NEUROGAZE IN VIRTUAL REALITY: ASSESSING AN EEG AND EYE TRACKING INTERFACE
AGAINST TRADITIONAL VIRTUAL REALITY INPUT DEVICES

by

WANYEA BARBEL
B.S, Computer Science, University of Central Florida, 2024

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in the Department of Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2024

Major Professor: Joseph J. LaViola

ABSTRACT

NeuroGaze is a novel Virtual Reality (VR) interface that integrates electroencephalogram (EEG) and eye-tracking technologies to enhance user interaction within virtual environments (VEs). Diverging from traditional VR input devices, NeuroGaze allows users to select objects in a VE through gaze direction and cognitive intent captured via EEG signals. The research assesses the performance of the NeuroGaze system against conventional input devices such as VR controllers and eye gaze combined with hand gestures. The experiment, conducted with 20 participants, evaluates task completion time, accuracy, and cognitive load through the NASA-TLX surveys and user preference through a post-evaluation survey. Results indicate that while NeuroGaze presents a learning curve, evidenced by longer average task durations, it potentially offers a more accurate selection method with lower cognitive load, as suggested by its lower error rate and significant differences in NASA-TLX scores, physical demand, and temporal subscales. This study highlights the viability of incorporating biometric inputs for more accessible and less demanding VR interactions. Future work aims to explore a multimodal EEG-Functional near-infrared spectroscopy (fNIRS) approach, further develop machine learning models for EEG signal classification, and extend system capabilities to dynamic object selection, highlighting the progressive direction for the use of Brain-Computer Interfaces (BCI) in virtual environments.

Dedicated to all my family, friends and colleagues who never stopped believing in me.

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

1. 3DUI—Three-Dimensional User Interface: a user interface that interacts with three-dimensional space, allowing users to manipulate 3D objects or navigate through 3D environments using various input devices.

2. AR—Augmented Reality: technology that overlays digital content (such as images, sounds, and text) in the real world, enhancing users' perception of their surroundings through devices like smartphones, tablets, or AR glasses.

3. BCI—Brain-Computer Interface: a direct communication pathway between a brain and an external device, enabling control of the device using brain signals, often measured via EEG for non-invasive BCIs or implanted electrodes for invasive BCIs.

4. EEG—Electroencephalography: a method for recording the brain's electrical activity using electrodes placed along the scalp. It measures voltage fluctuations resulting from ionic current flows within the brain's neurons and is widely used in neuroscience, cognitive psychology, and BCI applications.

5. FNIRS—Functional Near-Infrared Spectroscopy: a non-invasive imaging technique that measures brain activity by detecting changes in blood oxygenation and blood volume within the cortex, using near-infrared light.

6. HCI—Human-Computer Interaction: a multidisciplinary field of study focusing on the design of computer technology, particularly the interaction between humans (the users) and computers. HCI studies designing, evaluating, and implementing interactive computing systems for human use.

7. MR—Mixed Reality: a blend of physical and digital worlds; MR creates new environments where physical and digital objects coexist and interact in real time. It is often seen as a spectrum encompassing augmented reality (AR) and virtual reality (VR).

8. POC—Proof of Concept: a demonstration, typically in the early stages of development, to verify that certain concepts or theories can be achieved in development. In technology, a POC tests a prototype of an idea to validate its feasibility, functionality, or potential for further development.

9. VE—Virtual Environment: a digital space computer technology generates, simulating a real or imagined environment. VEs are interactive, allowing users to navigate and manipulate virtual objects or perform tasks. They are commonly used in VR systems.

10. VR—Virtual Reality: a simulated experience that can be like or completely different from the real world. VR technology creates a fully immersive environment, often engaging sight, sound, and increasingly touch through specialized hardware like VR headsets, gloves, and treadmills.

CHAPTER ONE: INTRODUCTION

1.1 Motivation

From my earliest memories, I've been captivated by the idea of existing in a virtual world — an enhanced version of our reality possible through digital innovation. My journey as a mixed reality engineer, crafting experience in virtual reality (VR), is the closest I've come to creating this *New World*. Yet, the challenge persists: the input devices currently available to interact with these virtual worlds have either made me feel disconnected from the virtual environment (VE) or the controls are too cumbersome. This disconnect highlights an opportunity for innovation in interacting and engaging with these virtual environments.

Selection within virtual environments is where I feel the most disconnected from the VE due to the need to remember specific hand gestures mid-game or maintain a constant arm motion to point and select objects. The discovery of brain-computer interfaces (BCI) through *Toward Brain-Computer Interfacing* by Guido Dornhege led me to wonder what applications the technology might have in virtual environments. Dornhege's comprehensive discussion covers a spectrum of BCI methodologies, from invasive to non-invasive approaches, alongside an in-depth look into BCI techniques and software. The book outlines promising BCI research and the potential trajectory of the future. This exploration made me wonder if BCIs are key to revolutionizing interaction within virtual environments, offering a seamless, intuitive way to merge human intention with virtual worlds.

Existing research highlights the feasibility of using electroencephalogram (EEG) recordings from the brain combined with eye gaze can be used to interact with computers [18]. Other research utilizing VR investigates how EEG and eye gaze is used to evaluate engagement [3] or how eye movement artifacts found within EEG readings during VR simulations can be used to build artifact correction methods

[25]. In my literature reviews, I have found that research involving BCI in VR primarily uses VR as a medium to simulate environments that are not easily replicable in non-virtual environments to record EEG data. However, there is a gap in research specifically attempting to segment and classify artifacts in EEG streams with the intent of selection or interaction in VR. My goal for this thesis is to build on the scarce literature in the space of using non-invasive BCIs to interact and select objects in virtual environments with the hope of paving a pathway for future work to be done in this area of study.

1.2 Input Devices

1.2.1 NeuroGaze – Eye Tracking & Electroencephalogram

NeuroGaze is a novel selection method we developed that utilizes infrared (IR) cameras built into a VR head-mounted display (HMD) to track where a user is looking in a VE. NeuroGaze also uses EEG to determine when a user would like to select an object once they have both looked at it and thought about it. This technique aims to provide users with a more intuitive experience when selecting objects in VR while continuing to maintain the level of reliability and usability users would expect when using traditional interaction methods (e.g., VR controllers).

