106][7][107][108]. Such controllers use an RGBD camera capable of providing depth information to guide the robot arm to reach the target location. The controller works in the following manner. First, the camera captures the images of the surrounding environment of the robot arm, then an image processing algorithm such as object identification, and object localization is applied to extract the relevant information from the captured images. Once the relevant information is extracted, that information is utilized to understand and interpret the surrounding environment into controller-understandable information. Based on the interpreted visual information, the controller makes decisions and generates control signals to actuate a robot arm for achieving the required task. These decisions include object tracking, path planning, obstacle avoidance, generating grasping configuration for robot end-effecter, and any other specific tasks depending on the necessity of the BCI application.

Brief summary of State-of-the-art Object Recognition technique: The key component of the vision-based controller is the image processing module. The image processing technique used in the vision-based controller serves different purposes like object classification, object localization, and object segmentation or semantic segmentation based on the application where it is applied. However, such challenging computer vision tasks come under a big umbrella called object recognition. With the advent of deep learning techniques, and convolutional neural networks, the object recognition task is mostly dominated by these two techniques. Here, we provide a brief description of the state-of-the-art object recognition strategies relevant to the present thesis work.

Girshick *et al.*[109] in 2014 proposed a promising approach to object recognition based on region-specific convolutional neural network (CNN) features, such a family of algorithms is popularly known as R-CNN or Region-based Convolutional Neural Network (R-CNN). The R-CNN uses three distinct modules to recognize any object from a digital image. The modules include

1. Region Proposal: This module is responsible for proposing the regions where the object may be located and for creating candidate bounding boxes. An offline-based algorithm " Selective Search" is used to propose the regions here.

2. Feature Extractor: This module uses Deep Convolutional Neural Networks to extract relevant features from each of the proposed regions.

3. Classifier: This final layer uses a classifier to assign a class label to each of the proposed regions based on the features obtained in the previous module.

However, the R-CNN object recognition method is a sequential method and employs a pipeline architecture, where the previously defined three modules work sequentially with a multistage pipeline. Hence the training process is slow. Training the Deep CNN for a large number of proposed regions is also computationally expensive. Overall the process becomes slow and not a good choice for real-time operation. As an alternative, Girshick *et al.*[110] proposed the Fast R-CNN network to speed up the process. Unlike the R-CNN which takes the input (proposed regions) for the CNN sequentially, the Fast R-CNN takes the entire image and the proposals in a single forward pass. The network also employs a RoI pooling layer at the end of the architecture to extract the specific feature related to each of the candidate bounding boxes. The output of the CNN reaches two separate branches, where one of the branches contains a softmax layer to predict the class label and another branch contains a regressor that regresses through the bounding box. Hence, the Fast R-CNN simultaneously predicts the object class and objectness score at the same time.

The structure of the network was further updated to increase the training speed as well as to increase the detection accuracy. The update was mainly concerned with the region proposal process, and a separate new module called Region Proposal Network (RPN) was introduced in cascade with the Fast-RCNN network. The combined architecture of RPN and Fast-RCNN is popularly known as Faster-RCNN[111]. The RPN is also a deep CNN structure that takes the output of a pre-trained CNN (such as VGG-16, ResNet[112][113]) as its input and provides the multiple region proposals along with their class label. The region proposals are nothing but bounding boxes with pre-defined shapes. The class label associated with each bounding box is binary in nature, i.e., RPN indicates whether the object is present in the region or not. Each proposed region is passed to the Fast-RCNN for further classification and accurate bounding box regression. This network takes the proposed regions as input and extracts their features using RoI (Region of Interest) pooling or RoI align. The Fast-RCNN predicts class probabilities for each proposed region and refines the bounding box coordinates to obtain accurate object detections. It is apparent from the above discussion that instead of using an external region proposal algorithm (like Fast-RCNN), Faster R-CNN combines the RPN and object detection network into a single architecture. By sharing convolutional features between the RPN and the detection network, Faster R-CNN achieves end-to-end training and further improves both accuracy and efficiency compared to Fast R-CNN.

1.6.2.3 Model Predictive Controller

Model Predictive Control (MPC) is a control strategy that utilizes a predictive model of the system's behavior to optimize control actions over a future time horizon[114]. First, a mathematical model of the system's dynamics is developed from the experimental data to describe the dynamic

behavior of the system. Based on the present state and control input of the system, the model predicts the behavior of the system over a specified time. An objective function with certain input-output constraints is optimized to obtain the optimal control input. Once the control input is applied to the system, the state of the system gets updated, which becomes the present state of the prediction model.

Such Model-based predictive controllers are currently being used to design the auxiliary controller in closed-loop BCI for their superior performance in terms of precision and accuracy [115][116][117][118].

1.7 State-of-the-art BCI-based Control Schemes

The BCI has become an emerging field in the last ten years, hence there exists a large number of literature in this domain. However, here we provide a review of recent BCI works which focus on the design of closed-loop BCI systems from the control-theoretic perspective and are relevant to the main theme of the present thesis.

In 2016 Meng *et al.* [119]developed a novel approach to cursor control (and also position control of an end-effecter) using two sequential steps. The first step utilizes the left/right MI for cursor control in the x-direction on a horizontal plane. In the second phase, the authors employed both hand-movement together to control the movement of the cursor (end-effecter) in the y-direction of the horizontal plane. Next, they arranged to hover over the target object using Mu-power. The position of the cursor/end-effecter is displayed as a feedback signal to the user for necessary corrective actions in subsequent steps.

Meng *et al.*[120] in their other work, proposed an interesting approach to controlling the position of a cursor on a computer screen. They employed overt spatial attention (OSA) to control the movement of a cursor on the left/right side of the screen depending on the subject's alertness towards the left/right ends of the screen. After the cursor hits the left/right bar of the screen, it would have a change of color to signify success. Once the cursor control in the horizontal direction is over, the user utilizes OSA to control the movement of the cursor in the up-down direction. A third strategy to control the motion of the cursor in a virtual plane perpendicular to the screen is to employ a left/right motor imagery signal to move the cursor outward/inward on the plane. An alternative strategy is to control the left/right and up/down movement of the cursor by motor imagery and motion in the perpendicular plane by OSA control.

Now, in the context of BCI-based robot manipulation, Tariq *et al.*[121] proposes the following sequence of actions to control a robot using brain commands. The schematic diagram of the system is provided in Fig. 1.16.

- 1. High-level control/planning decisions are generated by the BCI system.
- 2. The robot has its own sensory system and actuators to plan and execute high-level commands using low-level control signals by local controllers.
- 3. On-screen selection of the target /navigational direction by BCI-based system using P300. This is done by flashing the desired direction and activating by P300.

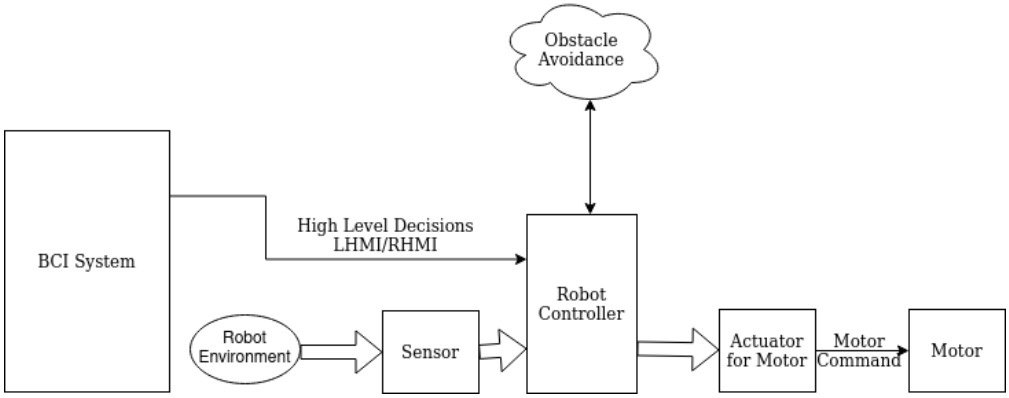

Figure 1.16: Schematic diagram of BCI-based control strategy proposed by Tariq *et al.*

Liu *et al.*[122] proposed a novel scheme where left-arm motor imagery is utilized to lift the robotic limbs and right-hand motor imagery to move down the limbs. Additionally, they utilized inverse kinematics to determine the angular positions of the joints of the 6 degrees of freedom manipulator. Lastly, they employed a novel neural-net-based position controller, that optimizes a given control objective to reduce to optimize the angular velocity and angular acceleration of each joint. The scheme is presented in Fig.1.17

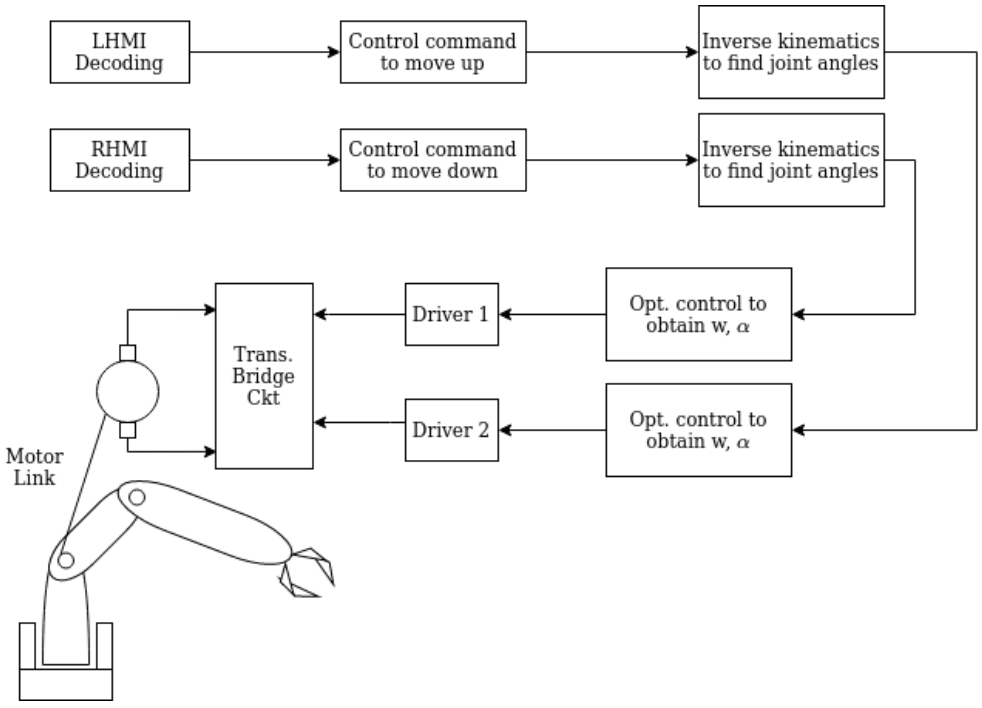

Figure 1.17: Schematic diagram of BCI-based control strategy proposed by Liu *et al.*

Tang *et al.*[106] proposed an interesting technique to control the movement of an end-effecter in a plane parallel to the plane of a camera mounted on the moving robotic arm. They introduced an on-off type control strategy by utilizing the following principle.

- 1. The left/Right movement of the end-effecter on the coronal plane (parallel to the plane of the camera) is done by left/right-hand motor imagery.

- 2. The motion of the end-effector is stopped when the camera detects the object of intent on the coronal plane.

Another work by Zhang *et al.*[7] proposes the following strategy to mentally control a robot arm using a custom-designed neural interface.

1. The subject selects the objects using SSVEP.
2. The subject uses motor imagery to control the robot arm.
3. There are 6 types if movements involved in grasping control: left, right, up, down, open and close.
4. By image analysis, the robot determines the displacement in left, right, up, down, opening, and closure of end-effecter.
5. Additionally, the robot plans the same movement patterns of the end-effecter.
6. Now, the probability of the end-effecter to grasp object due to up, down, left, right etc. movement are compared from the visual inspection data both by the Kinect sensor and the subject. A data fusion algorithm is run to obtain the optimized constraints.

The work undertaken by Xu *et al.*[107] deals with switched mode dual control strategy where BCI-based control is employed at an early stage to position a robot end-effecter close enough to a targeted object. After the end-effecter reaches the neighborhood of the target position, the manual to automatic camera-based position control is switched on. In the BCI-based control paradigm, the subject employs left-hand motor imagery and right-hand motor imagery to control the motion of the end-effector at 45° around the targetted position of the end-effecter. In other words, the LHMI and RHMI are transformed into velocity vectors at an angle of +45° or -45° respectively with respect to a reference line in the plane of motion of the end-effecter. The end-effecter is moved in a zig-zag path toward the pre-defined target. Once the end-effecter comes within close vicinity of the target object, the switch over to automatic control using vision-based movement planning is instigated. Under this scheme, a camera mounted at the corner of the robot-work table is turned on to capture the image of the object, localize it by image pre-processing and segmentation, and a control strategy is developed to bring the gripper towards the target object. Fig. 1.18 provides a schematic overview of the overall system involving the camera, robot arm, the subject, and the distributed control strategy.

Sorbello *et al.*[123] proposed an interesting scheme of shared control to control the position of the end-effecter. First, the subject selects the on-screen navigational commands, (such as move horizontal, vertical, left, right, gripper open/close, and the like by using P300 signals. A camera mounted on the top of the robot is used to determine the direction of navigation of the end-effector intelligently. In case the direction indicated by the camera matches with the one selected by the subject using P300, the action is selected, but not executed. Before execution of the command, a second step of confirmation is obtained by the bio-feedback. The "bio-feedback module" inputs 3 parameters: Attention, Focus, and Intent, and evaluates a metric as

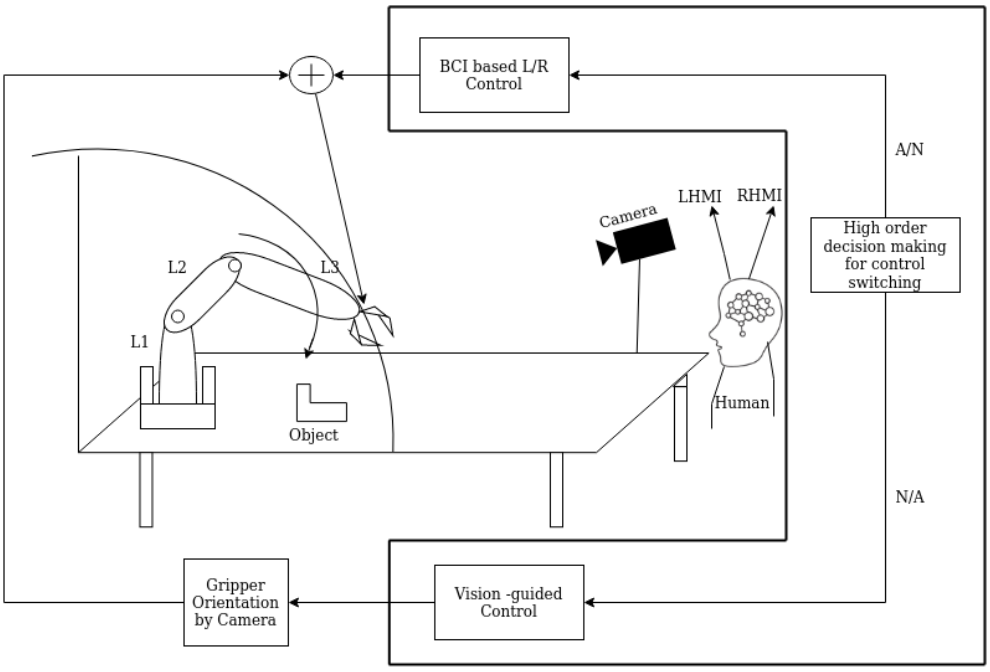

Figure 1.18: Schematic diagram of BCI-based control strategy proposed by Xu *et al.*

the weighted sum of the above 3 parameters. In case the weighted sum crosses a pre-defined threshold, the action chosen by the subject is executed. If the weighted sum goes below the threshold, the action selected by the subject is abandoned.

Wang *et al.* [124] presented an interesting approach to object selection through a camera using SSVEP. The scheme employs a camera mounted on top of a robot arm to explore the objects within the field of view, which are transferred to a computer monitor for selecting the target object using SSVEP. Here for each object, a bounding box is developed. These bounding boxes flicker at different frequencies to represent the identity of the individual objects. A subject intending of selecting a specific object focuses on the item and the flickering frequency is picked up by the subject through SSVEP. Once the object is selected, the navigation of the robot arm is automated by a camera-based position control system, The distance of the object from the camera is calibrated using the size of the object. The smaller the size, the closer the object with the robot arm.

Lillio *et al.*[125] proposed a novel strategy for shared control to develop a BCI-controlled assistive manipulator. Here the entire task of controlling the manipulator is partitioned into two components: i) BCI-based human interaction and ii) Intelligent autonomous action. In the BCI-based human interface, the subject selects three items in sequence: i) Object selection, ii) Robotic Action selection, iii) Action execution. Selections are done using P300. The BCI modality of the subject is now complete, and the control policy now is transferred to robot automation. In the automated phases, a Kinect-based camera is used for object recognition and mouth recognition. This information is passed to the waypoint generator which generates a reference trajectory to be followed by the robot. A task priority-based inverse kinematics algorithm is employed which takes care of the obstacles present in the environment and communicates

with a low-level robot joint velocity controller. The joint velocity controller controls the torque of each joint of the robot and manipulates it towards the desired direction.

Batzianoulis *et al.* [126] proposed an interesting approach for shared control by utilizing the ErrP signal and Inverse reinforcement learning (IRL) paradigm. Here the subject has its own choice of trajectory planning of a robot, for instance, obstacle avoidance and trajectory planning toward the fixed target. Each time the robot reaches an obstacle before reaching the destination, the subject releases an ErrP signal. The decoded ErrP signal is used to move away the robot from the obstacle without violating the planned trajectory of the robot. An IRL algorithm is employed to change the trajectory following the user's preferred trajectory of motion.

Duan *et al.*[108] in a recent paper presented a novel scheme for shared control with a hybrid BCI system. They employed a camera-based capturing of robots' movement. The camera back-plane and the human retinal backplane are parallel. So, the information extracted by the camera is the same as could be obtained by the subject, if he attempted to see the robot by his naked eyes. The computer screen displaying camera view about the robot's trajectory of motion includes provisions for generating navigational commands for the robot using SSVEP. Navigational commands include turn right/turn left/ grasping etc. After the SSVEP-driven commands are issued to the robotic device, the robot gets aware of the task it has to do with. Next, motor imagination is carried out to turn the motor clockwise/counter-clockwise. The switching from SSVEP to MI is performed by thresholding of a parameter.

Yuan *et al.*[127] developed a novel approach for BCI-based navigational planning of a mobile robot. A camera mounted on top of a mobile robot is used to capture scenes including obstacles present in the environment. The camera output is visible to a subject through a computer screen. The camera front plane and the computer screen are placed in parallel, so the subject has the same impact due to the onscreen video, rather than watching the robot motion directly. A SSVEP based selection of robotic actions is included in the front panel of the computer. The control decisions are taken by the subject are communicated to the robot wirelessly, to perform the required task. The control decision includes avoid obstacles and move towards the targeted goal.

Zeng *et al.*[128] provide a novel approach to BCI and camera-based object positioning and gripping. The camera back-plane is kept in parallel with the plane containing several objects around the robot arm. Naturally, the image acquired, when displayed on the computer screen, all objects seen by the camera take specific positions/coordinates on the screen. A second camera is mounted on the screen to develop the gaze-detection system. The gaze detection system determines the direction of the gaze of the user to select one of several objects in the local neighborhood of the robot. Later the subject controls the speed of the robot by repeated ERD/ERS and the direction of the movement of the robot using gaze control. When the gripper reaches above the object, the gripper vertically gets down to the object and grips it, and takes its precise position.

Tidoni *et al.*[129] proposed a faster and easier way to navigate a walking bipedal humanoid robot by combining SSVEP brain signals and audio-visual feedback from the robot. The task was to 1. Navigate a humanoid robot towards a target object, 2. Pick the object 3. Place

the object in the desired location. Participating subjects used their SSVEP signal to navigate the robot in the desired direction (left, right, forward, backward and stop movement) and also to select the desired object to be picked and desired location where the object to be placed. A camera is placed on the robot's head which captures the live feed of the robot's local environment and transmits the feed to a remote computer. The feed of the camera is visible to the user through a computer. Now the subject mentally issues navigational commands to the robot by observing the robot's movement on the computer screen. While navigating the robot through mental commands, the subject receives the footstep sound of the robot synchronous to the actual robot movement. It was observed that subject could navigate the robot with the increasing speed and accuracy while they receive audio and visual feedback from the robot through footstep sound and camera respectively. The authors also placed mirror near the robot, through which the robot it self became visible to the subject. Such placement of the mirror enhanced the accuracy of operation.

Fransicso *et al.*[130] Introduced a novel approach for EEG-BCI based on-off control of home appliances. In the said scheme, the subject selects control commands on a computer screen to communicate his/her desire to activate external devices (such as a light, fan, AC, TV and the like). The automatic selection of control commands is performed using P300 brain signal. Once a control command is selected, it is transformed into a speech signal by a text-to-speech translator. A Google Assistant is activated, once the speech is generated with a codeword used to activate Google Assistant. The rest of the communication is actuated by Google Assistant. For example, if the message needs to be communicated to a subject by phone, a phone call to the recipient may be initiated. In case the speech needs to be communicated to the speaker, the voice message is sent to the subscriber by phone. In case a TV or an AC needs to switched ON/OFF, an IR device is activated to be used as a remote controller through a Wi-Fi device to switch ON/OFF the TV/AC or similar devices.

Mao *et al.*[131] proposed a novel strategy for brain-robot interaction by fusing human and machine intelligence. They attempted to develop an intelligent system to navigate a mobile humanoid robot in a fixed workspace through BCI-based control commands. The workspace is captured by a camera mounted on the head of the robot. The captured frames about the local neighborhood of the robot is mapped onto a computer screen. The motivation here is to make the robot move along a user-intended path. This is done by generating two brain signals P300 and SSVEP in sequence. To make the problem simplified, let us assume the robot is desired to move to a fixed object position. This is realized by liberating P300 through flashing the desired object on the screen. However, in case, a trajectory of the robot is required, the first object the robot needs to reach is commanded by generating P300 (by flashing the selected object on screen) and the motion of the robot therefrom is controlled by generating SSVEP by gazing at a flickering source. Such movement of robot from one location to the other is needed in many applications like transportation, assembling in factory automation, and fire extinguishing at several places. The schematic diagram of the control strategy is given in Fig.1.19

Mengfan *et al.*[132] extended their previous work by judiciously selecting the 2-step motion commands of a robot using N200 and P300 signals. The first step involves the selection of

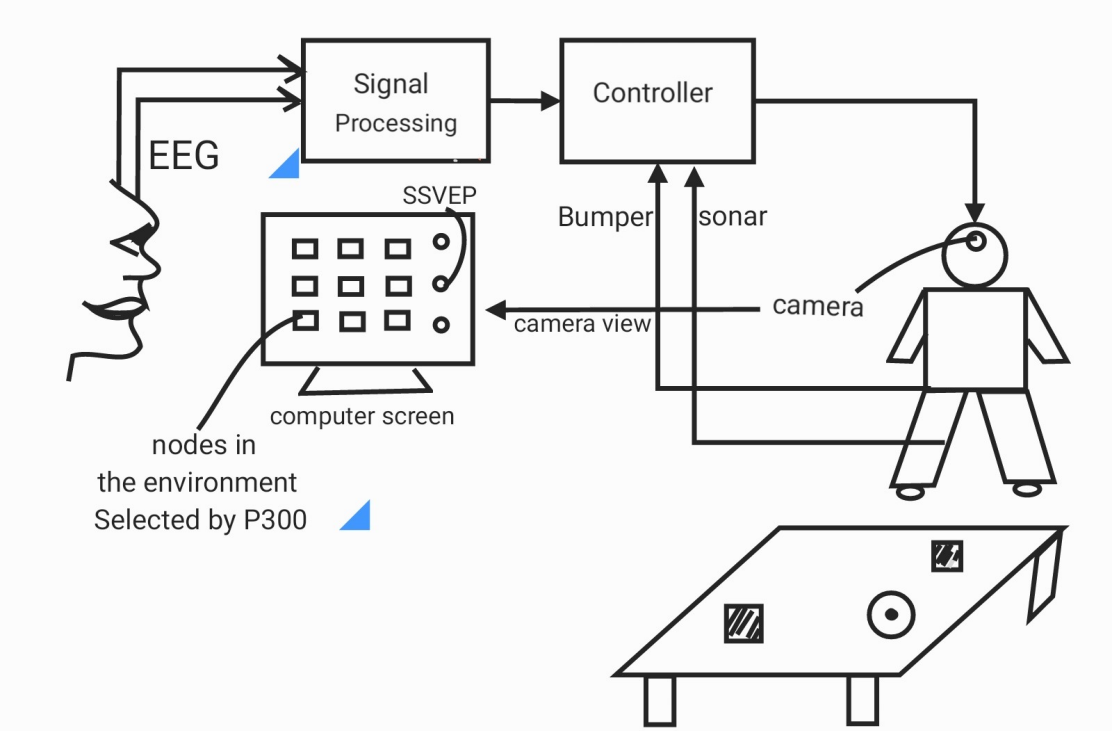

Figure 1.19: Schematic diagram of the BCI strategy proposed by Mao *et al.*

forward motion, while the latter involves one of the three possible motions like continuous, discrete or stepwise. In both cases, they utilize N200 followed by P300 signals for command generation. The classification accuracy is here increased by 2 parallel classifiers: SVM and FDA and the classifier decision is communicated to the robot when the response of the classifiers is matched.

The robotic neural prosthesis is an effective way to restore the function of disabled or amputated limbs but poses a restriction on the subject to receive somatosensory feedback. Such drawback is overcome by reanimating the original limb of the subject instead of using a robotic prosthesis. A novel strategy following the above protocol is proposed by Liu *et al.*[133], where they used a custom-designed low power, miniaturized, tactile force sensor, and an electro goniometer to encode tactile and proprioceptive feedback signals respectively from the re-animated limb. The subject wears the sensors which encode the feedback signal from the limb and transfers it to a bidirectional invasive brain-machine interface (BMI) which in turn generates a frequency/pulse-modulated stimulation to actuate a specific brain region. The BMI interface was designed using SoC technology, which integrates a neural activity recorder, electrical stimulator, analog-to-digital converter, ultra-wideband transceivers, and a digital signal processor for suitable stimulus signal generation.

As an improvised method of post-stroke rehabilitation, Romero-Laiseca *et al.*[134] proposed a novel brain-machine interface where the participating subjects perform pedaling imagery to trigger a motorized exercise bike which is highly effective for gait and lower-limb rehabilitation. The system is beneficial for post-stroke patients who can not voluntarily actuate a movement but now become able to do it using the proposed brain-machine interface. They focused on

minimizing the computational cost of the system using a few strategies which includes the use of i) Fast Fourier Transform(FFT) followed by inverse FFT for the pre-processing of EEG signal, ii) Riemannian geometry based feature extraction, iii) pairwise feature proximity for feature selection and using low-cost LDA classifier for pedaling motor imagery classification. The highest accuracy achieved by the two post-stroke patients was as high as 41.67 % and 91.7% although the average accuracy achieved by them were bit low. Significant improvement in the actuating time shows the efficacy of the system for its longer use.

Another efficient way of decoding the neural signals for the generation of suitable control signals in a closed-loop BCI system is proposed by Dangie *et al.*[135]. The method is particularly suitable for decoding the non-stationary brain signal in an online BCI task. Instead of using a traditional Kalman filter-based decoder which uses batch estimation for updating its parameters, here the researchers have proposed an adaptive version of the Kalman filter which continuously updates its parameter through a stochastic gradient descent algorithm. The proposed adaptive Kalman filter (AKF) updates the estimation of the state matrix and observation matrix at each time step. The efficiency of the algorithm is validated using the center-out cursor movement task in a 2D virtual plane and the brain function is mimicked using a simulated model of cortical neurons. The algorithm results in a faster learning process and increased success rate. The algorithm proves to be more robust to the sudden disturbances than it's non-adaptive version.

Kyung-Huan Shim *et al.*[136] proposes a novel technique of brain-machine interface-based robot reaching and grasping tasks using a custom-designed neural interface and a 3D vision camera. The neural interface translates the human intention captured by the EEG signal into robot commands using a novel convolutional neural network. The proposed CNN uses a smaller receptive field of 3x3 dimension to extract the minimal difference between the imagined movements of different parts of the same limb. The lower receptive field has the advantage of increasing the spatial resolution of EEG feature vectors. The proposed CNN also rules out the usage of any max-pooling layer that considers only the active features and neglects other important features, which may cause the robot to follow any unwanted command. The experiment consists of reaching and grasping an object with the mentally guided robot arm and pouring a beverage from the grasped object. All the tasks are carried out by five mentally imagined commands (Right/Left MI, twisting wrist imagery etc.). The imagined commands decoded by the neural interface actuate the robot-end-effecter to reach the correct region where the target object is located and then a 3D Kinect camera determines the exact location of the target object using a YOLO object detection algorithm and guides the robot end-effecter to grasp the target object. Results obtained in this study reveal that the proposed CNN yields an online accuracy of 90% and also performs better than other CNN-based decoding algorithms that use the maxpool in their architecture.

Recently Lu *et al.*[137] proposed a novel shared control-based strategy using an auxiliary controller to drive a car with mental commands. The auxiliary controller was designed based on the model predictive control strategy. The proposed Model Predictive Controller(MPC) ensures driving safety by properly adjusting the control commands issued by the human driver. It periodically tracks the output of the neural interface which translates the brain command into the motion control command of the robot. The entire system including the neural interface and MPC

was realized on embedded hardware. The proposed strategy was tested through a hardware-in-loop driving setup with two relevant driving tasks (Task1. Keep the vehicle on the road, Task2. Avoid obstacles. The human driver issues mental commands using an SSVEP interface to navigate a virtual car in right/left and forward directions. The MPC receives the human navigational command and generates suitable control motion commands for the vehicle by satisfying a number of safety constraints. The controller optimizes the control signal in such a way that the original navigational intent of the subject is preserved. The result shows that the accuracy of the mentally driven cars increases significantly where a accuracy of almost 100% was achieved in both the driving tasks described above.

Lu *et al.*[116] extended their previous work to develop a more robust controller that optimizes the motion command of the vehicle based on the human driver behavior model along with the host and preceding vehicle dynamics model. The controller aims to provide maximum rear seat safety by maintaining a threshold safe distance from the preceding vehicle and by maximizing the "minimum time to the collision". The controller attempts to preserve the original navigational command of the human driver while maintaining the driver's comfort during the journey. The most notable accomplishment of this study is the development of a human-driver behavior model which consists of a driver behavior model and a brain control behavior model. These two models along with the longitudinal model of the host and the preceding vehicle are exploited by the MPC to generate a suitable motion control command that ensures riding safety. The model and controller performance was tested under three different scenarios. In the first two scenarios, the driver has the task of following a preceding vehicle moving with different profiles of acceleration and in the third scenario the subject needs to issue an emergency brake command to avoid collision. The "vehicle follow" task achieves the highest 100% accuracy whereas the emergency braking task shows a significant reduction in response time. However, the system performance is still to be evaluated in the presence of actual physical vehicle.

A similar approach of using a Model Predictive auxiliary controller in a Brain-machine interface setup can be seen in a recent research paper by Fuizon He *et al.*[115]. They used a model predictive controller for the safe operation of a bain commanded mobile robot. The proposed controller modulates the control signal issued by the human operating subject to provide a safe and collision-free trajectory for the mobile robot. It also overcomes the shortcomings of the direct BCI control (without any auxiliary controller) that may lead to poor performance due to i. longer response time, ii. no guaranteed safety and iii. the higher number of control commands required to navigate the robot. The MPC uses a virtual model of a mobile robot to estimate the system behavior and generates a sequence of optimal control signals by satisfying the safety constraints at each time instant. The scheme was tested in a virtual environment where the subject uses SSVEP brain signal to issue different navigational commands which get optimized by the proposed model predictive controller. The accuracy of the robot navigation task reached 100% without any incident of collision with the obstacles and the task completion time was also reduced significantly.

The use of a model predictive controller in a BMI-based prosthesis control task (described above) requires the mathematical model of the system, which is often very complex in nature

due to the inclusion of a complex human brain model and other bio-physiological models in the control loop. Pan *et al.*[138] provided an interesting solution to it by designing a model-free controller which is entirely data-driven and does not require an explicit system model. However, the proposed controller was used in the feedback path of a closed-loop brain-machine interface to generate a suitable control command to actuate a device that can stimulate a specific brain region so that the subject can mentally guide the robot to achieve the desired response(such as a specific trajectory). They considered a simulated model of the cortical circuit for single-joint movement and used a model-free controller to generate artificial sensory feedback to actuate the group of perceived position vector (PPV) neurons of the cortical circuit to achieve their target firing rate which is required to control the robotic prosthesis. The controller performance was compared with a model predictive controller and the result shows that although the MPC controller shows slightly better performance but the proposed model-free controller has a faster response.

Shared autonomy in a BCI-based robot manipulation task has the intrinsic advantage of increasing the task accomplishment accuracy of the robot and also minimizing the cognitive fatigue of the human operator by reducing the human involvement in repetitive tasks. A generalized hierarchical structure for achieving suitable shared autonomy in a BCI system was proposed by Akinola *et al.*[139]. The proposed hierarchical system is universal in nature and hence can be adopted in any BCI-based robotic system. It aims to leverage the fullest power of each agent(human and robot) present in the system by subdividing the task into smaller sub-tasks and assigning those sub-tasks to the best agent (human/robot) for that specific sub-task. The sub-tasks involving high-level decisions are carried out by the human agent while the low-level tasks such as navigation in the local environment, and object grasping are carried out by the robot agent autonomously using its onboard sensors. It ensures that human involvement is only required when uncertainty about a decision arises. The performance of the system was evaluated in a home-like environment where a mobile robot has to perform certain tasks like food delivery, table cleaning, etc. The robot receives the higher level navigational commands from the human subject through an SSVEP-based neural interface and navigates to the target location using SLAM. A vision-based camera is then actuated to identify the desired object in the target location and a suitable grasping strategy is applied to grasp the desired object, The experiment achieved a task completion accuracy of 96.2%.

The shared-control-based strategy adopts various modalities to convey the user intent to the auxiliary controller. Among them, SSVEP is widely used for its high information transfer rate and high detection accuracy. Flickering sources on the screen are used to evoke the SSVEP in the subject's brain. In the vision-based strategy, SSVEP is often utilized by the subject to provide the necessary navigational command to the robot by observing its surrounding environment through a camera whose live video is presented on a screen. In all the cases the camera feed and the SSVEP sources are represented separately. However, a more interesting approach was proposed by Yang *et al.*[140] where they embedded the flickering sources directly on the objects visible in the live camera feed. A stereo-vision camera located in a fixed frame is utilized to capture the surrounding environment of a Baxter robot and identifies the nearby objects through

color segmentation. Subjects having the intent to select a particular object focus on the flickering source virtually embedded in it. A decoder interface was designed to decode the SSVEP response of the brain, which enables the system to identify the target object. Once the object is identified, its location (after suitable coordinate transformation) is conveyed to a vision-based controller which guides the robot to reach the target object. The controller is also responsible for avoiding collision with obstacles by coordinating the self and task motion of the robot. They also proposed a novel adaptive algorithm to identify objects in different lighting conditions. Apart from the above contribution they introduced the least mean square method in the co-ordinate calibration task between robot and the camera. The experiment was validated using an object-picking task where two subjects participated and achieved an accuracy of 100% and 80% respectively.

Generally, shared control methods used in BCI applications are broadly categorized into two types i.e. sequential shared control and all-time shared control. In the first case, the autonomous controller and the human subject share the task in sequential order. The human subject first delivers high-level commands such as the destination to reach and the autonomous controller performs the above task with maintaining certain conditions. The main drawback of such systems is the user's inability to intervene in the task while the autonomous controller is executing the task,i.e. once the task is assigned to the auxiliary/autonomous controller the user can no longer issue any further command until the controller becomes free. Such phenomena reduce the sense of authority of the user. Deng *et al.* [104]proposed a novel shared control scheme for the continuous sharing of tasks between humans and the auxiliary controller in a virtual wheelchair control application. A Bayesian shared controller was proposed to intelligently combine brain-actuated control and autonomous control introduced by the auxiliary controller to generate an optimal steering command for the wheelchair.

1.8 Scope of the Thesis

In recent years, there has been a growing interest in Brain-Computer Interface (BCI) due to its extensive applications in areas such as neural rehabilitation for patients with neuro-motor disorders, mind-controlled interfaces for gaming, cognitive driving, defense applications, and more. One particular challenge in neural rehabilitation using BCI involves effectively controlling the position of the robot arm's end-effector. A large number of works have been carried out addressing this challenge in the last few years, which have enriched the BCI community. Despite the promising results demonstrated by existing works, they are not without their limitations and drawbacks. Three major limitations are mainly observed;

First, most of the existing works use an ON-OFF control strategy to control a robot arm using mental commands. To illustrate, let's consider the process of turning a robotic link to a desired angle. This requires mentally activating an electromechanical motor to initiate the turning motion at a constant speed for a specific duration until the desired position is reached. Subsequently, the motor is deactivated for a pause, resulting in an on-off control strategy akin to traditional control theory. While this approach suffices for controlling individual links, the

complexity increases when multiple links need to be turned in a specific sequence. Hence, sequential control task becomes increasingly difficult with the increase in the number of robot links. Moreover, such an ON-OFF control strategy yields a large steady-state error, and large positional overshoot, resulting in a degradation of performance in terms of control theoretic parameters.

Second, some of the existing works are found to be using PID and Model Predictive Control based auxiliary controllers to aid the decision-making process of the participating subject and to generate optimized control commands based on the user intent to actuate a robot arm. Such auxiliary controllers are introduced inside the control loop to mitigate the gap between desired performance and actual performance. However, such a controller requires a complete mathematical model of the system, which is often highly complex in nature due to the inclusion of a complex human brain model and other bio-physiological models in the control loop. Obtaining an accurate mathematical description of the entire system is also not feasible in most of cases. Hence, the controllers fail to deliver satisfactory performance without complete knowledge of system dynamics. Additionally, such controllers can hardly model the individual subject's mental intentions about the high-level commands, i.e., the controllers can not be customized based on user preference.

Third, certain recent literature has introduced vision-based controllers to guide robot arms within a specific environment. These controllers utilize a camera to capture the surroundings and identify objects within that environment. However, in many of these studies, subjects are still responsible for planning the robot's trajectory or aligning the end-effector with the target object using their mental commands. The vision-based controller primarily aids in collision avoidance during the robot's movement and assists in grasping the target object once the end-effector is aligned with it. Notably, no specific autonomous grasping policy has been developed in these works. Very few works use some image processing techniques to identify the objects present in the environment and create an interface to enable the subject to directly choose the desired object, but in all those literature the researchers have depended on the existing object recognition approach, which is unable to identify the target object and grasp it from multiple overlapping objects that may appear within the camera's field of view. However, in real-life situations, it is common to encounter such overlapping objects in the user's domestic workspace. Attempting to manually grasp these objects solely through cognitive effort is exceedingly challenging for humans.

The thesis aims at handling the above problems by 3 distinctive approaches, presented in Chapters 2-4. Chapters 2 and 3 attempt to overcome the first two drawbacks of the existing control strategies while Chapter 4 overcomes the last drawback mentioned above.

Chapter 2 provides a new approach to brain-actuated controller design for BCI-based rehabilitative applications with an aim to reduce steady-state positional error, and settling time and also attempt to improve the relative stability of the proposed control systems by judiciously choosing the parameters of the controller. The chapter focuses on the issue of velocity modulation for individual links of a robot arm when crossing the desired target position. This modulation involves gradually reducing speed amplitude and even reversing direction to achieve

tighter control over the end-effector's position. The chapter introduces a two-loop cascade control approach, where the outer loop handles position control and the inner loop focuses on speed control. This configuration improves time-response performance, settling time, and positional overshoot, while also addressing steady-state error. The proposed control strategy includes a customizable brain-actuated controller that generates commands based on the occurrence of a P300 signal triggered by the subject in response to positional error zero crossings. Stability analysis, using root contour plots, confirms the stability of the brain-actuated control system under varying controller parameter values. Additionally, relative stability analysis is conducted to determine optimal parameter settings that simultaneously minimize the settling time and maximize the DC gain. The root locus analysis also finds the effect of the initial velocity of the robot link on the stability of the overall system. The above chapter also provides a thorough comparison of the controller performance with other hybrid-BCI-based position control schemes in terms of four performance metrics i.e. success rate, steady-state error, peak-overshoot, and settling time.

Chapter 3 explores the use of fuzzy logic in controller design, building upon the previous work presented in Chapter 2. The aim is to extract more information from the BCI system itself regarding positional overshoot/undershoot, enabling the controller to enhance positional accuracy in steady-state. While the P300 signal released by the experimental subject during the control system test phase indicates zero-error crossings in the positional offset, it does not provide information about the magnitude of positional error resulting from end-effector movement. To address this, the proposed fuzzy control policy utilizes two parameters: error magnitude and sign. The magnitude of error is assessed by determining if the end-effector or desired link crosses user-defined landmark positions located around the target position. Although fixing landmarks adds complexity, it aids the BCI system in approximating the magnitude and sign of error. A fuzzy-rule-based system is then designed to adjust the speed of the selected link or end-effector in the desired direction based on the error's sign and magnitude. The control scheme is validated with the participation of twelve volunteers.

The aforementioned works place the burden on the user to manually create complex trajectories for the robot arm using mental commands in order to reach and grasp specific objects. This process demands a significant cognitive load on human subjects, particularly for the precise position control needed to align the end-effector accurately with the desired object for successful grasping. Additionally, both works utilize a control scheme that focuses on controlling one link of the robot at a time. Consequently, rather than directly controlling the motion of the end-effector, the subjects are responsible for controlling each individual link separately.

Chapter 4 introduces a novel shared controller that addresses the limitations mentioned earlier. By leveraging vision-based techniques, this controller enables the robot to autonomously reach and grasp the desired object with the minimum number of brain commands. Consequently, in this scheme, the subject is relieved from the task of planning a complex trajectory for the robot link to align it with the desired object. The proposed scheme allows subjects to select their desired object using mental commands and a vision-based controller guides the robot arm to autonomously reach and grasp the target object. To achieve precise grasping of the target object by the robot gripper, a novel CNN-based robotic grasp detection network called Overlapping

Object Grasp Net (OOGNet) is utilized. This network generates a grasp rectangle, bounding box, and object class for each object present in the image, thereby associating each predicted grasp with its corresponding object. In order to enable simultaneous object and grasp detection, a custom loss function is proposed to ensure that each estimated grasp is correctly affiliated with its corresponding object. Additionally, the OOGNet network is capable of grasping the desired object even in situations where the object is partially overlapped by other objects. This capability enables the robot to accurately identify and grasp the target object, even in complex and cluttered environments. This autonomous movement of the robot arm also facilitates precise control that surpasses what can be achieved solely through human effort. In the chapter, a comparison is made regarding the cognitive workload experienced by subjects while operating a robot using three different BCI strategies, including the proposed strategy. The workload assessment utilizes the NASA-TLX questionnaire developed by NASA Ames Research Center.

Chapter 5 conducts a thorough self-review of the thesis, evaluating its content and findings. It also explores opportunities for extending the thesis in both theory and practice, considering the limitations and strengths of the research.

Chapter 2

Design and Analysis of Brain-Actuated Position Control of a Robot Arm

Brain-Computer interfacing (BCI) has currently added a new dimension in assistive robotics. Existing brain-computer interfaces designed for position control applications suffer from two fundamental limitations. First, most of the existing schemes employ open-loop control, and thus are unable to track positional errors, resulting in failures in taking necessary online corrective actions. There are examples of a few works dealing with closed-loop electroencephalography (EEG)-based position control. These existing closed-loop brain-induced position control schemes employ a fixed order link selection rule, which often creates a bottleneck preventing time-efficient control. Second, the existing brain-induced position controllers are designed to generate a position response like a traditional first- order system, resulting in a large steady-state error. This paper overcomes the above two limitations by keeping provisions for steady-state visual evoked potential (SSVEP) induced link-selection in an arbitrary order as required for efficient control and generating a second-order response of the position-control system with gradually diminishing overshoots/undershoots to reduce steady-state errors. Other than the above, the third innovation is to utilize motor imagery and P300 signals to design the hybrid brain-computer interfacing system for the said application with a gradually diminishing error-margin using speed reversal at the zero-crossings of positional errors. Experiments undertaken reveal that the steady-state error is reduced to 0.12%. The paper also provides a thorough analysis of the stability of the closed-loop system performance using the Root Locus technique.

2.1 Introduction

Brain-Computer Interfaces(BCIs) are currently gaining increasing interest for its wide spread application in neuro-motor prosthesis, neural rehabilitation, and mind-driven command generation for robot control, car control and other applications.BCI technology captures the human motor-intention to translate the thoughts into commands and actuates the robot to execute a mentally planned complex task. A BCI framework provides a non- muscular channel of communication with the outer world to enhance the quality of life of people suffering from brainstem stroke, neuro-muscular disorder, and the like, with an aim to rehabilitate them back to their normal lives.

Hybrid BCI [141] is a widely used name in BCI technology. Generally, it refers to multiple modalities of acquisition of brain activities, including functional near infrared spectroscopy (fNIRS), functional magnetic resonance imaging (fMRI), electroencephalography (EEG), electrocorticography (E-Cog), and the like. The second category of hybridization includes utilizing brain activity acquisition response in the presence of other physiological modalities, such as muscle activity acquisition by electromyography (EMG), eye-movement acquisition by electro-occulography, and the like. In this paper, we, however, used the phrase "hybrid BCI" to refer to utilizing multiple brain signals within a single modality of brain-activity acquisition, i.e., EEG. Each brain signal refers to one specific cognitive functionality of the brain, such as motor movement planning, responding to rare/infrequent (error) stimuli, and communicating the subject's choice (among alternatives) to the BCI system. It is, however, important to note that hybrid BCI may involve multiple EEG signals to decode a single cognitive activity of the brain [142] with an aim to improve reliability in the detection of the cognitive task undertaken.