1.2.2 Eye Gaze & Hand Tracking

Eye gaze and hand tracking is a selection method that uses eye tracking for users to *hover* over objects they would like to select and hand tracking to select or interact with objects. The *trigger* for these hand-tracking actions is usually done utilizing gestures. Gestures use specific hand movements and positions to specify a command in a VE [19]. Specific to our evaluation, participants used a pinch gesture to select objects within the VE. Recently, this technique has been showcased by the Apple Vision Pro for all selections within the AR environment.

1.2.3 VR Controllers

VR controllers are handheld devices that allow users to interact with a virtual environment (VE). They are designed to be used as selection devices, enabling users to navigate menus, select items, and interact with objects within a VE. For selection, participants will use the ray-casting interaction technique, casting a ray from the end of the VR controllers. This technique allows participants to determine where to interact with the VE, and the controller triggers are used to select objects. VR controllers give users precise control over selection and manipulation tasks but can become physically tiring when used for extended periods [20].

CHAPTER TWO: LITERATURE REVIEW

2.1 Input Modalities in Virtual Reality

2.1.1 VR Controllers

Early VR systems relied on basic input devices such as keyboards, mice, or joysticks, which did not provide a natural or intuitive way to interact with VEs. The development of dedicated VR controllers marked a significant advancement, offering users a more immersive and hands-on VR experience. Studies have investigated the efficiency and accuracy of selection interactions using VR controllers, highlighting their effectiveness for selecting and manipulating objects in 3D space [2]. Over the years, research has expanded on these devices in new and exciting ways to build upon the initial VR controllers.

An example of this can be seen in Whitmire et al. [30], who developed a unique handheld controller, denoted the *Haptic Revolver*, utilizing an actuated wheel. As participants spin a haptic wheel on the controller to move around the VE, the controller provides users with tactical feedback that can be interchanged with different wheels. This allows users to experience a variety of accurate haptic information depending on the VR scene. Although haptics is outside of the scope of this Thesis and the NeuroGaze system does not include haptic feedback, it is important to recognize the use of haptics to provide users with a more intuitive and immersive experience.

Fahmi et al. [8] investigated the overall user experience of VIVE VR controllers compared to a Leap Motion Controller (LMC) and Senso Gloves. LMCs use an infrared monochromic camera that periodically takes pictures [8]. An observation area is set, and when a user's hand enters this area, the hand gestures are recorded, and an action is performed in VR. Senso Gloves are wireless devices that use Inertia Measurement Unit (IMU) sensors to observe hand and finger movement. This device also includes haptic feed in the form of vibrators in the fingertips, back of the hand, and wrists [23]. When a user

experience study was conducted, high Likert scale values for the VIVE VR controllers suggested that participants felt that VR controllers were very interactive and the controls were easy to learn and use. When ideating on using EEG effectively as a part of the NeuroGaze input device, the first challenge was leveraging a user's previous experience with input devices such as VR controllers. Although NeuroGaze does not use VR controllers, the system should still feel intuitive and easy to use. Studies like Fahmi et al. show that if NeuroGaze is designed with the same level of interactivity, intuitiveness, and satisfaction as VR controllers, participants would be more inclined to report a positive user experience. Intuitiveness and ease of use are only a fraction of the user experience; the input system should also perform tasks successfully and effectively. Luong et al. [21] evaluate the interaction performance of VR controllers compared to hand gesture recognition. Only the VR controllers utilized the virtual pointer metaphor, which alleviates a major shortcoming of the input devices: interacting with objects that are out of reach [2]. This study showed that for the interactions, participants preferred VR controllers with ray cast because of the precision they could get. Additionally, these participants performed better when evaluated on speed and accuracy than when using the virtual hand gesture. It was decided that NeuroGaze would use some ray casting as the interaction metaphor of choice due to its speed and intuitiveness. However, physical strain is a common complaint when using ray pointing with VR controllers for longer durations. In research evaluating this input device and metaphor, participants start to hold the controller closer to their body to reduce fatigue [2]. For this reason, it was determined that the NeuroGaze system would not use VR controllers despite its intuitiveness. Looking deeper into how prolonged use of input devices can cause fatigue, we investigated input devices compatible with the ray-casting metaphor [2] but did not induce as much physical effort as VR controllers over longer durations. The following steps in developing the NeuroGaze system pointed toward eye tracking as a possible host for the ray-casting metaphor.

2.1.2 Eye Gaze

Eye gaze uses eye-tracking technology to measure a user's gaze direction, where the person is looking, and the movement of the user's eyes. This input method is usually achieved with infrared cameras that illuminate the eye by refracting light off the cornea and the retina to create distinct patterns called glints. Software like RemoteEye, developed by Hosp et al. [11], uses glint detection algorithms to calculate the gaze point and achieve high-speed eye-tracking capabilities. When applied to VR as an input device, an entire genre of interaction technique presents itself. Novel techniques like the ones evaluated by Piumsomboon et al. [24] show us the versatility of the eye gaze input device. In this evaluation, three selection techniques based on eye gaze were evaluated. The first, *Duo-Reticles*, uses one eye-gaze and one inertial reticle to perform selection. *Radial Pursuit* selects cluttered objects with smooth pursuit by creating a sphere of influence where the user looks and all objects within slowly separate for users to choose. *Nod and Roll* is a hand gesture-based interaction based on the vestibulo-ocular reflex. These techniques, in combination, highlight the four primary natural types of eye movements [27] that should be studied when constructing an eye gaze-based technique for selection. These include (1) *saccade*, a quick eye movement with a fixed end target (demonstrated by Duo-Reticles), (2) *smooth pursuit*, a smooth eye movement towards a moving target (demonstrated by Radial Pursuit), (3) vestibular-ocular reflex (VOR), an automatic eye movement that counters head movement when a user's gaze is fixed on a target (demonstrated by Nod and Roll), and finally (4) vergence when our eyes converge or diverge to look at targets of varying distance. Participants were tasked with finding a matching picture as quickly and accurately as possible. Participants were then asked to answer a 7-point Likert scale usability questionnaire to rank the input devices with statements like "It felt natural the use" and "I felt satisfied using it." The results showed that the Duo-Reticle was highly favored over the other input devices. When considering eye gaze as a part of the NeuroGaze input device, we needed to decide what technique to use to improve the usability of this input device and which would be best for our experimental design. At an early stage in our

experimental design, we wanted to focus on moving targets, but as we learned more about EEG, we realized that limiting kinematic activity, especially head movement, would be greatly beneficial to the success of a BCI. The evaluation of input methods that take advantage of saccade [10], such as Duo-Reticles, showed us how NeuroGaze could use the eye's natural ability to rapidly update new stationary targets. With this in mind, we were curious how quickly the eye could update to a new stationary target when compared to existing input methods, like VR controllers, that depended on the speed of a user's arm to achieve low completion times.