EEG is preferred in BCI design for its non-invasiveness, faster temporal response, and low cost [143]. This work employs EEG to capture the subject's motor intention. Existing works on mind-controlled external devices utilize a few selected brain signals, including steady-state visually evoked potential (SSVEP), P300, motor imagination (MI), and error-related potential (ErrP) for position control of artificial human appendages/external manipulators [144][131][145][146][147]. For example, P300 has successfully been used for goal/destination selection of a mobile robot/wheelchair [148], [149]. SSVEP has been utilized for direction control of wheelchairs [150]. Motor imagery is used mostly in position control applications to actuate an external device based on right/left hand-motor imagination [151][152]. Lastly, ErrP is infrequently used to determine the occurrence of errors, particularly when the manipulator crosses a fixed (predefined) target position [19]. SSVEP has been proven to be the most promising brain pattern in the BCI technology [153]. It is elicited from the visual cortex of the occipital lobe as a response to some visual stimulus that flickers continuously at a certain frequency in the range of [6, 30] Hz. The SSVEP constitutes signal rhythms at the target frequency and its harmonics [154], and thus is useful in communicating the subject's choice to the BCI system when the options corresponding to the stimuli are flickered at selected frequencies.

When a subject undertakes MI, a signal called event-related de-synchronization (ERD) followed by event-related synchronization (ERS) originates from the motor cortex as mu and beta rhythms. ERD refers to a relative decrease in the signal power in the μ and the β-bands during

motor imagination in comparison to the resting state of the brain [155], [156]. On completion of the motor imagination task, the signal power of the β-band increases until the power level matches the average value of the resting potential. The later signal is referred to as ERS.

Lastly, P300 is an important event-related potential (ERP) that appears in the subject's EEG recording when they focus their attention on some significant but rare stimulus [157], [158]. The subject must respond to the target stimulus in either a covert or overt manner in order for P300 to be evoked. This modality is characterized by a positive deflection in the EEG voltage waveform around 250–500 ms (on an average of 300 ms) after the observation of the target stimulus. P300 can be recorded most prominently over the midline of the brain. This ERP is often used to monitor the subject's attentiveness [159].

In the context of BCI, most of the existing BCI-based control applications are open-loop and require generating on-off commands for motors and actuators to switch on and off as needed by the users. Even in the traditional closed-loop BCI system mental-motor switching or ON-OFF control is employed to activate or deactivate the external device(electrical motor). Such a system utilizes visual feedback to reduce positional errors, but due to the ON-OFF control strategy, it always yields a non-zero steady-state error. Because of the above limitations of the ON-OFF control strategy, a continuous control that keeps track of the positional error to generate the control command for the motor is needed. This type of brain-actuated controller modulates the visually attended positional error into suitable control commands to reduce positional error toward zero within a small settling time. Generally, such controllers are referred to as auxiliary controllers as they attempt to translate high-level motor intention into low-level control commands.

A brief survey of the auxiliary control policy is attended here. Unlike traditional position controllers, the auxiliary controller is placed in both forward and feedback paths [115][116][117][118]. Controllers placed in the forward path generally use the following control strategy; i) PID control strategy,[100][101] ii) Shared Control strategy[160][161], and iii) Model predictive control (MPC) strategy[115][116]. In both the PID and MPC-based strategy a mathematical model of the system is assumed and the controllers attempt to provide an optimal control signal to actuate the external device based on the user intent. In MPC the controller is also able to satisfy safety constraints and certain input/output constraints.. On the other way, the shared control strategy uses an external sensor such as camera/Kinect sensor etc. to obtain information about the surrounding environment of the robot, which is combined with the user's original intent to generate a suitable navigational command for the external agent (robot).

One fundamental problem in model-predictive control lies in the complexity of modeling the bio-physical aspects of the controller. Additionally, the MPC can hardly model the individual subject's mental intentions about the high-level commands. The shared controller, on the other hand, losses the sense of authority of the user due to its autonomous nature of executing the assigned task. Because of the above limitations, we in the present chapter attempt to model an alternative brain-actuated controller capable of translating user intention into motor commands to achieve desired performance in stability and steady-state error irrespective of the individual difference among the subjects.

In closed-loop position control applications, we often come across a feedback signal for the automatic detection of zero positional error with respect to a desired step input of the angular position command. Both P300 [162] and ErrP [163] have been employed by previous authors to address the problem. The ErrP used in the present context, often referred to as observational ErrP, which is elicited from the medial frontal region with a negative deflection of approximately 250 ms after the subject observes a machine (or a person) to commit errors. Here, committing an error refers to crossing the desired angular position/set-point of the position-control experiment. ErrP, in general, is a reliable feedback signal. One of the early applications in this regard is Motor Imagery(MI) based actuation of a robotic arm and de-actuation by sensing the onset of Error-related Potential(ErrP) from the subject's brain. The MI signal is released when the subjects intend to move a link of a robot arm and ErrP is released when they inspects the robot to cross the target location. However, its magnitude is diminished in subjects with spinal cord injury and schizophrenic disorders [164], [165]. ErrP amplitude and latency also suffer from inter-subject and inter-trial variability [166], [[167]. Lastly, ErrP is best elicited with discrete events [16]. Here the robot motion is continuous which makes the ErrP signal less suited to being used as feedback marker. Because of the above limitations of ErrP, we prefer to use P300, which is released approximately 300 ms from the onset of positional zero-crossing errors, experienced by the subject. Additionally, the limitation of this scheme is fixed order link selection for position control and large steady-state positional error.

In the current work, SSVEPs corresponding to different flickering frequencies have been used to randomly select a link of the robotic manipulator. Motions of the manipulator links are then activated by the subject's motor imagination. When the selected link reaches the desired target position, P300 is elicited in the subject's EEG. The P300 is used to freeze the current motion of the robotic link. Since elicitation and detection of P300 waveform requires a finite (non-zero) time, the robotic link crosses the target position by a small angle before its motion is stopped. An auxiliary controller is designed to align the link with the target with high precision. Now the controller actuates the link movement in the reverse direction of the last motion, i.e., towards the target location. Throughout the motion of the link, the controller maintains a gradually diminishing speed. When the link reaches the target again, P300 appears once again and the aforementioned process is repeated. Although P300 is released by the subject for both forward and backward zero error crossings, here the zero-crossings during forward motion are represented by positive impulses and zero-crossing during backward motion by negative impulses. This cyclic operation continues as long as the steady-state error drops down to a threshold value determined by the user. Evidently, this process can align the link with the target with a high degree of accuracy. The BCI framework developed here consists of 4 successive stages. First, the acquired EEG signals are pre-processed to make them free from noise and artifacts. Second, the relevant features are extracted from the pre-processed signals. Third, the different classes of the experimental data points are determined by suitable pre-trained classifiers. Finally, control commands are generated to actuate the manipulator to serve the desired requirement.

The original contribution of the work lies in the architectural design of the hybrid BCI-based closed-loop position control system with special emphasis on the modeling of the auxiliary po-

sition controller from the response analysis of the control system. The model presumes an expected time-varying response of the position-controller like that of stable second-order dynamics due to the impulsive occurrence of error input at the time of target crossing and thus determines the transfer function of the position controller in the Laplace-domain. Given the transfer function of an armature-controlled DC motor (for one link of the robot), a stability analysis of the closed-loop position control system, involving both the motor and the controller, is performed using the Root-Locus analysis [168]. The analysis reveals an interesting observation that the stability margin of the closed-loop system is determined by the initial choice of maximum speed of the robotic links. The proposed method of analytical formulation of the auxiliary controller is novel in the BCI-based position-control system. Secondly, a simple but elegant feature selection algorithm is proposed to automatically select the best feature set from a pool of features with the aim to minimize the distance between pairs of intra-class data points and maximize the distance between pairs of inter-class data points for the selected feature set. An evolutionary algorithm is used to handle the said optimization problem. Third, it includes a provision for random ordered link-selection by SSVEP based BCI, instead of a fixed ordered link selection adopted in [19]. Above all, the proposed method reduces the steady-state error drastically (0.12%) in comparison to the one presented in [14], thus justifying its scope in high-precision rehabilitative appliances.

The chapter has 9 sections. Section 2.2 provides a general overview of the complete scheme. Controller design and stability analysis are examined in Section 2.3. Section 2.4 deals with signal processing, feature extraction, and classification of used brain signals. Experimental details are presented in Section 2.5. Statistical analysis and controller performance analysis are given in Sections 2.6 and 2.7, respectively. Comparison with similar works is undertaken in Section 2.8. Finally, the concluding remarks are listed in Section 2.9.

2.2 System Overview

This section provides an overview of the BCI-based position control for a multi-link robot arm. The robot arm used here has 6 degrees of freedom (DOF) with a maximum reach of 580 mm in any arbitrary direction. The frame assignments for all the joints along with the directions of positive angular motions about the z-axis are shown in Fig.2.1. The current work uses only the first three links, L1, L2, and L3, of the robot arm. Fig. 2.2 illustrates the complete position control scheme. Three different brain signals, including SSVEP, ERD/ERS, and P300 are employed here to control the position of the robot arm in its workspace.

An overview of the BCI-based position control scheme from a control theoretic point of view is given in Fig. 2.4. The scheme involves the link selection of a robot arm randomly as determined by the user based on the positional shift needed for the end-effecter. Link selection is done by using Steady-state visual evoked potential(SSVEP). After a link is selected, the subject performs Left/Right-hand motor imagination (MI) to turn the selected link counterclockwise or clockwise respectively. Once the robotic link crosses the target position, the subject releases a P300 signal, and the BCI system on detecting the P300 signal, commands the selected robotic link to turn oppositely with exponentially reduced speed. Once the subject decides that the

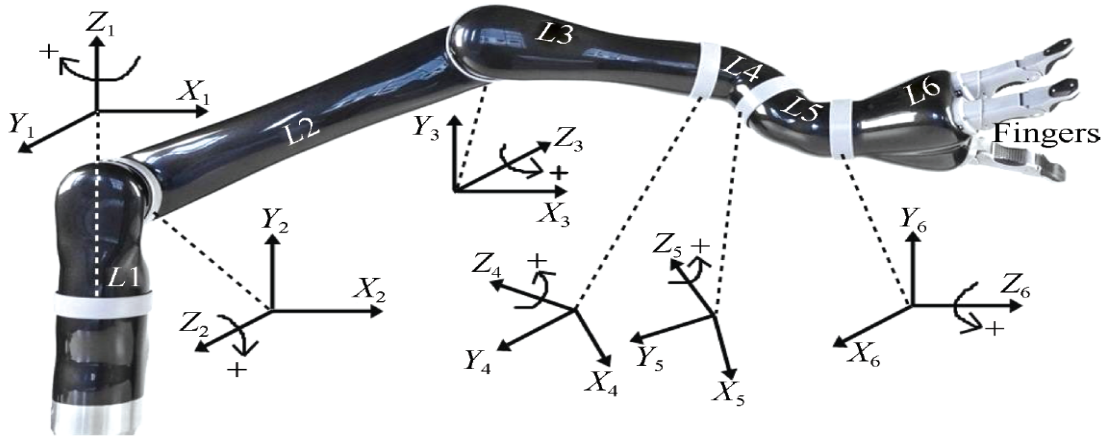

Figure 2.1: Frame assignments of a 6 Link Jaco Robot Arm

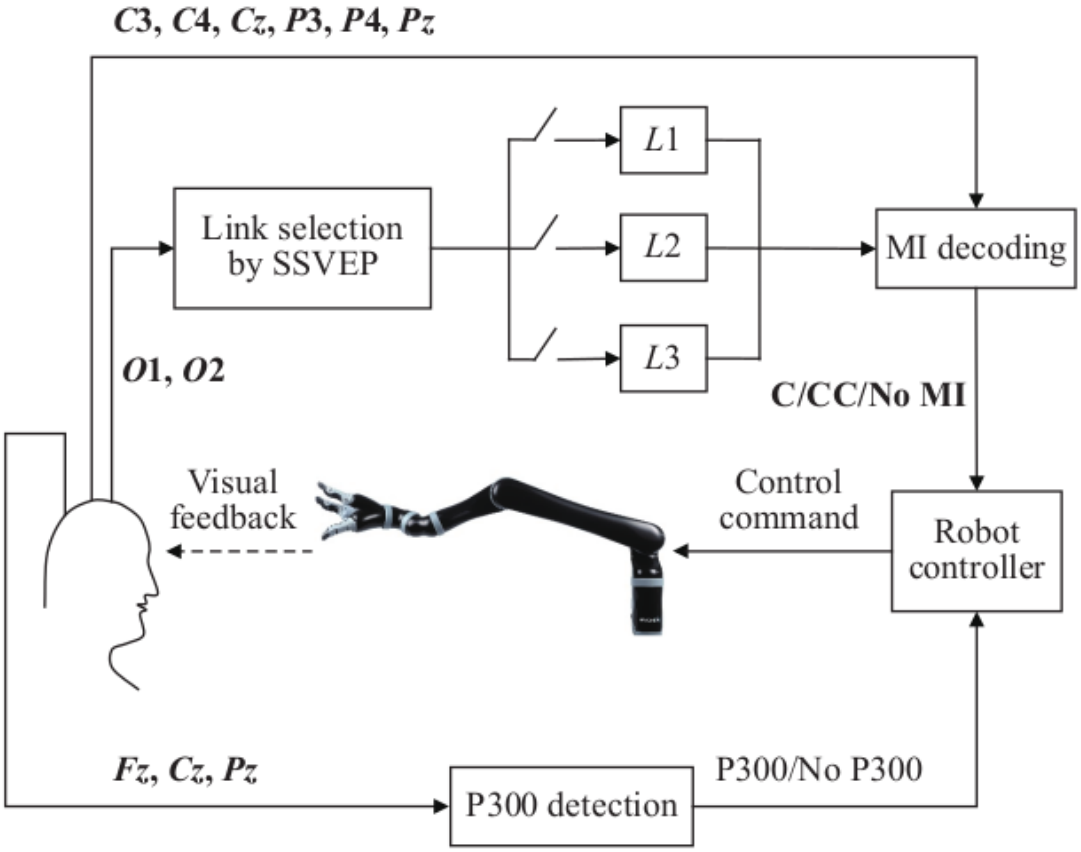

Figure 2.2: Schematic diagram of complete position control scheme

steady-state error for the link is small enough, he/she stops paying attention to the link and selects the next link he feels is important for the position control of the end-effecter. The link selection is continued one by one until the end-effecter reaches the target position. The flowchart of the sequence of actions is given below in Fig. 2.3

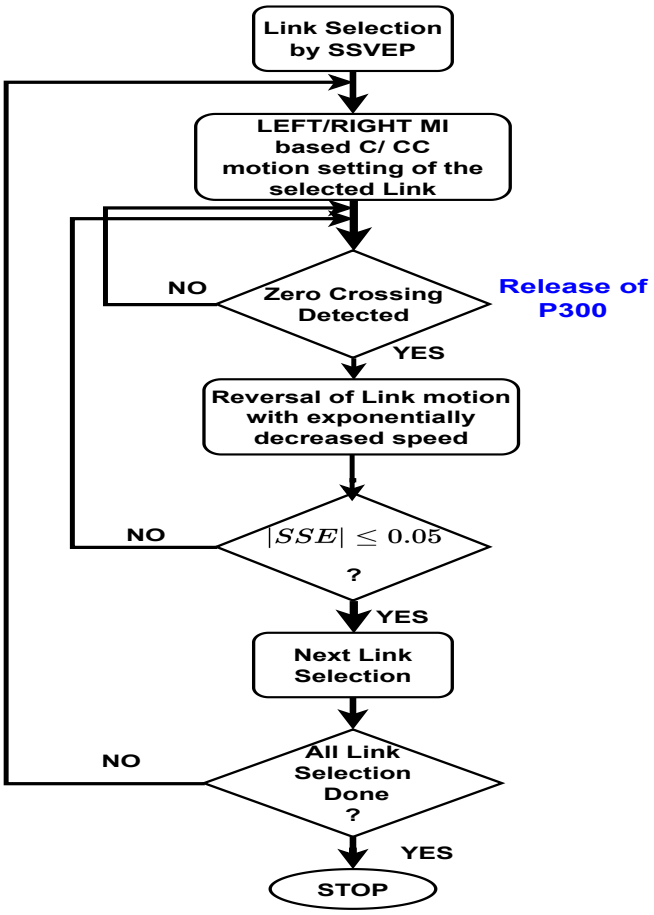

Figure 2.3: Flowchart of Sequence of Actions

Here, we propose a 2-loop position control system, where the inner loop, also called the stabilization loop, is used to enhance the stability of the complete 2-loop system because the negative feedback in the inner loop undertakes velocity control with a target to control the position of the end-effector of a robot arm. The velocity reference of the inner loop is generated by the outer loop with the help of the brain-actuated controller.
The brain-actuated controller starts functioning with the P300 response of the human subject. The subject releases P300 on observing the zero crossings of the positional error signal. The P300 signal released by the subject is then utilized to modulate the velocity of the position controller.

2.2.1 Link Selection by SSVEP Detection

Existing research on BCI-controlled robotics utilizes a fixed order of link selection, i.e., the robot follows a fixed temporal sequence of link selection irrespective of the target position of the end-effecter [19]. In many real-world position-control applications, dynamic link selection based on

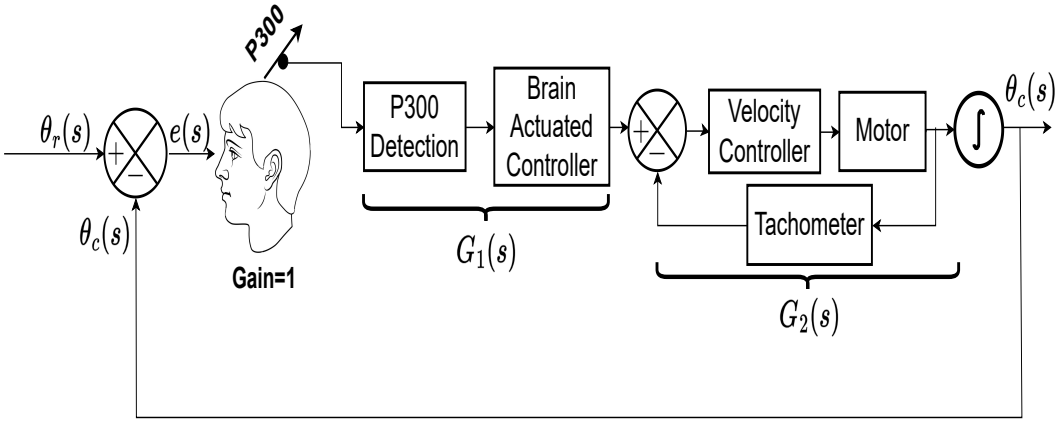

Figure 2.4: Schematic diagram of Proposed BCI system from the control-theoretic perspective

the users' choice is required to improve the speed and accuracy of the position-control task. This chapter aims at developing a dynamic order of link selection. One fundamental hindrance with dynamic link selection is the lack of communication to the robot about the choice of the current link by BCI means. This has been overcome here by attaching light emitting diodes (LEDs) to individual links, flickering at different frequencies. The subject needs to stare at one of the links carrying a LED, flickering at a fixed frequency, which in turn releases a special brain signal, called SSVEP. The SSVEP is frequency modulated at the flickering frequency and yields a large amplitude at the flickering frequency of the specific LED, which the subject pays attention to. The subjective interest of link-selection thus becomes apparent from the frequency spectra of the SSVEP.

2.2.2 Motion Activation by MI Decoding

Selection of a link by the subject indicates that the subject prefers to activate the link for clockwise/counterclockwise turning using motor imagery signals. The ERD/ERS motor imagery signal has been used to direct the motion of the link in the desired direction. The ERD/ERS signal exhibits a de-synchronization in the μ- and the β-band power, followed by a synchronization in the β-band power, and thus takes a "v"-like wave-shape. Because of the contra-lateral connectivity between limbs and the brain, ERD/ERS is released by the left motor cortex for right-hand motor imagery (RHMI) and by the right motor cortex for left hand motor imagery (LHMI). Suppose the subject wants to move the link clockwise by RHMI and counter-clockwise by LHMI. Thus it is preferred to detect LHMI (RHMI) from the right (left) brain lobe for efficient decoding of the motor imageries.

Decoding of ERD/ERS requires determining the ground-truth and then identifying similar ERD/ERD traces from the experimental instances. One simple method to construct the ground truth is to take average of the available ERD/ERS traces from the experimental instances [169], where averaging is performed at the sample points over all instances. The averaged trace is defined as the ground truth ERD/ERS. The other ERD/ERS traces in the pool that satisfy the Gaussian characteristic at all the sample points are regarded as positive instances for the true

ERD/ERS class. The Gaussian criterion is given below for the sake of convenience. Let Av_i be the average value of all ERD/ERS traces at sample point i, and σ_i be the standard deviation of all the traces at sample point i. Then an unknown ERD/ERS trace is presumed to be close enough to the ground truth if

$$\frac{|x_i^j - Av_i|}{\sigma_i} \leq 3 \forall i \tag{2.1}$$

where, x_i is amplitude of the j-th ERD/ERS at sample point i. The true class of ERD/ERS thus can easily be obtained. To identify the training instances for the false class, the false negative ERD/ERS instances are considered, and the above steps of positive instances are repeated. After the 2 classes of the ERD/ERS traces are generated, we need to extract certain ERD/ERS features, which together with the class label represent a sample training instance. Hzorth parameters, adaptive auto-regressive parameters (AAR) and discrete wavelet co-efficient (DWT) are few useful features, which have received wide publicity in BCI research. Common spatial pattern (CSP) features have also shown promising performance in LHMI/RHMI classification tasks [31]. We would use CSP features for MI classification in this paper.

2.2.3 Alignment With the Target by P300 Identification

Once the MI signal is released by the subject's brain, one selected link of the robot starts turning in the clockwise/counter-clockwise direction, and as a consequence, the end-effecter/ link at some time point t, counted from the onset of the ERD/ERS signal crosses the fixed (pre-defined) target position. This phenomenon where the end effecter crosses the target position acts as an infrequent stimulus to the subject, causing him/her to release the P300 signal. In other words, release of the P300 signal from the subject's brain in the present context, is a clear indication that the end-effecter/link crossed the target position, and thus needs a corrective action. Here, the release of P300 is used as corrective feedback to the BCI system to turn the robotic link in the reverse direction of its current movement direction with a gradual decay of its angular speed. The process is continued until the positional error goes below a threshold as mentally desired by the subject. The continued reduction in speed, and reversal of motion around the target-point, effectively results in a reduction in the peak overshoot and steady-state error. The pseudo-code for the proposed scheme of position control of a robotic arm is given below in algorithm 1. The code is self-explanatory.

2.3 Controller Design and Stability Analysis

Traditional BCI-based position control paradigms realize open-loop control using MI. As a consequence, the performance of the position control system cannot offer desired performance. A closed-loop feedback control system realization using BCI is a necessity to provide the desired performance in position control, such as reduced steady-state error and peak overshoot. The proposed control strategy can reduce both the steady-state error and peak overshoot than the ones obtained in open-loop position control. The peak overshoot is reduced by taking into account the occurrence of the P300 signal. Once the desired robotic link crosses the targeted position

Algorithm 1: Pseudo-code for the proposed position control scheme

Begin
Initialize: Initialize angular speed ω of each link = $\omega_0 \times e^{-\lambda t}$, where ω_0 is the initial angular speed, $\lambda(>0)$ is the decay factor, and t = iteration; ε= lower limit of angular speed;
Repeat:
1. Use SSVEP decoding to select a link of the user's choice;
2. Use MI decoding to find the direction of motion (clockwise/ counter-clockwise) of the selected link with a pre-selected angular velocity ω;
While $SSE > \varepsilon$ **do Begin**
3. Continue moving the link until P300 is detected, implying that the selected link crossed the mentally-imagined target position;
5. After the link crosses the mentally imagined target position, stop it temporarily, and reverse the motion of the link with the magnitude of angular speed $\omega = \omega^{'} \times e^{-\lambda t}$, where $\omega^{'}$ = the angular speed of the link just before zero-positional crossing;
End-While;
Until: the movement of all desired links is over;
End.

(target point in the case of translational movement along a line, target angle in the case of rotational movement on a plane, and target plane in the case of rotation of the link from one plane to the other), and then by reversing the motion of the link. The steady-state error here depends on the response time of the P300 signal and the time required for robotic motion. The steady-state error can be reduced by gradually reducing the speed of the motor exponentially with time t, and reversing motor speed each time the robotic link crosses the target position.

Let the i-th link at time t be at position $c(t)$ and r be the visually fixed target position for the same link. Then error at time t is defined as follows:

$$e_i(t) = r - c(t) \tag{2.2}$$

A typical P300 response is generated by the brain whenever $e_i(t)$ crosses zero value, i.e., $e_i(t)$ is slightly positive or negative. It is important to mention here that in classical control theory [44], the control signal $u_i(t)$ for the i-th link is a the function of error $e_i(t)$. However, in a BCI-based position control, we do not have an absolute measure of the magnitude of error as the error is recognized visually by the brain from the zero-crossings of $e_i(t)$, and P300 can only ensure the occurrence of zero-crossings without having any information about the magnitude of the error. Thus, we need a different formulation of the controller, which is given below.

Let v_0 be the initial velocity of the controller before the release of the first P300 signal. On receiving the P300, the velocity of the controller is reversed and decreased exponentially until the occurrence of the next P300 signal. The process is repeated until the subject feels that the error magnitude is too less and steady-state error is nothing to reach. The velocity profile set by the brain-actuated controller is used as the reference input to the inner loop. The integrator at the output of the inner loop provides the angular position of the motor, which is fed back to the positional error detector to detect the positional error. The schematic architecture of the

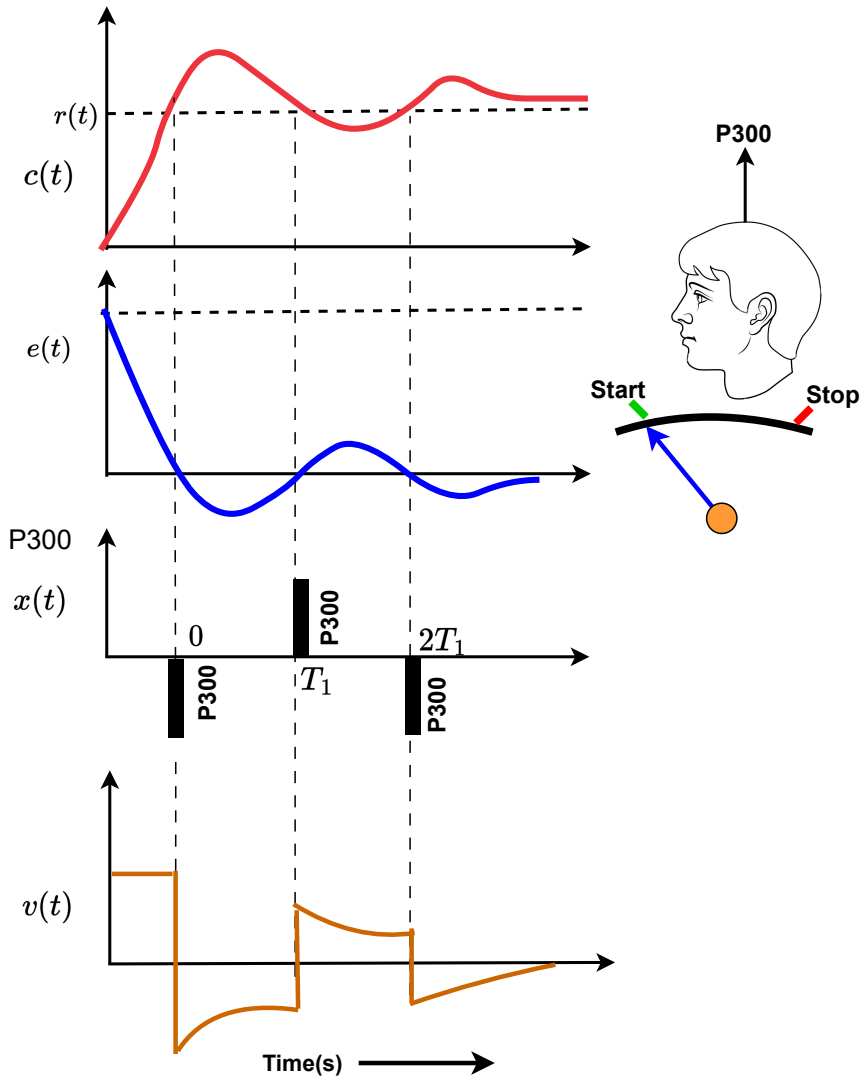

Figure 2.5: Variation of link velocity ($v(t)$) due to the occurrences of the P300 signal.

proposed system is given in Fig.2.4.

In Fig.2.4 the outer loop is used to sense the zero crossings in positional error which is detected by the experimental subject to return P300 in response to the zero crossings. The brain-actuated controller generates the control command for the velocity setting for the inner loop. The design of the brain-actuated controller is outlined here.

2.3.1 Design of Brain-Actuated Controller

It is evident from Fig.2.5 that the output velocity profile can be expressed as a sum of decaying exponential signal within the interval $T_1 = T + \Delta T$, where T is the time interval between two consecutive P300 generations and ΔT is the time taken by the system to identify P300 (for the sake of brevity ΔT is not shown in the figure. Now for n number of P300 input signal the velocity profile $v(t)$ will exhibit n cycles of oscillation. Considering $T + \Delta T = T_1$, the above phenomenon

can be mathematically expressed as the following

$$v(t) = \sum_{k=1}^{n} (-1)^k v_0 e^{-\lambda t} [u(t-(k-1)T_1) - u(t-kT_1)] \tag{2.3}$$

As the v_0 term in the above expression is constant, hence the above expression reduces to

$$v(t) = v_0 \sum_{k=1}^{n} (-1)^k e^{-\lambda t} [u(t-(k-1)T_1) - u(t-kT_1)] \tag{2.4}$$

Now the input sequence can be expressed as n number of P300 impulses ;

$$x_1(t) = -\delta(t) + \delta(t-T_1) - \delta(t-2T_1) + ... + \delta(t-(n-1)T_1) \tag{2.5}$$

The above equation can be written in a more generalized way using the following notation;

$$x_1(t) = \sum_{k=1}^{n} (-1)^k \delta(t-(k-1)T_1) \tag{2.6}$$

Now the transfer function of the mental controller can be obtained from the input-output relation defined in Eq.2.4 and Eq.2.6.

The transfer function $C(s)$ is obtained by taking the ratio between the Laplace transform of the above two equations with zero initial condition.

$$G(s) = \frac{\mathcal{L}(v(t))}{\mathcal{L}(x(t))} = \frac{V(s)}{X(s)} \tag{2.7}$$

Now the $V(s)$ and $X(s)$ can be found using the following way First considering Eq.2.4 and taking the Laplace transform, we get

$$\begin{aligned} V(s) &= v_0 \sum_{k=1}^{n} (-1)^k [\frac{e^{-(k-1)(s+\lambda)T_1}}{s+\lambda} - \frac{e^{-k(s+\lambda)T_1}}{s+\lambda}] \\ &= \frac{v_0}{s+\lambda} \sum_{k=1}^{n} (-1)^k e^{-(s+\lambda)kT_1} (e^{(s+\lambda)T_1} - 1) \\ &= -\frac{v_0}{s+\lambda} (e^{(s+\lambda)T_1} - 1) \sum_{k=1}^{n} (-1)^k e^{-(s+\lambda)kT_1} \end{aligned} \tag{2.8}$$

The rightmost term of the above equation can be expressed as a sum of an alternating GP series whose first term is $-e^{-(s+\lambda)T_1}$ and the common ratio is $-e^{-(s+\lambda)T_1}$. Hence the Equation 6 can be simplified as below,

$$V(s) = -\frac{v_0}{s+\lambda} (e^{(s+\lambda)T_1} - 1)(-e^{-(s+\lambda)T_1}) (\frac{1-(-e^{-(s+\lambda)T_1})^n}{1-(-e^{-(s+\lambda)T_1})}) \tag{2.9}$$

Now considering $n = odd$ (Two successive P300 create an oscillation, but for the system designed by considering n of oscillations, the last even-numbered P300 (returning P300) does not actuate the controller ,i.e., as n starts from 1, the system always receives an odd number of P300

impulses), Eq.2.9 takes the following form;

$$V(s) = -\frac{v_0}{s+\lambda}\left(\frac{1-e^{(s+\lambda)T_1}}{1+e^{(s+\lambda)T_1}}\right)(1+e^{-(s+\lambda)nT_1}) \tag{2.10}$$

Using Pade's approximation[170];

$$\begin{aligned} V(s) &= -\frac{v_0}{s+\lambda}\frac{1-\frac{1-(s+\lambda)T_1/2}{1+(s+\lambda)T_1/2}}{1+\frac{1-(s+\lambda)T_1/2}{1+(s+\lambda)T_1/2}}(1+e^{-(s+\lambda)nT_1}) \\ &= -\frac{v_0T_1}{2}\left(1+\frac{1-(s+\lambda)nT_1/2}{1+(s+\lambda)nT_1/2}\right) \\ &= -\frac{-2v_0}{n(s+2/nT_1+\lambda)} \\ &= -\frac{-2v_0}{n(s+2/(n(T+\Delta T))+\lambda} \end{aligned} \tag{2.11}$$

Now, the Laplace transform of the input sequence is obtained by taking the Laplace transform of both side of Eq.2.6;

$$X(s) = -1+e^{-sT_1}-e^{-2sT_1}+....+e^{-(n-1)sT_1} \tag{2.12}$$

which can be approximated as a GP series whose first term is -1 and the common ratio is $(-e^{-sT_1})$. The sum of the series is given below when $n = odd$;

$$X(s) = -\frac{1+e^{-nsT_1}}{1+e^{-sT_1}} \tag{2.13}$$

Again using the Pade's approximation we get;

$$\begin{aligned} X(s) &= -\frac{1+\frac{1-nsT_1/2}{1+nsT_1/2}}{1+\frac{1-sT_1/2}{1+sT_1/2}} \\ &= -\frac{1+sT_1/2}{1+nsT_1/2} \\ &= -\frac{1+sT/2+s\Delta T/2}{1+nsT/2+ns\Delta T/2} \end{aligned} \tag{2.14}$$

As ΔT is very small, we can safely ignore $s\Delta T$ in the numerator. The final form of the expression is given below;

$$X(s) = -\frac{s+2/T}{n(s+2/(n(T+\Delta T))}\left(\frac{T}{T+\Delta T}\right) \tag{2.15}$$

The transfer function $G(s)$of the mental controller is obtained by putting the values in Eq.2.7 as received from Eq.2.11 and Eq.2.15. $G(s)$ is given below;

$$G(s) = \frac{2v_0\alpha(s+2/nT_1)}{(s+2/T)(s+\lambda+2/nT_1)} \tag{2.16}$$

where $\alpha = \frac{T+\Delta T}{T}$ and $T_1 = T + \Delta T$. For very small value of ΔT, α is taken as unity. The final transfer function is given as

$$G(s) = \frac{2v_0(s+2/nT_1)}{(s+2/T)(s+\lambda+2/nT_1)} \tag{2.17}$$

The mental transfer function has poles at two different locations of negative real axis $Pole = -2/T, -(\lambda+2/n(T+\Delta T)$ and a $Zero = -2/n(T+\Delta T)$.

2.3.2 Design of the Inner Stabilization Loop

The inner loop is realized using a proportional plus derivative controller in cascade with the plant which here is an armature-controlled DC motor. The PD controller is selected for the following two reasons. The proportional control for the given second-order plant results in an overshoot, which is avoided by adding a derivative control which in anticipation of a large error rate increases the damping of the system resulting in a decrease in the overshoot. Secondly, the PD controller also increases the speed of the response.

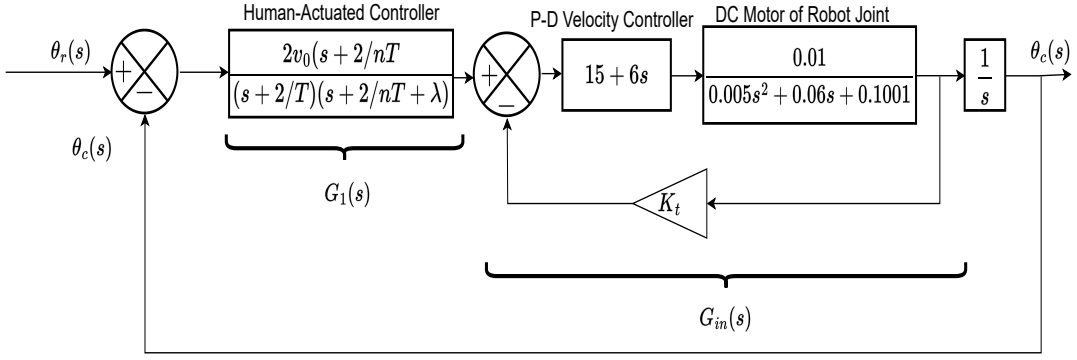

Figure 2.6: Block diagram of proposed control scheme

2.3.2.1 Design of PD controller

Links of the 3-DOF robot arms are connected through the rotary joints which are controlled by the armature-controlled DC motors. The parameters of the motor are given in Table2.1.

Table 2.1: Parameters of the DC motor used in the experiment

Parameter	Description	Value
J	Armature Inertia	0.01 N-m-s$^2/rad$
b	Armature viscous friction	0.1 N-m-s/rad
K	Mechanical Gain	0.01 N-m/A
L	Armature inductance	0.5 H
R	Armature resistance	1.2 Ohm

The transfer function of the motor is given below,

$$\begin{aligned} M(s) &= \frac{K}{(Js+b)(Ls+R)+K^2} \\ &= \frac{0.01}{0.005s^2+0.06s+0.1001} \end{aligned} \tag{2.18}$$

The motor has two poles at -9.99 and -2 on the negative real axis.
The PD controller here acts as a velocity controller and receives the velocity feedback from the motor using a tachometer. The transfer function of the PD controller is given below,

$$C(s) = K_p + K_d s \tag{2.19}$$

The block diagram of the system is given in Fig. 2.6 where an integrator is used to obtain the position from the velocity response of the motor. The PD controller inside the stability loop is tuned to satisfy the criteria of zero steady-state error and $< 1s$ settling time of the system. The tuned values of the parameters are found to be $K_p = 15$ and $K_d = 6$. The stability loop $G_{in}(s)$ is obtained with the following equation

$$\begin{aligned} G_{in}(s) &= \frac{M(s)C(s)}{1+K_t M(s)C(s)} \\ &= \frac{0.06s+0.15}{0.005s^2+0.09s+0.175} \\ &= \frac{12(s+2.5)}{s^2+18s+35} \end{aligned} \tag{2.20}$$

where the K_t is the gain of the tachometer. The step response of the velocity loop is given in Fig. 2.7.

2.3.3 Stability Analysis of the Complete Loop

The outer closed-loop system involves 3 parameters λ,n, and T. Naturally, the stability of the system is expected to depend on the choice of the parameters. An exhaustive test of stability is needed for the optimal choice of λ for a given value of n and T. The variable n is user dependent as the user decides when he/she should no longer continue participation in the experiment as plant response (position) reaches a steady-state value. However, the parameter T is approximately known priory. Here T denotes the delay between two successive P300 occurrences. Typically T lies between $1s$ to $3s$. Hence, we consider the variation of T in the range [1,3] and the variation of n in the range [1 3 5] to check the parameters of instability (if any) in the said range. This is done in two phases. In the first phase, we draw root-contour plots of the system by keeping 2 variables out of λ, T,n constant. It is evident from the root contours that the stability of the system increases with the increasing value of λ. However, it is apparent from the transfer function of the system that the settling time of the system also gets affected by the λ. Hence an optimal value of λ must be obtained for satisfactory performance of the system.
In the second phase, we plot the ratio $\frac{K}{T_s}$ (where K is the DC gain of the system and T_s is the

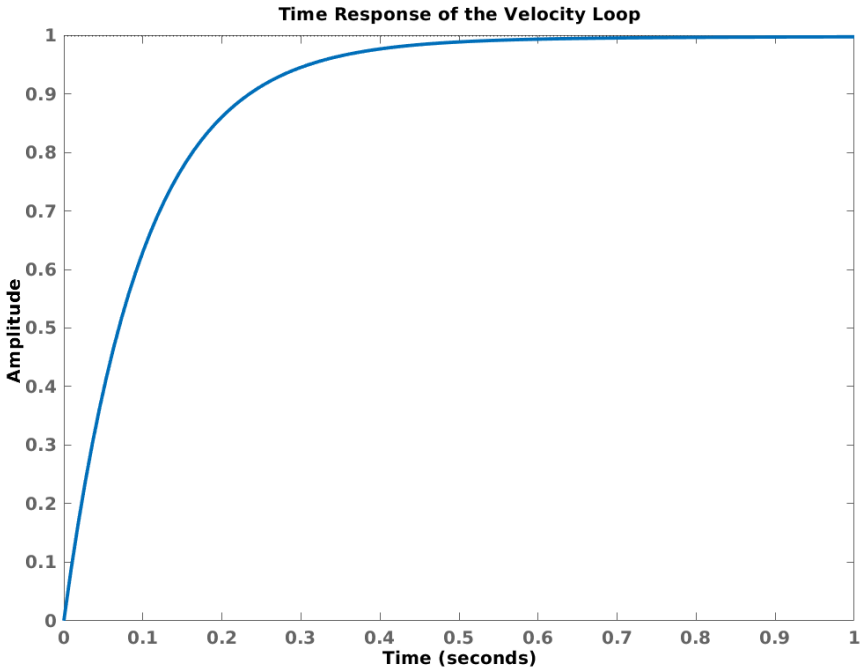

Figure 2.7: Step Response of the velocity loop

settling time of the system) and search for the optimal value of the λ for which the above ratio is highest. The surface plot of the ratio defined above for different values of n and λ is given in Fig. 2.9. The plot is obtained by varying n and λ for a given value of T.

The stability of the complete system is provided in Fig.2.8 through root contours for a specific value of n, T and with increasing value of λ, but the motor pole at -14.6(marked with red box) location does not get altered for different combination of n and T, hence for the sake of brevity this pole is not explicitly shown in the root contours for different combination of controller parameters presented through Fig.2.9 to 2.17 . The root contours are plotted with increasing values of λ for a given value of n and T. The DC gain of the system is reported in the figure by the notation K. It is evident that for a wide range of variations of the controller parameters, the system remains stable. However, the relative stability greatly varies with the parameter variation. For a given value of n and T, the stability margin is found to be increasing with the increasing values of λ. Another important observation is the dependency of the system's relative stability on the choice of initial velocity. The individual root locus for a specific choice of n, T, and λ is stable for a certain range of DC gain K, which is numerically equal to double of the initial velocity. i.e. $K = 2v_0$, hence the DC gain explicitly depends on the initial velocity v_0. As an example, a specific parameter combination ($n = 3$, $T = 1$, and $\lambda = 0.5$) yields the stability margin $0 < K < 23.56$ for the system to be stable, i.e. the initial velocity v_0 must be below $23.56/2 = 11.78$ to maintain the stability criteria.

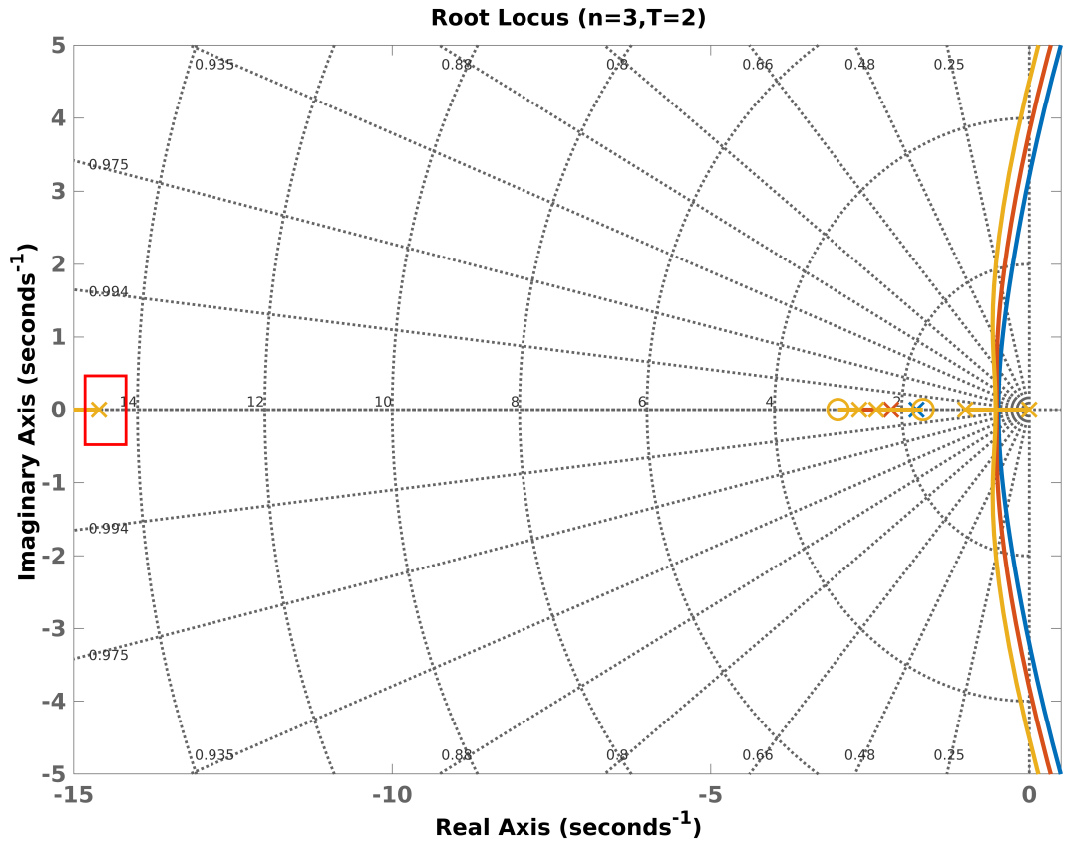

Figure 2.8: Root Locus of the complete system with increasing values of λ.