Luro et al. [20] evaluated VR controller accuracy compared to an eye gaze technique using the previously mentioned ray casting metaphor to perform aiming selection tasks [2]. To determine how demanding the input devices were, a raw NASA-TLX survey was distributed to participants. Participants also completed a system usability scale (SUS) survey. Before evaluation, participants were asked to complete an eye-tracking calibration test. Then, participants were asked to use their eye gaze to look at a target of interest and use VR controllers to select the object. The same evaluation was performed with VR controllers using a ray-casting technique. The evaluation results showed that the eye gaze input method performed on par with VR controllers with low statistical significance between task completion times across input devices. The survey results showed participants felt that the eye gaze input method produced a lower Physical Demand and Effort than VR controllers. At this point in our literature review, we determined that eye tracking is not only as performant as VR controllers but produces less physical demand and effort than VR controllers when selecting a fixed target. Under varying fields of view (FOV), eye gaze performed better than head-gaze-based selection [18, 24]. Based on our literature reviews, eye tracking seemed to be the most accurate, intuitive, and least cognitively intensive way to allow users in a VE to show intent for selecting an item. With BCIs still in our minds, we turned towards research in this area and the best techniques, hardware, and strategies for integration with eye gaze, given the limitations and scope of this Thesis.

2.1.3 Brain-Computer Interfaces

Brain-computer interfaces harness neural signals, usually translated into commands that can interface with a computer system. The area of study has broadened itself in the last 15 years, and its applications have increased in range, including technologies like robotics, Internet of Things (IoT), and VR [25]. In recent years, human testing of invasive neural interfaces, such as intracortical recordings, has been done. This process involves implanting neural interfaces composed of biocompatible materials to cure a user's existing limitations or enhance them. While it is important to be cognizant of invasive interfaces, the scope of this Thesis is strictly focused on signal acquisition using non-invasive methods (e.g., EEG).

An electroencephalogram, or EEG, is a non-invasive recording of electrical activity along the scalp. Specifically, EEG measures the voltage fluctuations resulting from ionic current flows within the neurons in the brain. [15]. EEG data has been used in various use cases, such as medical diagnostics for neurological disorders [13], cognitive and behavior and behavior research, and BCIs. With many areas of research using EEGs, standards must be practiced when placing electrodes on the scalp, or it becomes difficult to compare EEG data across evaluations. The collection of EEG data is dependent on the electrode system chosen.
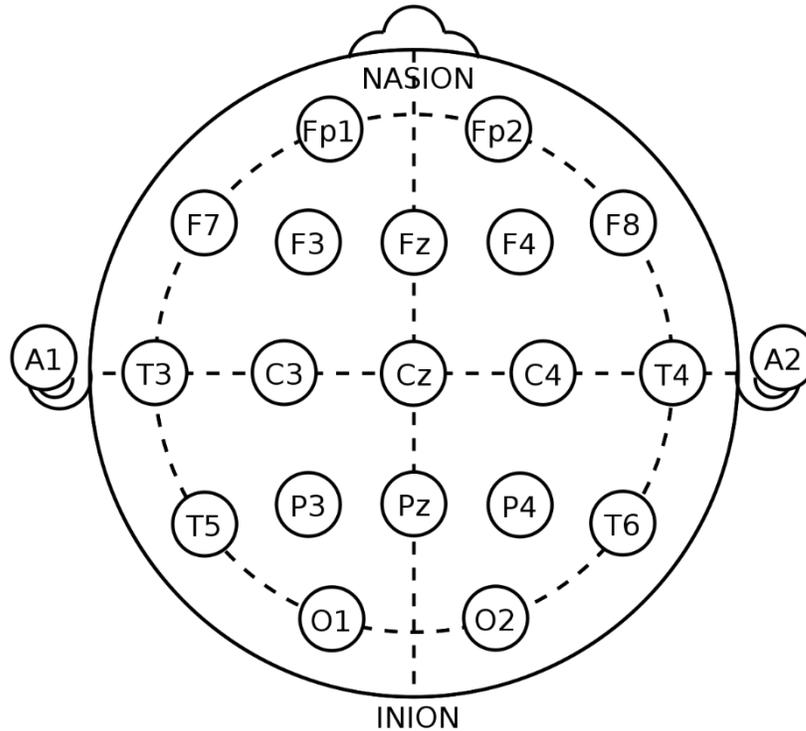
Figure 1: Electrode locations of International 10-20 system for EEG recordings.

The 10-20 electrode system, or International 10-20 system, is an internally recognized method to describe and apply the location of scalp electrodes in the context of an EEG test or experiment [15]. In Figure 1, we can observe a top-down view of the head where the nasion, the depressed area between the eyes, is the front of the head, and the Inion, the crest of the back of the skull, is the back of the head. The "10" and "20" refer to the 10% or 20% spacing between each adjacent electrode, starting from ear to ear. The starting letter is used to identify a lobe or area of the brain: central (C), occipital (O), parietal (P), temporal (T), and pre-frontal (Fp). The following number represents the side of the brain, whereas even numbers (2,4,6,8) represent the electrodes on the right side of the brain, and odd numbers (1,3,5,7) represent the left side of the brain. With standardized electrode placement, researchers build upon previous works with little variances in electrode placement on a participant's scalp. It should be noted that other systems exist, such as 10/5 and 10/10, but are not internationally recognized, nor do they have noticeable performance increases when evaluating the quality of the EEG data [13]. The 10-20 system is the current

standard for collecting EEG data, and NeuroGaze complies with this standard. We do not plan to diverge from this standard until other electrode placement standards are established or new positioning systems [14] are created and tested thoroughly.