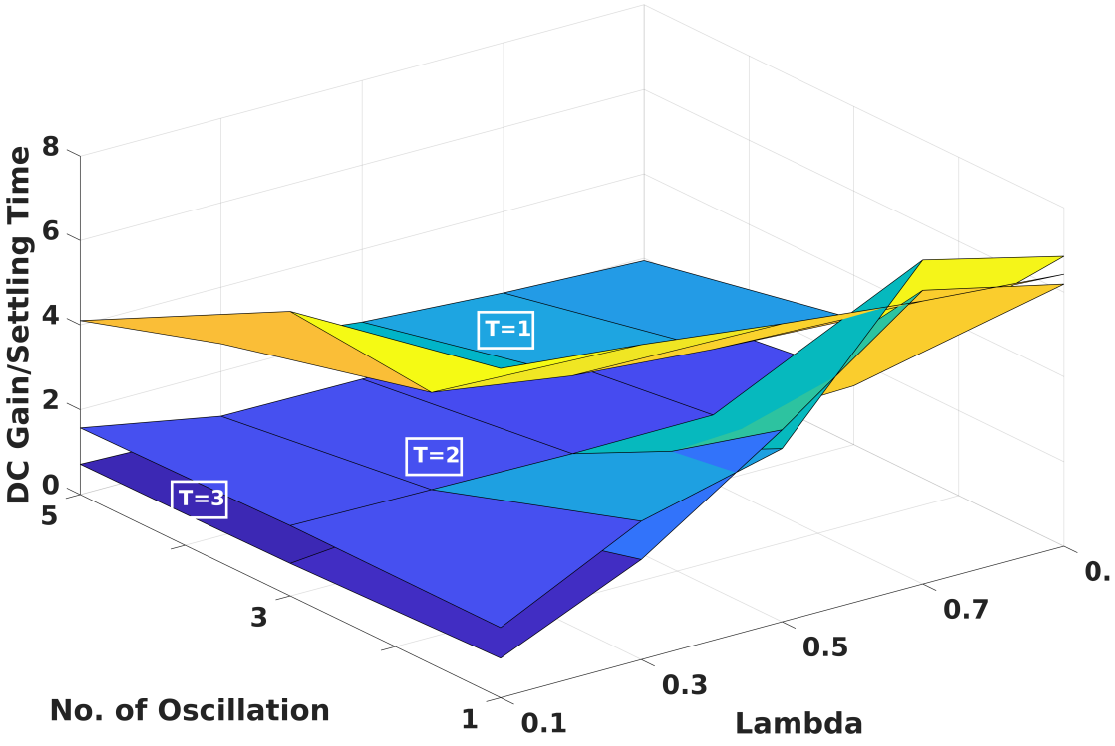

Figure 2.9: Surface plot of the $\frac{K}{T_s}$ ratio for different values of n,T and λ

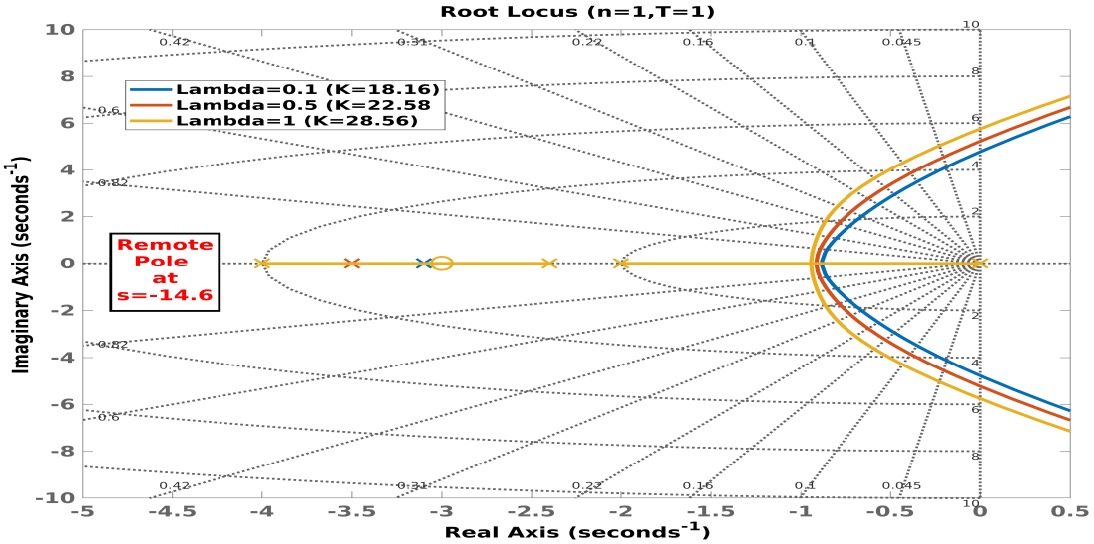

Figure 2.10: Root-Contour plots of the overall system for n=1,T=1 and increasing value of lambda

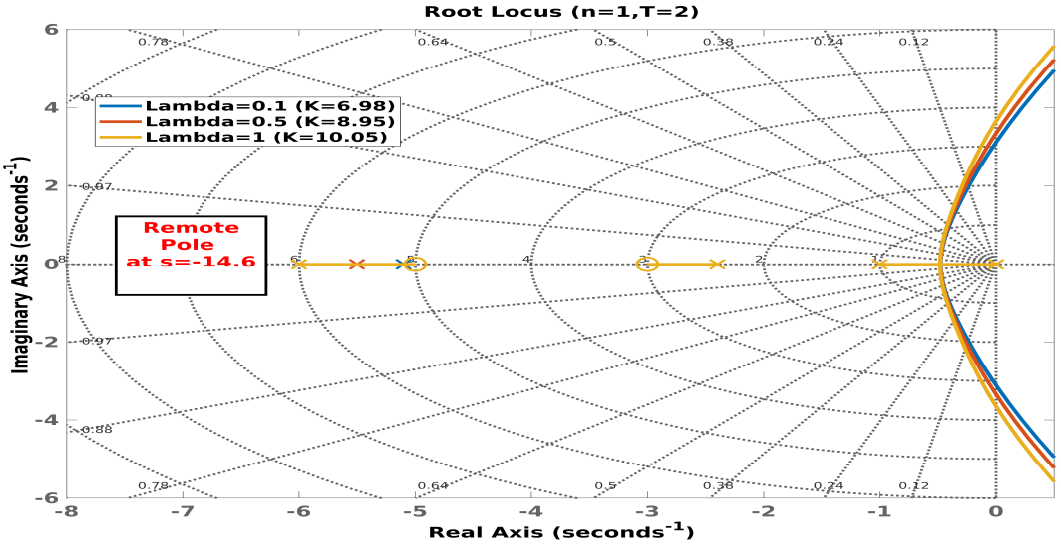

Figure 2.11: Root-Contour plots of the overall system for n=1,T=2 and increasing value of lambda

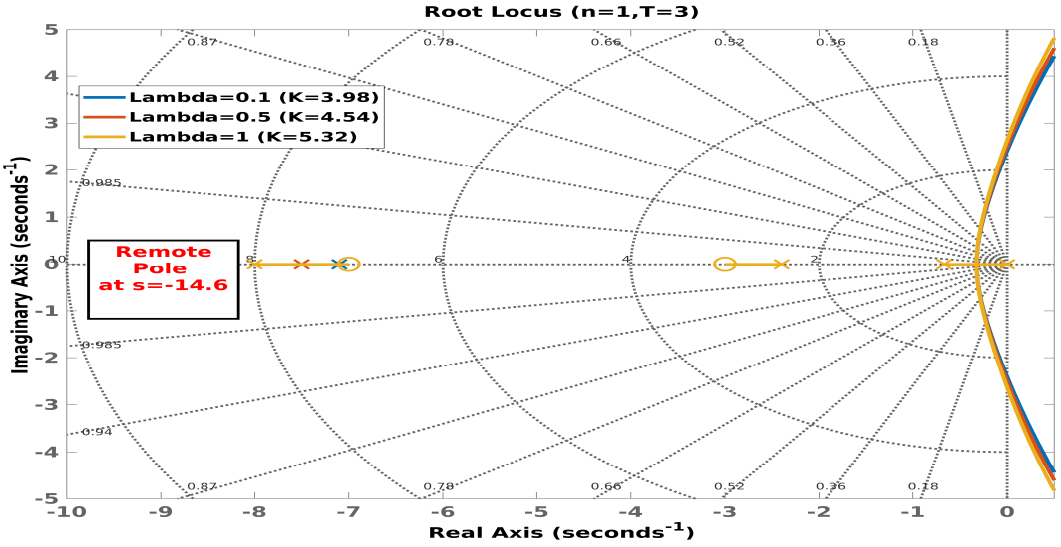

Figure 2.12: Root-Contour plots of the overall system for n=1,T=3 and increasing value of lambda

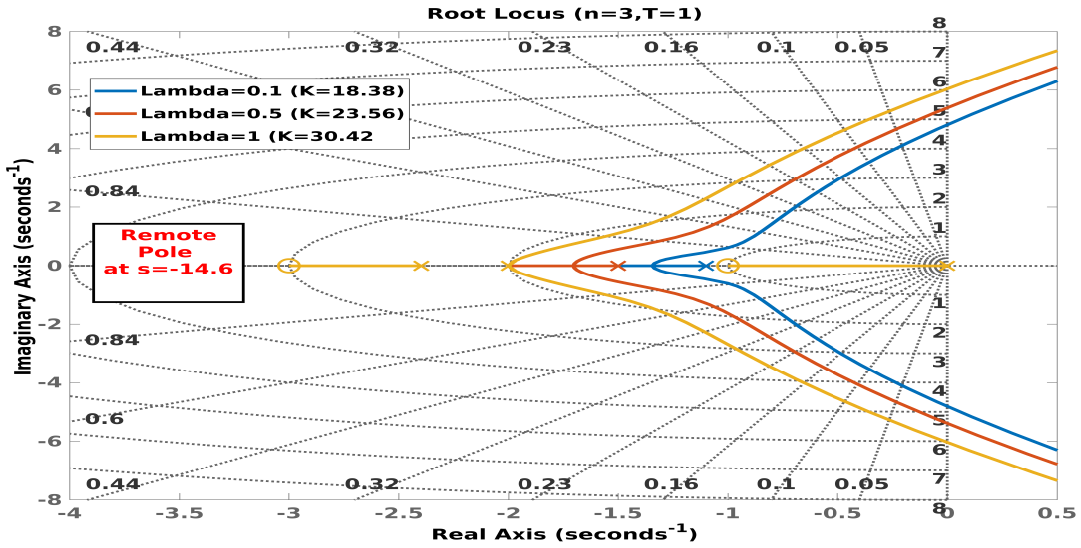

Figure 2.13: Root-Contour plots of the overall system for n=3,T=1 and increasing value of lambda

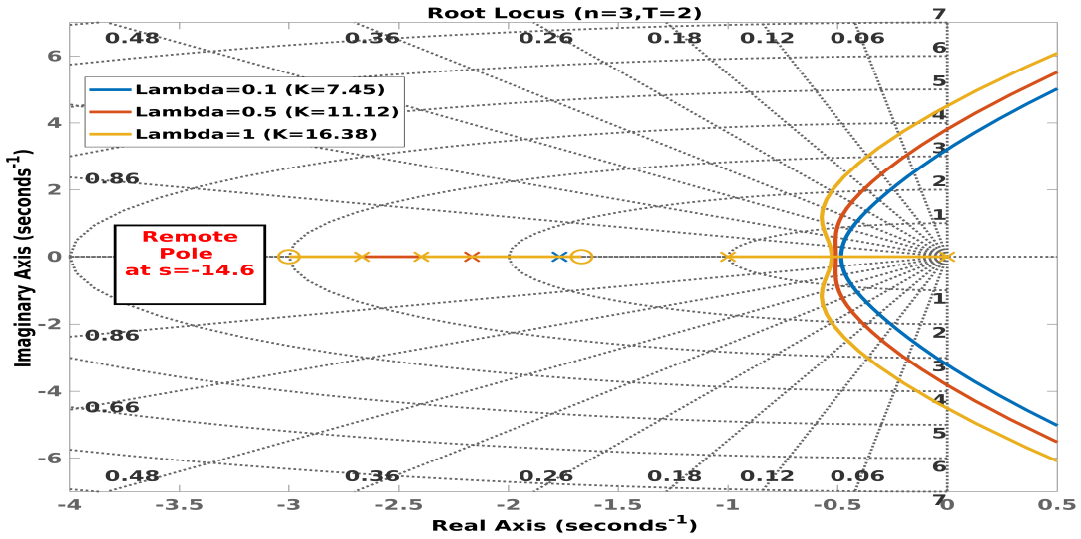

Figure 2.14: Root-Contour plots of the overall system for n=3,T=2 and increasing value of lambda

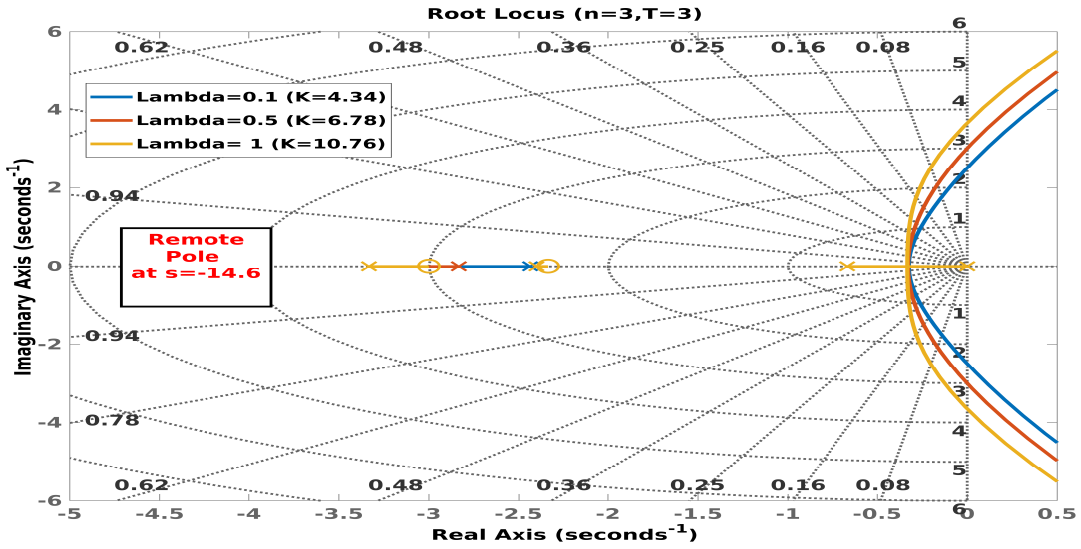

Figure 2.15: Root-Contour plots of the overall system for n=3,T=3 and increasing value of lambda

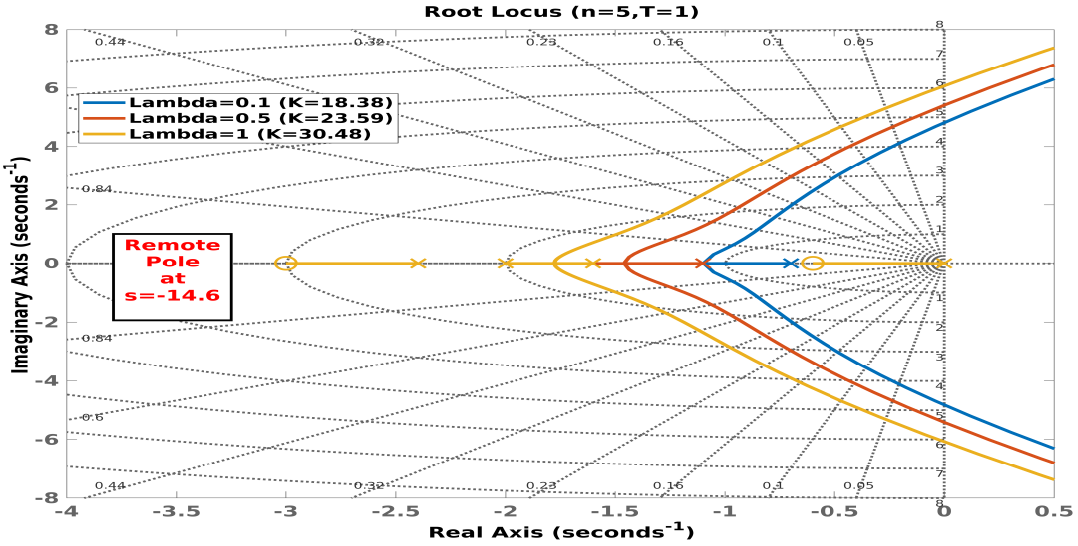

Figure 2.16: Root-Contour plots of the overall system for n=5,T=1 and increasing value of lambda

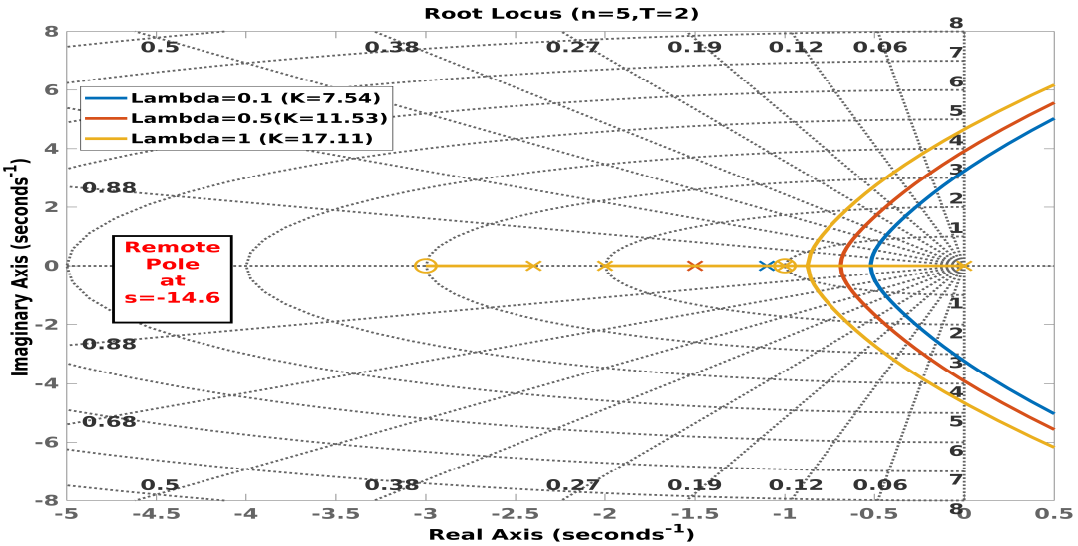

Figure 2.17: Root-Contour plots of the overall system for n=5,T=2 and increasing value of lambda

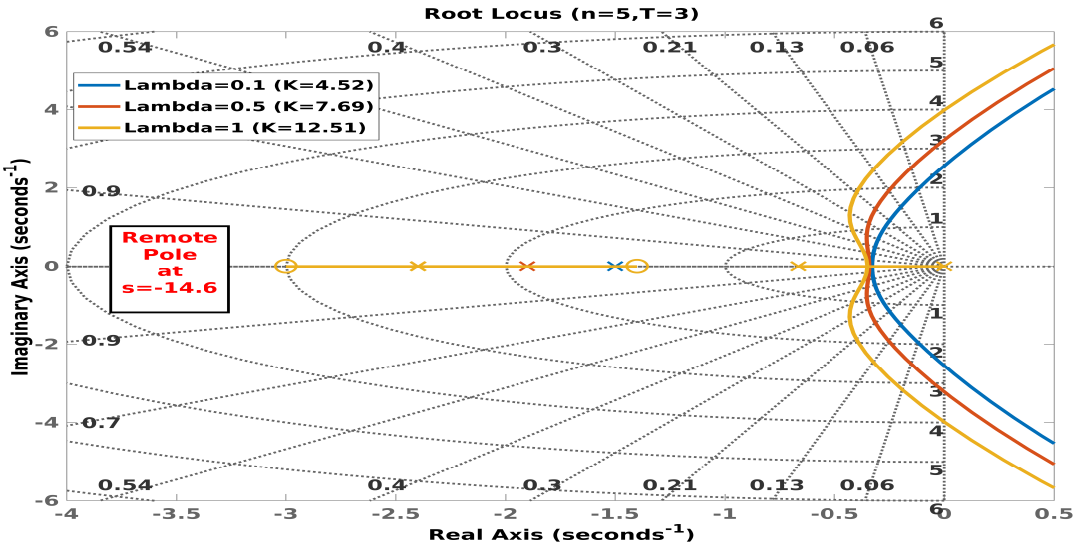

Figure 2.18: Root-Contour plots of the overall system for n=5,T=3 and increasing value of lambda

2.4 Brain Pattern Decoding

This section narrates the steps involved in processing the EEG signals with the ultimate aim to recognize the BCI signals: SSVEP, MI, and P300. The SSVEP signal originates from the visual cortex region of the occipital lobe. The electrodes $O1$ and $O2$ of the international 10–20 electrode placement system are nearest to that brain region used for the purpose of SSVEP decoding. The origin of the MI signals, on the other hand, is the parietal cortex and sensory-motor cortex regions. The electrodes located nearest to this region are $C4$, $C3$, Cz, $P3$, $P4$, and Pz. Thus, these 6 electrodes are employed for MI signal classification. Lastly, the P300 signal appears with a relatively larger amplitude over the midline of the brain, thereby facilitating the use of Fz, Cz, and Pz electrodes for its identification. Thus in this study, a total of 9 electrodes, including Fz, $C3$, $C4$, Cz, $P3$, $P4$, Pz, $O1$, and $O2$, are employed for EEG signal acquisition.

2.4.1 Pre-processing

After the acquisition, the EEG trials are filtered spatially by means of common average referencing (CAR) to remove common-mode noise, including thermal noise, power line interference, undesired physiological signals, etc., which appears uniformly across all the EEG electrodes [33]. Here, the sample-wise average of all the channels is subtracted from the signal samples of each channel at each time instant. Although there exist other sophisticated methods of noise and artifact removal, CAR has been chosen because of its low computational overhead in comparison to other existing filtering algorithms [171].

2.4.2 SSVEP Detection

The following steps are followed to detect the SSVEP brain signal liberated by the subject while gazing at the flickering sources mounted over the links of the robot arm.

2.4.2.1 SSVEP Preprocessing

The spatially filtered EEG signals are passed through a band-pass filter of passband 0.1–30 Hz, realized with a 6th order elliptical filter of 1 dB passband ripple and 60 dB stopband attenuation. The reason behind the choice of the elliptical filter is that it provides sharp roll-off characteristics and good attenuation of ripples in both the pass and the stop bands.

2.4.2.2 Feature Extraction

For the purpose of SSVEP detection, power spectral density (PSD) estimates the three flickering frequencies and two harmonics of each of those frequencies that are used as the EEG signal features. In this paper, the Yule Walker method of auto-regressive (AR) spectral estimation is employed, where the AR model of the input signal is used to determine the PSD [172]. The AR based method is chosen over the conventional periodogram method primarily because of two significant advantages. First, for signal-to-noise ratio (SNR) greater than 0 dB, this method provides better frequency resolution than traditional periodogram methods. Second, this method

is free from distortions due to side-lobe leakage effects which are inherent in the periodogram approach. According to the Yule Walker method, the acquired EEG signal is described by an AR model, where AR model parameters depend only on the previous output samples of the system. Therefore, the acquired n point sequence of EEG signal $y(n)$ can be described as a linear combination of the previous output of the system with the introduction of an error term $\Delta(n)$ where $\Delta(n) = N\{0, \sigma_E^2\}$ represents a Gaussian noise with mean zero and variance σ_E^2.

For previous j samples, $y(n)$ can be represented as

$$y(n) = -\sum_{k=1}^{j} a(k)z^{-k}y(n) + \delta(n) \tag{2.21}$$

where $y(n)$ indicates the nth sample of the input signal and $a(k)$ denotes the AR parameters where $k = [1, j]$, and j denotes the model order of the system. Here AR parameters are estimated using the least mean square method (LMS). Rewriting Eq.(2.21) yields

$$H(f) = \frac{y(n)}{\delta(n)} = \frac{1}{1 + \sum_{k=1}^{j} a(k)z^{-k}} \tag{2.22}$$

For estimating the PSD of the original EEG signal, the Power Spectral Density of the white noise (error term) must be known, which is found to be its variance σ^2. Therefore, the power spectral density estimate reduces to computation of Eq.2.23 given below:

$$P_y(f) = \frac{\sigma^2\delta(n)}{|1 + \sum_{k=1}^{j} a(k)e^{-jk\omega t}|} \tag{2.23}$$

In the present context, three flickering frequencies (7 Hz, 10 Hz, and 12 Hz) are considered. The spectral power of each of the 3 frequencies with their 2 harmonics is considered as a feature vector. For each frequency, a bandwidth of 0.5 Hz below and above the stimulus frequency is taken with a resolution of 0.1 Hz. Each stimulus frequency with its two harmonics generates 11 × 3 = 33 features for a single electrode. The total features generated in a single electrode for all the frequencies are 33 × 3 = 99. The study considers two electrodes for SSVEP detection, hence a total of 99 × 2 = 198 features are generated in a single epoch. We reduce the set of features into a small set using the following feature selection algorithm.

2.4.2.3 Evolutionary Feature Selection

The motivation of feature selection is to identify the smallest possible set of features, which should ideally be independent of each other, but are sufficient to characterize the classes of the given training instances. Such characterization is necessary for recognition of the classes from the selected features/attributes of the training instances. Existing literature on feature selection is primarily aimed at selecting features based on their linear independence[173]. An alternative method, perhaps, is to select features so as to maximize inter-class separating distances and minimize intra- class separating distances. This requires construction of objective functions, ensuring the above requirements, and a time-efficient search algorithm that identifies the smallest

possible set of features that satisfies the objective functions jointly. Let, $f_{i,j,k}$represent the feature i of the data point k lying in class $G(j)$. Also consider the parameters: $f_{i,j,l}$and $f_{i,j',l}$ where the suffixes carry similar meaning as defined for $f_{i,j,k}$. Suppose the training instances include M features for each data point, and $s(\leq M)$ denotes the number of selected features. Let Obj_1 and Obj_2 denote 2 objective functions, representing the respective measures of intra-class and inter- class separating distances between pairs of data points. In the case of Obj_1, the distance is computed between $f_{i,j,k}$ and $f_{i,j,l}$ for data points k and l both lying in class G_j for j in $[1,R]$. In the case of Obj_2, the distance is evaluated between $f_{i,j,k}$ and $f_{i,j',l}$ lying in different classes G_j and $G_j^{'}$, respectively. For the present application, we need to minimize Obj_1 and maximize Obj_2

$$Obj_1 = \sum_{\forall j}\sum_{\forall i}\sum_{\forall k,l\in G_j,k\neq l} \|f_{i,j,k}, f_{i,j,l}\| \tag{2.24}$$

$$Obj_2 = \sum_{\forall i}\sum_{G_j,G_k\in G\,\forall k,\in G_j,\forall l\in G_l} \|f_{i,j,k}, f_{i,j',l}\| \tag{2.25}$$

where

$$G = \bigcup G_r, r \in \{j,l,...\} \tag{2.26}$$

and $\|.\|$ denotes the Euclidean norm. For the sake of simplicity and convenience, the objective functions presented in (2.24) and (2.25) are combined to form a single objective function (2.27), the minimization of which would serve the purpose.

$$Obj_3 = \frac{Obj1}{\delta + Obj_2}, \delta \geq 0 \tag{2.27}$$

Here, δ, a small positive number, is introduced in (2.27) to limit Obj_3 to a finite value, particularly when Obj_2 approaches zero. In the present circumstance, we select δ as a very small positive number (10–6), so that it has no influence on Obj_3, as $Obj_2 >> \delta$ in (2.27), for such a setting of δ.

Although there are several meta-heuristic algorithms to minimize (2.27), we select the well-known particle swarm optimization (PSO) algorithm, primarily for its small code, small convergence time, small run-time complexity, and most importantly the authors' familiarity with it [36] over a decade. The parameters of the PSO algorithm used include $swarmconfidence = 2.0$, $self-confidence = 2$, and $inertialweight = 0.729$ based on the authors' experience [174].

2.4.2.4 Classification

A three-stage hierarchical linear support vector machine (LSVM) classifier has been employed for the purpose of SSVEP classification. Though any other standard pattern classifier could have served the purpose, LSVM is selected for its high classification accuracy and low computational overhead [175]. Moreover, LSVM requires a smaller training time as compared to other classifiers such as naive Bayesian or multi-layered Perceptron. Here SSVEP detection is performed in three distinct stages of binary classifications. In the first stage, the EEG trial is classified to check the presence of SSVEP in the selected time window. On finding the presence of SSVEP, the second stage of classification is performed so as to determine if the trial corresponds to frequency

f_1. In the final stage, the non-f1 trials are further classified into the two classes corresponding to frequencies f_2 and f_3, respectively.

2.4.3 ERD/ERS Detection

2.4.3.1 Pre-processing

In this case, EEG trials filtered spatially by CAR, are again filtered by a BPF of passband 8–24 Hz. The BPF is designed by utilizing a 6th-order elliptical filter of 1 dB passband ripple and 60 dB stopband attenuation.

2.4.3.2 Feature Extraction

For MI detection, we extract common spatial pattern (CSP) features. CSP is an optimized spatial filter, which aims at minimizing intra-class variance and maximizing inter-class variance of the filtered signals[176]. Let $X_i$be a $(p \times q)$ matrix representing band-pass filtered EEG data of class i, $i = 1,2$, where p and q respectively denote the number of channels and time-slices used in a trial for data acquisition. Let C_1 and C_2 be the spatial covariance matrices for classes 1 and 2, respectively, where $C_1 = X_1X_1^T$ and $C_2 = X_2X_2^T$. CSP attempts to determine the optimal spatial filter vector $w = [w_i]$, where w_i is the weight of the $i-th$ channel, such that the ratio of variances of the spatially filtered signals wX_1 and wX_2, given by $J_{CSP}(w)$ is optimized (maximized/minimized), where

$$J_{CSP} = (wX_1).(wX_1)^T : (wX_2).(wX_2)^T \tag{2.28}$$

$$= w(X_1X_1^T)w^T : w(X_2X_2^T)w^T \tag{2.29}$$

$$= wC_1w^T : wC_2w^T \tag{2.30}$$

The optimization of (2.30) is solved by the general eigenvalue decomposition (GEVD) technique. In fact, the principal components corresponding to the largest and the smallest eigenvalues of $A = C_2^{-1}C_1$act as the desired spatial filters corresponding to the maximum and the minimum variances. The singular value decomposition (SVD) technique is employed next to obtain the CSP filter $w = U^T$ by representing the matrix $AbyUDU^T$, where D is a diagonal matrix. The logarithm of the variance of CSP projections, i.e., $log(wCw^T) = log(var(wX))$ for $X = X_1$ and $X_2$are then used as CSP features of 2 classes.

The classical CSP algorithm outlined above works exceptionally well when the acquired EEG signals have large signal to noise ratio. However, because of the non-stationarity of the EEG, the same algorithm may not work well universally across all subjects [40]. Particularly, it suffers from high sensitivity to noise, over-fitting and in-sensitivity to spectral information of the used EEG samples [40]. The sensitivity to noise and over-fitting are eliminated by adding suitable regularizing constraints [38] in the CSP objective function (2.30). To utilize discriminating wave shapes and/or spectral information of RHMI and LHMI, there are 3 alternatives: i) using CSP features along with temporal [177] and spectral features [178] of EEG for classification; ii) undertaking CSP in narrow sub-bands of the useful frequency spectrum for MI, and then se-

lecting the best set of features from the CSP features in b sub-bands using a mutual information based feature section [50] hereafter called filter bank CSP (FBCSP); and iii) considering both magnitude and phase of the EEG samples in the CSP formulation [179] to derive optimal CSP features. Here, we adopt both ii) FBCSP and iii) magnitude-phase CSP (MPCSP) independently, and compare their relative performance with classical CSP in the experiment section. A brief outline of the newer approach [179]is given below.

The classical CSP formulation takes into account of the amplitudes of the EEG time samples, disregarding the phases of the EEG signals. In [179], the authors considered both the amplitude and phase of the EEG signal to obtain more reliable CSP features, responsible for improving the classification accuracy for the 2-class classification problem. In their formulation, the objective function appears similar to (2.30) with C_1 and C_2 replaced by C_1^* and C_2^* respectively defined in the complex plane. Later they adopted the Lagrange multiplier technique to optimize an objective function (2.31) equivalent to (2.30), containing C_1^* and C_2^* in place of C_1 and C_2, respectively.

$$L(\lambda, w) = \bar{w} C_1^* w - \lambda(\bar{w}^T C_2^* w - 1) \tag{2.31}$$

where $\bar{w}$ is the complex conjugate of w . The optimization of L with respect to w returns $C_2^{*-1} C_1^* \bar{w} = \lambda \bar{w}$. As $M = C_2^{*-1} C_1^*$ is a complex matrix, they adopted symmetric singular value decomposition (SSVD) for eigenvalue decomposition. In SSVD, for a singular matrix of $(p \times q)$ we have a unitary matrix $U^{'}$, such that $P = U^{'} C^* U^{'T}$, where $C^* = diag[\sigma_1, \sigma_2, ...\sigma_p]$ with $\sigma_i \geq 0$, where σ_i is the eigenvalue. Consequently, for the square symmetric matrix $A = C_2^{*-1} C_1^*$ we obtain $M = WD\bar{W}^T$ where D is a diagonal matrix and $\bar{W}^T$ is the desired CSP matrix. In classical CSP, the principal component analysis (PCA) is employed to determine the principal components corresponding to the largest and the smallest eigenvalues of $A = C_2^{*-1} C_1^*$.

A non-linear PCA and conformal mapping are required here to determine the largest and the smallest eigenvalues of the complex matrix $A = C_2^{*-1} C_1^*$, the details of which are available in [179]. The CSP features thus obtained, in conjunction with a standard linear discriminant analysis(LDA) classifier improves the classification accuracy to more than 98% at the cost of additional computational overhead.

2.4.3.3 Classification

This study makes use of a 2-stage radial basis function kernelized support vector machine (RBF-SVM) classifier, where the first stage categorizes the feature vector of an EEG trial in the presence/absence of MI in the trial. The trials found to contain MI are classified in the second stage into LHMI and RHMI.

2.4.4 Signal Processing for P300 Detection

2.4.4.1 P300 Preprocessing

The CAR-filtered P300 signal is passed through a BPF of passband 0.1–10 Hz. The filter is implemented with the 6th-order elliptical filter of 1 dB pass-band ripple and 60 dB stop-band

attenuation.

2.4.4.2 P300 Feature Extraction

Here, the adaptive autoregressive parameters (AAR) are utilized as the features of the EEG trials. The AAR model can efficiently represent the stochastic and non-stationary nature of EEG signals owing to the time-varying characteristics of the AAR coefficients [180] A j^{th} order AAR modelis represented by, AAR(j), where the AR parameters are evolved with time using a recursive-least-squares (RLS) algorithm [181] with an update-coefficient set to a small number ($= 0.008$) to facilitate only small changes in consecutive iterations.

For offline sessions, here, P300 trial is captured for the duration of $2s$. Considering a sampling rate of $200Hz$, a total of 400 data samples are collected during a single trial. Here, a 6th order AAR model is considered, hence $6 \times 400 = 2400$ AAR features are obtained. AAR features are averaged with a moving window of length = 60 features (50 ms). So, the dimension of the feature vector for each electrode after window averaging is 40. Considering three electrodes, F_z,C_z, P_z we thus have a total of $40 \times 3 = 120$ features.

2.4.4.3 Feature Selection

The PSO-based feature selection algorithm introduced before is employed now to select the most discriminating features for 2 classes, representing P300 present or absent in a given time-window.

2.4.4.4 Classification

The selected features are submitted to an LSVM classifier to recognize the presence or absence of P300 in the selected window.

2.5 Experiments and Results

This section presents the experimental protocol and the main results obtained while conducting the experiment. Performance of the SSVEP, MI, and P300 classifiers have been presented here in terms of four metrics viz. classification accuracy, true positive rate, false positive rate, and computation time.

2.5.1 The Experimental Framework

Subjects: Ten volunteers were chosen for the experiment. The medical history of the volunteers shows no evidence of any critical illness or any other surgery undergone in the near past. Among 10 subjects, 6 are male and 4 are female and all of them are in the age group of 24–30. A consent form was duly signed by them stating their willingness to participate in the experiment. All other safety and ethical issues were maintained according to the Helsinki Declaration of 1970, revised in 2000 [182].

The EEG System Used: A 19-channel EEG device manufactured by Nihon-Kohden was used to acquire the EEG signal from the subjects. Electrodes were placed according to the standard 10–20 electrode placement system, which uses $A1$ and $A2$ as the reference electrodes and FP_z is the common ground (Fig. 2.19). A built-in notch filter of 50 Hz frequency eliminates the power-line disturbances. The device acquires an EEG signal at a sampling rate of 200 Hz.

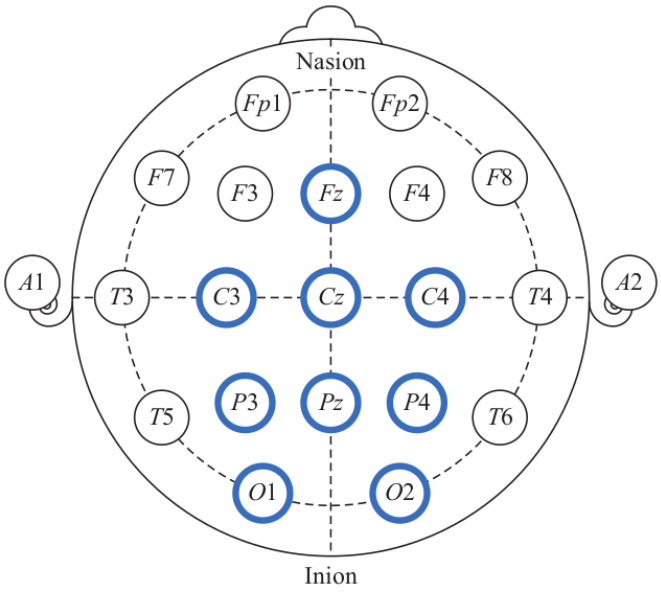

Figure 2.19: Bold (blue) circles representing selected electrode positions in the international 10–20 electrode placement array.

2.5.2 The Training Session

The training was offered in offline mode with the help of a power-point (PPT) stimulator. The first slide includes a fixation cross to make the subject alert to the stimulator. The second slide includes a flickering source mounted on one link of a robot arm to help the subject reproduce the SSVEP at the source frequency. The simulator includes a 3-link robot arm, each with a provision for flickering at different frequencies, with the ultimate aim to select the link based on the frequency contained in the SSVEP produced by the subject. The subject is asked to gaze at the particular flickering source mounted on the robot link. Once the SSVEP is recognized, the corresponding link responsible for the SSVEP is highlighted with a green color. The third slide includes commanding the subject to produce left/right motor imagery for clockwise/ counter-clockwise rotation respectively of the link. The last slide contains a virtual scenario where a particular robot link crosses a fixed target. The subject on observing this is expected to yield a P300 signal. The above sequence of four slides is repeated 60 times across one week's time. One typical instance of the training session, comprising a stimulus presentation followed by robotic actions in response to the stimuli is given in Fig.2.20

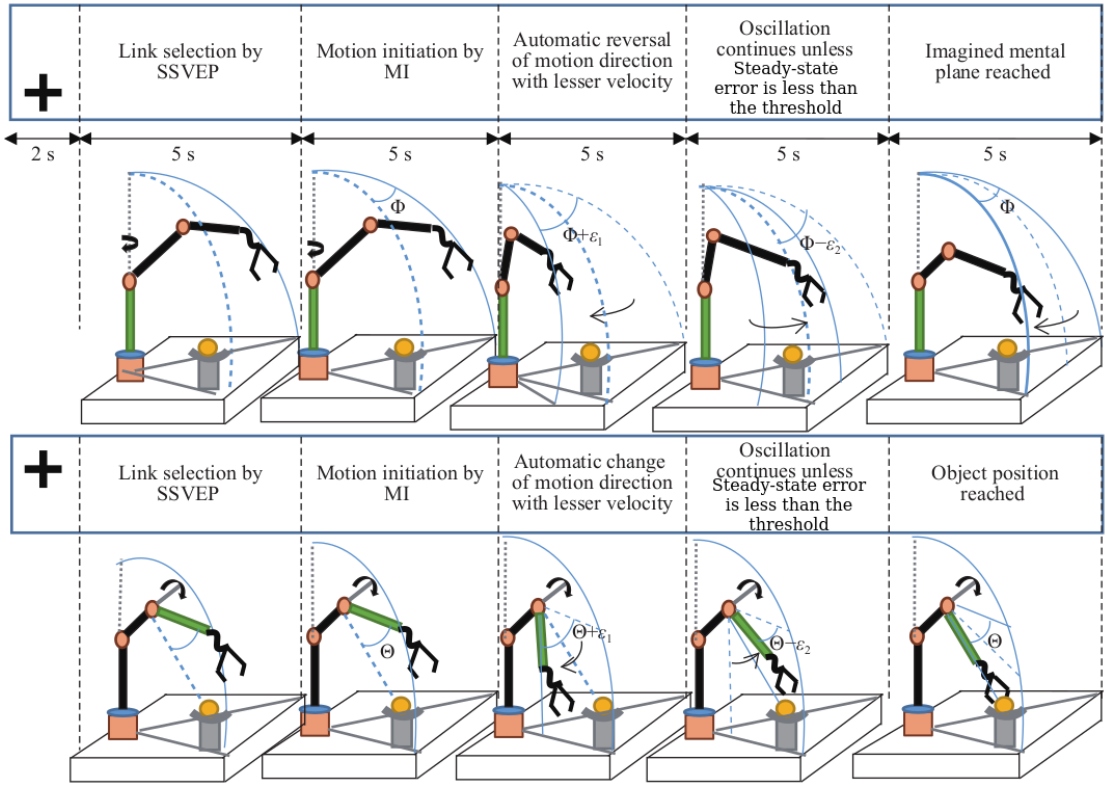

Figure 2.20: Stimulus description (header of the figure) with robotic actions in sequence (left to right of the first row followed by left to right of the second row) for a specific control task

2.5.3 The Testing Session

The basic difference between the training and the test session lies in the phenomenon that the training is imparted in simulation mode, while the test session is performed online with the real robot. Although the difference is small, the test session usually is more complex, as the subject has to plan the three steps of operations: link selection by SSVEP, MI to move the selected robot arm, and P300 generation on observing a positional error (i.e., the link crosses the target position), without having a reference on time-limits/intervals. The link selection protocol and link movement direction protocol are illustrated in Tables 2.2 and 2.3, respectively. During the real-time testing session, each of the three brain signals is observed with a moving time window of 1 s. Hence all the signals are captured throughout the length of the window but an exception is followed in case of MI detection. MI signal is observed through the entire length of the window but only the last 0.2 s of the signal is taken into account[183].

2.5.4 Observed Waveforms/Traces

The acquired P300, SSVEP, and ERD/ERS traces for 5 distinct trials on a subject with their population average trace are given in Figs. 2.21, 2.22, and 2.23, respectively. In each case, the population average is obtained by taking the average of all available instances. It is evident from Fig.2.21 that positive peaks of P300 are generated around 250 ms to 350 ms, counted from the onset of the target stimuli (at the 0th second), whereas Fig.2.22 refers to the SSVEP

response corresponding to a frequency of 7 Hz. It is interesting to note that the band power of SSVEP has shown a significant rise around 7 Hz. In Fig.2.23, the ERD plots show a sharp fall-off in magnitude at approximately 400 ms time-point, and the signal power is restored at approximately around 550 ms. Here ERD is quantified as a percentage change of power at each sample point relative to the average power in the reference interval [184].

Table 2.2: SSVEP-based link selection protocol

Link Number	Color of LED	Frequency of flicker(Hz)
Link 1	Red	7
Link 2	Green	10
Link 3	Amber	12

Table 2.3: MI-based motion activation protocol

Desired link movement	Corresponding MI
Clockwise rotation of any link	RHMI
Counter-clockwise rotation of any link	LHMI

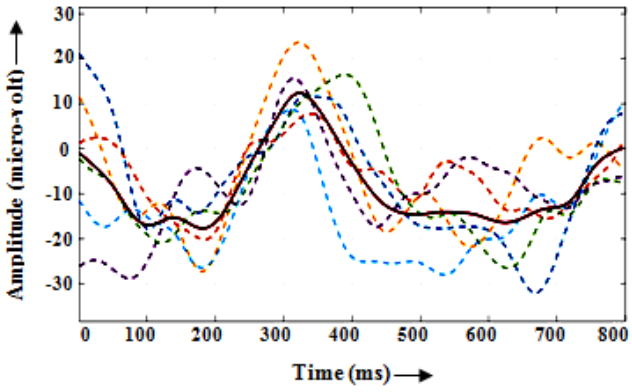

Figure 2.21: P300 waveform for 5 subjects represented by dotted lines and the population average of the signals represented by a red solid line after acquisition from C_z electrode, and filtering by 6th order elliptical filter in band 0.1–10Hz.

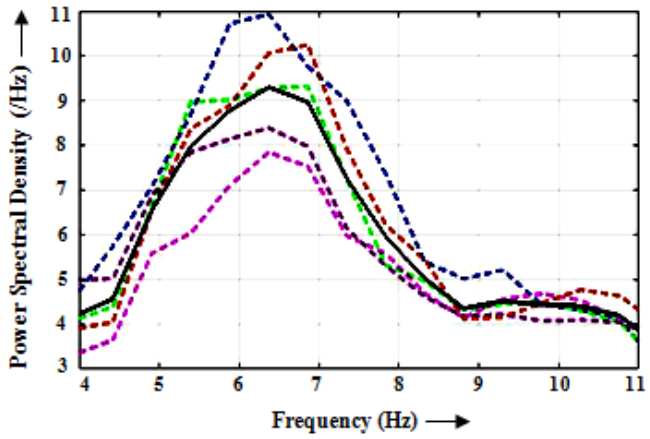

Figure 2.22: PSD plot of SSVEP at 7 Hz frequency for 5 subjects represented by dotted lines and the average of the signals represented by a black solid line after acquisition from channel O_1.

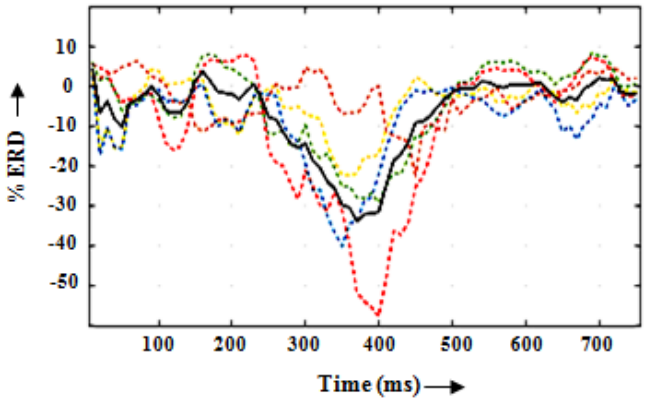

Figure 2.23: Percentage ERD plot of 5 subjects taken at C_3 electrode for right arm motor imagery with the population average represented by a solid black line after filtering in [8–11] Hz

2.5.5 Validation of the Decoders

In this experiment, 3 different methodologies are proposed to decode the three different EEG signals: SSVEP, MI, and P300. For the purpose of evaluating the performances of these decoders, 5 well-known performance metrics have been employed which are classification accuracy (CA), true positive rate (TPR), false positive rate (FPR), computational time (CT), and Cohen's kappa (κ) co-efficient. The formal definitions of these metrics are presented below for ready reference.

- CA: The fraction of the total number of instances that are correctly identified by the decoder.
- TPR: The ratio between the correctly detected positive instances and the total number of positive instances.

Table 2.4: Comparison of Different CSP-Based Classifiers

Algorithm	CA%	Kappa	Training Time(s)
Classical CSP + RBF-SVM	94.1	0.89	0.4232
FBCSP + LDA	97.3	0.86	3.2658
MPCSP+RBF-SVM	98.6	0.92	3.8249

- FPR: The fraction of the total number of negative instances that the decoders identify as positive ones.
- CT: The time taken by the trained decoder in order to generate the inference about the presence/absence of the concerned signal in a single trial EEG.
- Cohen's kappa: The inter-rater reliability parameter for the categorical items and also is a more robust parameter than percent classification accuracy [185].

For each subject, 7 instances are acquired of which 6 instances selected randomly are used for classifier training and the remaining one for classifier testing.

2.5.5.1 Validation of MI Decoder

Table 2.4 reports the performance evaluation results of the CSP based MI decoder which is required to identify the desired direction of link motion. It is apparent from the Table that the MPCSP feature selection method along with the radial basis function (RBF) kernelized SVM (RBF-SVM) classifier outperformed its nearest competitor FBCSP feature selection + linear discriminant analysis (LDA) classifier by a margin of 1.3% in terms of classification accuracy. A higher inter-rater reliability value ($\kappa = 0.92$) is obtained for MPCSP + RBF + SVM at the cost of increased training time. As the training session is performed offline, the relatively larger training time of MPCSP compared to FBCSP does not affect the real-time performance of the MPCSP-RBF-SVM classifier with its nearest competitor FBCSP-LDA.

Now considering Table 2.5, the best CA was attained in the case of the fourth subject which is 99.2%. The average metric values obtained are: CA = 98.1%, TPR = 0.93, FPR = 0.04, CT = 0.422 s, and kappa = 0.89. Here, the inter-subject variance of the results is represented as the coefficient of variation (CV), which is measured as the ratio of observed mean and observed variance. The CV value for classification accuracies is found to be 0.01 whereas the CV value of kappa over the different subjects is found to be 0.03.

2.5.5.2 Validation of the SSVEP Decoder

It is seen from Table 2.5 that the SSVEP decoder offered average CA, TPR, FPR, and CT of 95.9%, 0.92, 0.04, and 0.091 s, respectively. The average kappa value obtained is 0.91. The best CA of 96.8% was obtained for the fourth subject and the best kappa value is found to be 0.94 for the first subject. CV values of classification accuracy and kappa values over the different subjects

Table 2.5: Performance of the Proposed EEG Classifier Algorithms in Multiple Runs for Different Subjects

	Metric	Sub 1	Sub 2	Sub 3	Sub 4	Sub 5	Sub 6	Sub 7	Sub 8	Sub 9	Sub 10	Avg
SSVEP	CA(%)	95.5 (±0.42)	96.2 (±0.48)	95.8 (±0.32)	96.8 (±0.51)	95.4 (±0.35)	95.9 (±0.44)	95.1 (±0.53)	96.1 (±0.30)	96.3 (±0.49)	95.8 (±0.62)	95.9
	TPR	0.94	0.94	0.91	0.93	0.92	0.92	0.90	0.93	0.94	0.92	0.92
	FPR	0.02	0.03	0.05	0.03	0.04	0.03	0.05	0.05	0.04	0.03	0.04
	κ	0.94	0.93	0.90	0.93	0.89	0.92	0.89	0.92	0.91	0.91	0.91
	Time(s)	0.095	0.091	0.096	0.102	0.086	0.092	0.084	0.093	0.083	0.094	0.091
MI	CA(%)	98.3 (±0.38)	98.6 (±0.24)	97.8 (±0.43)	99.2 (±0.32)	97.1 (±0.39)	98.4 (±0.51)	96.2 (±0.28)	98.5 (±0.26)	97.3 (±0.31)	98.7 (±0.47)	98.1
	TPR	0.96	0.94	0.93	0.94	0.93	0.92	0.95	0.93	0.92	0.93	0.93
	FPR	0.03	0.04	0.05	0.03	0.04	0.05	0.07	0.05	0.05	0.04	0.04
	κ	0.95	0.90	0.87	0.92	0.90	0.89	0.87	0.86	0.86	0.88	0.89
	Time(s)	0.462	0.32	0.459	0.421	0.398	0.396	0.402	0.453	0.429	0.411	0.422
P300	CA(%)	94.1 (±0.51)	92.6 (±0.23)	91.8 (±0.28)	94.3 (±0.20)	93.2 (±0.31)	93.7 (±0.38)	90.5 (±0.32)	93.4 (±0.49)	94.9 (±0.36)	92.4 (±0.42)	93.3
	TPR	0.88	0.89	0.93	0.93	0.90	0.89	0.91	0.94	0.92	0.90	0.90
	FPR	0.03	0.04	0.03	0.03	0.05	0.02	0.02	0.04	0.03	0.04	0.03
	κ	0.89	0.87	0.90	0.94	0.91	0.85	0.88	0.91	0.92	0.88	0.89
	Time(s)	0.121	0.110	0.106	0.109	0.098	0.105	0.113	0.102	0.118	0.107	0.108

are found to be 0.005 and 0.018, respectively. It is apparent that such values are significantly on the lower side.

2.5.5.3 Validation of P300 Decoder

The average values of CA, TPR, FPR, kappa, and CT of the P300 decoder obtained are 93.3%, 0.90, 0.03, 0.89, and 0.108 s, respectively with the best CA of 94.9% obtained for the 9th subject. CV values of classification accuracy are noted to be 0.01 and that of kappa is noted to be 0.02.

2.6 Statistical Validation

ERD/ERS trials depicted in Fig.2.23 are statistically validated with the criterion mentioned in (2.1). Sample points of the depicted trials are compared with the sample points of the population mean trial (Ground truth) to check if they conform to Gaussian criteria.

P300 trials represented in Fig.2.21 are statistically validated with the population mean latency obtained from all trials of all subjects participating in the study. The mean latency of P300 is found to be 360 ms. The latency of 6 trials presented in Fig.2.21 is given in Table 2.6. The particular time instant when the highest peak of the P300 occurs is considered as the latency of the signal. Population mean latency is found to be 360 ms. One sample t-test [186] is used to statistically validate the trials with the population mean. The required null hypothesis is expressed as follows:

$$H_0 : \mu = \hat{X} \tag{2.32}$$

where μ is the population mean and $\hat{X}$ is the sample mean. A confidence level of 95% with degrees of freedom $(df = 5)$ is considered for obtaining the p-value. Table 2.7 provides the results of the One-sample t-test on P300 trials. The p value obtained here clearly indicates that the assumed null hypothesis is true. Hence, the represented samples belong to the same population.

Table 2.6: LATENCY OF THE REPRESENTED P300 TRIALS

Trial No	Latency(ms)
Trial 1	364
Trial 2	356
Trial 3	362
Trial 4	356
Trial 5	371
Trial 6	395

SSVEP trials presented in Fig.2.22 are also statistically validated using the One-sample t-test against the population mean amplitude revealed by power spectral density. The amplitude of the highest peak occurring in 6.5–7.5 Hz frequency range is considered for comparison. The population mean amplitude is found to be 9.1 dB. Peak amplitudes of the trials are given in Table

Table 2.7: RESULT OF One-sample t-test ON P300 TRIALS

Sample mean	Population mean	SD sample	t-stat	p-value(95%)	Result
367.33	360	14.66	1.22	0.2752	Not Significant

2.9. The null hypothesis is considered as same as considered in the previous case. The details of the results are given below in Table 2.8.

Table 2.8: RESULT OF t-test APPLIED ON SSVEP TRACES

Sample mean	Population mean	SD sample	t-stat	p-value(95%)	Result
9.28	9.2	1.30	0.13	0.89	Not Significant

Table 2.9: AMPLITUDE OF THE REPRESENTED SSVEP TRIALS

Trial No	Amplitude
Trial 1	7.8
Trial 2	8.2
Trial 3	9.3
Trial 4	10.2
Trial 5	10.9

The p-value (two-tailed, 95% confidence with $df = 4$) obtained in this case is 0.89, which clearly indicates that depicted trials are not statistically different; thus the assumed null hypothesis proves to be true in this case.

The performance of the classifiers is validated using the Friedman statistical test [187]. The Friedman test is a non-parametric statistical test. It ranks the classification algorithms for each dataset based on classification accuracy. The classifier with the highest classification accuracy gets the lowest rank of 1. For the i-th dataset and j-th algorithm, r_{ij} designates the relative rank of the classifier. The total rank of a classifier is evaluated by summing all the ranks it received for all the datasets. R_{ij}, the total rank of the j-th classifier is given in (2.33).

$$R_j = \sum_{i=1}^{N} r_j^i \tag{2.33}$$

where N denotes the number of datasets. For each subject, we consider a dataset. The test considers the null hypothesis, which assumes that the performances of all the classifiers are equivalent, so their rank sum should be equal. Under the null hypothesis, Friedman statistics are distributed as χ with k˘1 degrees of freedom. Here k denotes the number of classifiers used in the study.

The Freidman statistic is calculated by

$$\chi_F^2 = \frac{12}{Nk(k+1)} \sum_{j=1}^{k} (R_j)^2 - 3N(k+1) \tag{2.34}$$

where N = number of databases, and k = no. of competitive classifiers. Now using N = 10, k = 5, and ranks obtained from Table 2.11, Table 2.12, and Table 2.13, the value of χ_F^2 is determined separately for each of the three categories of signal and compared with the critical value of the chi-square obtained with a 95% confidence level and 4 degrees of freedom. The χ_F^2 values obtained along with the critical value are presented in Table 2.10.

Table 2.10: Comparison Table of chi-square Values

Signal Category	χ_F^2 value obtained from test	Value of $\chi_{4,0.95}^2$ obtained from chi-square distribution	Null Hypothesis Accepted/Rejected
SSVEP	35.76	9.48	Rejected
MI	38.56		Rejected
P300	36.86		Rejected

It is evident from the table that in each case value obtained from the Friedman test exceeds the critical value, so the null hypothesis that all the classifiers are equivalent is discarded. Hence the performance of the classifiers is evaluated by their cumulative ranks. The classifier with the smallest rank has the best performance. It is apparent that the RBF-SVM classifier has the lowest cumulative rank-sum in each case and performs best in the study.

Table 2.11: Rank Table of Classifiers used in SSVEP Detection

Sub ID	Friedman Statistical Test for SSVEP									
	LSVM	Rank	QDA	Rank	LDA	Rank	k-NN	Rank	BPNN	Rank
1	95.5	1	92.2	2	91.2	3	89.7	4	86.2	5
2	96.2	1	94.5	2	92.4	3	90.4	4	86.4	5
3	95.8	1	92.5	2	90.7	3	87.6	4	84.8	5
4	96.8	1	93.1	2	90.3	3	88.1	4	85.2	5
5	95.4	1	95.2	2	91.0	4	92.4	3	89.6	5
6	95.9	1	92.8	2	90.3	4	91.8	3	87.1	5
7	93.1	2	96.0	1	91.6	3	88.9	5	90.2	4
8	95.1	1	91.2	2	89.6	4	90.8	3	86.3	5
9	96.3	2	96.9	1	94.0	3	92.2	4	88.8	5
10	95.8	1	93.6	2	89.1	4	91.9	3	86.5	5
Total		12		18		34		37		49

Table 2.12: Rank Table of Classifiers used in MI Detection

Sub ID	Friedman Statistical Test for Motor Imagery									
	RBF-SVM	Rank	LSVM	Rank	QDA	Rank	LDA	Rank	k-NN	Rank
1	98.3	1	96.1	2	95.2	3	93.1	4	92.3	5
2	98.6	1	95.3	2	94.1	3	92.5	4	91.4	5
3	97.8	1	93.7	2	92.8	3	91.0	4	89.8	5
4	99.2	1	95.1	2	94.4	3	92.3	4	90.3	5
5	97.1	1	93.7	2	92.4	3	90.1	4	89.5	5
6	96.2	2	96.8	1	94.3	3	91.1	4	88.4	5
7	98.4	1	93.1	2	91.8	3	91.1	4	86.9	5
8	98.5	1	93.5	2	92.0	3	90.8	4	89.1	5
9	97.3	1	93.5	2	91.5	4	91.6	3	90.2	5
10	98.7	1	92.8	2	91.6	3	90.3	4	89.1	5
Total		11		19		31		39		50

Table 2.13: Rank Table of Classifiers used in P300 Detection

Sub ID	Friedman Statistical Test for P300 Detection									
	LSVM	Rank	QDA	Rank	LDA	Rank	k-NN	Rank	BPNN	Rank
1	94.1	1	92.0	2	91.4	3	88.2	4	86.4	5
2	92.6	1	90.1	2	88.3	4	89.2	3	85.2	5
3	91.8	1	88.6	2	87.2	3	87.1	4	83.3	5
4	94.3	1	93.2	2	91.0	3	89.3	5	89.8	4
5	93.2	1	91.5	2	89.4	4	90.0	3	87.8	5
6	93.7	1	89.9	2	87.2	3	82.4	4	82.2	5
7	90.5	1	90.1	2	88.1	3	86.3	4	83.4	5
8	93.4	1	93.0	2	89.3	3	88.6	4	86.0	5
9	94.9	1	93.7	2	90.6	3	87.2	4	85.1	5
10	92.4	1	90.2	3	91.8	2	86.9	4	82.9	5
Total		10		21		31		39		49

2.7 Controller Performance Analysis

The performance evaluation of the proposed method of robot arm position control is done based on four popularly used metrics, i.e., steady-state error, peak overshoot, settling time, and success rate. The formal definitions [19] of these metrics are given below for ready reference.

1. Steady-State Error (SSE): The difference between the steady state position of the end effector of the robot and the visually fixed position of the subject, in the limit as time goes to infinity.

Table 2.14: Controller Performance for different values of n and T

	n=1				**n=3**				**n=5**			
	SR%	**Mp%**	**SSE%**	**Ts (s)**	**SR%**	**Mp%**	**SSE%**	**Ts (s)**	**SR%**	**Mp%**	**SSE%**	**Ts (s)**
T=1	86.5	4.3	0.32	4.12	90.2	4.1	0.18	6.66	90.4	4.1	0.10	8.10
T=2	89.9	3.1	0.20	5.16	92.1	3.2	0.12	9.89	92.5	3.1	0.09	14.75
T=3	91.5	2.9	0.15	6.9	92.6	3.0	0.12	13.08	92.8	2.8	0.08	20.30

2. Peak Overshoot (MP): The maximum positional shift in response with respect to the (desired) steady-state position of the end-effector. It is expressed as a percentage of the final value. It is the first maximum peak

$$\%M_p = \frac{CP_p - CP_{ss}}{CP_{ss}} \times 100 \tag{2.35}$$

where CP_p is the final response, and CP_{ss} is the steady-state response of the system for step input.

3. Settling Time (Ts): It is the time taken by the response to reach and stay within 2% of the steady-state value or the desired value.

4. Success Rate (SR): It is defined as the ratio between the number of successful attempts by the subject to the total number of attempts. A trial is considered successful when the subject is able to reach the target within the positional tolerance of 2%.

The performance of the system is given in Table 2.14 for different choices of n and T by the user, while the parameter λ was taken based on the criteria mentioned in section 2.3.3, i.e. to maximize the ratio between the DC gain of the system and Settling time. It is apparent that the combination $[n = 2, T = 2]$ achieves the optimal performance with respect to all the parameters. It achieves a success rate of 92.1% with an overshoot of 3.2% and a moderate settling time of 9.89s. However, some other combinations achieved slightly higher accuracy and lower overshoot (ex. combination with $[n = 5, T = 3]$, $[n = 3, T = 2]$) but at the cost of a significant increase in settling time. The steady error was also found to be 0.12 with this combination, which can be considered to be significantly small for the BCI-based position control application. Few combinations providing smaller steady-state error again yield higher settling time. On the other hand, combinations providing smaller settling times yield lower success rates and higher steady-state error such as combination $[n = 1, T = 1]$. Hence, here we consider the performance obtained from the combination $[n = 3, T = 2]$ as our benchmark to compare the performance of our proposed system with other state-of-the-art methods. A qualitative comparison between the proposed method and the other three control strategies employing only MI, MI+ErrP is provided in Table 2.15.

As observed from Table 2.15 the proposed scheme performs better than the scheme employing the motor imagery detectors. The results show how the inclusion of the P300-based error detector has led to a drastic improvement in the results. The steady-state error has improved

Table 2.15: RELATIVE PERFORMANCE ANALYSIS

Performance Metrics	Only MI	MI+ErrP	Proposed Scheme
SSE(%)	7.73	2.1	0.12
Mp(%)	5.4	4.9	3.2
Ts(s)	35	31	9.89
SR(%)	48.3	85.6	92.1

by 7.61% with the error being closer to zero for the proposed scheme. In addition, the settling time has also considerably been reduced from 31 s to 9.89s as the initial speed for the proposed scheme is relatively higher, and so the first zero-crossing occurs much earlier. In the MI + ErrP scheme, the initial speed of the links has a moderate value in avoiding peak overshoot. The success rate of the proposed scheme also is improved with respect to that of MI + ErrP.

2.8 Comparison With Existing Literature

There exist quite a few papers involving BCI where EEG-based robot manipulation has been used successfully. Most of the cited references use an open-loop control strategy to control the position of the end-effecter. Those were achieved using different BCI signals like SSVEP [187], MI [152], [188]. All of them use a controller which essentially works in on/off control mode, and no feedback path is introduced between the robot and the human. Very recent works have used the P300 brain pattern for mobile robot navigation [147] and for the movement of a rehabilitative external agent [149]. All of the works are exposed to high positional error and a large value of peak overshoot because of the absence of any feedback mechanism from the robot to the human subject. The above approaches also need a rigorous amount of subject training to achieve a satisfactory calibration. This paper proposes a novel idea of minimizing bi-directional error and peak overshoot of the controller by incorporating a closed-loop control strategy. It also reduces the scope of subject-dependency by using Event-Related Potential P300 for stopping further movement of the arm. For a particular link operation, the subject has to perform the MI task once, and position alignment with the target will be achieved automatically by means of P300, for which the subject only has to concentrate on the object he has visually fixed. Such use of a closed-loop control strategy for BCI-based robot control is novel in the literature. The flexible link selection scheme also provides some definite advantages over the end-effecter-based control scheme of the manipulator. First, it is observed that linear movement of the end-effecter (translation along three coordinate axes) leads to the alignment singularity [192] condition when operated in a large range. It is not possible for the subject to resolve the singularity condition without any external intervention. Second, due to the lack of flexibility in controlling each degree of freedom, an overall movement of six joints becomes necessary even when the task can be achieved by turning only a single link. It is evident from relative performance analysis that the steady-state error in the case of the proposed approach is reduced drastically.

The comparison of the work with Hybrid BCI systems designed for position control by

Table 2.16: Comparison With Other Hybrid BCI Based Position Control

Study by	Nature of work	Hybridization principles used	Techniques used	Performance
Gao et al., 2017 [189]	EEG based open loop robot position control for writing words	ERD based motor switching in the first phase, testing of teeth clenching in the second phase by EMG, and finally SSVEP based movement direction control in the third phase.	MI: BPF, Mu rhythm extraction and thresholding; Teeth clenching: DWT, energy value, and thresholding. SSVEP: DWT, canonical correlation analysis, and thresholding.	Mean decoding accuracy of writing task = 93%.
Zeng et al., 2017 [190]	Gaze-sensitive BCI for position control of a robot arm	MI based motor switching and eye tracking	CSP features + LDA classifier	Classification accuracy = 85% ± 6.3%; Maximum positional error ±50.
Bhattacharyya et al., 2017 [19]	Closed-loop position control of a robot arm with a fixed order of link activation, irrespective of target position	ERD/ERS based motor activation and ErrP based stopping followed by turning by a fixed offset angle	MI: BPF, wavelet coefficients, correlation based feature selection, linear SVM based classification; ErrP: CAR-filtering, AAR parameter extraction, linear SVM based classification	Steady-state error =6.67%; Settling time = 31 s; Success rate = 85.62%.
Chen et al., 2020 [191]	Hybrid BCI based quadcopter robot control	MI and EOG (electro-occulogram) based hybridization for quad-copter navigationcontrol	MI: BPF, CSP feature extraction and SVM classification of LHMI and RHMI. EOG: Pre-processing, CSP filtering and hierarchical multi-class SVM classification. Combining EOG and EEG classes intelligently for motion control of the quadcopter in 3D space	Classification results for each control instruction > 96%, ITR > 45 bits/minute for all control actions.
Proposed study	Closed loop position control of a robot arm with provisions for link selection at random and near zero steady-state error	SSVEP-based link selection, ERD/ERS based motor activation, and P300 based error detection and reversing motion at zero crossings with gradually diminishing speed for any link till steady-state error goes below a threshold	MI: MP-CSP features + RBF kernelized SVM classifier P300:AAR features + Evolutionary feature selection + LSVM classifier SSVEP: Spectral power features + Evolutionary feature selection + LSVM classifier	Steady-state error =0.12%; Settling time = 9.89 s; Success rate = 92.1%.

robots is presented in Table 2.16. It is clear from the Table that the proposed study and the study by Bhattacharyya *et al.* [19] only address the problem of a closed-loop control strategy. The proposed study has improved steady-state error, settling time, and success rate, and so outperforms [19] significantly.

2.9 Conclusions and Future Directions

This paper introduced a new approach for position control of a robot's end-effecter by judiciously controlling the positions of the individual links of the arm. The choice of individual link selection and their position control is left to the user. The individual link selection is performed by the user by noticing the flickering LED mounted on the link. In fact, each link has one LED mounted over it to flicker at a fixed frequency. If the subject releases a P300 from one of the links, it's inferred that the subject prefers to use the link in the next time slot for position control.

Apart from BCI-based link selection, the other important attributes of our work lie in ERD/ERS based motor planning of the previously selected link, and a P300-induced automatic stopping and speed-reversal, where at, the target position is reached by an individual link. A thorough modeling and analysis of the controller.

performance undertaken in the study reveals that the proposed BCI based control is stable with low steady-state error (0.12%), low peak-overshoot (3.2%), and relatively lower settling time (9.89 s).

An analysis of the root locus of the overall system reveals that the stability margin of the proposed system is contingent on the initial choice of the maximum speed of the robotic links. The classifiers chosen have high classification accuracy and their ability to work in the presence of noise proves their elegance in the present study.

Above all, the proposed system outperforms all existing and reported works utilizing BCI-based position control with respect to both classifier and controller performance, thus justifying its utility in rehabilitative aids for people with neuro-motor disability. Future works may involve i) designing alternative control strategies to reduce subjective cognitive load; ii) improving classifier design, particularly MI classifiers following[183], [193]; and iii) removing ocular artifacts following [194] to develop robust, noise-insensitive BCI based control systems.

Chapter 3

Unertainty Handling in Brain-Actuated Position Control using Fuzzy Logic

This section addresses a novel approach to position control of a robot arm by utilizing 3 important brain signals, acquired with the help of an EEG interface. First, motor imagery signal is employed to activate the motion of a robotic link. Second, the Error Related Potential signal is acquired from the brain to stop the motion of the robotic link, when it crosses a predefined target position. Third, the approximate magnitude of the positional error is determined by steady-state Visual evoked potential signal, acquired by noting the nearest flickering lamp that the robotic link has just crossed. The novelty of the present research is to decode the approximate magnitude of the positional error. Once the approximate magnitude and sign of the positional errors are obtained from the mental assessment of the experimental subject, the above two parameters are fed to a fuzzy position controller to generate necessary control commands to control the position of the end-effecter of the robotic link around the pre-defined target position. Experiments undertaken confirm a low percentage of overshoot and small settling time of the proposed controller in comparison to those published in the current literature.

3.1 Introduction

Brain-computer Interface (BCI) is currently gaining increasing potential for its wide-spread applications in rehabilitative robotics. People with neuro-motor disability such as Amyotrophic Lateral Sclerosis (ALS), partial paralysis and the like require assistive support to perform their regular day-to-day-works, such as delivery of food[195], medicines[196] etc. by an artificial robotic device, where the patients themselves can control the movements of the robot arm, their pick-up and placements etc. by mind-generated control commands. Neuro-prosthesis is one of the most active area of BCI research for its inherent advantage to rehabilitate people with degenerative neuro-motor diseases. Early research on neuro-prosthetics began with the pioneering contribution of Pfurtscheller[197][10], who experimentally could first demonstrate the scope of one fundamental brain signal, called Motor Imagery, technically titled as Event Related De-synchronization followed by Event Related Synchronization (ERD/ERS). This signal appears in the motor cortex region of a person, when he/she thinks of moving his/her arms/legs or any voluntarily movable organs. Several researchers have utilized this signal for mind-driven motion-setting to a mobile robot[198], local navigating device[199],[200] artificial robotic arm[201][202][203], and many others. However, using ERD/ERS signal alone can switch on or switch off a device, and thus can only be used for open-loop applications.

In order to utilize the ERD/ERS in closed-loop position control applications, we need additional brain signals. Several research groups [204][205][19], have taken initiatives to utilize the benefits of Error-Related Potential (ErrP) and/or P300 signals to develop a generic platform for closed- loop position control applications. It is important to mention here that the ErrP signal is liberated from the z-electrodes, located at the midline of our scalp, when a subject himself commits any motion-related error and/or finds a second person or a machine to commit similar errors. The ERD/ERS and ErrP signals have been employed in a number of robot position control systems to set in motion of the robotic motor on emergence of the MI (ERD/ERS signal) and switch off the motor of the robot arm, when the robotic link crosses a fixed target position. However, the primary limitation of such position control schemes is on-off control strategy, which according to classical control theory results in large steady-state error [206].

To overcome this problem, several extensions to the basic control strategies have been proposed in the recent past[207][19]. In [207], the authors developed a new strategy to reduce large steady-state error by commanding the robot to turn in reverse direction at a relatively lower speed than its current speed and also sensing the second, third P300, when the target is crossed several times by the end-effecter. Such scheme can result in reduced steady-state error but at the cost of extra settling time.

The present research can reduce both steady-state error and settling time as happens to be in case of classical control strategy by assessing the sign and magnitude of positional error from the subject's brain. However, as the magnitude of error is approximate, a fuzzy controller is a more appropriate option in contrast to a traditional controller. A set of fuzzy rules are proposed to infer the position of the end-effecter from the approximate magnitude and sign of positional errors. Traditional Mamdani-type fuzzy reasoning is employed to yield the fuzzified end-effecter positions. In case a number of fuzzy rules fire synchronously, the union of the inferences are

considered. Finally, a defuzzifier is used to get back the controlled position of the end-effecter. The proposed approach thus is unique and remained unknown to the BCI research community.

The chapter is divided into 5 sections. In Section 3.2, we provide the principles adopted for position control using magnitude and sign of error, captured from the acquired ErrP and SSVEP signals. Section 3.3 deals with a discussion on processing of the acquired brain signals to make them free from noise and extraction of certain features from the pre-processed signals for classification. Section 3.4 deals with fuzzy controller design. Section 3.5 covers the experimental issues, and also narrates the main results justifying the claims. A list of conclusions is included in Section 3.6.

3.2 Principles Adopted in the Proposed Position Control Scheme

This section provides the principles of position control using 3 brain signals: i) Motor imagery to actuate the motion of a robotic link, ii) stopping the robotic link by sensing the ErrP signal, iii) assessing the magnitude of positional error from the flickering Light Emitting Diode (LED) closest to the stopping position. It is indeed important to mention here that assessment of the magnitude of error by SSVEP introduced here is novel and primary contribution of the present research. The sign and the magnitude of positional error together helps in generating the accurate control action for the position control application. The principles of the BCI-based position control scheme is given in Fig.3.1. It is noteworthy from Fig.3.1 that the controller receives both sign and magnitude of error to generate the control signal. However, the exact measure of the magnitude of error cannot be performed easily for practical limitation in placement of SSVEP sources continuously along the trajectory of motion of the robotic end-effecter. To overcome the present problem, an approximate assessment of the positional error is evaluated in 5 scales: NEAR ZERO, SMALL POSITIVE, LARGE POSITIVE, SMALL NEGATIVE and LARGE NEGATIVE using fuzzy membership functions[208]. The control signal about position of the end-effecter is also fuzzified in the same 5 scales. Such assessment helps in generating fuzzy inferences about the degree of memberships of control signals in multiple fuzzy sets. The schematic diagram of the overall position control scheme is presented in Fig.3.2. It is indeed important to mention here that a fuzzy system usually is much robust in comparison to traditional rule based expert systems as it takes care of aggregation of the inferences obtained from firing of multiple rules simultaneously by taking fuzzy union of the generated inferences. The defuzzification of the overall inference returns the signal back in the real domain. There exist several defuzzification procedures. Here, the center of gravity(CoG) defuzzification is used for its simplicity and wide popularity in fuzzy research community [209].

3.3 Signal Processing and Classification of Brain Signals

This section provides an overview of the basic signal processing, feature extraction and classifier design aspects for the proposed application.

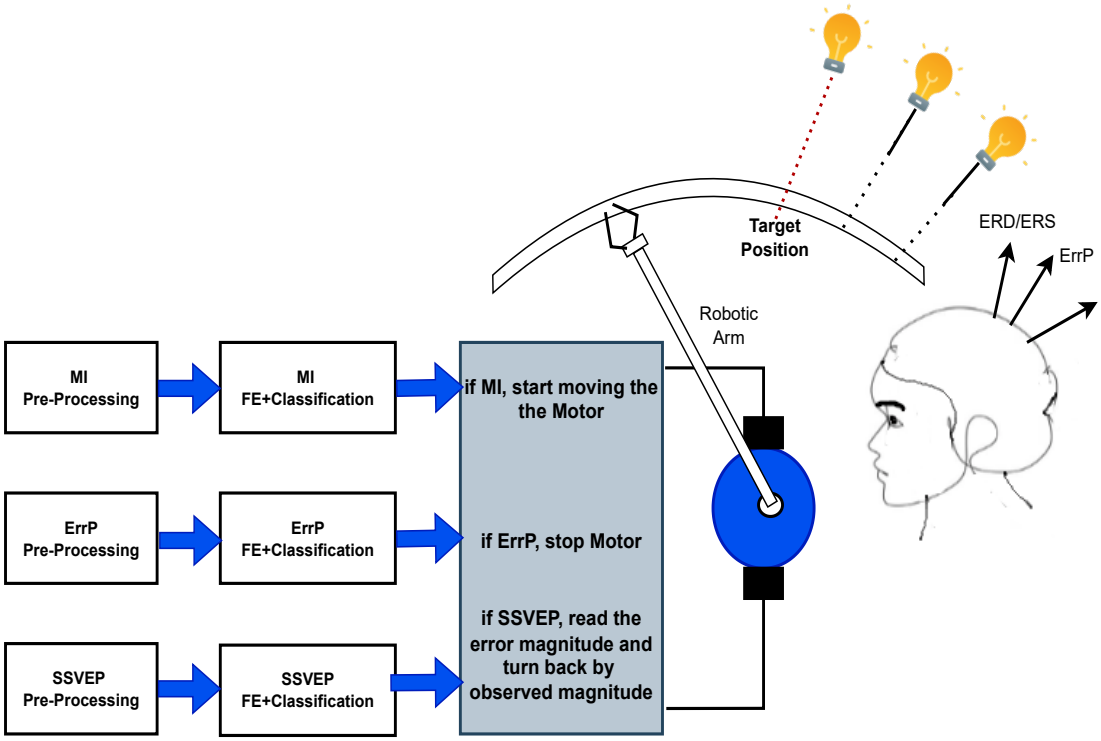

Figure 3.1: Overview of BCI based position control scheme

3.3.1 ERD-ERS Feature Extraction and Classification

For ERD-ERS feature extraction, we need to take as many as 500 offline instances of motor imagery (MI) signals acquired from the motor cortex regions of the subject. These 500 instances of MI signals are examined manually to identify around 300 true positive (v-shaped) and around 200 false negative (non-V or V with inadequate depth) instances. Both the true positive and false negative instances are then sampled at a fixed interval of time, and the mean and variance of the signals at each sampled point is evaluated. Let, at a given sample point s_i we obtain 300 values from 300 true positive curves. Now a Gaussian model is constructed for each sample point s_i, with mean=m_i and standard deviation σ_i. The sample values that lie within $m_i \pm 3\sigma_i$

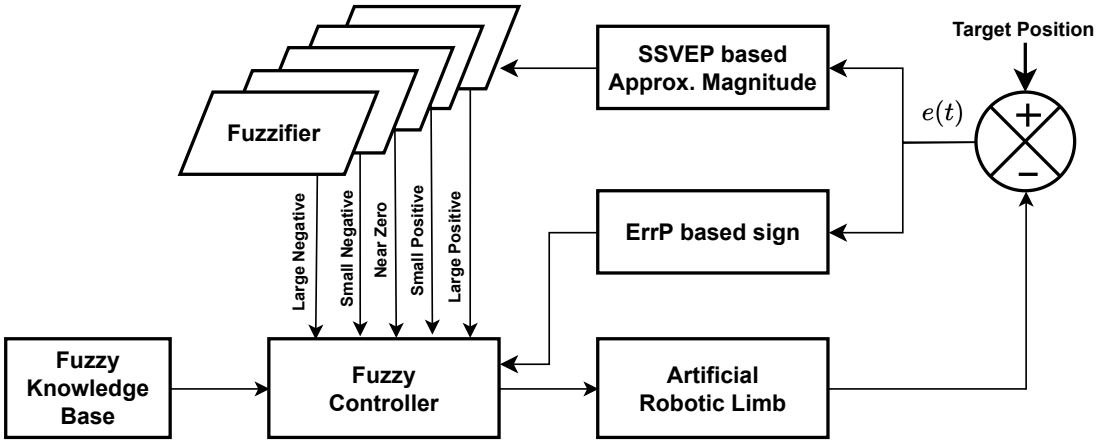

Figure 3.2: Schematic diagram of proposed position control system

are used and the rest are discarded. Thus for each time position in the training samples, we accommodate selected values of the existing trials. Similarly, we undertake selective sample values from a pool of 200 EEG false negative instances. These true positive and false negative instances of the ERD/ERS signals are used subsequently to train a classifier. In this paper, Common Spatial Pattern (CSP), which is widely used in BCI literature as an optimized spatial filter [210], is employed to evaluate the data co-variance matrices for the 2 classes to effectively project the training samples into CSP features. These CSP features are then transferred to a 2 level classifier to recognize the positive and negative motor imagery (MI) signals.

For classification of MI and resting conditions (also called NO motor imageries), the following steps are followed. Let X_1 and X_2 be m x n matrices, where m and n respectively denote number of EEG channels and number of time samples. Let $C1$ and $C2$ be the spatial covariance matrices given by $C1 = X_1X_1^T$ and $C2 = X_2X_2^T$ for positive (MI) and false negative classes. The motivation of CSP is to obtain filter vector $\mathbf{w}$, such that the scalar $\mathbf{wC_1w^T/wC_2w^T}$ is maximized. Once optimal value of vector w is evaluated, the variance of CSP projections: $\mathbf{wX_1}$ and $\mathbf{wX_2}$ are utilized as CSP features of 2 classes. Any traditional linear classifier, such as Linear Discriminant Analysis (LDA) or Linear Support Vector Machine (LSVM), and the like can be used for classification of the MI signals from the resting states. Here, the authors employed Kernelised Support Vector Machine(KSVM) with Radial Basis function (RBF) kernel for its proven accuracy in high dimensional non-linear classification[211].

3.3.2 ErrP Feature Extraction and Classification

Previous research on ErrP feature extraction reveals that the characteristics of ErrP signal can be better captured by time-domain parameters, such as Adaptive Autoregressive (AAR) coefficients [19]. This inspired the authors to utilize AAR features for the detection of ErrP. In the present research, AAR parameters are extracted from approximately 500 ErrP instances and 500 resting states in offline training phase. A q order AAR expresses each EEG sample as a linear combination of past q samples along with an error term characterised by zero mean Gaussian process. AAR co-efficients are estimated using Least Mean Square(LMS) algorithm with an update parameter of 0.0006. For an EEG signal of 1s duration (200 samples), an 6^{th} order AAR generates $6 \times 200 = 1200$ AAR parameters which are used as the feature vector of the EEG trial.An LSVM classifier is then developed to determine the unique set of weights of the classifier to classify the ErrP and non-ErrP instances in real-time.

3.3.3 SSVEP Detection

For detection of SSVEP, the occurrence of the peak power at the flickering frequency of the stimulus is checked. To test this, the maximum power in the PSD is searched over the frequency spectrum. If there is a single peak power occurring at the flickering frequency, then SSVEP is confirmed.In this study we estimated the spectral power density through Welch's modified periodogram method[212]. Power spectral density is obtained for each stimulus frequency and their first two harmonics. We considered an interval of 1Hz below and above the stimulus frequency to obtain the PSD. Once the PSD values associated to each SSVEP stimulus frequency

are obtained, we search for the frequency that has highest PSD value. The frequency having the highest frequency value is considered as the target stimulus.

3.4 Fuzzy Controller Design

The novelty of the current paper is to determine the controller response from the approximate measure of magnitude of error. Here, the occurrence of the error signal is determined from the occurrence of ErrP signal. Now, to measure the magnitude of the error signal, a set of flickering light sources are placed at regular intervals. All these sources flicker at disjoint frequencies. When the subject observes the robotic arm crossing the target position, he is supposed to yield an ErrP signal from the z-electrodes. Almost simultaneously, he is supposed to release an SSVEP signal. Generally, people suffering from neuro-motor diseases have relatively poor reflex, and so they take longer time to respond to flickering visual signals. In order to alleviate this problem, light sources flickering at different frequencies are placed around their trajectory of the end-effecter. Here, the subject has to pay attention to the nearest flickering source, close enough to the terminal position of the end-effecter. Here, the flickering signal of the sources have frequencies in the ascending order of their distances from the predefined target position. This makes sense in the way that larger is the distance of the flickering source from the target position, the larger is the frequency of the source. A set of fuzzy quantifiers is employed to quantify the measure of the positional error in 5 grades: NEAR ZERO(NZ), SMALL POSITIVE(SP), LARGE POSITIVE(LP), SMALL NEGATIVE(SN) and LARGE NEGATIVE(LN). A knowledge- base comprising a set of rules that map the fuzzified errors into fuzzy control signals is then utilized to derive the control signals for each fired rule. The union of the fuzzy control signals is taken, and the result is de-fuzzified to get back the actual value of the control signal.

3.4.1 Fuzzy Reasoning in the Control Problem

Consider the fuzzy production rules:

- ***Rule 1:*** If x is A_1 then y is B_1
- ***Rule 2:*** If x is A_2 then y is B_2

  $\vdots$
- ***Rule n:*** If x is A_n then y is B_n

Here x,y are linguistic variables in the universes X and Y respectively. $A_1, A_2, ..., A_n$ are fuzzy sets under the universe X and $B_1, B_2, ..., B_n$ are fuzzy sets under the universe Y. Let $x = x'$ be a measurement. We compute the fuzzy inference for the given measurement $x = x'$ by the following steps.

Step 1: Compute: $\alpha_1 = Min(\mu_{A_1}(x'), \mu_{B_1}(y))$, $\alpha_2 = Min(\mu_{A_2}(x'), \mu_{B_2}(y))$,..., $\alpha_n = Min(\mu_{A_n}(x'), \mu_{B_n}(y))$.

Step 2: Evaluate the overall fuzzy inference $\mu_{B'}(y) = Max(\alpha_1, \alpha_2, ..., \alpha_n)$.

After the fuzzy inference $\mu_{B'}(y)$ is evaluated, we compute the centroid of it by 'centre of gravity' method[213].

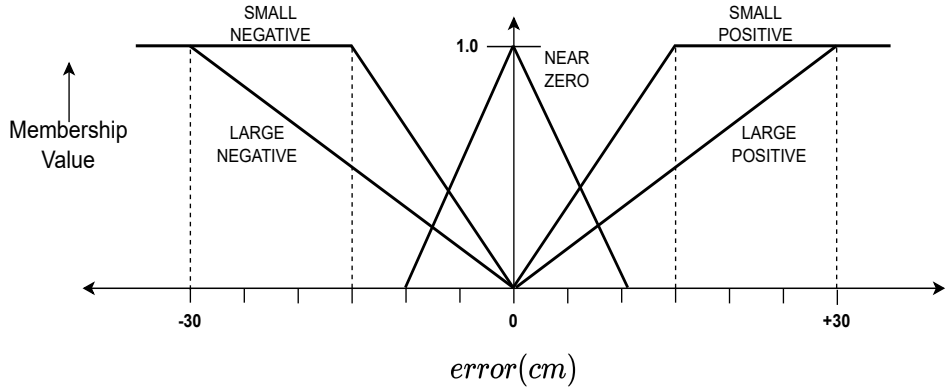

Figure 3.3: Membership function: error

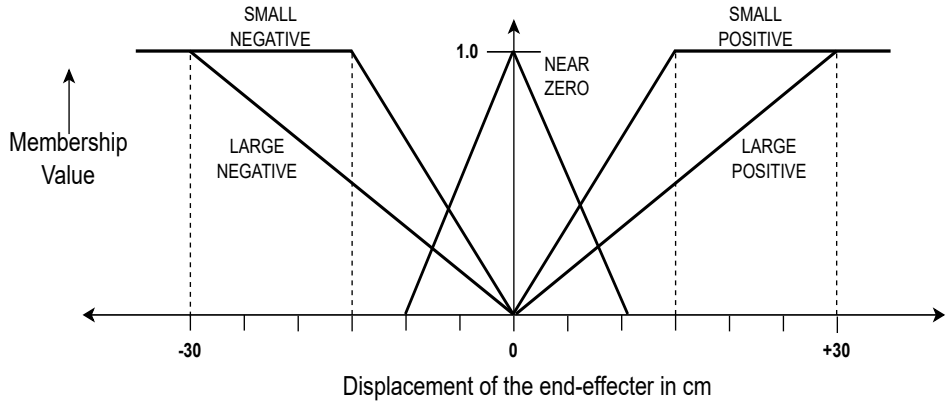

Figure 3.4: Membership function: angular displacement

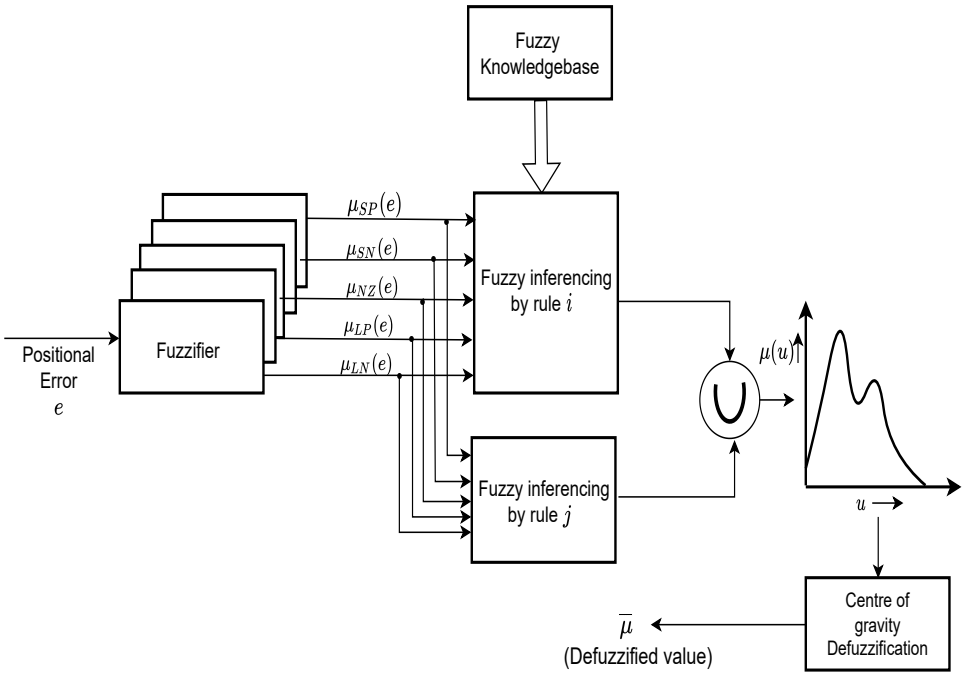

Figure 3.5: Architecture of the proposed fuzzy controller

In the present control problem, x is error and y is displacement of the end-effecter. The fuzzy rules constructed for the position control system are triggered appropriately depending on magnitude and sign of error signal and the selected rules on firing generate inferences, the union of which is the resulting control signal, representing displacement of the end-effecter. The fuzzy membership functions involving error is SMALL POSITIVE etc. and angular displacement is SMALL POSITIVE etc. are given in Fig.3.3,Fig.3.4 and architecture of the proposed fuzzy controller is given in Fig.3.5. The list of fuzzy rules used for the generation of control signals is given below.