Larsen et al. [18] recently published a paper evaluating selection in VR using a similar input method as NeuroGaze. The paper showcased a proof of concept (POC) called *SSVEP-Speller, which uses eye-trackers to preselect* a subsection of a virtual 3D keyboard to select a letter. EEG measures the temporal difference between the time it takes to close the eyes completely during a blink. Electrodes were placed over the occipital region at O1, Oz, O2, P3, P7, Pz, P8, and P4 to record EEG data of the visual stimuli. To collect data for their temporal blink classifier, participants were instructed to wear the EEG headset while holding a VR headset to their head and blink naturally with both eyes at about a rate of 60 blinks per minute. During the evaluation, participants were tasked with looking at letters they wanted to select while varying flickering levels occurred, turning their screen completely black to simulate a blinking sensation. The results showed that these researchers successfully classified intervals of blinking to minimize jittering that occurs during blinking during selection with eye gaze. This paper educated us on how an EEG data stream can be negatively impacted due to unwanted blinking artifacts. While we did not develop a trained machine-learning model for classifying noise and intent artifacts, this paper solidified our decision to find software that would handle this for the NeuroGaze system. In the initial pilots of the NeuroGaze system, we experienced issues with jittering due to blinking, which was handled for us in the Meta Quest Pro HMD. Independent of the Meta Quest Pro HMD, future work could benefit from software like Blink [1], designed to manage noise potentially inferring with eye tracking data. This would allow us to explore different, more compatible VR headsets.

Putze et al. [26] present research that is closest to our latest version of the NeuroGaze input device. Interestingly, their selection device was not designed for a 3D stereoscopic display but rather in a 3D flat screen-based environment. This distinction is crucial, emphasizing the different challenges and

interaction dynamics in less immersive settings. Their system, denoted *EEG+GAZE*, consists of a Tobii X60 eye tracker with a sampler frequency of 60 Hz and a BrainProducts 32-electrode EEG cap. Participants had 29 electrodes positioned respective of the 10-20 system at the following locations: Fp1, Fz, F3, F7, FT9, FC5, FC1, C3, T7, CP5, CP1, P3, Pz, P7, O1, Oz, O2, P4, CP6, CP2, Cz, C4, T8, FT10, FC6, FC2, F4, F8, FP2 and reference electrode positioned at Fz. EEG+GAZE used gaze to detect when a user is likely and unlikely to select an object. This is achieved by monitoring when users slowly move their eyes toward an area of interest versus quickly scanning the screen. During the window, when a user's gaze seems to want to interact with an object, the EEG headset records the temporal window. To collect data to train the classifier for the eye gaze and EEG, participants were tasked with following the key on a monitor with their eyes, and in the moments when the object's speed increased and slowed down, the data coming from these two devices created a window of interest. The evaluation consisted of 10-fold cross-validation for each participant, and the result was averaged. The data shows that their classifier produced a lower $F_{0.2} - Score$, which was the lower tolerance, measured in the number of windows and milliseconds. This suggests that their EYE+GAZE systems classifier had minimal false positives and, thus, more accurate classification for event-based evaluation scenes [26].

Continuing to investigate other novel BCIs, we were curious if we could use different modalities in conjunction with a BCI to accomplish our selection task. Riccio et al. present various research works using BCI, specifically independent of eye gaze. This is done to inform researchers on developing novel BCIs for users with late-stage amyotrophic lateral sclerosis (ALS) to interact with their environment. Individuals with late-stage ALS can suffer from the deterioration of oculomotor control and thus cannot use visual modalities as a medium to interact with their environment [28]. For this reason, Riccio intentionally avoids using eye gaze with BCIs and alternatively chooses to present auditory BCIs, tactile BCIs, and independent visual BCIs. For auditory BCIs, 16 published papers were discussed, specifically event-related potentials (ERP) based BCI for communication. Most of the studies presented users with acoustic

stimuli (target stimuli and non-target stimuli) through auditory cues and were asked to concentrate on target stimuli. These auditory cues consisted of tones, environmental sounds, and words, and the user's attention to either stimulus was recorded using attention artifacts in an EEG stream. The studies showed high variability across different tasks and subjects within each experiment, primarily due to varying focus levels within participants. Six papers were discussed for tactile BCIs, focusing on the feasibility of steady-state somatosensory-evoked potentials (SSSEPs) and somatosensory ERP-based BCIs [28]. Using vibrations, tactile stimuli were delivered to different body parts, such as fingertips. Like auditory BCIs, users were evaluated on their ability to focus on targets based on tactile cues, and their EEG data was recorded to monitor how the brain responded to tactile stimulation. Tactile and auditory BCIs performed similarly and heavily relied on the user's ability to focus on the target stimuli when presented with stimuli of the same type through a separate input channel. Gaze-independent visual BCI included research that utilized covert attention and rapid derail visual presentation (RSVP) to classify intent by focusing without looking at the object of interest. Across experiments, high accuracy was achieved; however, the high cognitive load expressed by participants suggests a possible performance decrease with prolonged use of these techniques. Although the final design of the NeuroGaze system would not be ideal with users with ALS, it opens possibilities for individuals with other physical limitations that do not affect eye movement. Individuals suffering from severe motor impairments due to spinal cord injuries or stroke survivors with limited limb modality could benefit from our system. By leveraging EEG signals and the ability to control gaze, these individuals could interact with various applications and devices, enhancing their ability to communicate and interact with their surroundings without needing physical input devices.

This literature review has shown our journey for determining the best possible design of the NeuroGaze system by breaking apart this new input device's subcomponents. VR controllers present us with the ray-casting interaction metaphor and the physical strain problem. Eye gaze provided us with comparable saccade interaction methods, like Duo-Reticles, and evaluation with fixed points in a 3D environment

for our system evaluation. Finally, an overview of the research done in BCI in 2D and 3D environments shows us that it is possible to classify these variances in neuron voltage. By combining these elements, we have designed the NeuroGaze input device to challenge the efficiency, accuracy, and cognitive ease with which BCIs can operate in virtual environments.