- ***Rule 1:*** If error is SMALL POSITIVE then angular displacement is SMALL POSITIVE.
- ***Rule 2:*** If error is SMALL NEGATIVE then angular displacement is SMALL NEGATIVE.
- ***Rule 3:*** If error is NEAR ZERO then angular displacement is NEAR ZERO.
- ***Rule 4:***If error is LARGE NEGATIVE then angular displacement is LARGE NEGATIVE.
- ***Rule 5:***If error is LARGE POSITIVE then angular displacement is LARGE POSITIVE.

3.5 Experiments and Results

This section first describes the experimental protocol in a detailed way and represents the major outcomes of the experiment in subsequent stages. Key details of the experiment are highlighted below,

3.5.1 Subjects

Twelve people within a age group of 18-40 years (mean age 32) voluntarily participated in the study . None of them had any prior experience with BCI training. Out of the twelve volunteers , 6 were male, 6 were female. The objective and procedure of experiment were made clear to the volunteers before conducting the experiment and a consent form stating their willingness to participate in the study was duly signed by them. The experiment was conducted in adherence to the Helsinki Declaration 1970 later revised in 2000[182].

3.5.2 EEG system

EEG data were acquired from the volunteers using a 19 channel EEG amplifier device made by the company Nihon-Kohden. The EEG system has sampling rate of 200Hz and comes with built in notch filter at 50Hz frequency. EEG electrodes were placed over the scalp by following the international 10-20 electrode placement convention[214]. Out of the total 19 electrodes, we used six electrode positions (C_3, C_4, C_z over the motor cortex and P_3, P_4, Pz over the parietal lobe) to acquire the Motor Imagery brain signals. For the SSVEP and ErrP brain signals, we used $\{O_1, O2\}$ and $\{Fz, P_z\}$ electrode positions respectively.

3.5.3 Training Session

We conducted the training session throughout the 15 days with a repetitions of 3 sessions in a day for each subject.Inter session gap of 10 minutes were provided. Each session consists of 50 trials, resulting 150 trials for a subject in a day.Each trial contains the visual instruction to be followed by the participating subjects.

Visual instructions are presented before the subject through a robotic simulator. The robotic simulator virtually represents a robotic limb capable of producing clock/anti-clockwise movement around a specially designed fixed frame. The frame has markings of various positions over it along with the target position and LEDs are mounted near the frame against each positional markings. The LEDs flicker with a constant frequency but are different from each other.

A trial starts with a fixation cross that appears as a visual cue and asks the subject to remain alert for the upcoming visual cues. It stays on the screen for 2s duration. The next visual cue contains an instruction to perform either LEFT or RIGHT arm motor imagery for clockwise/anti clockwise movement of the robotic limb. The next visual cue contains a scenario where the moving link commits an error by crossing the target location, hence the subject develops ErrP brain pattern by observing the error. The next scenario illustrates a condition where the end-effecter of the moving link crossed the target position. Now, Subjects are instructed to focus their gaze on the flickering LED nearest to the present position of robot end-effecter, focusing on the flickering source generates a SSVEP signal modulated by the source frequency in the subjects' brain. Timing diagram of stimulus presentation is depicted in Fig. 3.6.

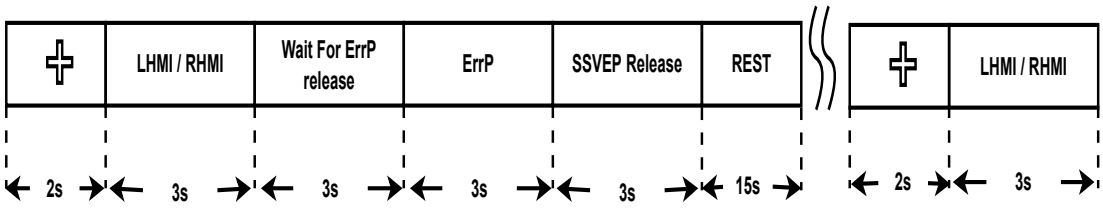

Figure 3.6: Stimuli diagram of training session

3.5.4 Testing Session

The major difference between training session and testing session lies in the medium of operation. In contrast to the training session, which is conducted offline using a robotic simulator, the testing session is performed in real time with the physical robot. This session is more complex than training session as the subject participating in this session does not receive any visual instruction to perform the required mental task. Hence, the subjects need to plan the three steps of action (viz. link movement, target selection and gazing on the nearest flickering source) themselves without any visual guidance. A timing diagram presented in Fig. 3.7 shows the time taken by each module during real time operation. During the real time operation, we used a window of 1s duration to acquire the MI signal and SSVEP signal, whereas ErrP was acquired through the windows of 250ms.

Table 3.1: Comparative Study of Different ERP Detection Methods

Brain Pattern Detection	Feature Extraction + Classifier	Performance Metrics			
		CA (%)	TPR	FPR	kappa
MI Detection	CSP+KSVM-RBF	91.31	0.89	0.04	0.84
	CSP+LSVM	90.11	0.89	0.05	0.83
	CSP+QDA	87.19	0.85	0.06	0.79
	DWT+KSVM-RBF	84.45	0.83	0.07	0.84
	DWT+QDA	88.56	0.89	0.05	0.80
	Hjorth+KSVM-RBF	82.38	0.81	0.09	0.75
	Hjorth+QDA	80.62	0.80	0.09	0.72
ErrP Classifier	AAR+LSVM	92.71	0.91	0.04	0.82
	AAR+LDA	90.18	0.85	0.06	0.80
	Temporal Feature+ANN	83.13	0.82	0.07	0.76
	Temporal Feature+LDA	80.52	0.79	0.08	0.74
	SWLDA	91.23	0.90	0.04	0.81
SSVEP Classifier	PSD(Welch)+Threshold	92.89	0.92	0.04	0.86
	PSD(Welch)+LSVM	93.80	0.93	0.03	0.85
	FFT	88.81	0.87	0.05	0.78
	CCA	94.96	0.94	0.02	0.88

CSP=Common Spatial Pattern

KSVM-RBF=Kernelised Support Vector Machine with Radial basis function kernel

LSVM=Linear Support Vector machine,DWT=Discrete Wavelet Transform

QDA=Quadratic Discriminant Analysis,LDA=Linear Discriminant analysis

ANN=Artificial Neural Network, CCA=Canonical Correlation Analysis

3.5.5 Results and Discussions

The results of the current experiment are presented in three stages. First, we provide a comparative analysis between the performance of the proposed feature extraction and classifier combination and other widely used methods in BCI literature.The performance is evaluated by averaging the performance of all the subjects over all the sessions during the testing phase. In the second stage, we provide performance analysis of all the subjects that participated in the testing session, and the performance of the proposed fuzzy controller is presented in the third stage.

Performance of the brain signal detection methods are evaluated on the basis of four metrics - Classification Accuracy (CA), True Positive Rate (TPR), False Positive Rate (FPR) and Cohen's kappa index (κ) as used in [207].

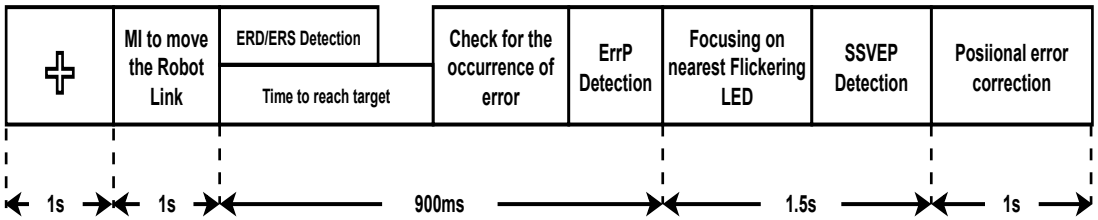

Figure 3.7: Timing diagram of testing session

- CA: Classification accuracy (CA) indicates the fraction of trials in the test data that have been correctly classified by the system. It is computed from the confusion matrix as below;

$$CA = \frac{TP+TN}{TP+TN+FP+FN} = p_a \tag{3.1}$$

  where TP, TN, FP, and FN represent True Positive, True Negative, False Positive and False Negative respectively.

- TPR: True Positive Rate (TPR) is defined as the fraction of test trials that are correctly detected as the positive instances out of the total positive instances. Numerically it is computed as below;

$$TPR = \frac{TP}{TP+FN} \tag{3.2}$$

- FPR: False Positive Rate(FPR) is defined as the fraction of instances that are misclassified as the positive class but originally belong to the negative class out of all negative instances.The computational formula of FPR is given below;

$$FPR = \frac{FP}{FP+TN} \tag{3.3}$$

- kappa(κ): Cohen's kappa index is a inter rater reliability measure of categorical items and it is used to assess the reliability of the classifier. It is computed as

$$\kappa = \frac{p_a - p_e}{1 - p_e} \tag{3.4}$$

  where p_e is the chance of agreement that is expected and p_a is actual percentage of agreement. The random accuracy, p_e, is calculated as

$$p_e = \frac{(TN+FP)(TN+FN)+(FN+TP)(FP+TP)}{(TP+TN+FP+FN)^2} \tag{3.5}$$

Performance of MI detection is presented in the first phase of Table 3.1. Along with the proposed Feature Extraction and Classifier combination (CSP + RBF SVM), we considered six other combinations to compare the performance. It is evident from the table that the proposed feature extraction+classifier combination worked best in our case yielding an average accuracy of 91.31% with average TPR,FPR and kappa of 0.89, 0.04 and 0.84 respectively.

ErrP detection and SSVEP detection performances are compared with other relevant methods and results are presented in the second and third phase of Table 3.1. It is observed that average ErrP detection accuracy is achieved as high as 92.71% followed by the TPR ,FPR and kappa of 0.91, 0.04 and 0.82. Clearly, the present ErrP detection scheme outperforms the other methods by a significant margin.

We see a similar result in SSVEP performance, where the present SSVEP detection method achieves a moderately high detection accuracy of 92.89% with the TPR=0.92,FPR=0.04 and

kappa=0.86. Although, CCA here performs a little better than our proposed detection method, but still we choose the proposed method for the major advantage of being computationally very inexpensive, hence most suitable for real-time operation.

Table 3.2: Subjectwise Motor Imagery Detection Result

Subject	**Performance Metrics(MI Detection)**				
	CA% $\pm std$	TPR	FPR	Kappa(κ)	Time(s)
Sub1	92.82±2.39	0.92	0.03	0.86	0.602
Sub2	93.96±1.82	0.92	0.03	0.91	0.549
Sub3	94.39±1.06	0.93	0.02	0.92	0.553
Sub4	89.81±2.21	0.86	0.03	0.81	0.608
Sub5	87.84±1.89	0.86	0.06	0.84	0.574
Sub6	94.49±1.84	0.92	0.04	0.89	0.579
Sub7	92.16±1.95	0.91	0.03	0.81	0.601
Sub8	91.87±1.26	0.89	0.04	0.83	0.583
Sub9	94.23±1.93	0.93	0.03	0.81	0.559
Sub10	93.12±1.28	0.93	0.05	0.84	0.552
Sub11	86.82±4.28	0.87	0.08	0.78	0.548
Sub12	84.23±3.73	0.85	0.07	0.76	0.571

Table 3.3: Subjectwise ErrP Detection Result

Subject	**Performance Metrics(ErrP Detection)**				
	CA% $\pm std$	TPR	FPR	Kappa(κ)	Time(s)
Sub1	94.81±1.05	0.93	0.03	0.82	0.109
Sub2	94.52±1.01	0.94	0.04	0.90	0.113
Sub3	91.86±2.09	0.90	0.03	0.79	0.108
Sub4	93.47±0.98	0.92	0.04	0.81	0.121
Sub5	94.31±0.77	0.95	0.04	0.78	0.111
Sub6	93.28±1.46	0.92	0.03	0.93	0.107
Sub7	90.63±2.58	0.91	0.03	0.81	0.118
Sub8	89.86±2.81	0.88	0.03	0.85	0.105
Sub9	92.19±1.63	0.90	0.02	0.86	0.118
Sub10	90.25±2.28	0.90	0.06	0.78	0.108
Sub11	89.11±3.13	0.90	0.06	0.79	0.113
Sub12	86.28±2.08	0.85	0.05	0.72	0.110

Performances of all the subjects participated in the experiment are given in Table 3.2, Table 3.3, and Table 3.4. Each participant is evaluated through four metrics (CA, TPR, FPR, kappa(κ)) described earlier.Average classification time taken by the classifier during the testing time is also reported in the above tables. Table 3.2 reveals that the highest detection accuracy of MI brain pattern is achieved for the sixth subject (CA=94.49%) while the third subject shows the highest kappa value of 0.92 indicating highest reliability. As revealed from Table 3.3 and Table 3.4,the

Table 3.4: Subjectwise SSVEP Detection Result

Subject	Performance Metrics(SSVEP Detection)				
	CA% $\pm std$	TPR	FPR	Kappa(κ)	Time(s)
Sub1	93.88±0.89	0.92	0.02	0.91	0.091
Sub2	91.49±0.96	0.92	0.03	0.88	0.082
Sub3	91.90±0.93	0.90	0.04	0.82	0.095
Sub4	95.06±0.18	0.95	0.03	0.91	0.090
Sub5	95.27±0.27	0.94	0.02	0.92	0.086
Sub6	89.26±2.65	0.90	0.03	0.86	0.097
Sub7	92.43±1.03	0.91	0.03	0.87	0.092
Sub8	90.79±1.88	0.89	0.05	0.82	0.089
Sub9	93.72±0.98	0.94	0.05	0.81	0.103
Sub10	90.93±2.15	0.90	0.05	0.81	0.098
Sub11	85.89±5.05	0.84	0.08	0.72	0.089
Sub12	88.21±4.29	0.89	0.05	0.80	0.085

other two brain patterns, ErrP and SSVEP were detected with maximum accuracy of 94.81% and 95.27% respectively. The highest ErrP accuracy is observed with the first subject while the fifth subject shows the highest SSVEP accuracy. For the above two categories of signal, the highest kappa values are achieved as 0.93 and 0.92.

3.5.6 Comparison of System Performance

The overall position control performance of the system is evaluated using few popular metrics taken from control system literature. The metrics are Success rate, Steady State Error(SS error), Peak Overshoot and Settling Time [207][206] The metrics are given below with their formal definition.

- Success Rate: It defines the number of successful attempts made by the robot arm to reach the desired target position. An attempt is considered successful only when it reaches the target position and stays at that position.

- Steady-State error: It indicates the maximum positional deviation of the robot end effector from the desired position in the infinite time range.

- Peak Overshoot: The maximum deviation of the response from its desired position. It is expressed as percentage change from its final response.

- Settling Time: Time consumed by the robot arm to reach and stay within the 2% of steady state location.

Overall performance of the system is presented in Table 3.5. Results are averaged over all the subjects over all the testing sessions. Performance result is compared with five other relevant strategies. First the result is compared with the open loop control strategy solely based on Motor Imagery [215]. Success rate obtained in this case found to be (76.2%). Next, the proposed method is compared with four hybrid BCI control strategies, where researchers ,instead of relying on a single brain pattern ,used multiple brain signals to design a robust interface for mentally controlling a robot arm. We considered four different control strategies that used four different combination of brain signals (MI+SSVEP[216], MI+P300[147], MI+ErrP[19], MI+SSVEP+P300[207]). Comparison results are obtained by implementing the control strategies in our own BCI setup. It is evident from the Table 3.5 that our proposed method achieves

Table 3.5: Relative Performance Analysis

Strategies	**Performance Metrics**			
	Success Rate	SS Error(%)	Peak Overshoot(%)	Settling Time(s)
MI[215]	76.2	6.22	6.2	18
MI+SSVEP[216]	88.5	6.09	5.9	15
MI+P300[147]	84.3	3.21	4.5	13
MI+ErrP[19]	85.8	2.1	4.9	16
MI+P300+SSVEP[207]	90.2	0.31	4.2	20
Proposed Mehod	92.1	0.15	4.1	6

highest success rate (92.1%) among all the control strategies. It also ensured the lowest settling time(6s), steady-state error (0.15%), and peak overshoot (4.1%) among strategies under comparison. Although the present scheme shows improvement over all the fields considered in Table 3.5, the major improvement is considered to be the drastic reduction of settling time with simultaneous reduction of steady state error and peak overshoot. Hence, the proposed fuzzy BCI controller outperforms the rest of the control strategies by a significant margin.

3.6 Conclusion

This chapter claims to have utilized mentally generated sign and magnitude of positional error for automatic control of artificial robotic limb. The principles and realization of the above idea being novel in the realm of BCI, is expected to open up new direction of control strategies, parallel to traditional controllers, as both the (approximate) magnitude and sign of positional error is known beforehand. Because of approximate estimation of positional errors, the logic of fuzzy sets has been incorporated that could handle the approximations and yields good control accuracy with small peak overshoot below 4.1% and settling time around 6 seconds.

Chapter 4

Vision-Assisted Brain-Computer Interface for Autonomous Position Control and Grasping

A major drawback of a Brain-Computer Interface-based robotic manipulation is the complex trajectory planning of the robot arm to be carried out by the user for reaching and grasping an object. The present chapter proposes an intelligent solution to the existing problem by incorporating a novel Convolutional Neural Network(CNN)-based grasp detection network that enables the robot to reach and grasp the desired object (including overlapping objects) autonomously using an RGB-D camera. This network uses a simultaneous object and grasp detection to affiliate each estimated grasp with its corresponding object. The subject uses motor imagery brain signals to control the pan and tilt angle of an RGB-D camera mounted on a robot link to bring the desired object inside its Field-of-view presented through a display screen while the objects appearing on the screen are selected using the P300 brain pattern. The robot uses inverse kinematics along with the RGB-D camera information to autonomously reach the selected object and the object is grasped using the proposed grasping strategy. The overall BCI system outperforms other comparative systems involving manual trajectory planning significantly. The overall accuracy, steady-state error, and settling time of the proposed system are 93.4%, 0.05%, and 15.92 s, respectively. The system also shows a significant reduction of the workload of the operating subjects in comparison to manual trajectory planning-based approaches for reaching and grasping.

4.1 Introduction

People suffering from neuro-motor disabilities face great difficulty in locating and grasping objects even if the desired object is present within their reach. With the recent development of Brain-Computer Interface (BCI) technology and current state-of-the-art robotic arms, hands, and perception systems, it has been proved that these individuals with restricted mobility can interact with their environment to perform activities of daily living (ADL), including things like drinking water, opening doors, and other basic actions. BCI provides direct nonmuscular communication between the neural activity generated by the subject's brain and the outside world [155]. For electroencephalography (EEG) based non-invasive BCI, the brain signals are obtained by placing the electrodes on the surface of the subject's scalp which are then mapped to manipulate external devices such as humanoid robots [108, 217], virtual helicopters [218, 219], wheelchairs [220, 221], telepresence mobile robots [222, 223]. In the recent past, BCI has been successfully used for rehabilitation training of stroke patients [224, 225, 226], motor control of prosthetic limbs [227, 228, 229] and performing several activities of daily living [230, 231, 232]. However, accurate object grasping using only brain-commanded signals is still an open challenge because of the high degrees of freedom (DOF) and challenges arising from complex precise position control of the robot arm.

EEG based BCI can be categorized based on characteristic brain activity patterns. Among them Motor Imagery (Event-Related Desynchronization / Synchronization (ERD/ERS)) [233, 234], Steady State Visual Evoke Potentials (SSVEP) [235, 236] and P300 patterns[237] are widely used. Motor Imagery is extensively used for control of brain-actuated robot link control and navigation. However, the main drawback of the MI-based system is the rigorous subject training required. P300-based BCI is relatively easy to use for generating control signals without extensive training of the user. There are also traces of work where a hybrid modality employing two or more brain signals is used for robot link manipulation. But most of the previous research works employing MI and P300 are solely based on subjective control of the robot link where the participating subject mentally guides the robot arm to reach and grasp the desired object. The main drawbacks of those systems are two folds, first, the brain-commanded robot often misses the target object resulting a large positive or negative positional error. Second, it requires a large amount of subject training to accurately control the position of the robot arm to grasp the desired object. Here the subjects need to perform the complex trajectory planning of the robot arm in order to align the robot gripper with the desired object. It becomes extremely difficult for the human subject to perform such complex planning mentally and to control the position and orientation of the robot gripper to perfectly grip the desired object. Added to the above facts, such complex trajectory planning imposes a high cognitive load in the user's brain.

Literature shows that quite a few studies exist in the domain of EEG-based grasping control. In 2012, Hochberg *et al.* [238] and Collinger *et al.* [239] developed neural interface system-based control of robotic arms to perform three-dimensional reach and grasp movements for patients with tetraplegia. Later in the past few years, ample experiments is conducted to control the robot arm using brain signals for reaching and grasping task [240, 241, 242] In all the above cases, MI-based BCI protocol is used. These BCI systems required several weeks of

training sessions to learn direct motor control with high DOF. Such a rigorous training procedure often causes a large discomfort to the participants. Moreover, as these systems are not fully autonomous and the user controls the complete trajectory of the robot arm, their performance is limited for real-time practical applications and the cognitive load of the operating subjects is also very high. There also exists quite a few literature that uses P300 navigational signal to the robot. Spataro *et al.* [243] used P300 based EEG command to control a humanoid robot with the aim of reaching and grasping a glass of water; however the desired object can not be directly selected and the subject needs to mentally guide (using P300 based GUI) the manipulator to reach the object. Such drawback is also seen in a few other works as well [149, 244, 245]. As discussed, such a strategy imposes a high cognitive load on the subjects' brain. Recently, Rakshit *et al.* [207] proposed a SSVEP-based random order robot link selection and P300 based link movement seizing strategy to reduce the positional error. They found a drastic reduction in positional error compared to the other state-of-the-art literature [19], but here again, the entire robot arm trajectory is planned by the subject.

BCI based shared control strategy has also been studied in the past. Tang *et al.* [105] used a shared control strategy to grasp an object using robot hand but the method suffers from the fact that the objects used for the experiment were identical in nature (red cap bottles), which makes the method challenging for diverse objects present in the environment. Xu *et al.* [107] proposed a novel shared control strategy where subjects mentally guided a robot end-effector in a horizontal plane and once the end-effecter comes within close vicinity of the target object, the switch over to automatic control using vision-based movement-planning is instigated. Although they achieved the highest accuracy around 97% but the following points still need to be addressed there. First, the user can move the end-effector in only a horizontal plane, no control commands are assigned to move it in the vertical direction.Second, the scheme does not allow the user to select the target object priory, the subject still needs to mentally plan the trajectory of the end-effector and use motor imagery to reach the target object. In their continuation work, Xu *et al.* [246]extended their strategy for multiple objects and provided adaptive assistance to the participating subject. Assistance was provided by implementing autonomous trajectory correction and autonomous grasping during reaching and grasping of the target object respectively. The scheme still requires human intervention in the path planning of the end-effector. Grasping performance was evaluated using three identical objects scattered in the workspace, hence its performance for various objects in various scenes (overlapping and non-overlapping) is still to be explored. In [122] Liu *et al*. proposed a novel strategy of controlling a dual-arm robot using motor imagery and a Kinect sensor. The subject used their left and right MI to command a dual-arm robot to lift and drop a given object respectively. A PDNN-based neuro-dynamics optimization was used for solving the motion redundancy of the robotic arm. In [106] Tang *et al* proposed a BCI-based robot manipulation approach to quickly grasp an object using motor imagery and camera-based object detection technology. A camera is used here to capture the live feed of the robot environment which is visible to the user through a computer monitor. The subject observes the computer screen and uses left/right arm motor imagery to align the robot arm in such a way that the target object should come in the target area(centre of the camera view). The

YOLO object detection algorithm is used to get the information about the object inside the target area and the grasp command is executed thereafter. We recognize first, that aligning the target object with the target using mental commands is bit challenging for the patients and second, the performance of the system is still unknown for objects located spatially very close to each other (for the condition when more than one object come inside the target area). Recently Zeng *et al.* [128]proposed a novel shared controller which dynamically blends the user motion planning and autonomous motion planning to achieve a smooth and collision-free robot trajectory. The user continuously uses his/her gaze direction to move the end-effector in a desired direction over a horizontal plane and simultaneously performs motor imagination to modulate the speed of it. The subject gets assistance for most difficult part of the task. The strategy yields a maximum of 100% success rate in this context. However the paper focuses mostly on the reaching task and its performance(reaching+grasping) in presence of multiple overlapping objects is yet to be explored. The scheme also involves user intervention throughout the task (focusing gaze and performing MI simultaneously), which may increase the cognitive load of the novel participating subjects. Duan *et al.*[108] proposed an approach to manipulate wheeled robot using mental commands and computer vision. A camera mounted on the chest of the mobile robot extracts the information about the robot environment. The computer screen displaying camera-view about robot's trajectory of motion includes provisions for generating navigational commands for the robot using SSVEP, whereas MI commands are issued to accomplish the manipulation task such as grasping the object. The grasping phase was validated using an object which carries a color mark on its body which helps the vision system to distinguish the object from the background based on the color feature. Hence the proposed system performance is yet to be explored in real-life scenario, where multiple objects with different colours are present and it is also difficult to mark each of them with colour marker. Wang *et al.* [124] in a recent work used camera-based real-time feedback to navigate a telepresence robot and reach the desired object using SSVEP. The scheme employs a camera mounted on top of a robot arm to explore the objects within the field of view, which are transferred to a computer monitor for selecting the target object using SSVEP. Here for each object, a bounding box is developed. These bounding boxes flicker at different frequencies to represent the identity of the individual objects. A subject intending of selecting a specific object focuses on the item and the flickering frequency is picked up by the subject through SSVEP. Once the object is selected, the navigation of the robot arm is automated by a camera-based position control system.However if the objects are located spatially very close to each other, it might fall inside the same flickering bounding box making it difficult to grasp any one of them. Apart from that, prolonged attention over the SSVEP stimuli also causes mental and eye fatigue to the user. Similarly, Zhang *et al.* in [241] used a MI based shared control strategy to control the robot grasping, and in their another paper [7] they used SSVEP-based object selection along with the Kinect-based machine vision, the shared control strategy is used for the same purpose. Recently, Di Lillo P *et al.* [125] used a similar P300 and Kinect-based grasping strategy to control a manipulator in grasping an object. Li *et al.* developed a BCI based shared control strategy to navigate a humanoid robot by combining central vision tracking strategy and two different brain signals N200 and P300 [247]. Batzianoulis *et*

al. [126] proposed an interesting approach for shared control by utilizing the ErrP signal and Inverse reinforcement learning (IRL) paradigm. Here the subject has its own choice of trajectory planning of a robot, for instance, obstacle avoidance and trajectory planning towards the fixed target. Each time the robot reaches an obstacle before reaching the destination, the subject releases an ErrP signal. The decoded ErrP signal is used to move away the robot from the obstacle without violating the planned trajectory of the robot. An IRL algorithm is employed to change the trajectory following the user's preferred trajectory of motion.

In all the literature stated above, the authors have relied on existing object recognition strategy which is not capable of detecting multiple overlapping objects present in the field of view of the camera. In real-life scenarios, such multiple overlapping objects can frequently be found in the domestic workspace of the user, and grasping those objects using only manual cognitive effort is immensely difficult for the human user.

In this chapter, we aim to provide an intelligent solution to this problem of brain-actuated object grasping with the help of a camera mounted on a robot arm to localize the object on the computer monitor and autonomously control the motion of the arm to accurately grasp the object. This work employs a 6-DOF robot arm and a Microsoft Kinect which can be used as a depth-sensing device in association with an RGB camera. The surrounding environment is visible to the subject on the computer screen through the real-time feed of the Kinect. The Kinect is mounted on the robot arm so that it can move in accordance with the arm. The subject uses feet motor imagery to choose between the pan and tilt motion of the camera and hand motor imagery to change the pan and tilt angle of the camera(by moving the 1st/5th joint of the robot arm) until the desired object comes into the field of view of Kinect and appears in the screen. A standard pre-trained 1-D CNN classifier is used to decode the hand and feet motor imagery signal. A deep learning-based masking algorithm is used to estimate the object masks in the environment and to compute the centroid of all the objects appearing on the screen. To choose the desired object, the centroid of the objects appearing on the screen is flashed randomly. Once the centroid of the target object is flashed, the subject releases a P300 signal, indicating his/her choice of the desired object. After the desired object is known to the system, the robot arm automatically moves closer to the object and we employ a novel Overlapping Object Grasping Network(OOGNet) to estimate the proper grasping rectangle and finally, grip the desired object by a parallel-plate gripper mounted at the end of the last link of the arm.

The main contributions of this chapter are summarized as follows

- The chapter presents a novel scheme for brain-commanded object grasping to localize, select and grasp the desired object in a multi-object scene. Our proposed strategy of shared cognitive control allows subjects with neuro-motor disabilities to grasp objects in their surroundings accurately and reliably with minimal cognitive effort.

- The chapter proposes a CNN-based novel robotic grasp detection network named Overlapping Object Grasp Net(OOGNet), which is capable of grasping the desired object even if the object is partially overlapped by other objects. The proposed grasping model outperforms the baseline algorithms by a large margin.

- Here, the subject is relieved from planning a complex trajectory for the robot link to align it with the desired object as the entire reaching and grasping phase of the robot is made autonomous in the present work. Hence, the proposed scheme requires very little subject training compared to existing state-of-the-art algorithms and reduces the overall workload of the subject.
- The proposed strategy significantly improves the success rate while minimizing the settling time and positional-steady error of the system. Autonomous navigation of the robot toward the desired object and the proposed OOGNet-based grasping strategy yielded superior performance.

In addition to this, this chapter provides a comparative analysis of the workload imposed on the performing subject while implementing different BCI schemes. Due to the limited cognitive processing ability of the human brain, workload analysis becomes a necessary method to evaluate the advantages of any BCI scheme over the others [248][249]. Such comparison also provides a tool of assessing the match between mental cost and system performance [250]. Here we adopt a NASA-TLX-based workload analysis technique to compare our proposed BCI scheme with the two other state-of-the-art BCI schemes that use manual trajectory planning [251]. NASA-TLX-based workload assessment has greatly been adopted in the field of BCI for years and also proves to be an effective way of workload assessment [252][130].

4.2 System Overview

Our setup consists of a 6-DOF robot arm (Model: ABB IRB 120), with a payload capability of 3 kg and a maximum reach of 3 ft, mounted beside a human subject. The links of this manipulator are connected by rotary joints allowing only rotational movements. A Microsoft Kinect sensor(RGB camera with depth sensing device) is placed on the 5th axis of the robot arm, in such a way that one can change the camera's pan and tilt by rotating the 1st and 5th joints of the manipulator respectively. The Kinect provides live RGB feedback of the surrounding environment to the subjects via an LCD monitor placed next to them. Multiple objects of different classes are arranged in a variety of layouts, in the vicinity of the robot arm with some objects overlapping with others. We aim to solve the task of locating, identifying and grasping the desired object (within the reach of the robot arm) with minimal human intervention. A complete overview of the system is presented in Fig.4.1. For the sake of simplification, we have taken the following liberties in our set-up:

- Although some objects may overlap with each other, each object is clearly visible from the initial position by changing the camera's Field of View (FOV).
- Different objects belonging to the same class are indistinguishable in nature.
- Grasping an object does not require re-arrangement of other overlying objects.

We have divided the problem into several sub-tasks.

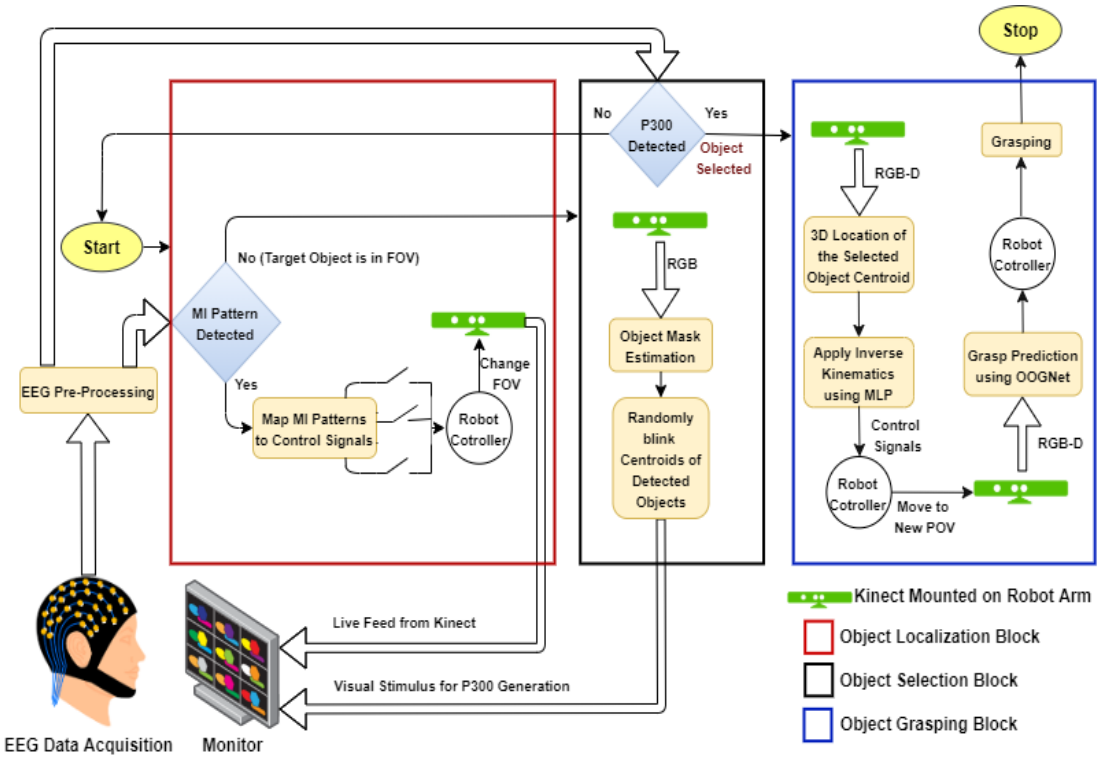

Figure 4.1: Complete overview of the proposed scheme.

1. ***Locating the object:*** Although the user can see the objects physically in the environment, all of them are not visible on the monitor as the Field-of-view(FOV) of the Kinect is limited. So there may arise a need to move the Kinect to bring the desired object into its FOV. We have thus developed an algorithm that allows the subject to use MI brain signals to change the Kinect's pan and tilt based on the position of the desired object. The right and left arm motor imaginations are mapped to the clockwise and anti-clockwise movement respectively of the selected joint (1st/5th) , while feet imagery is used to toggle the selection between 1st and 5th joint. Hence a subject first uses feet imagery to select the desired joint followed by the left and right arm imagery to rotate the joint in desired direction. The present selection of the joint is displayed on the screen for the convenience of the participating subject. Motor imageries are decoded by detecting each unique ERD/ERS MI pattern [233, 234] using a CNN classifier.
   Once the object fully enters the Kinect's FOV, the user stops the Motor imagination. If no MI pattern is detected in two consecutive time windows, the algorithm terminates, indicating to the system that the object has been successfully located. This stipulation reduces system sensitivity to both classification and user errors. The algorithm of the process is given in Algorithm 2.

2. ***Choosing the object:*** Depending on the layout, there may be multiple objects visible on the screen together with the desired one. In order to tell the system which object to grasp, we need to identify all the objects and select the desired one out of them. Our algorithm uses a state-of-the-art network Mask-RCNN [253] to detect objects in the image and calculate the object centroids from their segmentation masks. The centroids are then

flashed in a random sequence following the oddball paradigm [237]. When the desired centroid flashes, the subject gazing at that object elicits a P300 response which is detected by a CNN classifier to identify the target object for the system. The target object's class as predicted by Mask-RCNN is recorded for automatic identification in a later step. If no P300 signal is detected, the system returns to the starting state. Algorithm 3 explains the above procedure.

Algorithm 2: Algorithm for Object Localization

```
COUNT ← 1;
Default Joint ← 1st joint (Pan);
while COUNT ≤ 2 do
    Provide Cue to Subject to Start MI;
    while Time within TIME WINDOW do
        Read EEG data;
    Provide Cue to Subject to Stop MI;
    Classify EEG for MI Tasks;
    if MI not detected then
        COUNT ← COUNT+1;
    else
        COUNT← 1;
        switch MI pattern do
            Feet: Toggle between Pan and Tilt;
            Left hand: Tilt Up/Pan Left;
            Right hand: Tilt Down/Pan Right;
```

Algorithm 3: Algorithm for Object Selection

```
Perform Object Detection and Centroid Calculation;
Generate Random Sequence S of Detected Objects;
for Objects in S do
    Blink Centroids and Read EEG Data;
    if P300 detected then
        Choose Current Object;
        Exit Loop;
```

3. ***Grasping the object:*** The system can now identify the desired object. But the end effector/gripper is still far away from its target to grasp it. We solve the automatic grasping problem in two steps:

   (a) ***Positioning:*** For any position (within the reach of the arm) of the desired object, the gripper first automatically moves closer to the object. Once the desired object is selected, the system calculates its real-world 3D position from the 2D coordinates of the centroid and the 2D depth map produced by the kinect. We apply inverse kinematics to move the gripper to a new position located a small fixed distance above

the desired object, with the Kinect tilted downwards to provide a new point of view (POV) from the top. This allows the next step to be independent of object's position relative to the initial position of the gripper.

(b) ***Estimating Gripper Configuration:*** The positioning step allows us to formulate the final task of grasping as the problem of calculating the gripper configuration from the object image. Jiang *et al.* [254] and Lenz *et al.* [255] have shown that a seven-dimensional grasp orientation for a parallel plate gripper can be represented by a grasp rectangle parameterized by its position, width, height and orientation. So, the problem is reformulated as the task of predicting a grasp rectangle from the object's RGB-D image. In our case the top view may show multiple objects on the screen along with the desired one with possible overlapping layouts. We need to re-select the target object from the new POV as the previously calculated 3D position is not reliable in terms of identifying the object across different points of view of the camera, especially in case of close and overlapping object arrangements. Further, due to the proximity between objects, the problem cannot be reduced to a single object or a simple multi-object non-overlapping grasp detection task like [254, 255, 256, 257, 258, 259, 260, 261], as there arises a need for affiliation between object and grasp prediction.

Hence,our proposed network OOGNet generates a grasp rectangle, a bounding box and a class prediction for each object in the multi-object scene, thus ensuring a reliably high accuracy even in overlapping scenarios. Since objects of the same class are indistinguishable in our setup, the target object is re-identified automatically, by matching the object classes predicted by OOGNet with the previously recorded class of the object chosen in the selection stage. Once the target object is identified, the corresponding grasp rectangle is converted to a gripper configuration to grasp the desired object.

4.3 Method

4.3.1 EEG Data Analysis

The raw EEG data is usually noisy and contains a lot of irrelevant information. Hence, appropriate task-specific filtering techniques are applied to the EEG signal before classification.

4.3.1.1 Motor Imagery

The EEG signal is band-pass filtered at 8-24 Hz to isolate the ERD/ERS phenomenon associated with MI brain patterns. Because of the effectiveness of Common Spatial Patterns (CSP) [38] in discriminating Motor Imagery tasks, we use CSP filters in a multiclass one vs rest scheme to project the EEG data along directions that maximise the differences between MI classes. The

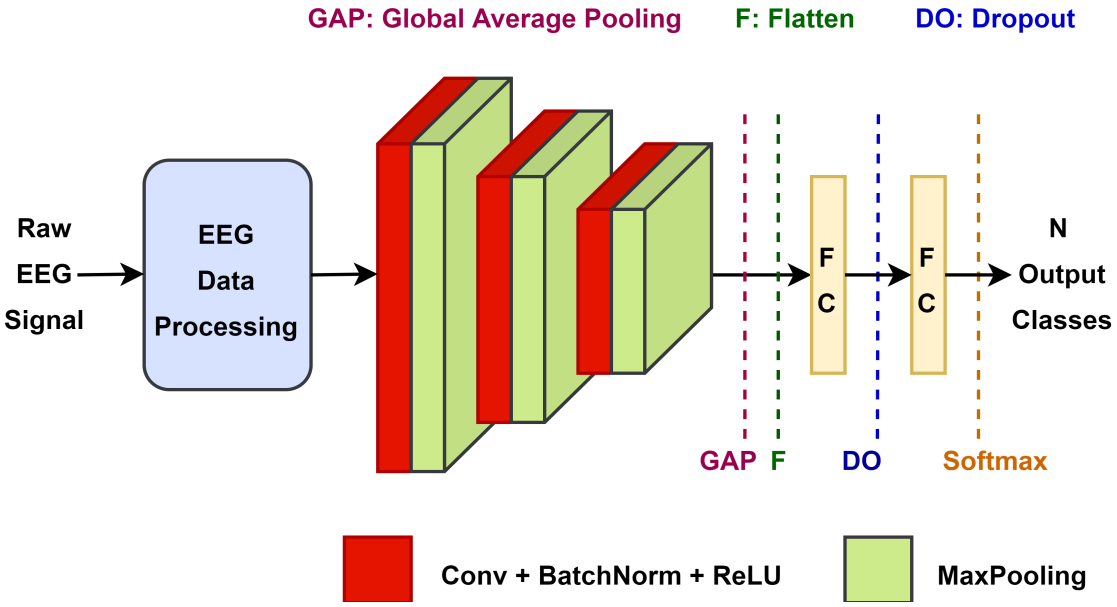

Figure 4.2: The common EEG classification network architecture. Two separate instances of this network are used to classify MI and P300 signals with different network dimensions.

processed EEG signal (Y) is expressed as,

$$Y = WX \tag{4.1}$$

where, X is a Channels (C) $\times$ Time (T) matrix of the band-pass filtered EEG data and W is the $L \times T$ CSP projection matrix, with L spatial filters.

4.3.1.2 P300 ERP:

P300 brain signal is characterised by a positive going peak around 300 ms after the onset of the target stimulus. A Chebyshev type I bandpass filter is used to filter the raw EEG signal between 1-10 Hz, to reduce the background noise. Next, the information relevant to the P300 ERP is isolated from the EEG data using Principal Component Analysis (PCA) [262]. PCA maps the signal to a lower dimensional space by extracting the K Eigenvectors from the EEG data that contain the most information for P300 responses. The lower dimensional signal S is expressed as,

$$S = PX \tag{4.2}$$

where, X is a Channels (C) $\times$ Time (T) matrix of the band-pass filtered EEG data and P contains the K eigenvectors as rows.

4.3.1.3 Feature Extraction and Classification

Convolutional Neural Networks (CNNs) have become extremely popular for EEG classification tasks due to their much higher accuracy compared to traditional linear classifiers. The processed 2D EEG data matrix is fed into 3 Convolutional and maxpooling layers to learn high-level infer-

ences from the data and a 1D feature vector is generated from the feature map by Global Average pooling and Flatten operations. The resulting vector is classified by a series of fully connected (FC) and dropout layers. The pooling and dropout operations prevent over-fitting. Two different instances of CNN having the same architectures and layers (as stated above) is used to classify Motor Imagery and P300 brain patterns respectively. The common CNN architecture used for the classification purpose is shown in Fig.4.2. Minor adjustment is done where the final FC layer contains 3 neurons for MI classification and 2 neurons for P300 detection. For Motor Imagery, no MI pattern is detected if the probability of a particular EEG input does not exceed 0.5 for any of the classes.

4.3.2 Object Detection and Centroid Calculation

In order to select our desired object, we first need to identify all the objects present in an image. Such object identification is carried out by a state-of-the-art object detection network, called Mask R-CNN [253] that can detect objects in a variety of closely positioned and overlapping layouts. For each input RGB image, the network predicts the class, bounding box and segmentation mask for every object visible in the image. We calculate the centroid (X_c, Y_c) of each object as,

$$X_c = \sum_i (X_i/n) \;\; Y_c = \sum_i (Y_i/n) \tag{4.3}$$

where (X_i, Y_i) is the position of the i-th pixel in its segmentation mask which contains n pixels in total.

4.3.3 Gripper Alignment

The automatic positioning step allows us to reduce the cognitive load on the subject by making the system perform the precise position control task of reaching and grasping the desired object autonomously. The Kinect placed on the fifth axis of the robot arm, is equipped with an RGB camera and a depth sensor, which together provide the X and Y coordinates of the target object's centroid (obtained in the object detection and centroid calculation step) and its distance from the sensor. The centroid coordinates (x', y', z') measured from the Kinect frame are then transformed with reference to the base of the robot arm as shown in Fig.4.3. The new coordinates of the centroid (x, y, z) with respect to the base frame can be obtained from Equation 4.4.

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} R & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} I & c \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \\ 1 \end{pmatrix} \tag{4.4}$$

where, $\mathbf{c} = (c_x, c_y, c_z)$ is the position vector from the origin of the base frame to the kinect frame and $R = [r_{ij}]$ is the rotation matrix, determined from the configuration (position and orientation) of the fifth link with reference to the base frame. Once the position of the centroid is known in terms of the base frame, the robot arm reaches the object in two stages.