CHAPTER THREE: SOFTWARE & SYSTEM DESIGN – NEUROGAZE

3.1 System Overview

NeuroGaze is an input device proof of concept (POC) developed explicitly for selection in a VE. NeuroGaze aims to empower users with a selection device that allows them to interact in VR in a new way to improve accuracy and efficiency compared to traditional input devices, such as VR controllers. Our system uses eye trackers built into the Meta Quest Pro HMD to activate a *hover state* cue on objects they can interact with in the scene. When a user's eye gaze is broken by one of these interactable objects, the object slowly and continuously grows until it reaches its maximum scale. Conversely, when a user's eye gaze moves off an interactable object, the object will shrink continuously until it is returned to its original scale. The user's eye gaze interacts with objects in the scene using the ray cast interaction metaphor through a white line renderer to represent the midpoint between the user's eyes at any moment. NeuroGaze uses the Emotiv EPOC X EEG headset for the EEG data stream and the EmotivBCI program to create training profiles that collect EEG data, handle EEG noise sanitization, and classify EEG artifacts. The Unity Game engine and Meta XR All in One SDK were used to complete the development of NeuroGaze, integrating the hardware components and software solutions to create a seamless and intuitive user experience within VE. This integration allows NeuroGaze to provide a hands-free and accurate method for interacting with and selecting objects in VR, showcasing improvement over conventional input devices regarding user preference and interaction fidelity. The implementation of NeuroGaze represents a novel approach to VR interaction, leveraging the precision of eye tracking and the sophistication of EEG data analysis to challenge existing input devices. By utilizing the Emotiv EPOC X EEG headset and EmotivBCI software, NeuroGaze minimizes noise and classifies user intentions based on neural activity, demonstrating the potential of combining multiple biometric inputs for enhanced control within a VE.

Pilot studies were used to ideate and iterate over our software design and apparatus. These studies comprised eight people with little to no experience with VR systems. Our objective was to learn what drawbacks users felt towards the NeuroGaze system so we could identify them early in the development process. These were the most significant lessons learned:

1. Individuals with more hair must put their hair up in less comfortable positions to accommodate the EEG node placement of the EEG headset and the ergonomics of the EEG and VR headset.

2. Without the eye-casting ray and the "growing and shrinking" metaphors, users have a tough time telling which interactable they were looking at and could select.

3. One eye-casting ray per eye distracted users, and they expressed frustration. On occasion, each of the users' eyes could be looking at different objects, causing double selections.

    a. Averaging distance and forward vector between eyes and updating the ray to originate from this position was more intuitive and performant for users.

4. In our EEG training loop:

    a. Giving users a countdown well before starting the Emotiv training software gave them time to recall their unique thought and increased the quality of the EEG training data.

    b. Allowing users to think of their thought for selection leverages their unique experiences and generates EEG artifacts that are easier to classify for selection. Standardizing what users should think about (e.g., crushing the object) decreased the quality of the user's EEG training data and should be avoided.

    c. Completely train one mental command (e.g., neutral state or shrink state) before moving on to the next generates artifacts that are easier to classify, forming a solid EEG baseline.

d. Individuals who believed they were selecting the interactable object during training seemed to yield higher-quality EEG data.

Throughout the development process, a user-centered design philosophy [17] was emphasized, ensuring that NeuroGaze not only advances the technological capabilities of VR interfaces but also addresses users' practical needs and preferences. By focusing on ease of use, accessibility, and the reduction of physical exertion, NeuroGaze aims to make virtual interactions more enjoyable and less tiring, particularly for users who may find traditional input devices cumbersome or completely inaccessible. Active sensors from both the EPOC X and Quest Pro were used, as the user is required to wear both headsets simultaneously. This input device was chosen to be multimodal to leverage the advantages of both devices. By decoupling the main input channel (EPOC X) from the secondary channel (eye gaze), we are hoping to reduce the cognitive load [24] that would be required of these input channels individually. Using EEG or eye gaze alone can be problematic due to stimuli that might gain a user's visual attention or cause unwanted artifacts in the EEG data stream. This would lead to forced error and reduce the usability of this input device. Using a multimodal approach, we aim to reduce error by using redundancy flags. One flag (via a *Hover State* from eye gaze) is used for intent, and another flag is used to trigger selection (via pull command from EEG stream), given that both flags are active. Increasing the number of input channels should increase recall rates [17] and make a more successful selection. The specific combination of these modalities should be catered to the task [17].

The design of the NeuroGaze system included many constraints that challenged our initial concept during the ideation phase. The lack of ergonomic compatibility between an EEG headset and the Meta Quest Pro proved challenging. Three different EEG headsets (EPOC X[1], MUSE II[2], and Insight II[3])

---

[1] https://www.emotiv.com/products/epoc-x
[2] https://choosemuse.com/products/muse-2
[3] https://www.emotiv.com/products/insight

were all tried during pilot studies to evaluate comfort and usability, with the EPOC X ultimately being chosen for our final evaluation design. We believe that the comfort lost by wearing both headsets simultaneously will be compensated with the ability to interact with items completely independent of limbs that would cause discomfort from repetitive motions. Unlike NeuroGaze, the physical ergonomics of VR controllers require users to undergo repeated isotonic contractions that can be tiring, uncomfortable, and frustrating to use over time. Similarly, the pinch gesture required from the Eye Gaze + Hand Gesture input device requires users to make isotonic contractions if they choose to raise their arm before the pinch gesture or isometric contractions by keeping their hands at waist level while performing the pinching gesture [30]. Either way, an opposition motion is required by opening and closing fingers to select objects. NeuroGaze does not require these types of flexions; only a rotation of the head and body is needed, which the other two input devices also demand. Users with physical limitations can potentially gain a sense of autonomy by using our system to interact with VEs without the intervention of a third party.