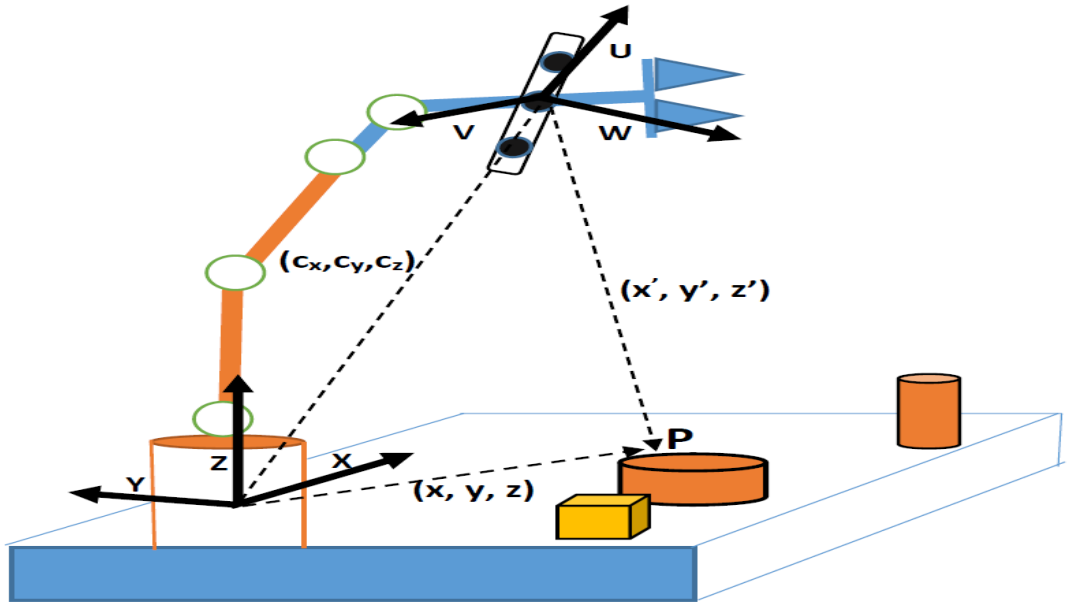

Figure 4.3: Relative position of the object from kinect frame and robot base frame.

In the first stage, the arm approaches the destination location $(x, y, z - \delta_z)$, where δ_z is a fixed offset distance above the centroid position along Z-axis. The robot arm employs an inverse kinematic model, which utilizes the D-H parameters of the robot arm to find the required joint movements for reaching the destination. Since axes of the last three joints intersect at a point in the IRB 120 robot, only the first three joints contribute towards determining the position. For the destination position and the initial position of the end effector, $(x_d, y_d, z_d)^T$ and $(x_0, y_0, z_0)^T$ respectively, the required joint movements of the first three joints can be obtained from the following expression.

$$\begin{pmatrix} x_d \\ y_d \\ z_d \end{pmatrix} = {}^0Q_i \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} \tag{4.5}$$

where ${}^0Q_i = {}^0T_1 . {}^1T_2 ... {}^{i-1}T_i$ and $i = 1, 2, 3$. Here ${}^{i-1}T_i$ is the transformation matrix for link i. Thus for a given destination co-ordinate, required joint movement is obtained by knowing the ${}^{i-1}T_i$ from below expression;

$$\begin{pmatrix} x_d \\ y_d \\ z_d \end{pmatrix} = \left({}^0T_1 . {}^1T_2 ... {}^{i-2}T_{i-1} \right) . {}^{i-1}T_i . \begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix} \tag{4.6}$$

Once the destination is reached, the 5th link is tilted downwards to orient the Kinect in a fixed angular offset with the vertical, to obtain a top view of the desired object to calculate the gripper configuration using our grasp detection network.

After the grasp rectangle is predicted and the desired object is re-identified by its class, the second stage of the approach is initiated. The coordinates (x_c, y_c, z_c) of the center of the grasp rectangle, calculated with respect to the base frame, is chosen as the grasping point. The gripper opening is fixed according to the rectangle width. Now, the orientation of the grasp rectangle determines the rotation of the gripper about the normal to the image plane (Kinect

axis). An illustrative picture of the grasp rectangle is shown in Fig.4.4. Now, the robot uses its inverse kinematics to achieve the particular orientation. The last three joint variables θ_4, θ_5, θ_6 is obtained by solving the rotation matrix ${}^3R_6$ in a similar manner described in Eq (4.6), where ${}^3R_6 = {}^3R_4\,{}^4R_5\,{}^5R_6$. With this configuration, the gripper approaches the object along the normal and grasps the object as described in [255] and [254].

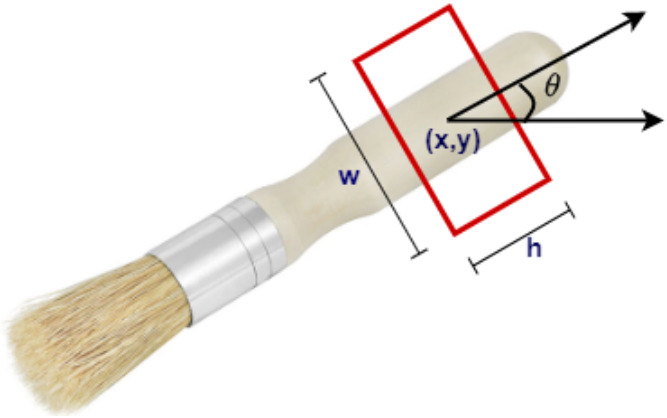

Figure 4.4: 5-D rectangular representation of a gripper configuration. Here, (x,y) denotes the center of the grasp rectangle, w, and h represent the width of the gripper opening and height of the gripper plates respectively and θ is the orientation of the grasp rectangle with respect to the horizontal direction

4.3.4 Grasp Detection

In our use case, we need to predict grasps for multi-object overlapping scenes. This is significantly more difficult than non-overlapping or single object cases, due to partial occlusion by overlapping objects and the need for affiliation between object and predicted grasp. The grasping success rates of previous works ([263], [264], [265]) in this domain are too low for use in reliable human assistant systems. The absence of a repository for the cited implementations together with the availability of depth information at our disposal, motivated us to design a novel grasp prediction network, that would ensure a high physical grasping accuracy for our use case. While the previous approaches use only RGB information, we use the RGB-D images from the Kinect and a deeper feature extractor to improve accuracy. In order to maintain a fast enough execution speed for user convenience, we predict only one grasp rectangle for each region of interest (ROI) instead of multiple rectangles, unlike the previous methods. Our proposed Overlapping Object Grasping Network (OOGNet) generates a grasp rectangle, bounding box, and object class for each object in the image thus associating each predicted grasp with its object. The architecture of the OOGNet is shown in Fig.4.5. Similar to [255], we represent a gripper configuration by a grasp rectangle (G) with 5 parameters as,

$$G = \{x, y, w, h, \theta\} \tag{4.7}$$

where (x, y) denotes the center of the grasp rectangle, h denotes the height of parallel plates, w denotes the maximum distance between parallel plates, and θ denotes the orientation of the

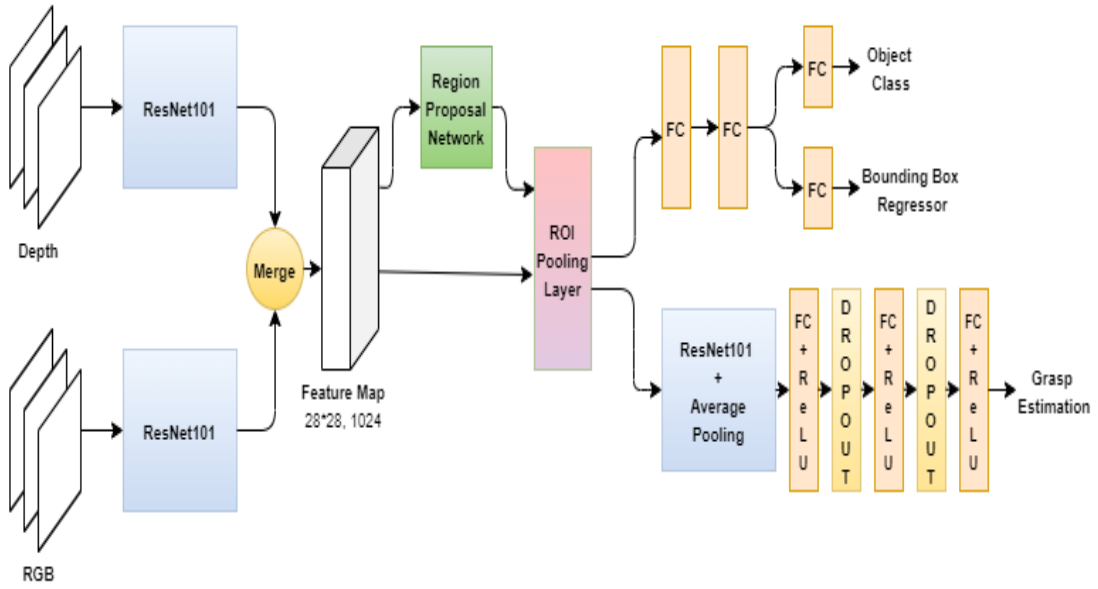

Figure 4.5: Architecture of our proposed Overlapping Object Grasping Network (OOGNet).The network takes an RGB-D image of multiple overlapping objects as input and predicts the class, bounding box and a 5-D grasp rectangle for each object in the image

grasp rectangle with respect to the horizontal axis of the image.

4.4 Grasp Detection Network

Our proposed network takes an RGB-D image as input and generates multiple ROI proposals for objects present in the image. Each ROI is then fed into three parallel branches that perform object classification, bounding box regression, and grasp prediction. The object classifier and bounding box regressor branches are similar to that of Fast R-CNN [110] in structure. The grasp predictor branch regresses to the 5 parameters of a grasp rectangle for each object class. The following sections detail the architecture of our network in two stages that describe the generation of object ROIs from the input image and the prediction of grasp rectangles from each ROI.

4.4.1 Object proposals:

The first stage of the network generates object ROI proposals from the input RGB-D image. The single-channel input depth image is converted to a 3-channel image by the grayscale to RGB conversion method. The 3-channel depth map and the RGB image are each fed into identical and parallel feature extractors. We use ResNet-101 [112] as the backbone of our feature extractor network. The skip connections in the Residual block allow us to use deeper networks that learn high-level features without degradation of accuracy. The feature extractor in this stage contains the first 23 layers of ResNet-101. Feature maps of size $28 \times 28 \times 512$ extracted from the depth and RGB inputs are concatenated to form a merged feature map of size $28 \times 28 \times 1024$. A Region Proposal Network (RPN) similar to that used by Faster R-CNN [111] is used to generate 9 (3

scales and 3 aspect ratios) Object ROI proposals for each location in the combined feature map. Each ROI is characterized by an objectness score (2 probabilities) and 4 parameters (x', y', w', h') denoting the bounding box location, where (x', y') specifies the top-left corner of the box and w' and h' denote width and height respectively. The RPN is trained in a similar fashion to [111] with the same loss function.

4.4.2 Grasp prediction branch:

Each variable-sized ROI generated from the RPN is fed into an ROI pooling layer together with the merged features to produce a smaller feature map of fixed spatial size (14×14). Three parallel branches share the pooled ROI feature map as input. The grasp branch contains a ResNet feature extractor that learns grasp-specific inferences from the object ROIs. Here the feature extractor contains the last 51 layers of ResNet-101. The grasp feature maps of size $(7 \times 7 \times 2048)$ from the last convolutional layer of ResNet-101 are pooled by an average pooling layer and fed into three fully connected layers with ReLU activation. Each fully connected (FC) layer except the final one is followed by a dropout layer to reduce overfitting. The final FC layer outputs $5 \times k$ grasp parameters, for the k object classes. Thus the grasp branch predicts a grasp rectangle for each class of object from the input object ROI.

4.4.3 Loss Function:

OOGnet generates three outputs, one from each branch. For each ROI, the classification branch predicts the softmax probabilities $p = (p_i | \forall i \in [0, k])$ of the object belonging to the k+1 classes; k object types and one background class denoting no object is present in the ROI. Here, p_i denotes the softmax probability of the object belonging to class i. The bounding box and grasp branches regress to the bounding box parameters $t_i = (x'_i, y'_i, w'_i, h'_i)$ and the grasp parameters $G_i = (x_i, y_i, w_i, h_i, \theta_i)$ respectively for each of the k object classes, where i indexes the i^{th} class.

The labels for each ROI include a ground truth class u, a ground truth bounding box regression target v, and a ground truth grasp rectangle regression target g. Extending the loss in [110] we define a multitask loss L_{total} on each ROI to jointly train for classification, bounding box regression, and grasp prediction.

$$\begin{aligned} L_{total}(p, u, v, t_i, g, G_i) = L_{cls}(p, u) + \lambda [u \geq 1] L_{box}(t_u, v) \\ + \lambda' [u \geq 1] L_{grasp}(G_u, g) \end{aligned} \tag{4.8}$$

Here, $L_{cls}(p, u) = -\log p_u$ is the classification loss. L_{box} is the bounding box loss defined over the predicted box parameters $t_u = (x'_u, y'_u, h'_u, w'_u)$ for the ground truth class u and the ground truth box parameter tuple v. The grasp loss L_{grasp} is added to the L_{cls} and L_{box} losses defined in Fast R-CNN [110] to simultaneously train for grasp predictions. L_{grasp} is defined over the ground truth grasp rectangle tuple $g = (g_x, g_u, g_w, g_h, g_\theta)$ and the predicted predicted grasp rectangle $G_u = (x_u, y_u, w_u, h_u, \theta_u)$ for the groundtruth class u. Both L_{box} and L_{grasp} are smooth L_1 losses.

For L_{grasp} the Smooth L_1 loss is expressed as,

$$L_{grasp}(G_u, g) = \sum_{j \in x,y,w,h,\theta} (smooth_{L_1}(G_u^j - g^j)) \tag{4.9}$$

where

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| \leq 1 \\ |x| - 0.5, & \text{otherwise,} \end{cases} \tag{4.10}$$

Smooth L_1 loss is used because of its robustness to outliers as pointed out by [110]. The L_{box} loss is defined similarly to L_{grasp}. The Iverson bracket indicator function $[u \geq 1]$ is defined as,

$$[u \geq 1] = \begin{cases} 1, & \text{if } u \geq 1 \\ 0, & \text{otherwise} \end{cases} \tag{4.11}$$

Labeling the background class as 0 together with the Iverson function allows the network to ignore the bounding box and grasp losses when the ROI is predicted to be the background. This is essential as there is no object and hence no graspable region in the image background. The hyper-parameters λ and λ' are the loss weights. They tune the relative weightages of the different loss terms. We use $\lambda = \lambda' = 1$ for our experiments. The ground-truth regression targets for both the bounding box and grasp rectangle are also normalized to have zero mean and unit variance.

4.5 Experiments

4.5.1 Experimental Protocol

This section describes the experimental protocol employed in this study and highlights the key steps of conducting the experiment.

Subjects : The present study employs ten volunteers showing no major illness in their recent medical history. Out of ten volunteers, six were male and four were female. All the volunteers belong to the age group of 18-35 with mean age of 30 years.The details of the experiment and its objective were made clear to all volunteers and a consent form, stating their interest to participate in the study, was duly signed by them. All the ethical and safety issues for employing human subject in the experiment is maintained according to Helsinki Declaration 1970 later revised in 2000 [266].

EEG System: EEG signal is acquired from the subjects using a 21-channel mobile EEG amplifier system. The amplifier has a sampling rate of 200Hz with built in notch filter at 50Hz frequency. The present experiment follows the international 10-20 electrode positioning system to place the EEG electrodes in the subjects' scalp. Electrode position $C3, C4, Cz$ placed over the motor cortex region and $P3, P4, Pz$ placed over the parietal region are used to capture the motor imagery. The electrode position M1-M2(mastoid process) are used as contra-lateral referencing

of all electrodes and Fpz is used as the ground position. P300 brain pattern is captured from the electrode position Pz, Cz, and Fz.

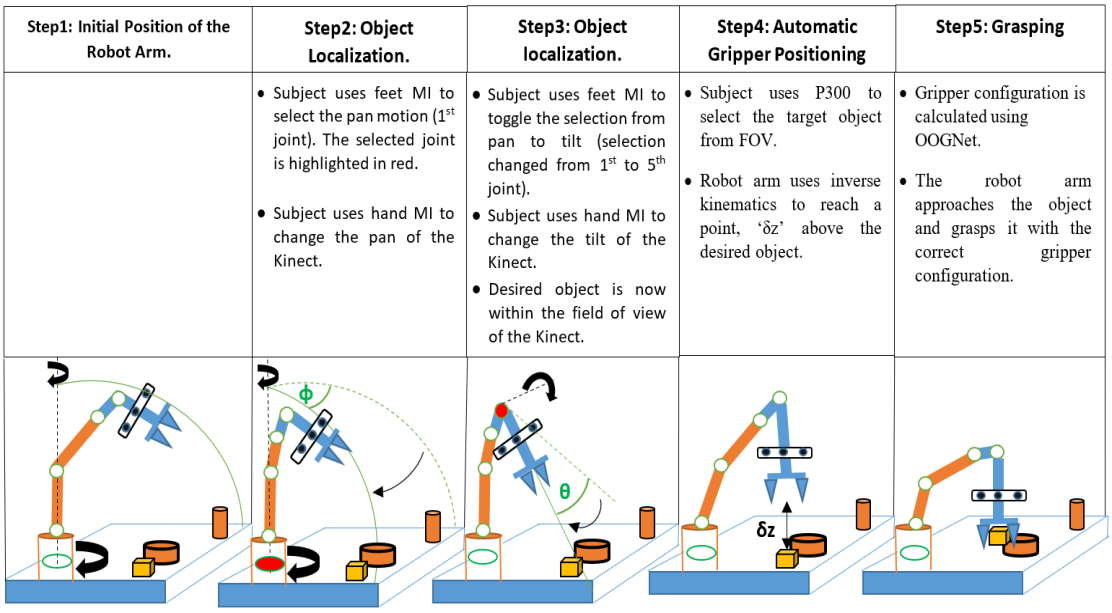

Figure 4.6: Different stages of the robot manipulation in the testing session.Robot joints are marked with green circles while the currently selected joint is highlighted in red along with their axis of rotation.The dotted lines represent the axis of rotation.

A figure depicting the different instruments used in the experiment is given in Fig.4.7.

Communication Protocol: The EEG headset is wirelessly connected with a computer through Bluetooth protocol. The computer runs a python API to capture the EEG data and processes it in real-time, while another computer(placed in front of the subject) runs a Python script to capture and process the Kinect data in real-time. Both the computers are connected to a server computer using TCP/IP (creating TCP sockets in both server and client), where computers connected with EEG and Kinect act as the clients. The server computer generates the control commands for the robot based on the information provided by the computers connected with EEG and Kinect. The control algorithm in the server is executed in the Robotstudio platform (by running an ABB Rapid language script) which again communicates with the physical robot controller(IRC 5 controller with Robotware version 6) using UDP (UDP socket). Once the generated control commands are sent to the robot controller, the robot joints are actuated and a specific task is executed.

4.5.2 Training session

Training data for the classifier are obtained from ten subjects with the repetition of five sessions for each subject with an inter-session gap of 30 minutes. Training session data are taken throughout the fifteen days. Each session contains thirty trials. Each of the trials contains a visual cue of instructions to the subject. The timing diagram of the visual instruction is illustrated in Fig 4.8. At the beginning of the trial, a fixation cross appears in the visual cue for the 2s followed by a blank screen of 2s. Now a visual cue containing the instruction of motor imagery appears on the

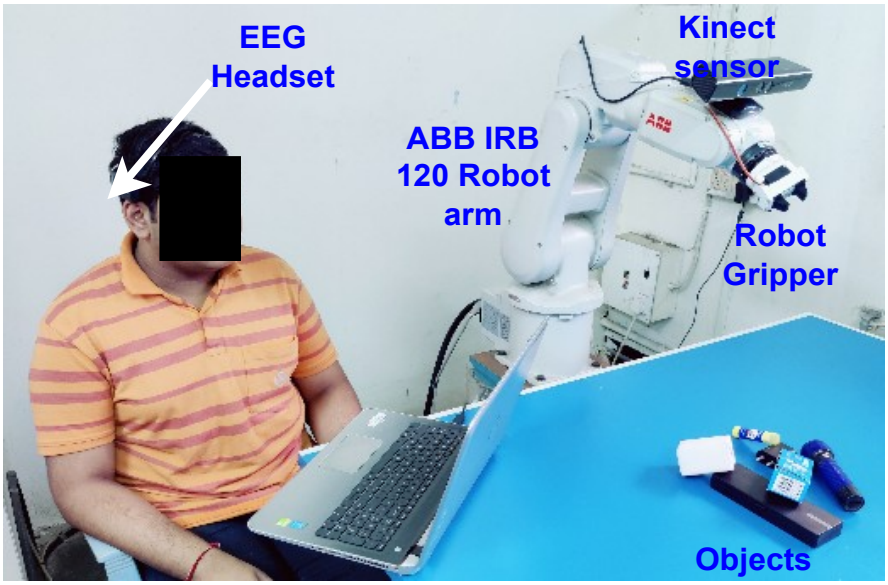

Figure 4.7: Depiction of experimental scenario and different instruments used in the experiment.

screen. The subject performs the motor imagery either to select the link (feet imagery) or to turn the virtual robot link in the given direction (left and right-hand imagery). Hence, the camera mounted on the robot arm changes its FOV and now focuses on the objects on the table. Next, the centre points of the objects start blinking in a random manner. A visual cue then appears on the screen to facilitate the P300 training data acquisition. The visual cue contains the instruction to the subjects to focus their gaze on a particular object. The subject develops the P300 brain response whenever the centre point of that object is flashed.

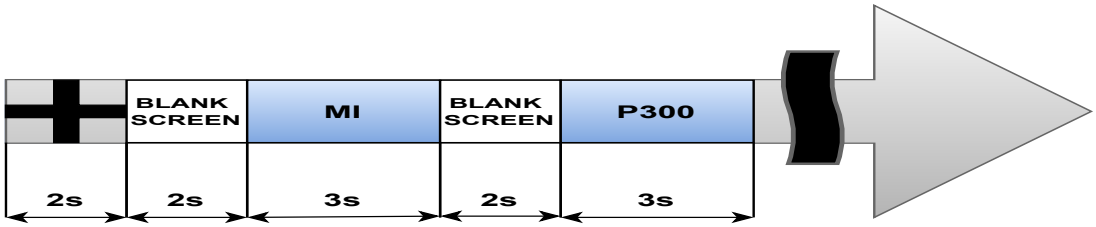

Figure 4.8: Timing diagram of Training Session.

4.5.3 Testing Session

The testing session is more complex as no visual cue is shown and subject has to plan two steps of operation (MI generation and developing P300) without any assistance. During the testing session all the signal modalities are acquired with a moving window of 1s. An exception is followed in case of MI signal acquisition, where the signal is acquired for 1s but last 0.2 s of signal samples are considered for classification [183]. For better understanding the different stages of the operation is illustrated through sketches in Fig 4.6 while the actual experimental scenario is depicted through Fig.4.9 to Fig.4.12.

Figure 4.9: Step1: The robot is at it's initial position and the subject uses his motor imagery and P300 to rotate the robot arm and select the desired object respectively.

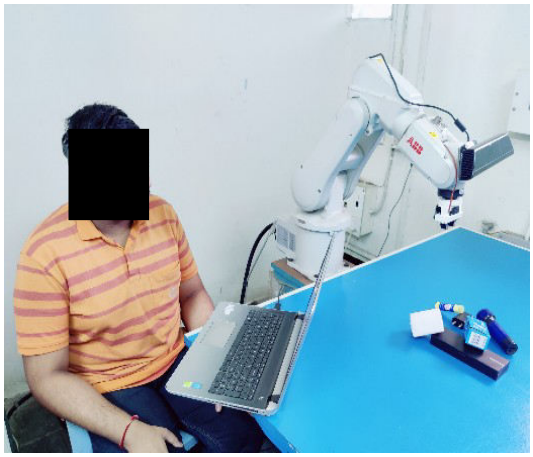

Figure 4.10: Step2: The robot is just above the desired object and determines the gripper configuration.

Figure 4.11: Step 3: The robot has successfully grasped the desired object.

Figure 4.12: Step4: Grasped object is picked up by the robot to place it in other place.

4.6 Detailed Experimental Procedure of Robotic Grasp Prediction

4.6.1 Datasets

There is no publicly available RGB-D dataset for robotic grasp detection in multi-object scenes. So, in order to train our model for overlapping multi-object scenes, we carefully collect a Multi-Object RGB-D dataset and annotate it manually. For every object in a single image, we annotate several possible grasp rectangles which are a comprehensive subset of all possible good candidates. We take several images of the same set of objects with different orientation and pose. We also include the affiliation between each grasp with corresponding objects using the index of the object bounding boxes. Example images from the multi-grasp dataset is shown in Fig.4.13 to 4.18.

Our Multi-Object Grasping Dataset contains 784 images with 3-5 different objects in each image. The objects are arranged in several overlapping and non-overlapping layouts. The dataset

consists of both RGB and depth images. We use the same Kinect as the depth sensor. There are in total of 17 classes and different instances of each class are indistinguishable in nature. The object bounding boxes and the grasp rectangles are manually annotated.

4.6.2 Pre-Training and Data Pre-processing

Similar to [267], we reuse the pre-trained weights of ResNet-101 on ImageNet [268] dataset to avoid over-fitting. The new layer weights are randomly initialized with a zero-mean Gaussian distribution with a standard deviation 0.02. The NaN values in the depth image are replaced with zeros. The depth image is converted to a 3-channel image using the grayscale to RGB conversion method and is rescaled to the 0-255 range. As both datasets are small, we perform extensive data augmentations by randomly rotating, translating and changing the background color for regularisation[269, 270, 271]. We also add noise, saturation, illumination and hue randomly, to make the system robust to real conditions.

4.6.3 Training

We train the entire network end-to-end using Pytorch framework on an NVIDIA GTX 1080 Ti GPU, with 16 GB dedicated memory, with CUDA-10 and cuDNN-7.5 installed. We randomly divide the Multi-Grasping dataset in 4:1 ratio for training and testing. There are 2016 object instances in training set and 758 object instances in the test set.

The training process is divided into two stages. First, we train the RPN using the input images and ground truth Object proposals as described in [111]. Next, the complete network is trained end-to-end. The pre-trained ResNet is fine-tuned using stochastic gradient descent (SGD) optimizer with the hyperparameters set as: initial learning rate = 0.0001, mini batch size = 16, momentum = 0.9, and the maximum number of epochs = 30. We divide the learning rate by 10 every 10000 iterations.

4.7 Results

4.7.1 Performance of EEG Classifier

The performance of the proposed EEG classifier networks is evaluated on the basis of four metrics - Classification Accuracy (CA), True Positive Rate (TPR), False Positive Rate (FPR), and Cohen's kappa index (κ) which are defined as -

$$CA = \frac{TP+TN}{TP+TN+FP+FN} = p_a \tag{4.12}$$

$$TPR = \frac{TP}{TP+FN} = Sensitivity \tag{4.13}$$

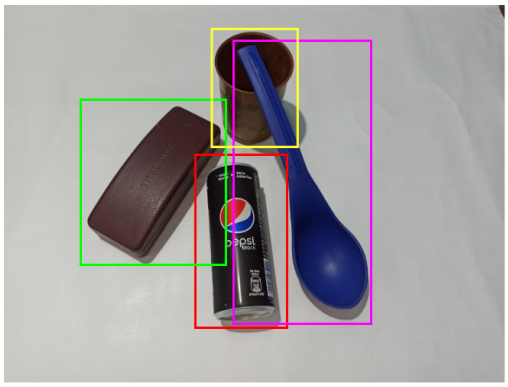

Figure 4.13: Example image 1 from multi grasp dataset

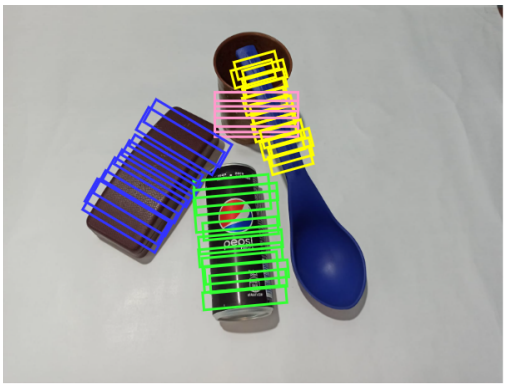

Figure 4.14: Example image 1 with multiple grasp rectangles

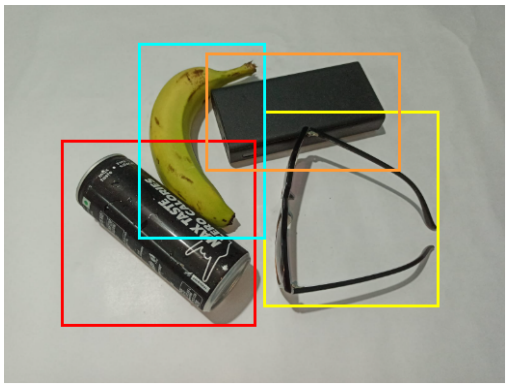

Figure 4.15: Example image 2 from multi grasp dataset

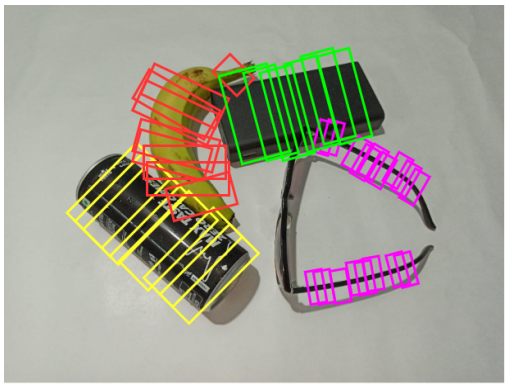

Figure 4.16: Example image 2 with multiple grasp rectangles

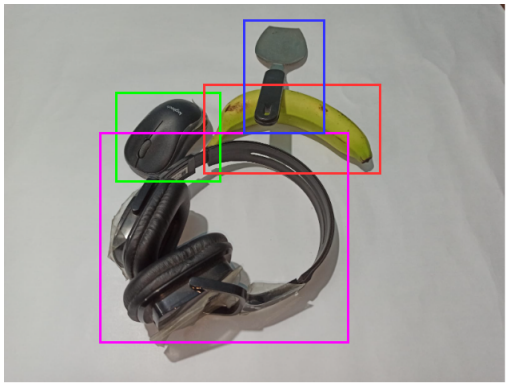

Figure 4.17: Example image 3 from multi grasp dataset

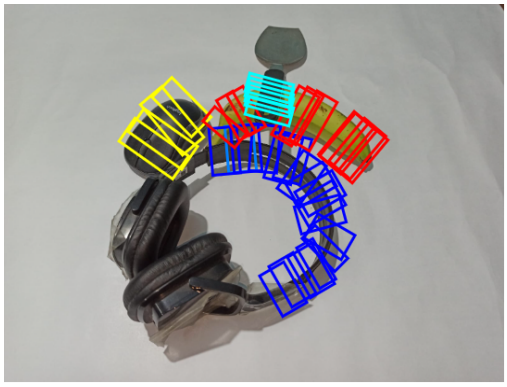

Figure 4.18: Example image 3 with multiple grasp rectangles

$$FPR = \frac{FP}{FP+TN} = 1 - Specificity \tag{4.14}$$

$$\kappa = \frac{p_a - p_e}{1 - p_e} \tag{4.15}$$

where, TP is the true positives, TN is the true negatives, FP is the false positives, FN is the false negatives, p_e is the chance of agreement that is expected and p_a is actual percentage of

agreement. The random accuracy, p_e, is calculated as

$$p_e = \frac{(TN+FP)(TN+FN)+(FN+TP)(FP+TP)}{(TP+TN+FP+FN)^2} \tag{4.16}$$

The Classification Accuracy shows the percentage of trials in the test data that have been correctly classified. TPR and FPR show the ability of the classifier to correctly detect the true positive and true negative instances out of total positive and negative instances respectively. Cohen's kappa index is a inter-rater reliability measure of categorical items and it is used to assess the reliability of the classifier.

It can be seen from Table 4.1 that the proposed classifier outperforms both linear and non-linear classification methods for both MI and P300 classification. While a standard CNN works better than linear classifiers, pre-selection of features followed by a CNN achieves the best results with CA, TPR, FPR, and κ values of 95.58%, 0.96, 0.05, 0.91 for MI and 96.30%, 0.91, 0.03 and 0.90 for P300. Better results for CSP and PCA compared to other techniques are expected as per our literature-supported intuitions about MI and P300 EEG respectively.

The metric values for 10 subjects have been reported in Table 4.2, for both MI and P300 classification, to highlight the interpersonal variance in the performance of our classifier. As we can see, the average values of CA, TPR, FPR and κ are 95.58%, 0.96, 0.05, and 0.91 respectively for MI classification, while for P300 detection, they are 96.3%, 0.91, 0.03 and 0.90 respectively. It is evident from the standard deviation values (written below the CA metric), that the inter-trial variation in the classifier performance is very small for both MI and P300 detection, indicating the high reliability and robustness of our proposed classifier, though the P300 detection system shows more reliability than MI detection system.

4.7.2 Statistical Validation of the Classifiers

Classifiers are statistically validated using Friedman statistical test. The Friedman test is a non-parametric test (does not hold the assumption that the data come from a normal distribution) that determines if there exists any significant difference between the classifier performance based on any selected parameter and ranks them according to it. Here, we have considered two different parameters, Accuracy and Reliability(kappa score), and performed the Friedman test separately for each of these parameters. The test considers a null hypothesis that assumes the performance of the classifiers under testing is equal based on the selected parameter, hence the sum of their ranks, which are assigned based on their performance, are also equal. Under the null hypothesis, Friedman statistic is distributed as χ with $n-1$ degrees of freedom, where n is the number of classifiers under testing. Mathematically the Friedman statistic is computed as below;

$$\chi_F^2 = \frac{12}{Ln(n+1)} \sum_{i=1}^{n} R_i^2 - 3L(n+1) \tag{4.17}$$

where L is the number of data-set (we considered data-set averaged over all the sessions for each of the participating subjects, hence L=10) , n is the number of classifiers under testing and R_i is the rank sum of the classifier which was determined by summing all the ranks it got from all the

Table 4.1: Comparative Study of Different EEG Classifiers

	Classifiers with Optional EEG Pre-processing	Performance Metrics			
		CA (%)	TPR	FPR	κ
MI	LSVM [272]	85.80	0.85	0.07	0.83
	KSVM-RBF Kernel [273]	87.55	0.86	0.06	0.86
	IT2FS [274]	90.54	0.88	0.06	0.88
	GT2FS [275]	90.65	0.89	0.04	0.87
	BPNN [276]	89.82	0.86	0.08	0.82
	CNN [277]	92.26	0.92	0.04	0.89
	`STFT` [278] + CNN	94.32	0.94	0.05	0.90
	`DWT` [279] + CNN	94.75	0.95	0.04	0.90
	CSP [38] + CNN	**95.58**	**0.96**	**0.05**	**0.91**
P300	SWLDA [44]	90.23	0.84	0.07	0.83
	LSVM [272]	90.81	0.86	0.05	0.86
	KSVM-RBF Kernel [273]	92.56	0.90	0.05	0.88
	BPNN [276]	89.80	0.84	0.04	0.86
	CNN [277]	93.95	0.90	0.04	0.90
	`ICA` [280] + CNN	95.12	0.91	0.04	0.90
	`MRMR` [281] + CNN	94.05	0.90	0.03	0.89
	PCA [262] + CNN	**96.30**	**0.91**	**0.03**	**0.90**

data-sets based on the performance on that data-set. The values of χ_F^2 is obtained separately for each category of signal (MI and P300) and compared with the critical value of the χ_F^2 ($\alpha = 0.95$). If the obtained value crosses the critical value, we conclude that a significant difference exists between the performance of the classifiers and the classifiers can be ranked as per the cumulative rank sum. The classifier having the lowest cumulative sum is considered as the best-performing classifier.

MI classifier validation: During the MI classification process, performance of the proposed classifier is compared with eight other classifiers, hence we consider n=9 and L=10 in this case. The statistical test is carried out in two phases, in the first phase we ranked the performance of the classifiers based on accuracy, and in the second phase we ranked them based on kappa score. The cumulative sum of the ranks are obtained and put into (4.17) separately for two cases and in each case the obtained χ_F^2 value exceeds the critical value. Detailed results are given in Table 4.3.

P300 classifier validation: Performance of the proposed P300 classifier is evaluated over 10 data-sets (L=10) and compared with seven other classifiers (n=8). The statistical test is carried out in the same manner as described above. The result is given in Table 4.3. It is evident from the result that obtained χ_F^2 value exceeds the critical value in each case.

As the obtained χ_F^2 value exceeds the corresponding critical value in every cases, we conclude that the null hypothesis is rejected in each case. Hence, the performance of the classifiers can be evaluated by their cumulative ranks and the classifier with the lowest rank has the best performance.

Table 4.2: Performance of the Proposed EEG Classifier Algorithms in Multiple Runs for Different Subjects

	Metric	**Sub 1**	**Sub 2**	**Sub 3**	**Sub 4**	**Sub 5**	**Sub 6**	**Sub 7**	**Sub 8**	**Sub 9**	**Sub 10**	**Avg**
MI Classifier CSP + CNN	CA(%)	96.75 (±1.31)	96.10 (±1.37)	94.05 (±2.21)	97.05 (±1.22)	92.85 (±0.85)	94.48 (±1.30)	95.90 (±2.28)	95.25 (±0.81)	97.59 (±0.32)	95.92 (±0.62)	95.58
	TPR	0.97	0.96	0.94	0.98	0.92	0.95	0.97	0.98	0.98	0.94	0.96
	FPR	0.03	0.04	0.06	0.04	0.06	0.06	0.05	0.07	0.03	0.03	0.05
	κ	0.93	0.92	0.88	0.94	0.86	0.89	0.92	0.90	0.95	0.92	0.91
	Time(s)	0.591	0.594	0.564	0.570	0.544	0.585	0.580	0.576	0.590	0.568	0.576
P300 Classifier PCA + CNN	CA(%)	95.0 (±0.21)	97.58 (±0.15)	95.8 (±0.44)	96.3 (±0.31)	98.1 (±0.25)	96.9 (±0.12)	94.7 (±0.25)	97.8 (±0.29)	93.9 (±0.40)	97.4 (±0.30)	96.3
	TPR	0.92	0.96	0.89	0.88	0.94	0.90	0.89	0.95	0.83	0.92	0.91
	FPR	0.04	0.02	0.03	0.02	0.01	0.03	0.04	0.02	0.05	0.01	0.03
	κ	0.82	0.94	0.89	0.94	0.98	0.90	0.83	0.96	0.80	0.96	0.90
	Time(s)	0.121	0.186	0.152	0.162	0.140	0.180	0.156	0.126	0.134	0.160	0.151

Table 4.3: Results of Friedman Statistical Test

Category	Parameter	χ_F^2 value obtained from test	Critical χ_F^2 Value	Null Hypothesis Accepted/ Rejected
MI	Accuracy	71.52	16.91	Rejected
	Kappa	61.66		Rejected
P300	Accuracy	65.13	15.50	Rejected
	Kappa	33.03		Rejected

In the MI classification process, our proposed classifier achieved the lowest cumulative rank of 13 for both the accuracy and kappa score. In the case of the P300 classification process, our proposed classifier got the lowest cumulative ranks of 13 and 12 for accuracy and kappa score respectively. Hence, in each case, the proposed classifier performs best among others.

4.7.3 Performance of Grasp Prediction Network

In order to evaluate the performance of our model on the Multi-Object Grasping Dataset for object overlapping scenes, we need to take into account both object detection and grasp rectangle regression performances, since object-grasp affiliation requires accurate classification and localisation of the objects in the image. For our task, we use an mAP based metric called mAPg defined in [263, 264, 265]. A detected object-grasp pair is labeled successful if:

1. the object is classified correctly and the predicted object bounding box has an IOU higher than 0.5 with the ground truth bounding box

2. the detected grasp is labeled as a good grasp according to the rectangular metric defined in [254], subject to the following criteria:

   - the difference between the predicted grasp angle (orientation) and ground truth grasp angle is less than 30.
   - the Jaccard Index(J) between ground truth grasp rectangle (g) and predicted grasp rectangle (G), as defined below, is more than 0.25.

$$J(G,g) = \frac{(g \cap G)}{(g \cup G)} \tag{4.18}$$

1. **Performance on Multi grasp dataset:** The previous works [263, 264, 265] on simultaneous object detection and grasp prediction have not made their code or architectural details public, ruling out any possibility of re-creation. While they have reported mAP scores on the VMRD dataset defined in [282], the absence of depth data prevents us from evaluating our network on VMRD. So, in order to provide proper context to the performance of our network in overlapping object scenes, we select a combination of a state-of-the-art object detector, Faster-RCNN [111] and two state-of-the-art grasp detection models,

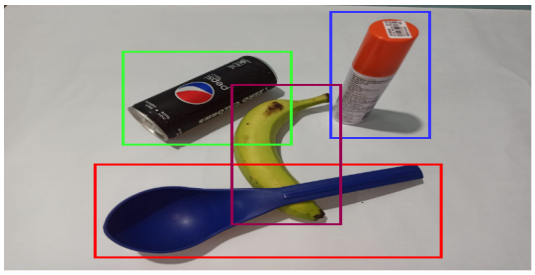

Figure 4.19: Example image 1 with predicted bounding boxes.

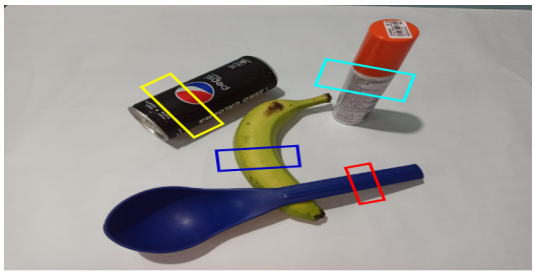

Figure 4.20: Example image 1 with predicted grasp on each corresponding object.

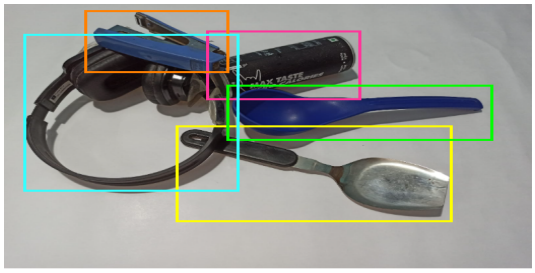

Figure 4.21: Example image 2 with predicted bounding boxes.

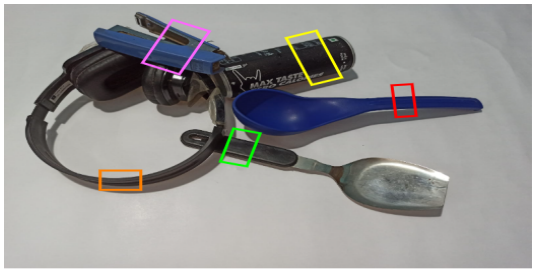

Figure 4.22: Example image 2 with predicted grasp on each corresponding object.

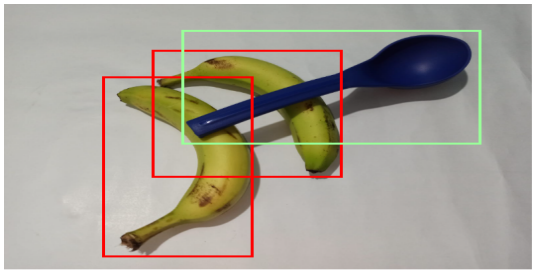

Figure 4.23: Example image 3 with predicted bounding boxes.

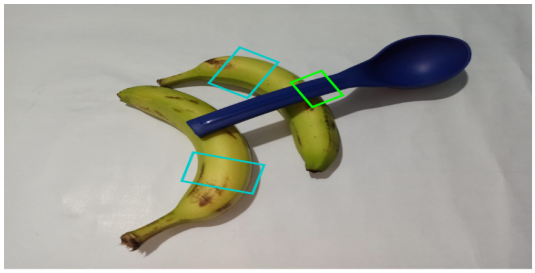

Figure 4.24: Example image 3 with predicted grasp on each corresponding object.

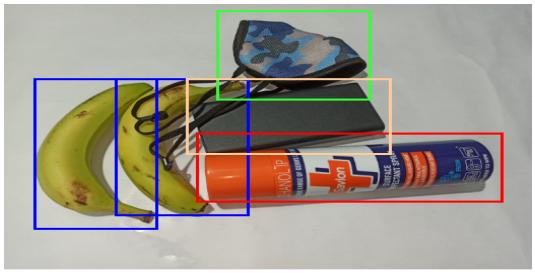

Figure 4.25: Example image 4 with predicted bounding boxes.

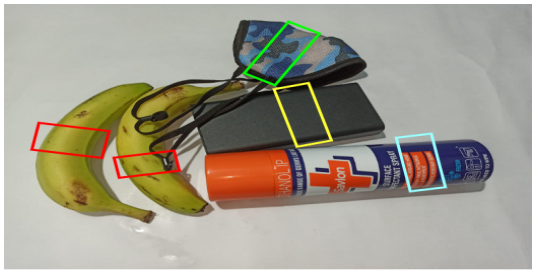

Figure 4.26: Example image 4 with predicted grasp on each corresponding object.

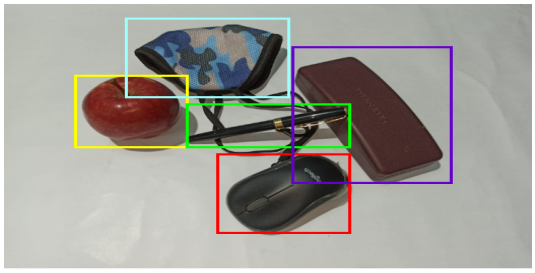

Figure 4.27: Example image 5 with predicted bounding boxes.

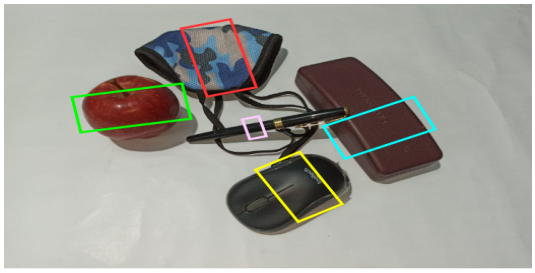

Figure 4.28: Example image 6 with predicted grasp on each corresponding object.

GR-Convnet [259] and FCGN [260] including all the architecture variations of the latter. Since these combined networks have no implicit grasp-object affiliation, the grasp rectangle with a confidence score higher than 0.25 and centre closest to the object bounding box center is associated with each detected object. Since our dataset contains RGB-D images, we evaluate the FCGN model, equipped for 3 channel inputs on both RGB and RGD modalities while the GR-Convnet is evaluated on RGB-D as intended by its authors.

Table 4.4 demonstrates that our model outperforms the other models by a large margin with an mAP score of 80.4%. Although the GR-ConvNet is shown to be quite effective at handling cluttered object scenes, our network still beats it in terms of performance. The improvement can be explained in part by the increased effectiveness of our model in dealing with the partial occlusion of objects in case of overlapping layouts, and in part by the simplistic object-grasp affiliation scheme used in the absence of an implicit association in case of the combination models. While FCGN has a higher mAP for RGB input compared to RGD, the mAP of the FCGN model (69.5%) has also increased from that reported by [263] on VMRD (54.5%). The reason for this increase may be attributed to the fact that the overlapping object layouts in our Multi grasp dataset are relatively less complex than that in VMRD due to our precondition for object visibility as stated in Section 4.2. An execution speed of 11.1 fps for our model is more than sufficient for user convenience. The improvement in speed over the FCGN model, in spite of having a deeper feature extractor, can be explained by our choice to generate only object proposals instead of predicting multiple object and grasp candidates. Performance of the grasp prediction of the OOGNet is represented with few example images in Fig.4.19 to Fig.4.28.

Table 4.4: Evaluation on Multi-Grasp Dataset

Algorithms	mAPg (%)	Speed (fps)
Faster-RCNN [111] (RGB) + GR-ConvNet [259] (RGB-D)	72.1	**46.9**
Faster-RCNN [111] (RGB) + (ResNet-50) FCGN [260] (RGB)	64.5	11.9
Faster-RCNN [111] (RGB)+ (ResNet-50) FCGN [260] (RGD)	63.3	11.9
Faster-RCNN [111] (RGB) + (ResNet-101) FCGN [260] (RGB)	69.5	10.2
Faster-RCNN [111] (RGB) + (ResNet-101) FCGN [260] (RGD)	68.2	10.2
OOGNet (RGB-D)	**80.4**	11.1

2. **Physical Evaluation:** In order to ascertain how well the grasp predictions translate to successful physical grasps in real world, we perform extensive experiments. In the experimental set-up, the robot arm is positioned a fixed distance above a plane surface containing either a single object or multiple objects in a variety of overlapping layouts, with the Kinect mounted on its 5th axis and tilted downwards. Our network takes an RGB-D image of the top view of the objects as input from the Kinect and generates a class prediction, bounding box, and grasp rectangle for each object in the image. For a multi-object layout, a particular object is selected to be grasped. The predicted grasp rectangle is converted to a gripper pose; the arm approaches and grasps the target object as described in [255] and

Table 4.5: Results of Physical Grasping Experiments

	Grasp Prediction Success rate		**Grasp Execution Success rate**	
	Single	**Multiple**	**Single**	**Multiple**
Banana	10/10	10/10	10/10	10/10
Headphone	10/10	10/10	9/10	9/10
Can	10/10	10/10	10/10	10/10
Stapler	10/10	10/10	10/10	10/10
Spectacle	10/10	10/10	10/10	10/10
Spoon	10/10	9/10	10/10	9/10
Box	10/10	10/10	10/10	10/10
Mask	9/10	9/10	9/10	8/10
Apple	10/10	10/10	10/10	10/10
Mouse	10/10	10/10	10/10	10/10
Torch	9/10	9/10	9/10	10/10
Glue Stick	10/10	10/10	10/10	10/10
Mobile Charger	10/10	9/10	10/10	9/10
Battery Box	10/10	10/10	10/10	10/10
Total	**98.6%**	**97.1%**	**97.8%**	**96.4%**

[254]. The experiment is performed on 10 different objects, with 25 trials for grasping each object in both single and multi-object overlapping layouts.

Table 4.5 shows the success rates for both the grasp prediction and execution for each object in single and Multi-object scenes in a benchmark scale of 10 as used in [52][55] . For single object cases, our model reached a 98.6% prediction success rate and a 97.8% success rate for physical grasping over all objects. On the other hand for Multi-object scenes, our network achieved a staggering 96.4% success rate for physical grasping and a 97.1% success rate for prediction. The results are calculated over 10 trials chosen randomly from the original 25 trials, to remove any biases present in the manual arrangement or positioning of the objects in single and multi-object settings. This highlights the effectiveness and reliability of our proposed network in performing real-world grasping tasks for a variety of object arrangements.

Table 4.6: Performance Comparison the Proposed System with Existing Hybrid Closed Loop BCI Schemes

Performance Metric	**MI+ErrP** [34]	**SSVEP+MI+P300** [33]	**Proposed Method**
Success rate (%)	85.6	90.2	**93.4**
Steady state error (%)	2.1	0.2	**0.05**
Settling Time (s)	31	24	**15.92**
Peak Overshoot (%)	4.9	4.2	**0**

Table 4.7: Online performance results

Sub	Method 1 [34]		Method 2 [33]		Proposed Method	
	Acc_{BCI}	Acc_{Sys}	Acc_{BCI}	Acc_{Sys}	Acc_{BCI}	Acc_{Sys}
S1	79.16	68.75	89.58	83.33	85.41	85.41
S2	85.41	79.16	81.25	81.25	91.66	87.5
S3	83.33	72.91	85.41	83.33	87.50	85.41
S4	85.41	79.16	87.50	79.16	91.66	89.58
S5	85.41	72.91	83.33	81.25	93.75	91.66
S6	87.50	79.16	75.00	72.91	95.83	91.66
S7	79.16	68.75	62.50	58.33	85.41	83.33
S8	83.33	72.91	81.25	75.00	89.58	87.50
S9	75.00	66.66	72.91	70.83	83.33	79.16
S10	81.25	70.83	85.41	79.16	93.75	89.58
Avg	82.50	73.12	80.41	76.45	89.79	87.08

Table 4.8: Comparison between different modules of the proposed system

Proposed Module	Average Execution Time (s)	Average Success Rate (%)	Human Involvement
Object Localization	7.18	94.1	Yes
Object Selection	2.85	95.0	Yes
Automatic Positioning	3.22	99.8	No
Grasping	2.64	96.2	No
Overall System Performance	15.92	93.4	-

4.7.4 System Performance

The overall position control performance of the BCI system is assessed here using few popular metrics arrived from control system literature. The metrics viz. success rate, settling time, peak overshoot, and steady-state error are considered to evaluate the system performance. Formal definitions of the metrics are given below.

Success rate: It expresses the number of successful attempts out of the total attempts made by the robot to reach the desired object. An attempt is regarded as successful only when the robot is able to grasp the desired object properly.

Steady state error: It indicates the maximum positional deviation of the robot end effector from the desired position in the infinite time range.

Settling Time: Time taken by the system to reach and stay within 2% of steady-state position.

Peak Overshoot: The maximum deviation of the response from its desired position. It is expressed as a percentage change from its final response.

Performance of the overall system is given in Table 4.6 and the online performance is provided in Table 4.7 . In both cases, the result is also compared with the performance of the two recent state-of-the-art works [207][19] which fall under the category of hybrid closed loop BCI and employs manual trajectory planning. Table 4.6 reports the overall success rate of the system by counting the number of times the subject is able to reach the desired position however the subjects are allowed to retake their decision if their intent is miss-classified at any stage. Table 4.6 focuses more on how the proposed control and planning method affects the overall system performance. On the other hand, Table 4.7 is obtained by following an online protocol that rejects the entire trial if miss-classification occurs at any stage of a trial and considers a trial to be successful only if all the stages of it are successful. The accuracy of the BCI and overall system are reported separately to provide insight to the readers on how the BCI performance affects the overall system accuracy.

Time taken by each module of the present work and their individual success rates are reported in Table 4.8. The last row denoting total system performance provides the average performance of the entire system when all the modules work together, which is not equal to the numerical average of each module. The above table also indicates the human involvement in each of the modules. Hence the time taken by the first two modules is greatly affected by human behavior whereas the time taken by the last two modules is affected by the velocity of the robot arm and the shape of the object selected by the subject. Sub-components of the system that govern the real-time behavior of the system are described below along with the execution time. Object localization module includes detection of Motor signals that require approximately 0.38 s including signal acquisition for 0.3s and classification time of 0.08s. The robot actuation time is 0.01s. The rest of the time is accounted for the control of the robot arm by the human subject to select the desired FOV. Next, the object selection step aims to choose the target object when multiple objects are present in the FOV. This includes object detection using Mask RCNN (0.1s), Centroid calculation (<0.01s) and P300 detection (0.58s= signal acquisition for 0.5s+classification time 0.08s). The rest of the time is taken by the human subject to decide which object he/she wants to choose. The Positioning step aims to estimate the 3D coordinates of the target object using a 2-D spatial location and a 2-D depth map. Co-ordinate estimation takes an average of 0.05s. Reaching the estimated position by the robot arm depends on the end effector velocity and the distance between the present and estimated position. Lastly, the final gripper configuration is estimated using our proposed OOGNet architecture. This module takes approximately 0.1s (11fps) and gripper actuation time is 0.01s The total gripping time depends on the size and shape of the object selected by the human.

It is apparent from Table 4.6 that the success rate of the proposed method is increased in a significant margin of 3.2% and from the traditional BCI-based success rate reported previously [207] . The overall success rate is found to be 93.4%. The steady-state error is also drastically reduced to 0.05% along with the settling time which is further reduced to 15.92s. The proposed method shows no overshoot or undershoots in either of the experiment due to the over-damped

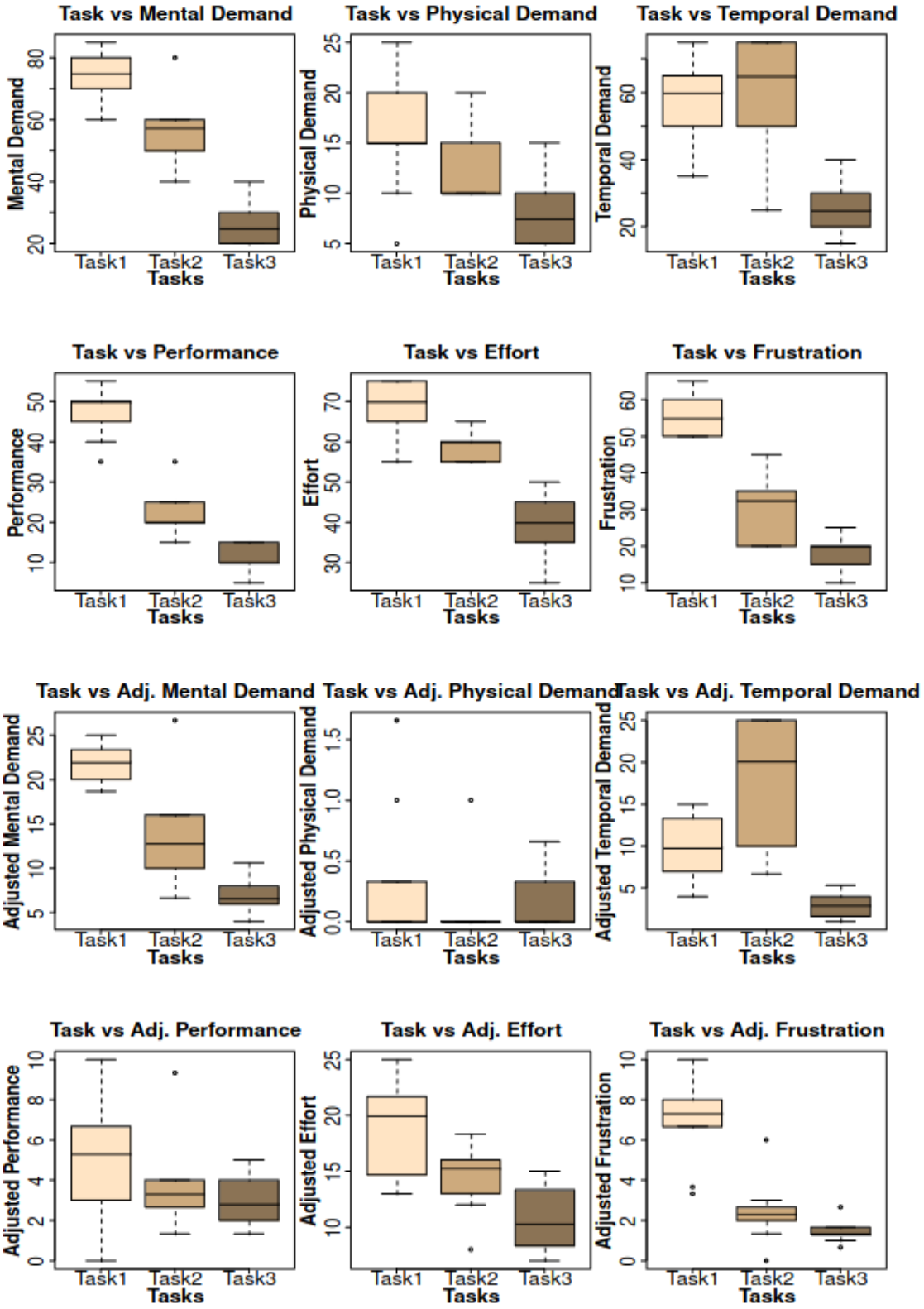

Figure 4.29: Box plot of sub-scales of the NASA-TLX study reported by ten participating subjects. The upper row represents the raw TLX scores whereas the lower row represents the adjusted TLX scores.

response of the robot end effector. The absence of human involvement in the gripper positioning phase and invoking autonomous positioning modules have eliminated the oscillation of the robot end-effector around the targeted object, which is otherwise reported in the existing literature. A similar trend is seen in online protocol results reported in Table 4.7 where the best overall system accuracy and BCI accuracy are found to be 87.08% and 89.79% respectively for the proposed method. Such an increase in BCI and overall accuracy may be attributed to the fact that the proposed method uses minimal human intervention hence minimizing the error that may arise from BCI decoding performance. The judicious selection of autonomous positioning and grasping strategy also contributed to making the system more robust and highly accurate compared to other methods reported in the table.

As per Table 4.8, the average success rate of the individual modules are found to be 94.1%, 95.0%,99.8% and 96.2% for Object localization, Object selection, Automatic positioning and Grasping phase respectively.It is also noticed that when the subject performs each module consecutively in a single run, the overall success rate of the system slightly reduces to 93.4% which is less than the numerical average of individual success rates.

Removing human interaction from the end-effector positioning and grasping phase has a significant effect on reducing the workload of the subject during real-time operation. The complex planning procedure of aligning the robot end effector with the desired object imposes a heavy workload on the subject. It also requires a significant amount of subject training and most of the novice subjects are not able to do it with the required accuracy. Experiments [19][207] involving such complex planning procedures are replicated in the laboratory environment and the workload of the subject while operating under those schemes is compared with the present proposed method. The main motivation behind providing this comparison study is to indicate the quantitative difference in workload associated with pure cognitive control based state-of-the art BCI approaches(the subject has to plan the entire robot trajectory for reaching and grasping) and our proposed scheme (shared control-based approach where reaching and grasping phase were made autonomous, minimizing overall human intervention).

Overall Workload assessment: Workload of the participating subjects was analyzed using the NASA-TLX questionnaire survey developed by NASA Ames Research Center that allows to asses the workload of the subjects operating various human-machine systems[251]. It assesses the workload using a multidimensional rating system with six sub-scales: Mental Demands, Physical Demands, Temporal Demands, Performance, Effort, and Frustration [252][130]. Each sub-scale is divided into 20 equal intervals which represents the score 0-100. Subjects provide ratings over each sub-scale for the task they were assigned. Here three BCI systems were compared, hence the subject provided the rating for three tasks. Once the subject finishes the rating, 15 pairwise comparisons between the sub-scales are presented to them, where subjects need to choose the sub-scale contributed most to their workload. The weight of a sub-scale is determined by the number of times it is chosen by the subject during a pairwise comparison task. The overall score of the test is found by computing the weighted average of the sub-scales with the weights determined above.

Here 10 participating subjects provided the rating for six sub-scales for each of the three BCI

tasks. Hence a total of 180 responses (10 x 6 x 3)were recorded. Average ratings of each BCI tasks for the six sub-scales are found by averaging the response over ten subjects. The result is shown in Fig 4.29. The adjusted rating of each sub-scale is also reported in the above figure. RAW TLX scores reported in the first row of the figure reveal that Task 1 [19]imposed the highest mental load and physical load on the subject, whereas Task 2[207] imposes the highest temporal load. Task 3 (proposed strategy) has been the lowest in all of the above categories and also demands the least effort from the subject to operate it. Task 3 is also found to have the highest performance rating and lowest frustration rating. Adjusted TLX scores show a similar pattern except for physical demand where it is found to be negligible in all three tasks. The overall adjusted TLX scores of Task1 and Task 2 are found to be $62.49 \pm 4.59 (mean \pm std)$ and 51.43 ± 5.58 respectively whereas the overall score of Task3 is found to be 23.99 ± 6.80 imposing a least cognitive load on the participating subjects compared to the other manual trajectory planning based state-of-the-art BCI robot manipulation techniques.

4.8 Conclusion

The main motivation of the present work was to relieve the subjects from the manual complex trajectory planning of the robot arm in a BCI-based robot control scheme. Complex trajectory planning is mainly involved in object reaching and grasping tasks, which are made autonomous in the present scheme. The idea facilitates the precise grasping of any mentally selected object without any human intervention hence reducing the cognitive load of the subject drastically. The chapter also proposed a CNN-based novel robotic grasp detection network to predict the accurate grasp in real time. The proposed network is able to work on overlapping scenes and uses simultaneous object and grasp detection. The overall performance of the BCI system is greatly improved from the recent state of the art where trajectory planning is entirely done by human subjects. As an example steady-state error of the system is reduced to 0.05% and the settling time is reduced to 15.92s. The results are substantiated by providing a comparison of the cognitive load of participating subjects for the proposed scheme and other recent BCI schemes. It was evident from the comparison that the present scheme imposes the least cognitive load on the subject and hence more suitable than the scheme involving manual trajectory planning. However, there exists an ample scope to further reduce the cognitive load of the subject by suitably predicting the human behavior of selecting any object at any stage of operation and assisting the human with autonomous navigational commands to reach the desired object. Such a learning mechanism will reduce the need for multiple P300 generation that is used here for selecting the desired object. Such a scheme uses fewer mental commands hence decreasing the cognitive load and simultaneously increasing the real-time accuracy of the system.

Chapter 5

Conclusions and Future Research Directions

This chapter aims at self-reviewing the targeted contributions of the individual chapters and also motivations of the entire work undertaken in the thesis. The pros and cons of the proposed techniques are reviewed, and possible techniques to be adopted to overcome the limitations are also outlined. Scope of extension of the works undertaken is also narrated here in detail.

5.1 Self-Review of the Thesis

The thesis proposes 3 distinct approaches for position control of a multi-link robot arm for possible rehabilitative applications of patients suffering from neuro-motor disabilities. Four distinct brain signals, including ERD/ERS, ErrP, P300, and SSVEP are utilized to generate necessary control commands for external activation of motors, positional overshoot detection, target object selection, and link selection of the robot arm.

Chapter 2 is concerned with an important problem of 3-dimensional position control of an end-effecter by controlling the position of individual links of a robot arm. The link selection here can be done randomly as desired by the human subject participating in the experiment. Link selection is here performed by acquiring the SSVEP response of the brain to different flickering frequency signal sources attached with the individual link. Once a specific link is selected for position control, the link actuation is accomplished by one brain signal, called ERD/ERS (or in general MI). The response to the mental desire for movement (or motor imagery (MI)) is translated to motion-related commands to undertake the initial clockwise/counter-clockwise motion of the selected link with a fixed pre-defined velocity. After the selected link crosses the desired target position, an ErrP and/or P300 are liberated indicating the possible positional error (or oddball situation). The BCI system on detecting the zero-crossing in positional error, commands the robotic link to reverse the direction and reduce its speed until positional error is less than a pre0defined threshold. The merit of the above scheme includes;

1. Zero/Low steady-state error.
2. Small Overshoot.
3. High Stability margin.
4. Giving user the choice to select the number of zero-crossing before terminating the link motion.
5. Robust design

The system introduced above utilizes only zero-crossing positions of the link/end-effecter, but cannot measure the magnitude of positional error at any instant of time. This limitation has been overcome in the second scheme introduced in Chapter 3.

The second scheme introduced in Chapter 3 is concerned with fuzzy control of the robot arm. The attractive part of this design lies in sensing the magnitude of positional overshoot or undershoot with respect to the final target position. An exact measurement of positional overshoot or undershoot is not possible, an approximate measurement is done and expressed as a fuzzy quantifier such as SMALL, LARGE or MEDIUM error. The fuzzified error is then utilized to determine the fuzzy angular displacement of the selected robotic link. The experiment uses a few landmark positions around the target and flickering sources were mounted over the landmark positions. The subject initiates the robot link movement with motor imagery brain commands and whenever the link crosses the desired target an ErrP signal is liberated from the brain to indicate the occurrence of error, which in turn freezes the link movement. Once the movement

is stopped, the subject focuses on the flickering LED mounted at the nearest landmark position to generate SSVEP response in their brain. The system identifies the magnitude of the error and the relative position of the end-effector with respect to the target position by decoding the corresponding SSVEP signal. The error magnitude and relative position of the end-effecter are utilized by the previously defined fuzzy controller to take the corrective action to bring back the end-effecter at the target position. The scheme proposed in Chapter 3 overcomes the limitations of Chapter 2 but it has a limited scope of application as it is designed for a fixed target position.

Chapter 4 overcomes the limitations of Chapter 3 as it takes care of user-fixed targets by camera vision. The scheme enables the subjects to select their desired object by using their mental commands. Additionally, the scheme made the reaching and grasping phase of the robot arm autonomous, hence the subjects were relieved from creating the trajectory of the robot arm with their cognitive effort. Such an autonomous movement of the robot also facilitates high precision control which couldn't be achieved only with human effort.
The scheme employs a depth-sensing camera mounted at the end of a 6-DOF robot arm to capture the surrounding environment of the robot. The user was given the liberty to change the camera position using their mental commands to bring the desired object into its field of view (FOV) by observing the camera feedback on a digital monitor placed near them. The subject used Motor Imagery signal to rotate the 1st and 5th link of the robot, which in turn changes the camera position to cover the entire 3D surrounding environment. Once the target object is visible on the monitor, the subject stops issuing mental commands (MI). The target object (in the presence of other objects) visible on the screen was selected using the P300 brain command. Now the vision-based auxiliary controller takes charge and guides the robot to reach the target position and grasp the desired object with high accuracy. The proposed control policy also helped to reduce the cognitive load of the participating subject to a great extent.

5.2 Possible Extension of the Current Research Outcome

The brain-inspired controller design presented in Chapter 2 is formulated as a linear system, and so is realized in s-domain using the transfer function approach. Such linear approximation of the controller has limitations from the point of view of stability analysis. There exists ample scope of extension of the brain-inspired controller model as a nonlinear system. A brief outline of the possible extension of the model is given below for the sake of the convenience of the readers and the completeness of the thesis.

Let θ_r be the reference angle and θ_c be the desired angle of motion of the target. Let the angular positional error signal $e = \theta_r - \theta_c$. We define 2 alternative non-linear models for speed-setting of the mentally actuated controller. Fig. 5.1 and 5.2 provide the schematic of controller dynamics and the closed-loop control structure. The control strategy can be expressed as below;

$$\dot{\theta_m} = K; e(t) \geq \delta \tag{5.1}$$

$$= -K; e(t) \leq \delta \tag{5.2}$$

$$= 0; -\delta < e(t) < \delta \tag{5.3}$$

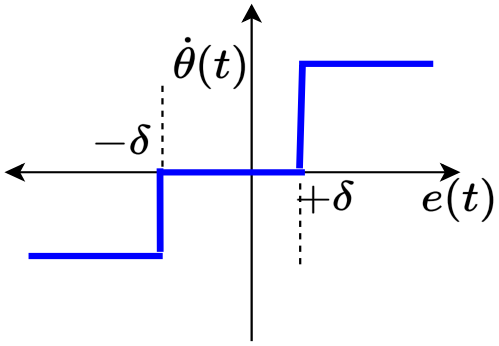

Figure 5.1: Schematic of Controller Dynamics 1

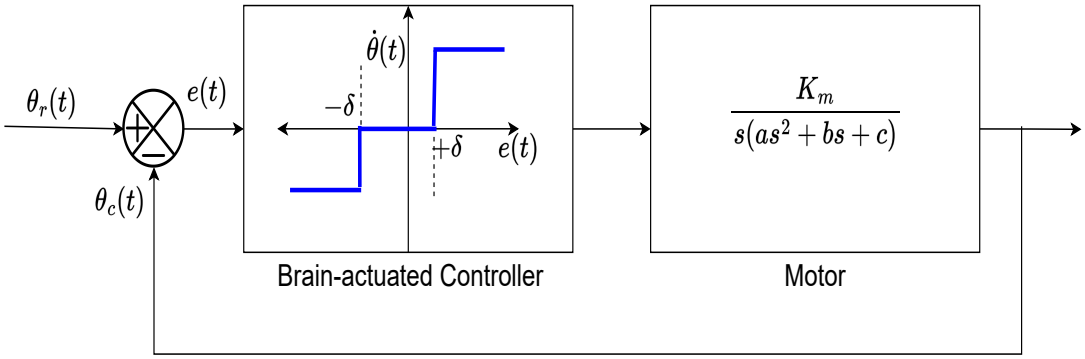

Figure 5.2: Closed-loop control structure with Controller 1

Alternatively, we can model controller speed by the following formulation. Figure 2 (a) and (b) provide an alternative formulation of the controller and the overall control loop.

$$\dot{\theta_m} = \alpha e(t); |e(t)| \geq \delta \tag{5.4}$$

$$= -K; e(t) \leq \delta \tag{5.5}$$

$$= K; e(t) \geq \delta \tag{5.6}$$

It is apparent from Fig. 5.2 and 5.4 that the stability of the closed-loop control would be greatly influenced by the controller dynamics. Several methods of stability analysis, including phase-trajectory analysis, Describing function approach, and Lyapunov stability approach are applicable to the present application.

The fuzzy logic approach for controller design is realized with Mamdani-type fuzzy reasoning. There is ample scope for controller modeling using Takagi-Sugeno approach. The future work will focus on Takagi-Sugeno modeling of the fuzzy controller. The merit of the Takagi-Sugeno controller lies in its inherent stability analysis, which, however, is not available for the Mamdani approach.

Secondly, the fuzzy logic approach proposed does not have a provision for rule adaptation. This requires provisions for autonomous learning when existing rules do not fire for lack of instantiation by the existing database.

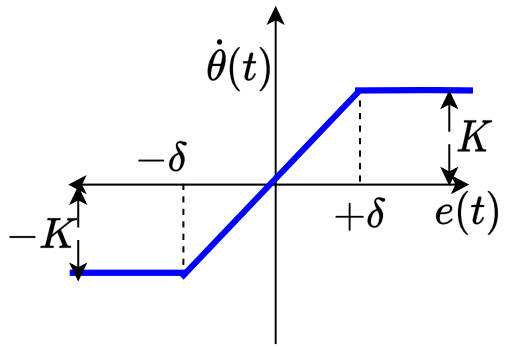

Figure 5.3: Schematic of Controller Dynamics 2

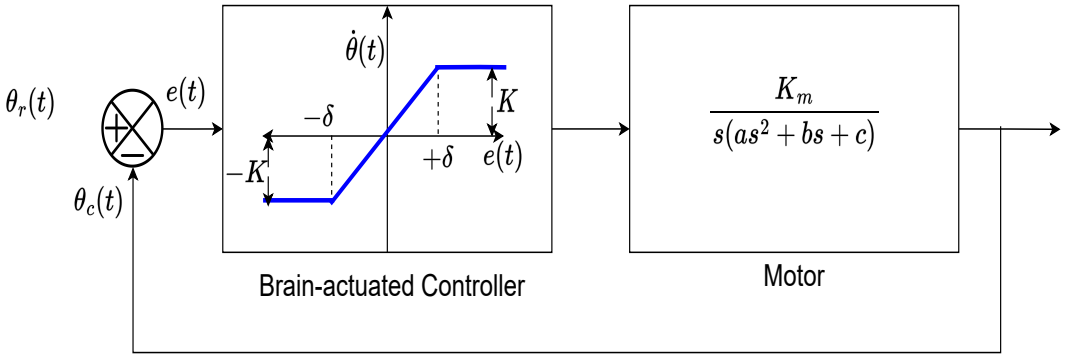

Figure 5.4: Closed-loop control structure with Controller 2

Thirdly, the fuzzy control strategy requires labeling the workspace of the robot to approximately determine the magnitude of the positional error signal. Such labeling is not feasible for a big workspace. So, alternative automation needs to be designed to determine the magnitude of positional error.

The third control strategy presented in Chapter 4 undertakes the control action in 2 phases. In the first phase, the BCI-system is utilized to make the system identify the target object, and then employ the auxiliary control strategy to reach the target position to grasp the desired object. Here, the human subject participating in the BCI-loop loses control in the second phase, as the auxiliary controller is given full autonomy to execute the control actions for reaching and grasping. The future extension of the work may offer more controlling power to the human subject for possibly better performance of the control system in accordance with the planning and decisions generated by the experimental subject. Conveying the decision of the subject to the auxiliary controller as and when needed, however, needs to be addressed in future research.

References

[1] Eric R Kandel, James H Schwartz, Thomas M Jessell, Steven Siegelbaum, A James Hudspeth, Sarah Mack, et al. *Principles of neural science*, volume 4. McGraw-hill New York, 2000.

[2] Endel Tulving and Fergus IM Craik. *The Oxford handbook of memory*. Oxford University Press, 2000.

[3] Michael A Arbib. *The handbook of brain theory and neural networks*. MIT press, 2003.

[4] Bernhard Obermaier, Gemot R Muller, and Gert Pfurtscheller. " virtual keyboard" controlled by spontaneous eeg activity. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 11(4):422–426, 2003.

[5] Reinhold Scherer, GR Muller, Christa Neuper, Bernhard Graimann, and Gert Pfurtscheller. An asynchronously controlled eeg-based virtual keyboard: improvement of the spelling rate. *IEEE Transactions on Biomedical Engineering*, 51(6):979–984, 2004.

[6] Jinyi Long, Yuanqing Li, Hongtao Wang, Tianyou Yu, Jiahui Pan, and Feng Li. A hybrid brain computer interface to control the direction and speed of a simulated or real wheelchair. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 20(5):720–729, 2012.

[7] Wenchang Zhang, Fuchun Sun, Chunfang Liu, Weihua Su, Chuanqi Tan, and Shaobo Liu. A hybrid eeg-based bci for robot grasp controlling. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3278–3283. IEEE, 2017.

[8] David Marshall, Damien Coyle, Shane Wilson, and Michael Callaghan. Games, gameplay, and bci: the state of the art. *IEEE Transactions on Computational Intelligence and AI in Games*, 5(2):82–99, 2013.

[9] Zhihua Wang, Yang Yu, Ming Xu, Yadong Liu, Erwei Yin, and Zongtan Zhou. Towards a hybrid bci gaming paradigm based on motor imagery and ssvep. *International Journal of Human–Computer Interaction*, 35(3):197–205, 2019.

[10] Gert Pfurtscheller, Christa Neuper, Christoph Guger, WAHW Harkam, Herbert Ramoser, Alois Schlogl, BAOB Obermaier, and MAPM Pregenzer. Current trends in graz brain-computer interface (bci) research. *IEEE transactions on rehabilitation engineering*, 8(2):216–219, 2000.

[11] Piotr Stawicki, Felix Gembler, and Ivan Volosyak. Driving a semiautonomous mobile robotic car controlled by an ssvep-based bci. *Computational intelligence and neuroscience*, 2016, 2016.

[12] Wendi Song, Xiangzhou Wang, Shuhua Zheng, and Yingzi Lin. Mobile robot control by bci based on motor imagery. In *2014 Sixth International Conference on Intelligent Human-Machine Systems and Cybernetics*, volume 2, pages 383–387. IEEE, 2014.

[13] Christoph Kapeller, Christoph Hintermüller, Mohammad Abu-Alqumsan, Robert Prückl, Angelika Peer, and Christoph Guger. A bci using vep for continuous control of a mobile robot. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 5254–5257. IEEE, 2013.

[14] Cong Wang, Bin Xia, Jie Li, Wenlu Yang, Alejandro Cardona Velez, Hong Yang, et al. Motor imagery bci-based robot arm system. In *2011 Seventh International Conference on Natural Computation*, volume 1, pages 181–184. IEEE, 2011.

[15] William J Gehring, Yanni Liu, Joseph M Orr, and Joshua Carp. The error-related negativity (ern/ne). *Oxford handbook of event-related potential components*, pages 231–291, 2012.

[16] Akshay Kumar, Lin Gao, Elena Pirogova, and Qiang Fang. A review of error-related potential-based brain–computer interfaces for motor impaired people. *IEEE Access*, 7:142451–142466, 2019.

[17] Markus Ullsperger, Adrian G Fischer, Roland Nigbur, and Tanja Endrass. Neural mechanisms and temporal dynamics of performance monitoring. *Trends in cognitive sciences*, 18(5):259–267, 2014.

[18] E Donchin. Is the p300 component a measure of context updating? behavioral and brain sciennce. *Science*, 11:357–374, 1988.

[19] Saugat Bhattacharyya, Amit Konar, and DN Tibarewala. Motor imagery and error related potential induced position control of a robotic arm. *IEEE/CAA Journal of Automatica Sinica*, 4(4):639–650, 2017.

[20] Lidia Ghosh and Amit Konar. Eeg-induced adaptation of controller parameter for closed-loop position control of the end-effecter in a robot arm. In *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pages 1–7. IEEE, 2019.

[21] Jing Wei Tan and Yimesker Yihun. Application of forearm fmg signals in closed loop modality-matched sensory feedback stimulation. *Journal of Bionic Engineering*, 17:899–908, 2020.

[22] Kotaro Nishikawa, Masayuki Shakutsui, Kentaro Hirata, and Masahiro Takaiwa. Development of pneumatic bellows myoelectric hand with tactile feedback. *IEEJ Transactions on Electronics, Information and Systems*, 137(1):127–135, 2017.

[23] Sharlene N Flesher, Jennifer L Collinger, Stephen T Foldes, Jeffrey M Weiss, John E Downey, Elizabeth C Tyler-Kabara, Sliman J Bensmaia, Andrew B Schwartz, Michael L Boninger, and Robert A Gaunt. Intracortical microstimulation of human somatosensory cortex. *Science translational medicine*, 8(361):361ra141–361ra141, 2016.

[24] Edoardo D'Anna, Giacomo Valle, Alberto Mazzoni, Ivo Strauss, Francesco Iberite, Jérémy Patton, Francesco M Petrini, Stanisa Raspopovic, Giuseppe Granata, Riccardo Di Iorio, et al. A closed-loop hand prosthesis with simultaneous intraneural tactile and position feedback. *Science Robotics*, 4(27):eaau8892, 2019.

[25] Anwesha Khasnobish, Monalisa Pal, Dwaipayan Sardar, DN Tibarewala, and Amit Konar. Vibrotactile feedback for conveying object shape information as perceived by artificial sensing of robotic arm. *Cognitive neurodynamics*, 10:327–338, 2016.

[26] Henri J Nussbaumer and Henri J Nussbaumer. *The fast Fourier transform*. Springer, 1981.

[27] Shaima Miqdad Mohamed Najeeb, Haider Th Salim Al Rikabi, and Shaima Mohammed Ali. Finding the discriminative frequencies of motor electroencephalography signal using genetic algorithm. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 19(1):285–292, 2021.

[28] Richard Wesley Hamming. *Digital filters*. Courier Corporation, 1998.

[29] Louis Litwin. Fir and iir digital filters. *IEEE potentials*, 19(4):28–31, 2000.

[30] Christos Stergiadis, Vasiliki-Despoina Kostaridou, and Manousos A Klados. Which bss method separates better the eeg signals? a comparison of five different algorithms. *Biomedical Signal Processing and Control*, 72:103292, 2022.

[31] Filipa Campos Viola, Stefan Debener, Jeremy Thorne, and Till R Schneider. Using ica for the analysis of multi-channel eeg data. *Simultaneous EEG and fMRI: Recording, Analysis, and Application: Recording, Analysis, and Application*, pages 121–133, 2010.

[32] Xinyang Yu, Pharino Chum, and Kwee-Bo Sim. Analysis the effect of pca for feature reduction in non-stationary eeg based motor imagery of bci system. *Optik*, 125(3):1498–1502, 2014.

[33] Dennis J McFarland, Lynn M McCane, Stephen V David, and Jonathan R Wolpaw. Spatial filter selection for eeg-based communication. *Electroencephalography and clinical Neurophysiology*, 103(3):386–394, 1997.

[34] Ömer Türk, Mesut Şeker, Veysi Akpolat, and Mehmet Siraç Özerdem. Classification of mental task eeg records using hjorth parameters. In *2017 25th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4. IEEE, 2017.

[35] Carmen Vidaurre, A Schlogl, Rafael Cabeza, Reinhold Scherer, and Gert Pfurtscheller. A fully on-line adaptive bci. *IEEE Transactions on biomedical engineering*, 53(6):1214–1219, 2006.

[36] Imali Thanuja Hettiarachchi, Thanh Thi Nguyen, and Saeid Nahavandi. Motor imagery data classification for bci application using wavelet packet feature extraction. In *Neural Information Processing: 21st International Conference, ICONIP 2014, Kuching, Malaysia, November 3-6, 2014. Proceedings, Part III 21*, pages 519–526. Springer, 2014.

[37] Zhonglin Lin, Changshui Zhang, Wei Wu, and Xiaorong Gao. Frequency recognition based on canonical correlation analysis for ssvep-based bcis. *IEEE transactions on biomedical engineering*, 53(12):2610–2614, 2006.

[38] Fabien Lotte and Cuntai Guan. Regularizing common spatial patterns to improve bci designs: unified theory and new algorithms. *IEEE Transactions on biomedical Engineering*, 2010.

[39] Varun Bajaj and Ram Bilas Pachori. Classification of seizure and nonseizure eeg signals using empirical mode decomposition. *IEEE Transactions on Information Technology in Biomedicine*, 16(6):1135–1142, 2011.

[40] Fabien Lotte. A tutorial on eeg signal-processing techniques for mental-state recognition in brain–computer interfaces. *Guide to brain-computer music interfacing*, pages 133–161, 2014.

[41] Abdel-Ouahab Boudraa and Jean-Christophe Cexus. Emd-based signal filtering. *IEEE transactions on instrumentation and measurement*, 56(6):2196–2202, 2007.

[42] Bruce Thompson. Canonical correlation analysis. 2000.

[43] Eric W Sellers, Theresa M Vaughan, and Jonathan R Wolpaw. A brain-computer interface for long-term independent home use. *Amyotrophic lateral sclerosis*, 11(5):449–455, 2010.

[44] Garett D Johnson and Dean J Krusienski. Ensemble swlda classifiers for the p300 speller. In *International Conference on Human-Computer Interaction*, pages 551–557. Springer, 2009.

[45] Nanying Liang and Laurent Bougrain. Decoding finger flexion from band-specific ecog signals in humans. *Frontiers in neuroscience*, 6:91, 2012.

[46] Julio Ortega, Javier Asensio-Cubero, John Q Gan, and Andrés Ortiz. Classification of motor imagery tasks for bci with multiresolution analysis and multiobjective feature selection. *BioMedical Engineering OnLine*, 15:149–164, 2016.

[47] Rebeca Corralejo, Roberto Hornero, and Daniel Álvarez. Feature selection using a genetic algorithm in a motor imagery-based brain computer interface. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 7703–7706. IEEE, 2011.

[48] Rifai Chai, Yvonne Tran, Ganesh R Naik, Tuan N Nguyen, Sai Ho Ling, Ashley Craig, and Hung T Nguyen. Classification of eeg based-mental fatigue using principal component analysis and bayesian neural network. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 4654–4657. IEEE, 2016.

[49] Dong Kang and Luo Zhizeng. A method of denoising multi-channel eeg signals fast based on pca and debss algorithm. In *2012 International Conference on Computer Science and Electronics Engineering*, volume 3, pages 322–326. IEEE, 2012.

[50] Kai Keng Ang, Zheng Yang Chin, Chuanchu Wang, Cuntai Guan, and Haihong Zhang. Filter bank common spatial pattern algorithm on bci competition iv datasets 2a and 2b. *Frontiers in neuroscience*, 6:39, 2012.

[51] Mikito Ogino and Yasue Mitsukura. Portable drowsiness detection through use of a prefrontal single-channel electroencephalogram. *Sensors*, 18(12):4477, 2018.

[52] Adham Atyabi, Martin H Luerssen, and David MW Powers. Pso-based dimension reduction of eeg recordings: implications for subject transfer in bci. *Neurocomputing*, 119:319–331, 2013.

[53] Alberto Ortega, Juan José Escobar, Julio Ortega, Jesús González, Alfredo Alcayde, Jorge Munilla, and Miguel Damas. Performance study of ant colony optimization for feature selection in eeg classification. In *International Conference on Bioengineering and Biomedical Signal and Image Processing*, pages 323–336. Springer, 2021.

[54] Christian Mühl, Camille Jeunet, and Fabien Lotte. Eeg-based workload estimation across affective contexts. *Frontiers in neuroscience*, 8:114, 2014.

[55] Kenji Kira and Larry A Rendell. A practical approach to feature selection. In *Machine learning proceedings 1992*, pages 249–256. Elsevier, 1992.

[56] Saugat Bhattacharyya, Anwesha Khasnobish, Somsirsa Chatterjee, Amit Konar, and DN Tibarewala. Performance analysis of lda, qda and knn algorithms in left-right limb movement classification from eeg data. In *2010 International conference on systems in medicine and biology*, pages 126–131. IEEE, 2010.

[57] Abdulhamit Subasi and M Ismail Gursoy. Eeg signal classification using pca, ica, lda and support vector machines. *Expert systems with applications*, 37(12):8659–8666, 2010.

[58] Suresh Balakrishnama and Aravind Ganapathiraju. Linear discriminant analysis-a brief tutorial. *Institute for Signal and information Processing*, 18(1998):1–8, 1998.

[59] Alaa Tharwat. Linear vs. quadratic discriminant analysis classifier: a tutorial. *International Journal of Applied Pattern Recognition*, 3(2):145–180, 2016.

[60] Obed Carrera-Leon, Juan Manuel Ramirez, Vicente Alarcon-Aquino, Mary Baker, David D'Croz-Baron, and Pilar Gomez-Gil. A motor imagery bci experiment using wavelet analysis and spatial patterns feature extraction. In *2012 workshop on engineering applications*, pages 1–6. IEEE, 2012.

[61] Chivalai Temiyasathit et al. Increase performance of four-class classification for motor-imagery based brain-computer interface. In *2014 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pages 1–5. IEEE, 2014.

[62] Irina Rish et al. An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46, 2001.

[63] Hua Wang, Yanchun Zhang, et al. Detection of motor imagery eeg signals employing naïve bayes based learning process. *Measurement*, 86:148–158, 2016.

[64] Arnab Rakshit, Anwesha Khasnobish, and DN Tibarewala. A naïve bayesian approach to lower limb classification from eeg signals. In *2016 2nd International Conference on Control, Instrumentation, Energy & Communication (CIEC)*, pages 140–144. IEEE, 2016.

[65] MNAH Sha'Abani, N Fuad, Norezmi Jamal, and MF Ismail. knn and svm classification for eeg: a review. In *InECCE2019: Proceedings of the 5th International Conference on Electrical, Control & Computer Engineering, Kuantan, Pahang, Malaysia, 29th July 2019*, pages 555–565. Springer, 2020.

[66] Christopher M Bishop and Nasser M Nasrabadi. *Pattern recognition and machine learning*, volume 4. Springer, 2006.

[67] Mervyn VM Yeo, Xiaoping Li, Kaiquan Shen, and Einar PV Wilder-Smith. Can svm be used for automatic eeg detection of drowsiness during car driving? *Safety Science*, 47(1):115–124, 2009.

[68] Mohammad-Parsa Hosseini, Amin Hosseini, and Kiarash Ahi. A review on machine learning for eeg signal processing in bioengineering. *IEEE reviews in biomedical engineering*, 14:204–218, 2020.

[69] LR Quitadamo, F Cavrini, L Sbernini, F Riillo, L Bianchi, S Seri, and G Saggio. Support vector machines to detect physiological patterns for eeg and emg-based human–computer interaction: a review. *Journal of neural engineering*, 14(1):011001, 2017.

[70] Ashkan Yazdani, Touradj Ebrahimi, and Ulrich Hoffmann. Classification of eeg signals using dempster shafer theory and a k-nearest neighbor classifier. In *2009 4th International IEEE/EMBS Conference on Neural Engineering*, pages 327–330. IEEE, 2009.

[71] Björn O Peters, Gert Pfurtscheller, and Henrik Flyvbjerg. Mining multi-channel eeg for its information content: an ann-based method for a brain–computer interface. *Neural networks*, 11(7-8):1429–1433, 1998.

[72] M Serdar Bascil, Ahmet Y Tesneli, and Feyzullah Temurtas. Spectral feature extraction of eeg signals and pattern recognition during mental tasks of 2-d cursor movements for bci using svm and ann. *Australasian physical & engineering sciences in medicine*, 39:665–676, 2016.

[73] Advait Balaji, Aparajita Haldar, Keshav Patil, T Sai Ruthvik, CA Valliappan, Mayur Jartarkar, and Veeky Baths. Eeg-based classification of bilingual unspoken speech using ann. In *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1022–1025. IEEE, 2017.

[74] Fabien Lotte, Laurent Bougrain, Andrzej Cichocki, Maureen Clerc, Marco Congedo, Alain Rakotomamonjy, and Florian Yger. A review of classification algorithms for eeg-based brain–computer interfaces: a 10 year update. *Journal of neural engineering*, 15(3):031005, 2018.