3.2 EEG Hardware

When deciding which EEG headset to use for the NeuroGaze system, our immediate concern was ensuring we chose a headset that gave us a budget-efficient high-fidelity EEG stream. In the context of this Thesis, we define high fidelity according to the following four metrics: signal clarity, high spatial (precisely locating brain activity) and temporal resolution (detecting deltas in brain activity in very short intervals of time), artifact minimization (signals not of cerebral origin; e.g., eye blinks and head movement), and producibility and reliability; that is, being able to have similar EEG data for the same user given the similar conditions. We wanted to keep our budget in the $1000 range or less due to this Thesis not being funded. Additionally, ergonomically, we needed to ensure that the EEG headset could be worn underneath the Meta Quest Pro VR HMD. If possible, we would also prefer the electrodes to be different from the position where the VR headset would hold the user. Although we are limited by these two pieces of hardware not designed for one another, reducing pressure points on the user's head would significantly

17

increase comfort and reduce unnecessary artifacts from discomfort. Getting data off these headsets was also a factor we had to consider. How many channels do we need to access? How can we export EEG data, and does the headset interface with any existing data collection and synthesis software? All these unknowns needed to be investigated to find the most "out of the box" apparatus and software architecture for our POC. We reviewed many different EEG headsets, but the following were the three we narrowed down for the final design for the NeuroGaze input device.

3.2.1 Emotiv Insight II

Emotiv is a pioneer in the field of neuroscience and has developed high-quality, accessible EEG technology. Their products are used worldwide for research, neuroeducation, and brain-computer applications, focusing on producing comprehensive brain monitoring solutions. Emotiv's commercial EEG headset was not only budget-friendly but also would aid with tracking the quality of the EEG data we collected through the EmotivPRO and EmotivBCI programs. The Insight II model from Emotiv has five channels with two reference sensors, focusing on critical areas of cognitive state monitoring. The polymer sensors are arranged according to the international 10-20 system, and the locations of the sensors are in the AF3, AF4, T7, T8, and Pz positions, as seen in *Figure 2*.
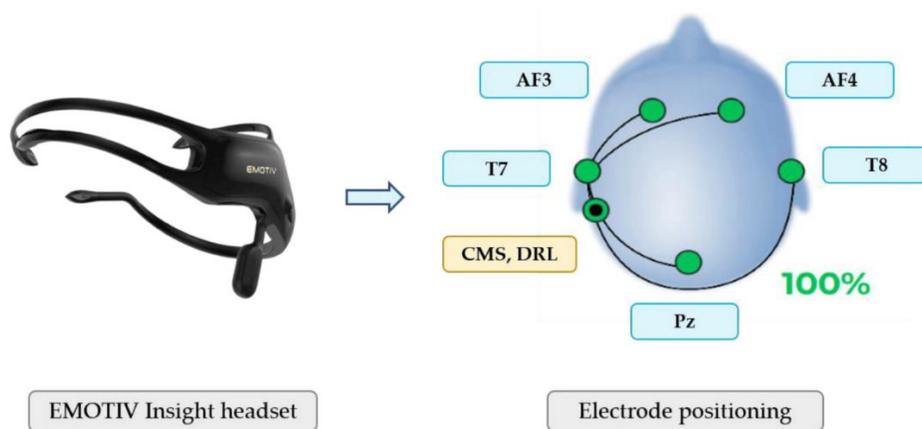
Figure 2: Emotiv Insight II (left) and electrode placement (right).

Source: Suárez, M. (2023)

These sensor locations focus on the frontal and parietal areas, important for cognitive and emotional insight. This headset has a sampling rate of up to 128Hz and connects to any computer that supports a Bluetooth 4.0 connection. We used this headset for a couple of weeks, and while the contact quality was good, this headset was highly uncomfortable to use for a more extended period. This is partially because the Meta Quest Pro HMD distributes its weight to the front of the headset, and the AF3 and AF4 electrodes are in this same position. Additionally, these two nodes were required to be a three-pronged contactor (See *Figure 3*), which was very uncomfortable and sometimes even painful to wear for a prolonged period while wearing the Meta Quest Pro. For this reason, we removed this headset from our apparatus and looked for alternatives for the NeuroGaze input device.

Figure 3: Emotiv Insight II polymer sensors: three-pronged (left), one-pronged (right).

3.2.2 Muse II

Muse also contributes to neurotechnology, offering devices that promote mental wellness through guided meditation and biofeedback. They offer the Muse II, a commercial EEG headband that is the more cost-efficient of the three EEG devices we were interested in using. Additionally, this device is a hand band, not a headset, so the device did not have any nodes or prongs that could be uncomfortable to wear.

The EEG headband comprises seven sensors (3 reference sensors, two forehead sensors, and one grounding sensor behind each ear) and a heart monitor (See *Figure 4*). The Muse II, worn like a pair of glasses, pairs nicely with the Meta Quest Pro HMD. Despite having sensors on the front of the headset, the flat design of these sensors prevents discomfort.
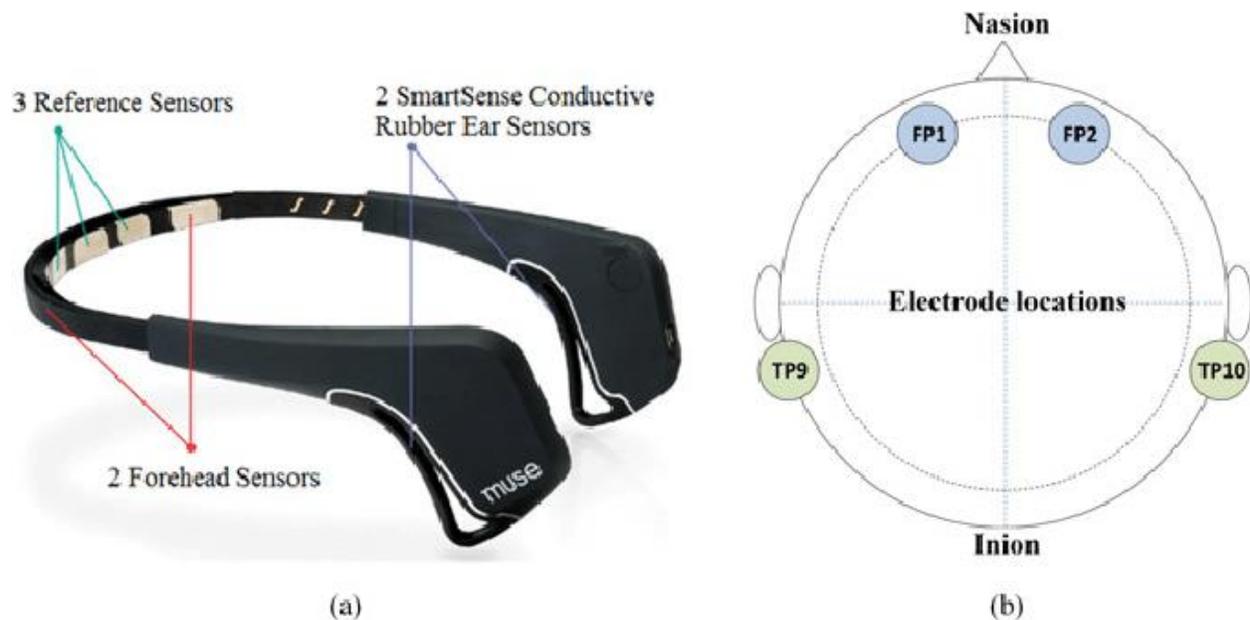
Figure 4: (a) Muse II EEG Headband with sensor locations, (b) electrode locations in International 10-20 system.