[75] SJ Pan and Q Yang. A survey on transfer learning. ieee transaction on knowledge discovery and data engineering, 22 (10), 2010.

[76] Si Si, Dacheng Tao, and Bo Geng. Bregman divergence-based regularization for transfer subspace learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(7):929–942, 2009.

[77] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*, pages 2200–2207, 2013.

[78] Guangquan Liu, Dingguo Zhang, Jianjun Meng, Gan Huang, and Xiangyang Zhu. Unsupervised adaptation of electroencephalogram signal processing based on fuzzy c-means algorithm. *International Journal of Adaptive Control and Signal Processing*, 26(6):482–495, 2012.

[79] Alois Schlögl, Carmen Vidaurre, and Klaus-Robert Müller. Adaptive methods in bci research-an introductory tutorial. In *Brain-Computer Interfaces: Revolutionizing Human-Computer Interaction*, pages 331–355. Springer, 2010.

[80] Ji Won Yoon, Stephen J Roberts, Matt Dyson, and John Q Gan. Adaptive classification for brain computer interface systems using sequential monte carlo sampling. *Neural Networks*, 22(9):1286–1294, 2009.

[81] Julie Blumberg, Jorn Rickert, Stephan Waldert, Andreas Schulze-Bonhage, Ad Aertsen, and Carsten Mehring. Adaptive classification for brain computer interfaces. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 2536–2539. IEEE, 2007.

[82] Bashar Awwad Shiekh Hasan and John Q Gan. Hangman bci: An unsupervised adaptive self-paced brain–computer interface for playing games. *Computers in biology and medicine*, 42(5):598–606, 2012.

[83] Guangquan Liu, Gan Huang, Jianjun Meng, Dingguo Zhang, and Xiangyang Zhu. Improved gmm with parameter initialization for unsupervised adaptation of brain–computer interface. *International Journal for Numerical Methods in Biomedical Engineering*, 26(6):681–691, 2010.

[84] Alberto Llera, Vicenç Gómez, and Hilbert J Kappen. Adaptive multiclass classification for brain computer interfaces. *Neural computation*, 26(6):1108–1127, 2014.

[85] Fabien Lotte. Signal processing approaches to minimize or suppress calibration time in oscillatory activity-based brain–computer interfaces. *Proceedings of the IEEE*, 103(6):871–890, 2015.

[86] Carmen Vidaurre, A Schlogl, Rafael Cabeza, Reinhold Scherer, and Gert Pfurtscheller. Study of on-line adaptive discriminant analysis for eeg-based brain computer interfaces. *IEEE transactions on biomedical engineering*, 54(3):550–556, 2007.

[87] Mehrnaz Kh Hazrati and Abbas Erfanian. An online eeg-based brain–computer interface for controlling hand grasp using an adaptive probabilistic neural network. *Medical engineering & physics*, 32(7):730–739, 2010.

[88] DR Lowne, Stephen J Roberts, and Roman Garnett. Sequential non-stationary dynamic classification with sparse feedback. *Pattern Recognition*, 43(3):897–905, 2010.

[89] Fatih Özyurt, Türker Tuncer, Engin Avci, Mustafa Koç, and İhsan Serhatlioğlu. A novel liver image classification method using perceptual hash-based convolutional neural network. *Arabian Journal for Science and Engineering*, 44:3173–3182, 2019.

[90] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[91] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323. JMLR Workshop and Conference Proceedings, 2011.

[92] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[93] Muhammad Tariq Sadiq, Muhammad Zulkifal Aziz, Ahmad Almogren, Adnan Yousaf, Siuly Siuly, and Ateeq Ur Rehman. Exploiting pretrained cnn models for the development of an eeg-based robust bci framework. *Computers in Biology and Medicine*, 143:105242, 2022.

[94] Hubert Cecotti and Axel Graser. Convolutional neural networks for p300 detection with application to brain-computer interfaces. *IEEE transactions on pattern analysis and machine intelligence*, 33(3):433–445, 2010.

[95] Ran Manor and Amir B Geva. Convolutional neural network for multi-category rapid serial visual presentation bci. *Frontiers in computational neuroscience*, 9:146, 2015.

[96] No-Sang Kwak, Klaus-Robert Müller, and Seong-Whan Lee. A convolutional neural network for steady state visual evoked potential classification under ambulatory environment. *PloS one*, 12(2):e0172578, 2017.

[97] Yousef Rezaei Tabar and Ugur Halici. A novel deep learning approach for classification of eeg motor imagery signals. *Journal of neural engineering*, 14(1):016003, 2016.

[98] Robin Tibor Schirrmeister, Jost Tobias Springenberg, Lukas Dominique Josef Fiederer, Martin Glasstetter, Katharina Eggensperger, Michael Tangermann, Frank Hutter, Wolfram Burgard, and Tonio Ball. Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, 38(11):5391–5420, 2017.

[99] Vernon J Lawhern, Amelia J Solon, Nicholas R Waytowich, Stephen M Gordon, Chou P Hung, and Brent J Lance. Eegnet: a compact convolutional neural network for eeg-based brain–computer interfaces. *Journal of neural engineering*, 15(5):056013, 2018.

[100] Xilin Liu, Milin Zhang, Andrew G Richardson, Timothy H Lucas, and Jan Van der Spiegel. Design of a closed-loop, bidirectional brain machine interface system with energy efficient neural feature extraction and pid control. *IEEE transactions on biomedical circuits and systems*, 11(4):729–742, 2016.

[101] Carmen Vidaurre, Christian Klauer, Thomas Schauer, Ander Ramos-Murguialday, and Klaus-Robert Müller. Eeg-based bci for the linear control of an upper-limb neuroprosthesis. *Medical engineering & physics*, 38(11):1195–1204, 2016.

[102] Kristin M Quick, Jeffrey M Weiss, Francesco Clemente, Robert A Gaunt, and Jennifer L Collinger. Intracortical microstimulation feedback improves grasp force accuracy in a human using a brain-computer interface. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 3355–3358. IEEE, 2020.

[103] Ramin Pashaie, Ryan Baumgartner, Thomas J Richner, Sarah K Brodnick, Mehdi Azimipour, Kevin W Eliceiri, and Justin C Williams. Closed-loop optogenetic brain interface. *IEEE Transactions on Biomedical Engineering*, 62(10):2327–2337, 2015.

[104] Xiaoyan Deng, Zhu Liang Yu, Canguang Lin, Zhenghui Gu, and Yuanqing Li. A bayesian shared control approach for wheelchair robot with brain machine interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(1):328–338, 2019.

[105] Jingsheng Tang and Zongtan Zhou. A shared-control based bci system: For a robotic arm control. In *2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS)*, pages 1–5. IEEE, 2017.

[106] Zhichuan Tang, Lingtao Zhang, Xin Chen, Jichen Ying, Xinyang Wang, and Hang Wang. Wearable supernumerary robotic limb system using a hybrid control approach based on motor imagery and object detection. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:1298–1309, 2022.

[107] Yang Xu, Cheng Ding, Xiaokang Shu, Kai Gui, Yulia Bezsudnova, Xinjun Sheng, and Dingguo Zhang. Shared control of a robotic arm using non-invasive brain–computer interface and computer vision guidance. *Robotics and Autonomous Systems*, 115:121–129, 2019.

[108] Feng Duan, Dongxue Lin, Wenyu Li, and Zhao Zhang. Design of a multimodal eeg-based hybrid bci system with visual servo module. *IEEE Transactions on Autonomous Mental Development*, 2015.

[109] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[110] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015.

[111] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 2015.

[112] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[113] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[114] James B Rawlings. Tutorial overview of model predictive control. *IEEE control systems magazine*, 20(3):38–52, 2000.

[115] Fujian He, Luzheng Bi, Yun Lu, Hongqi Li, and Ling Wang. Model predictive control for a brain-controlled mobile robot. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3184–3188. IEEE, 2017.

[116] Yun Lu and Luzheng Bi. Human behavior model-based predictive control of longitudinal brain-controlled driving. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1361–1374, 2020.

[117] Hongguang Pan, Baocang Ding, Weimin Zhong, Gautam Kumar, and Mayuresh V Kothare. Designing closed-loop brain-machine interfaces with network of spiking neurons using mpc strategy. In *2015 American Control Conference (ACC)*, pages 2543–2548. IEEE, 2015.

[118] Hongguang Pan, Wenyu Mi, Xinyu Lei, and Jun Deng. A closed-loop brain–machine interface framework design for motor rehabilitation. *Biomedical Signal Processing and Control*, 58:101877, 2020.

[119] Jianjun Meng, Shuying Zhang, Angeliki Bekyo, Jaron Olsoe, Bryan Baxter, and Bin He. Noninvasive electroencephalogram based control of a robotic arm for reach and grasp tasks. *Scientific Reports*, 6(1):38565, 2016.

[120] Jianjun Meng, Taylor Streitz, Nicholas Gulachek, Daniel Suma, and Bin He. Three-dimensional brain–computer interface control through simultaneous overt spatial attentional and motor imagery tasks. *IEEE Transactions on Biomedical Engineering*, 65(11):2417–2427, 2018.

[121] Madiha Tariq, Pavel M Trivailo, and Milan Simic. Eeg-based bci control schemes for lower-limb assistive-robots. *Frontiers in human neuroscience*, 12:312, 2018.

[122] Yiliang Liu, Wenbin Su, Zhijun Li, Guangming Shi, Xiaoli Chu, Yu Kang, and Weiwei Shang. Motor-imagery-based teleoperation of a dual-arm robot performing manipulation tasks. *IEEE Transactions on Cognitive and Developmental Systems*, 11(3):414–424, 2018.

[123] Rosario Sorbello, Salvatore Tramonte, Marcello Emanuele Giardina, Vincenzo La Bella, Rossella Spataro, Brendan Allison, Christoph Guger, and Antonio Chella. A human–humanoid interaction through the use of bci for locked-in als patients using neurobiological feedback fusion. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(2):487–497, 2017.

[124] Xingchao Wang, Xiaopeng Huang, Yi Lin, Liguang Zhou, Zhenglong Sun, and Yangsheng Xu. Design of an ssvep-based bci stimuli system for attention-based robot navigation in robotic telepresence. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4126–4131. IEEE, 2021.

[125] Paolo Di Lillo, Filippo Arrichiello, Daniele Di Vito, and Gianluca Antonelli. Bci-controlled assistive manipulator: developed architecture and experimental results. *IEEE Transactions on Cognitive and Developmental Systems*, 2020.

[126] Iason Batzianoulis, Fumiaki Iwane, Shupeng Wei, Carolina Gaspar Pinto Ramos Correia, Ricardo Chavarriaga, José del R Millán, and Aude Billard. Customizing skills for assistive robotic manipulators, an inverse reinforcement learning approach with error-related potentials. *Communications biology*, 4(1):1–14, 2021.

[127] Yuxia Yuan, Wenbin Su, Zhijun Li, and Guangming Shi. Brain–computer interface-based stochastic navigation and control of a semiautonomous mobile robot in indoor environments. *IEEE Transactions on Cognitive and Developmental Systems*, 11(1):129–141, 2018.

[128] Hong Zeng, Yitao Shen, Xuhui Hu, Aiguo Song, Baoguo Xu, Huijun Li, Yanxin Wang, and Pengcheng Wen. Semi-autonomous robotic arm reaching with hybrid gaze–brain machine interface. *Frontiers in neurorobotics*, 13:111, 2020.

[129] Emmanuele Tidoni, Pierre Gergondet, Abderrahmane Kheddar, and Salvatore M Aglioti. Audio-visual feedback improves the bci performance in the navigational control of a humanoid robot. *Frontiers in neurorobotics*, 8:20, 2014.

[130] Francisco Velasco-Álvarez, Álvaro Fernández-Rodríguez, and Ricardo Ron-Angevin. Brain-computer interface (bci)-generated speech to control domotic devices. *Neurocomputing*, 509:121–136, 2022.

[131] Xiaoqian Mao, Wei Li, Chengwei Lei, Jing Jin, Feng Duan, and Sherry Chen. A brain–robot interaction system by fusing human and machine intelligence. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27(3):533–542, 2019.

[132] Mengfan Li, Wei Li, Linwei Niu, Huihui Zhou, Genshe Chen, and Feng Duan. An event-related potential-based adaptive model for telepresence control of humanoid robot motion in an environment cluttered with obstacles. *IEEE Transactions on Industrial Electronics*, 64(2):1696–1705, 2016.

[133] Xilin Liu, Hongjie Zhu, Tian Qiu, Srihari Y Sritharan, Dengteng Ge, Shu Yang, Milin Zhang, Andrew G Richardson, Timothy H Lucas, Nader Engheta, et al. A fully integrated sensor-brain–machine interface system for restoring somatosensation. *IEEE Sensors Journal*, 21(4):4764–4775, 2020.

[134] Maria Alejandra Romero-Laiseca, Denis Delisle-Rodriguez, Vivianne Cardoso, Dharmendra Gurve, Flavia Loterio, Jorge Henrique Posses Nascimento, Sridhar Krishnan, Anselmo Frizera-Neto, and Teodiano Bastos-Filho. A low-cost lower-limb brain-machine interface triggered by pedaling motor imagery for post-stroke patients rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(4):988–996, 2020.

[135] Siddharth Dangi, Suraj Gowda, Rodolphe Héliot, and Jose M Carmena. Adaptive kalman filtering for closed-loop brain-machine interface systems. In *2011 5th International IEEE/EMBS Conference on Neural Engineering*, pages 609–612. IEEE, 2011.

[136] Kyung-Hwan Shim, Ji-Hoon Jeong, Byoung-Hee Kwon, Byeong-Hoo Lee, and Seong-Whan Lee. Assistive robotic arm control based on brain-machine interface with vision guidance using convolution neural network. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 2785–2790. IEEE, 2019.

[137] Yun Lu, Luzheng Bi, and Hongqi Li. Model predictive-based shared control for brain-controlled driving. *IEEE Transactions on Intelligent Transportation Systems*, 21(2):630–640, 2019.

[138] Hongguang Pan, Wenyu Mi, Mei Wang, and Jinggao Sun. The auxiliary controller design based on model-free control in the brain-machine interface. In *2018 Chinese Control And Decision Conference (CCDC)*, pages 3578–3583. IEEE, 2018.

[139] Iretiayo Akinola, Boyuan Chen, Jonathan Koss, Aalhad Patankar, Jake Varley, and Peter Allen. Task level hierarchical system for bci-enabled shared autonomy. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 219–225. IEEE, 2017.

[140] Chenguang Yang, Huaiwei Wu, Zhijun Li, Wei He, Ning Wang, and Chun-Yi Su. Mind control of a robotic arm with visual fusion technology. *IEEE Transactions on Industrial Informatics*, 14(9):3822–3830, 2018.

[141] Zhitang Chen, Zhongpeng Wang, Kun Wang, Weibo Yi, and Hongzhi Qi. Recognizing motor imagery between hand and forearm in the same limb in a hybrid brain computer interface paradigm: An online study. *IEEE Access*, 7:59631–59639, 2019.

[142] Erwei Yin, Timothy Zeyl, Rami Saab, Tom Chau, Dewen Hu, and Zongtan Zhou. A hybrid brain–computer interface based on the fusion of p300 and ssvep scores. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(4):693–701, 2015.

[143] Jonathan R Wolpaw. Brain–computer interfaces. In *Handbook of clinical neurology*, volume 110, pages 67–74. Elsevier, 2013.

[144] Jing Zhao, Wei Li, and Mengfan Li. Comparative study of ssvep-and p300-based models for the telepresence control of humanoid robots. *PLoS One*, 10(11):e0142168, 2015.

[145] Teodiano Bastos-Filho, Alan Floriano, Eduardo Couto, and Richard JM Godinez-Tello. Towards a system to command a robotic wheelchair based on independent ssvep–bci. In *Smart Wheelchairs and Brain-Computer Interfaces*, pages 369–379. Elsevier, 2018.

[146] Andres F Salazar-Gomez, Joseph DelPreto, Stephanie Gil, Frank H Guenther, and Daniela Rus. Correcting robot mistakes in real time using eeg signals. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 6570–6577. IEEE, 2017.

[147] Yang Yu, Zongtan Zhou, Yadong Liu, Jun Jiang, Erwei Yin, Nannan Zhang, Zhihua Wang, Yaru Liu, Xingjie Wu, and Dewen Hu. Self-paced operation of a wheelchair based on a hybrid brain-computer interface combining motor imagery and p300 potential. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(12):2516–2526, 2017.

[148] Ana Lopes, Joao Rodrigues, Jorge Perdigao, Gabriel Pires, and Urbano Nunes. A new hybrid motion planner: Applied in a brain-actuated robotic wheelchair. *IEEE Robotics & Automation Magazine*, 23(4):82–93, 2016.

[149] Filippo Arrichiello, Paolo Di Lillo, Daniele Di Vito, Gianluca Antonelli, and Stefano Chiaverini. Assistive robot operated via p300-based brain computer interface. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6032–6037. IEEE, 2017.

[150] Shili Sheng, Peipei Song, Lingyue Xie, Zhendong Luo, Wennan Chang, Shurui Jiang, Haoyong Yu, Chi Zhu, Jeffrey Too Chuan Tan, and Feng Duan. Design of an ssvep-based bci system with visual servo module for a service robot to execute multiple tasks. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2267–2272. IEEE, 2017.

[151] Luz Maria Alonso-Valerdi, Ricardo Antonio Salido-Ruiz, and Ricardo A Ramirez-Mendoza. Motor imagery based brain–computer interfaces: An emerging technology to rehabilitate motor deficits. *Neuropsychologia*, 79:354–363, 2015.

[152] Saugat Bhattacharyya, Shingo Shimoda, and Mitsuhiro Hayashibe. A synergetic brain-machine interfacing paradigm for multi-dof robot control. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(7):957–968, 2016.

[153] Yangsong Zhang, Peng Xu, Tiejun Liu, Jun Hu, Rui Zhang, and Dezhong Yao. Multiple frequencies sequential coding for ssvep-based brain-computer interface. *PloS one*, 7(3):e29519, 2012.

[154] Gernot R Müller-Putz, Reinhold Scherer, Christian Brauneis, and Gert Pfurtscheller. Steady-state visual evoked potential (ssvep)-based communication: impact of harmonic frequency components. *Journal of neural engineering*, 2(4):123, 2005.

[155] Jonathan R Wolpaw, Niels Birbaumer, Dennis J McFarland, Gert Pfurtscheller, and Theresa M Vaughan. Brain–computer interfaces for communication and control. *Clinical neurophysiology*, 2002.

[156] Anwesha Khasnobish, Amit Konar, Dewakinandan N Tibarewala, and Atulya K Nagar. Bypassing the natural visual-motor pathway to execute complex movement related tasks using interval type-2 fuzzy sets. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(1):91–105, 2016.

[157] Emanuel Donchin, Kevin M Spencer, and Ranjith Wijesinghe. The mental prosthesis: assessing the speed of a p300-based brain-computer interface. *IEEE transactions on rehabilitation engineering*, 8(2):174–179, 2000.

[158] Tanuka Bhattacharjee, Reshma Kar, Amit Konar, Anna Lekova, and Atulya K Nagar. A general type-2 fuzzy set induced single trial p300 detection. In *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6. IEEE, 2017.

[159] Angela Riccio, Luca Simione, Francesca Schettini, Alessia Pizzimenti, Maurizio Inghilleri, Marta Olivetti Belardinelli, Donatella Mattia, and Febo Cincotti. Attention and p300-based bci performance in people with amyotrophic lateral sclerosis. *Frontiers in human neuroscience*, 7:732, 2013.

[160] Rong Liu, Yong-Xuan Wang, and Lin Zhang. An fdes-based shared control method for asynchronous brain-actuated robot. *IEEE Transactions on Cybernetics*, 46(6):1452–1462, 2015.

[161] Elizabeth C Tyler-Kabara, John Downey, Jeffrey Weiss, Katharina Muelling, Arun Venkataraman, Jean-Sebastien Valois, Shervin Javdani, Herbert Martial, J Andrew Bagnell, Andrew Schwartz, et al. 201; brain-machine interface control of a robotic arm for object grasping is improved with computer-vision based shared control. *Neurosurgery*, 62:233, 2015.

[162] Saugat Bhattacharyya, Amit Konar, and DN Tibarewala. Motor imagery, p300 and error-related eeg-based robot arm movement control for rehabilitation purpose. *Medical & biological engineering & computing*, 52:1007–1017, 2014.

[163] Su Kyoung Kim, Elsa Andrea Kirchner, Arne Stefes, and Frank Kirchner. Intrinsic interactive reinforcement learning–using error-related potentials for real world human-robot interaction. *Scientific reports*, 7(1):1–16, 2017.

[164] Claude Alain, Heather E McNeely, Yu He, Bruce K Christensen, and Robert West. Neurophysiological evidence of error-monitoring deficits in patients with schizophrenia. *Cerebral Cortex*, 12(8):840–846, 2002.

[165] John G Kerns, Jonathan D Cohen, Angus W MacDonald III, Melissa K Johnson, V Andrew Stenger, Howard Aizenstein, and Cameron S Carter. Decreased conflict-and error-related activity in the anterior cingulate cortex in subjects with schizophrenia. *American Journal of Psychiatry*, 162(10):1833–1839, 2005.

[166] Francisco L Colino, Harvey Howse, Angela Norton, Robert Trska, Anthony Pluta, Stephen JC Luehr, Todd C Handy, and Olave E Krigolson. Older adults display diminished error processing and response in a continuous tracking task. *Psychophysiology*, 54(11):1706–1713, 2017.

[167] Patricia E Pailing and Sidney J Segalowitz. The error-related negativity as a state and trait measure: Motivation, personality, and erps in response to errors. *Psychophysiology*, 41(1):84–95, 2004.

[168] Katsuhiko Ogata. *Modern control engineering fifth edition*. 2010.

[169] Anthony Christopher Davison and David Victor Hinkley. *Bootstrap methods and their application*. Number 1. Cambridge university press, 1997.

[170] JL Basdevant. The padé approximation and its physical applications. *Fortschritte der Physik*, 20(5):283–331, 1972.

[171] Jürgen Kayser and Craig E Tenke. Hemifield-dependent n1 and event-related theta/delta oscillations: an unbiased comparison of surface laplacian and common eeg reference choices. *International Journal of Psychophysiology*, 97(3):258–270, 2015.

[172] Matthias Witkowski, Mario Cortese, Marco Cempini, Jürgen Mellinger, Nicola Vitiello, and Surjo R Soekadar. Enhancing brain-machine interface (bmi) control of a hand exoskeleton using electrooculography (eog). *Journal of neuroengineering and rehabilitation*, 11(1):1–6, 2014.

[173] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern recognition*. Elsevier, 2006.

[174] Swagatam Das, Ajith Abraham, and Amit Konar. Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives. *Advances of computational intelligence in industrial systems*, pages 1–38, 2008.

[175] Anwesha Khasnobish, Saugat Bhattacharyya, Amit Konar, DN Tibarewala, and Atulya K Nagar. A two-fold classification for composite decision about localized arm movement from eeg by svm and qda techniques. In *The 2011 International Joint Conference on Neural Networks*, pages 1344–1351. IEEE, 2011.

[176] Benjamin Blankertz, Ryota Tomioka, Steven Lemm, Motoaki Kawanabe, and Klaus-Robert Muller. Optimizing spatial filters for robust eeg single-trial analysis. *IEEE Signal processing magazine*, 25(1):41–56, 2007.

[177] Qibin Zhao, Liqing Zhang, et al. Temporal and spatial features of single-trial eeg for brain-computer interface. *Computational intelligence and neuroscience*, 2007, 2007.

[178] Rajesh PN Rao and Reinhold Scherer. Statistical pattern recognition and machine learning in brain–computer interfaces. In *Statistical signal processing for neuroscience and neurotechnology*, pages 335–367. Elsevier, 2010.

[179] Biswadeep Chakraborty, Lidia Ghosh, and Amit Konar. Designing phase-sensitive common spatial pattern filter to improve brain-computer interfacing. *IEEE Transactions on Biomedical Engineering*, 67(7):2064–2072, 2019.

[180] Ernst Haselsteiner and Gert Pfurtscheller. Using time-dependent neural networks for eeg classification. *IEEE transactions on rehabilitation engineering*, 8(4):457–463, 2000.

[181] Arnab Rakshit, Susenjit Ghosh, Amit Konar, and Monalisa Pal. A novel hybrid brain-computer interface for robot arm manipulation using visual evoked potential. In *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)*, pages 1–6. IEEE, 2017.

[182] General Assembly of the World Medical Association et al. World medical association declaration of helsinki: ethical principles for medical research involving human subjects. *The Journal of the American College of Dentists*, 81(3):14–18, 2014.

[183] Lianghua He, Die Hu, Meng Wan, Ying Wen, Karen M Von Deneen, and MengChu Zhou. Common bayesian network for classification of eeg-based multiclass motor imagery bci. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46(6):843–854, 2015.

[184] Gert Pfurtscheller and FH Lopes Da Silva. Event-related eeg/meg synchronization and desynchronization: basic principles. *Clinical neurophysiology*, 110(11):1842–1857, 1999.

[185] Alois Schlögl, Felix Lee, Horst Bischof, and Gert Pfurtscheller. Characterization of four-class motor imagery eeg data for the bci-competition 2005. *Journal of neural engineering*, 2(4):L14, 2005.

[186] Ton J Cleophas, Aeilko H Zwinderman, Ton J Cleophas, and Aeilko H Zwinderman. One-sample continuous data (one-sample t-test, one-sample wilcoxon signed rank test, 10 patients). *SPSS for Starters and 2nd Levelers*, pages 3–6, 2016.

[187] Xu Han, Ke Lin, Shangkai Gao, and Xiaorong Gao. A novel system of ssvep-based human–robot coordination. *Journal of neural engineering*, 16(1):016006, 2018.

[188] Kyuhwa Lee, Dong Liu, Laetitia Perroud, Ricardo Chavarriaga, and José del R Millán. A brain-controlled exoskeleton with cascaded event-related desynchronization classifiers. *Robotics and Autonomous Systems*, 90:15–23, 2017.

[189] Qiang Gao, Lixiang Dou, Abdelkader Nasreddine Belkacem, Chao Chen, et al. Noninvasive electroencephalogram based control of a robotic arm for writing task using hybrid bci system. *BioMed research international*, 2017, 2017.

[190] Hong Zeng, Yanxin Wang, Changcheng Wu, Aiguo Song, Jia Liu, Peng Ji, Baoguo Xu, Lifeng Zhu, Huijun Li, and Pengcheng Wen. Closed-loop hybrid gaze brain-machine interface based robotic arm control with augmented reality feedback. *Frontiers in neurorobotics*, 11:60, 2017.

[191] Chao Chen, Peng Zhou, Abdelkader Nasreddine Belkacem, Lin Lu, Rui Xu, Xiaotian Wang, Wenjun Tan, Zhifeng Qiao, Penghai Li, Qiang Gao, et al. Quadcopter robot control based on hybrid brain-computer interface system. *Sensors & Materials*, 32, 2020.

[192] K Koray Ayten, M Necip Sahinkaya, and Ahmet Dumlu. Optimum trajectory generation for redundant/hyper-redundant manipulators. *IFAC-PapersOnLine*, 49(21):493–500, 2016.

[193] Chuanwei Liu, Yunfa Fu, Jun Yang, Xin Xiong, Huiwen Sun, and Zhengtao Yu. Discrimination of motor imagery patterns by electroencephalogram phase synchronization combined with frequency band energy. *IEEE/CAA Journal of Automatica Sinica*, 4(3):551–557, 2016.

[194] Chung Song Kim, Jinwei Sun, Dan Liu, Qisong Wang, and Sung Gyun Paek. Removal of ocular artifacts using ica and adaptive filter for motor imagery-based bci. *IEEE/CAA journal of automatica sinica*, 2017.

[195] Jihyeon Ha and Laehyun Kim. A brain—computer interface-based meal—assist robot control system. In *2021 9th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–3. IEEE, 2021.

[196] Jihyeon Ha, Sangin Park, Chang-Hwan Im, and Laehyun Kim. A hybrid brain–computer interface for real-life meal-assist robot control. *Sensors*, 21(13):4578, 2021.

[197] Gert Pfurtscheller, Christa Neuper, GR Muller, Bernhard Obermaier, Gunter Krausz, A Schlogl, Reinhold Scherer, Bernhard Graimann, Claudia Keinrath, Dimitris Skliris, et al. Graz-bci: state of the art and clinical applications. *IEEE Transactions on neural systems and rehabilitation engineering*, 11(2):1–4, 2003.

[198] Yiliang Liu, Zhijun Li, Tong Zhang, and Suna Zhao. Brain–robot interface-based navigation control of a mobile robot in corridor environments. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50(8):3047–3058, 2018.

[199] Luca Tonin, Felix Christian Bauer, and José del R Millán. The role of the control framework for continuous teleoperation of a brain–machine interface-driven mobile robot. *IEEE Transactions on Robotics*, 36(1):78–91, 2019.

[200] Xin Chen, Yang Yu, Jingsheng Tang, Liang Zhou, Kaixuan Liu, Ziyuan Liu, Siming Chen, Jian Wang, Ling-Li Zeng, Jinfang Liu, et al. Clinical validation of bci-controlled wheelchairs in subjects with severe spinal cord injury. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:579–589, 2022.

[201] Xiaogang Chen, Bing Zhao, Yijun Wang, and Xiaorong Gao. Combination of high-frequency ssvep-based bci and computer vision for controlling a robotic arm. *Journal of neural engineering*, 16(2):026012, 2019.

[202] Anthony Casey, Hannan Azhar, Marek Grzes, and Mohamed Sakel. Bci controlled robotic arm as assistance to the rehabilitation of neurologically disabled patients. *Disability and Rehabilitation: Assistive Technology*, 16(5):525–537, 2021.

[203] Marco Vilela and Leigh R Hochberg. Applications of brain-computer interfaces to the control of robotic and prosthetic arms. *Handbook of clinical neurology*, 168:87–99, 2020.

[204] Xiaofei Wang, Hsiang-Ting Chen, Yu-Kai Wang, and Chin-Teng Lin. Implicit robot control using error-related potential-based brain-computer interface. *IEEE Transactions on Cognitive and Developmental Systems*, 2022.

[205] Iretiayo Akinola, Zizhao Wang, Junyao Shi, Xiaomin He, Pawan Lapborisuth, Jingxi Xu, David Watkins-Valls, Paul Sajda, and Peter Allen. Accelerated robot learning via human

brain signals. In *2020 IEEE international conference on robotics and automation (ICRA)*, pages 3799–3805. IEEE, 2020.

[206] IJ Nagrath and M Gopal. Control systems engineering, (2007), isbn: 81-224-2008-7. *New Age International Publishers*, pages 193–268.

[207] Arnab Rakshit, Amit Konar, and Atulya K Nagar. A hybrid brain-computer interface for closed-loop position control of a robot arm. *IEEE/CAA Journal of Automatica Sinica*, 7(5):1344–1360, 2020.

[208] Janusz T Starczewski. *Advanced concepts in fuzzy logic and systems with membership uncertainty*, volume 284. Springer, 2012.

[209] Hans-Jürgen Zimmermann. *Fuzzy set theory—and its applications*. Springer Science & Business Media, 2011.

[210] Fabien Lotte and Cuntai Guan. Spatially regularized common spatial patterns for eeg classification. In *2010 20th International Conference on Pattern Recognition*, pages 3712–3715. IEEE, 2010.

[211] Rihab Bousseta, Salma Tayeb, Issam El Ouakouak, Mourad Gharbi, Fakhita Regragui, and Mohamed Majid Himmi. Eeg efficient classification of imagined hand movement using rbf kernel svm. In *2016 11th International Conference on Intelligent Systems: Theories and Applications (SITA)*, pages 1–6. IEEE, 2016.

[212] Sarah N Carvalho, Thiago BS Costa, Luisa FS Uribe, Diogo C Soriano, Glauco FG Yared, Luis C Coradine, and Romis Attux. Comparative analysis of strategies for feature extraction and classification in ssvep bcis. *Biomedical Signal Processing and Control*, 21:34–42, 2015.

[213] Amit Konar. *Computational intelligence: principles, techniques and applications*. Springer Science & Business Media, 2006.

[214] Richard W Homan, John Herman, and Phillip Purdy. Cerebral location of international 10–20 system electrode placement. *Electroencephalography and clinical neurophysiology*, 66(4):376–382, 1987.

[215] Rihab Bousseta, I El Ouakouak, M Gharbi, and F Regragui. Eeg based brain computer interface for controlling a robot arm movement through thought. *Irbm*, 39(2):129–135, 2018.

[216] Nong Yan, Chang Wang, Yi Tao, Jinming Li, Kexu Zhang, Ting Chen, Ziwen Yuan, Xiangguo Yan, and Gang Wang. Quadcopter control system using a hybrid bci based on off-line optimization and enhanced human-machine interaction. *IEEE access*, 8:1160–1172, 2019.

[217] Javier Andreu-Perez, Fan Cao, Hani Hagras, and Guang-Zhong Yang. A self-adaptive online brain–machine interface of a humanoid robot through a general type-2 fuzzy inference system. *IEEE Transactions on Fuzzy Systems*, 2016.

[218] Alexander J Doud, John P Lucas, Marc T Pisansky, and Bin He. Continuous three-dimensional control of a virtual helicopter using a motor imagery based brain-computer interface. *PloS one*, 2011.

[219] Tianwei Shi, Hong Wang, and Chi Zhang. Brain computer interface system based on indoor semi-autonomous navigation and motor imagery for unmanned aerial vehicle control. *Expert Systems with Applications*, 2015.

[220] Keun-Tae Kim, Heung-Il Suk, and Seong-Whan Lee. Commanding a brain-controlled wheelchair using steady-state somatosensory evoked potentials. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2016.

[221] Zhijun Li, Suna Zhao, Jiding Duan, Chun-Yi Su, Chenguang Yang, and Xingang Zhao. Human cooperative wheelchair with brain–machine interaction based on shared control strategy. *IEEE/ASME Transactions on Mechatronics*, 2016.

[222] Carlos Escolano, Javier Mauricio Antelis, and Javier Minguez. A telepresence mobile robot controlled with a noninvasive brain–computer interface. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2011.

[223] Suna Zhao, Zhijun Li, Rongxin Cui, Yu Kang, Fuchun Sun, and Rong Song. Brain–machine interfacing-based teleoperation of multiple coordinated mobile robots. *IEEE Transactions on Industrial Electronics*, 2016.

[224] Ang Kai Keng and Guan Cuntai. Brain-computer interface in stroke rehabilitation. *Journal of Computing Science and Engineering*, 2013.

[225] Nikhil Sharma, Valerie M Pomeroy, and Jean-Claude Baron. Motor imagery: a backdoor to the motor system after stroke? *Stroke*, 2006.

[226] Nicholas Cheng, Kok Soon Phua, Hwa Sen Lai, Pui Kit Tam, Ka Yin Tang, Kai Kei Cheng, Raye Chen-Hua Yeow, Kai Keng Ang, Cuntai Guan, and Jeong Hoon Lim. Brain-computer interface-based soft robotic glove rehabilitation for stroke. *IEEE Transactions on Biomedical Engineering*, 2020.

[227] Andrew B Schwartz, X Tracy Cui, Douglas J Weber, and Daniel W Moran. Brain-controlled interfaces: movement restoration with neural prosthetics. *Neuron*, 2006.

[228] Meel Velliste, Sagi Perel, M Chance Spalding, Andrew S Whitford, and Andrew B Schwartz. Cortical control of a prosthetic arm for self-feeding. *Nature*, 2008.

[229] Kapil D Katyal, Matthew S Johannes, Spencer Kellis, Tyson Aflalo, Christian Klaes, Timothy G McGee, Matthew P Para, Ying Shi, Brian Lee, Kelsie Pejsa, et al. A collaborative bci approach to autonomous control of a prosthetic limb system. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2014.

[230] Sorin M Grigorescu, Thorsten Lüth, Christos Fragkopoulos, Marco Cyriacks, and Axel Gräser. A bci-controlled robotic assistant for quadriplegic people in domestic and professional life. *Robotica*, 2012.

[231] Sebastian Schröer, Ingo Killmann, Barbara Frank, Martin Völker, Lukas Fiederer, Tonio Ball, and Wolfram Burgard. An autonomous robotic assistant for drinking. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[232] Daniel Kuhner, Lukas DJ Fiederer, Johannes Aldinger, Felix Burget, Martin Völker, Robin T Schirrmeister, Chau Do, Joschka Boedecker, Bernhard Nebel, Tonio Ball, et al. Deep learning based bci control of a robotic service assistant using intelligent goal formulation. *bioRxiv*, 2018.

[233] Gert Pfurtscheller. Event-related synchronization (ers): an electrophysiological correlate of cortical areas at rest. *Electroencephalography and clinical neurophysiology*, 1992.

[234] Christa Neuper, Michael Wörtz, and Gert Pfurtscheller. Erd/ers patterns reflecting sensorimotor activation and deactivation. *Progress in brain research*, 2006.

[235] Leonard J Trejo, Roman Rosipal, and Bryan Matthews. Brain-computer interfaces for 1-d and 2-d cursor control: designs using volitional control of the eeg spectrum or steady-state visual evoked potentials. *IEEE transactions on neural systems and rehabilitation engineering*, 2006.

[236] Pablo Martinez, Hovagim Bakardjian, and Andrzej Cichocki. Fully online multicommand brain-computer interface with visual neurofeedback using ssvep paradigm. *Computational intelligence and neuroscience*, 2007.

[237] Lawrence Ashley Farwell and Emanuel Donchin. Talking off the top of your head: toward a mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology*, 1988.

[238] Leigh R Hochberg, Daniel Bacher, Beata Jarosiewicz, Nicolas Y Masse, John D Simeral, Joern Vogel, Sami Haddadin, Jie Liu, Sydney S Cash, Patrick Van Der Smagt, et al. Reach and grasp by people with tetraplegia using a neurally controlled robotic arm. *Nature*, 2012.

[239] Jennifer L Collinger, Brian Wodlinger, John E Downey, Wei Wang, Elizabeth C Tyler-Kabara, Douglas J Weber, Angus JC McMorland, Meel Velliste, Michael L Boninger, and Andrew B Schwartz. High-performance neuroprosthetic control by an individual with tetraplegia. *The Lancet*, 2013.

[240] Michele Barsotti, D Leonardis, C Loconsole, Massimiliano Solazzi, E Sotgiu, C Procopio, C Chisari, M Bergamasco, and A Frisoli. A full upper limb robotic exoskeleton for reaching and grasping rehabilitation triggered by mi-bci. In *2015 IEEE international conference on rehabilitation robotics (ICORR)*, pages 49–54. IEEE, 2015.

[241] Wenchang Zhang, Fuchun Sun, Jianhua Chen, Chuanqi Tan, Hang Wu, and Weihua Su. An asynchronous mi-based bci for brain-actuated robot grasping control. In *2017 International Conference on Computer Systems, Electronics and Control (ICCSEC)*, pages 893–898. IEEE, 2017.

[242] Jeong-Hyun Cho, Ji-Hoon Jeong, Kyung-Hwan Shim, Dong-Joo Kim, and Seong-Whan Lee. Classification of hand motions within eeg signals for non-invasive bci-based robot hand control. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 515–518. IEEE, 2018.

[243] Rossella Spataro, Antonio Chella, Brendan Allison, Marcello Giardina, Rosario Sorbello, Salvatore Tramonte, Christoph Guger, and Vincenzo La Bella. Reaching and grasping a glass of water by locked-in als patients through a bci-controlled humanoid robot. *Frontiers in human neuroscience*, 11:68, 2017.

[244] Batyrkhan Saduanov, Tohid Alizadeh, Jinung An, and Berdakh Abibullaev. Trained by demonstration humanoid robot controlled via a bci system for telepresence. In *2018 6th International Conference on Brain-Computer Interface (BCI)*, pages 1–4. IEEE, 2018.

[245] Jonathan Delijorge, Omar Mendoza-Montoya, Jose L Gordillo, Ricardo Caraza, Hector R Martinez, and Javier M Antelis. Evaluation of a p300-based brain-machine interface for a robotic hand-orthosis control. *Frontiers in Neuroscience*, 14, 2020.

[246] Yang Xu, Heng Zhang, Linfeng Cao, Xiaokang Shu, and Dingguo Zhang. A shared control strategy for reach and grasp of multiple objects using robot vision and noninvasive brain-computer interface. *IEEE Transactions on Automation Science and Engineering*, 2020.

[247] Xiaoqian Mao, Wei Li, Chengwei Lei, Jing Jin, Feng Duan, and Sherry Chen. A brain–robot interaction system by fusing human and machine intelligence. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27(3):533–542, 2019.

[248] Fred Paas, Juhani E Tuovinen, Huib Tabbers, and Pascal WM Van Gerven. Cognitive load measurement as a means to advance cognitive load theory. *Educational psychologist*, 38(1):63–71, 2003.

[249] Pavlo Antonenko, Fred Paas, Roland Grabner, and Tamara Van Gog. Using electroencephalography to measure cognitive load. *Educational Psychology Review*, 22(4):425–438, 2010.

[250] Naveen Kumar and Jyoti Kumar. Measurement of cognitive load in hci systems using eeg power spectrum: an experimental study. *Procedia Computer Science*, 84:70–78, 2016.

[251] Sandra G Hart and Lowell E Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Advances in psychology*, volume 52, pages 139–183. Elsevier, 1988.

[252] Aniana Cruz, Gabriel Pires, Ana Lopes, Carlos Carona, and Urbano J Nunes. A self-paced bci with a collaborative controller for highly reliable wheelchair driving: Experimental tests with physically disabled individuals. *IEEE Transactions on Human-Machine Systems*, 51(2):109–119, 2021.

[253] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2017.

[254] Yun Jiang, Stephen Moseson, and Ashutosh Saxena. Efficient grasping from rgbd images: Learning using a new rectangle representation. In *2011 IEEE International conference on robotics and automation*, 2011.

[255] Ian Lenz, Honglak Lee, and Ashutosh Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 2015.

[256] Ashutosh Saxena, Justin Driemeyer, and Andrew Y Ng. Robotic grasping of novel objects using vision. *The International Journal of Robotics Research*, 2008.

[257] Matei Ciocarlie, Kaijen Hsiao, Edward Gil Jones, Sachin Chitta, Radu Bogdan Rusu, and Ioan A Şucan. Towards reliable grasping and manipulation in household environments. In *Experimental Robotics*. Springer, 2014.

[258] Joseph Redmon and Anelia Angelova. Real-time grasp detection using convolutional neural networks. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[259] Sulabh Kumra and Christopher Kanan. Robotic grasp detection using deep convolutional neural networks. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[260] Xinwen Zhou, Xuguang Lan, Hanbo Zhang, Zhiqiang Tian, Yang Zhang, and Narming Zheng. Fully convolutional grasp detection network with oriented anchor box. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[261] Fu-Jen Chu, Ruinian Xu, and Patricio A Vela. Real-world multiobject, multigrasp detection. *IEEE Robotics and Automation Letters*, 2018.

[262] Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, pages 37–52, 1987.

[263] H. Zhang, X. Lan, S. Bai, X. Zhou, Z. Tian, and N. Zheng. Roi-based robotic grasp detection for object overlapping scenes. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019.

[264] H. Zhang, X. Lan, S. Bai, L. Wan, C. Yang, and N. Zheng. A multi-task convolutional neural network for autonomous robotic grasping in object stacking scenes. 2019.

[265] Dongwon Park, Yonghyeok Seo, Dongju Shin, Jaesik Choi, and Se Young Chun. A single multi-task deep neural network with post-processing for object detection with reasoning and robotic grasp detection. 2020.

[266] World Medical Association et al. World medical association declaration of helsinki. ethical principles for medical research involving human subjects. *Bulletin of the World Health Organization*, 79(4):373, 2001.

[267] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, 2016.

[268] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 2015.

[269] Jason Wang and Luis Perez. The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Networks Vis. Recognit*, 2017.

[270] Sebastien C Wong, Adam Gatt, Victor Stamatescu, and Mark D McDonnell. Understanding data augmentation for classification: when to warp? In *2016 international conference on digital image computing: techniques and applications (DICTA)*. IEEE, 2016.

[271] Connor Shorten and Taghi M Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 2019.

[272] Fei Pan, Baoying Wang, Xin Hu, and William Perrizo. Comprehensive vertical sample-based knn/lsvm classification for gene expression analysis. *Journal of Biomedical informatics*, 2004.

[273] Ujwala Ravale, Nilesh Marathe, and Puja Padiya. Feature selection based hybrid anomaly intrusion detection system using k means and rbf kernel function. *Procedia Computer Science*, 2015.

[274] Anisha Halder, Amit Konar, Rajshree Mandal, Aruna Chakraborty, Pavel Bhowmik, Nikhil R Pal, and Atulya K Nagar. General and interval type-2 fuzzy face-space approach to emotion recognition. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2013.

[275] Anuradha Saha, Amit Konar, and Atulya K Nagar. Eeg analysis for cognitive failure detection in driving using type-2 fuzzy classifiers. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2017.

[276] Vu NP Dao and VR Vemuri. A performance comparison of different back propagation neural networks methods in computer network intrusion detection. *Differential equations and dynamical systems*, 2002.

[277] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 2012.

[278] Daniel Griffin and Jae Lim. Signal estimation from modified short-time fourier transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1984.

[279] Arne Jensen and Anders la Cour-Harbo. *Ripples in mathematics: the discrete wavelet transform*. Springer Science & Business Media, 2001.

[280] Aapo Hyvärinen and Erkki Oja. Independent component analysis: algorithms and applications. *Neural networks*, pages 411–430, 2000.

[281] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238, 2005.

[282] Hanbo Zhang, Xuguang Lan, Xinwen Zhou, Zhiqiang Tian, Yang Zhang, and Nanning Zheng. Visual manipulation relationship network for autonomous robotics. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, 2018.

Arnab Rakshit
24/07/23

24.7.23

PROF. AMIT KONAR
CO-ORDINATOR
INTELLIGENT AUTOMATION
& ROBOTICS
E.T.C.E. DEPTT., J.U., KOL-32