Source: Liu, R. (2020)

To pull the stream and visualize the EEG data from the headset and store it, we attempted to use the open-source application Mind Monitor (See *Figure 5*). To classify data, we used the Edge Impulse software to import the raw data recorded from Mind Monitor and construct a neural network classifier. We collected data on thinking *right*, *left*, and neutral in 5-second increments per thought. The data is stored as one CSV per session that Edge Impulse can use for an 80/20 training-testing split. Once the model trained on our data, the model performed inference with our testing split with the highest accuracy of 40%. After further testing this pipeline, we concluded that the noise artifacts impeded the EEG data were clouding the data too much to classify intent with any confidence level.
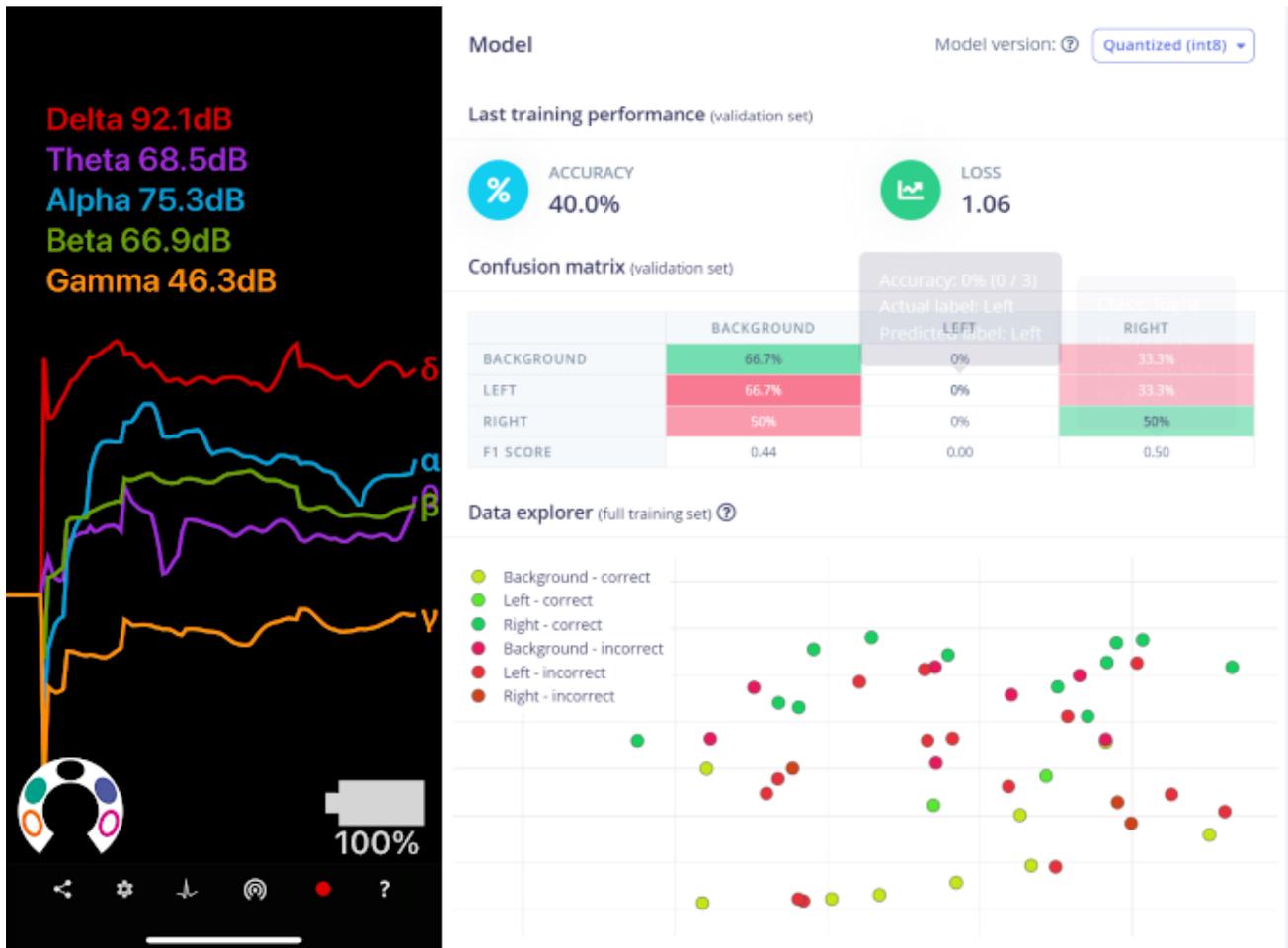
Figure 5: Mind Monitor visual of EEG data stream (left), inference performed on Edge Impulse model trained on EEG data from Mind Monitor (right).

The drawback of this headset is the limited channels we have access to for streaming EEG data. Additionally, the Muse SDK lacks any functionality for cleaning or classifying the data. If we selected this device, we would be required to collect ample data to train our own machine-learning classifiers for identifying noise artifacts (e.g., eye blinks, random kinematic movements) and identifying when a user wants to interact with an object in the scene. Due to time restrictions and the POC nature of the NeuroGaze input device, we did not want to dedicate resources to developing machine learning classifiers. Instead, we wanted to focus on finding the best configuration and techniques to use. For this reason, we

21

chose not to use this headset and looked towards less cost-efficient Emotiv headsets that can handle our data processing pipeline.

3.2.3 Emotiv EPOC X
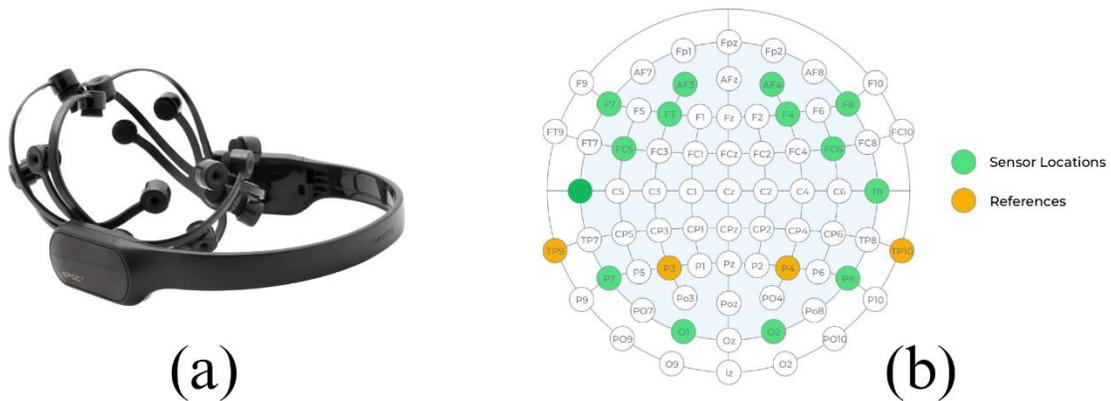
Figure 6: (a) Emotiv EPOC X EEG headset, (b) International 10-20 electrode placement for Emotiv EPOC X.

In selecting the EEG headset for the NeuroGaze system, the decision-making process extended beyond merely assessing technical specifications; it also encompassed considerations such as user comfort, ease of integration, and relevance to the application. When choosing the EEG headset for the NeuroGaze system, considerations extended beyond technical specifications to consider factors such as user comfort, integration ease, and application relevance to our evaluation task. The Emotiv EPOC X stands out for its comprehensive 10-20 system channel coverage (sensors at AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4, with references at TP9, P3, P4, TP10) and an extensive software ecosystem (See *EEG Software*). Although this headset is the most cost-efficient of the three candidates, we decided it was well worth it. Despite its functionality and integration possibility while being a commercial headset, the location of its electrode connections makes it ideal for compliance with the ergonomics of the Meta Quest Pro VR HMD. Additionally, the nodes are felt, and saline solution can be added through the refillable sensor hole on each sensor (See *Figure 19*). This design allows maximum comfort, even at pressure points, and

easy access to wetting dried-out electrodes with saline solution even after the user has put on the NeuroGaze system. During evaluation, participants are told to think of a motor command they would perform if they weren't stationary in a VE. With this in mind, we wanted to understand how effective the Emotiv EPOC X would be in classifying motor imagery and how internal motor acts are rehearsed in working memory [6]. Fakhruzzaman et al. [7] used the Emotiv EPOC X to attempt to classify motor imagery using BCI scenarios. Participants were told to imagine the motor action they would prefer during the following testing scenarios:

1. Left Hand Movement

2. Right Foot Movement

3. Simultaneous Left Hand and Right-Hand Movement

4. Simultaneous Left-Hand Movement and Head Nodding

5. Simultaneous Right Foot and Left Foot Movement

6. Simultaneous Right Foot Movement and Head Nodding

When attempting to classify the EEG data that correlate to these motor imageries, the Scenario 1 test results had a 76.67% success rate, Scenario 2 test results showed a 91.67% success rate, Scenario 3 yielded a 28.33% success rate, Scenario 4 got 13.33% success rate, Scenario 5 showed 60% success rate, and Scenario 6 success rate was 53.33% success rate. The main takeaway from this study is that the Emotiv EPOC X could not handle complex motor imagery classification, so we would need to limit participants to a single, consistent thought (e.g., thinking of selecting objects with their dominant hand). Additionally, the paper commented that while the classification for these tasks is not amazing, the EPOC X is sufficient for a consumer-grade product for POCs. Less commercial hardware should be used for further investment into our research question. Given our limitations, it was decided that the Emotiv EPOC X would be sufficient for our use case for this Thesis, and based on the results, further financial investment would be made.

3.3 EEG Software

When deciding which EEG software to use, we prioritize software that integrates well with the Unity Game Engine. We needed the classification of the user's intent to select an object in the VE to have as little latency as possible. This is because eye tracking will be used for a hover state, and if the classification is too delayed, the user could look at another object and accidentally select an object they didn't mean to. At this point, we had already decided that we were going to use the Emotiv EPOC X as our EEG device and Emotiv already has software built to stream, clean, and train the EEG data coming off their headsets. Using the EmotivBCI program, we made *training profiles* to associate EEG data with a user and train the model on specific *mental commands*. The Emotiv Launcher handled the authorization of our Unity program's access to these training profiles and all the data associated with them. With most of the software development efforts of building our own classification model alleviated, we could focus on integrating our Unity application with Emotiv's Cortex API.

3.3.1 EmotivPRO

The EmotivPRO software handles the raw stream of EEG data from the EPOC X. This program also handles EEG preprocessing. This program cleans all our raw EEG data with a high pass filter. All 14 channels on the headset can also be played back (See *Figure 7*). The raw stream is measured in microvolts converted directly from a 14-bit unsigned integer. The program has performance metric data that classifies and charts emotional states (e.g., Attention, Engagement, Excitement, Interest, Relaxation, Stress). We do not have these performance metrics in the NeuroGaze system, but it could be interesting to investigate in future works. This program was critical in handling the initial stages of the software pipeline and allowed us to remove unwanted artifacts from our data, like blinking and small kinematic movements. Although this software does not give us direct access to training-specific artifacts to remove from our EEG data stream, the EmotivPRO software is great for POC-type projects where little budget and short

time are both major contributors.
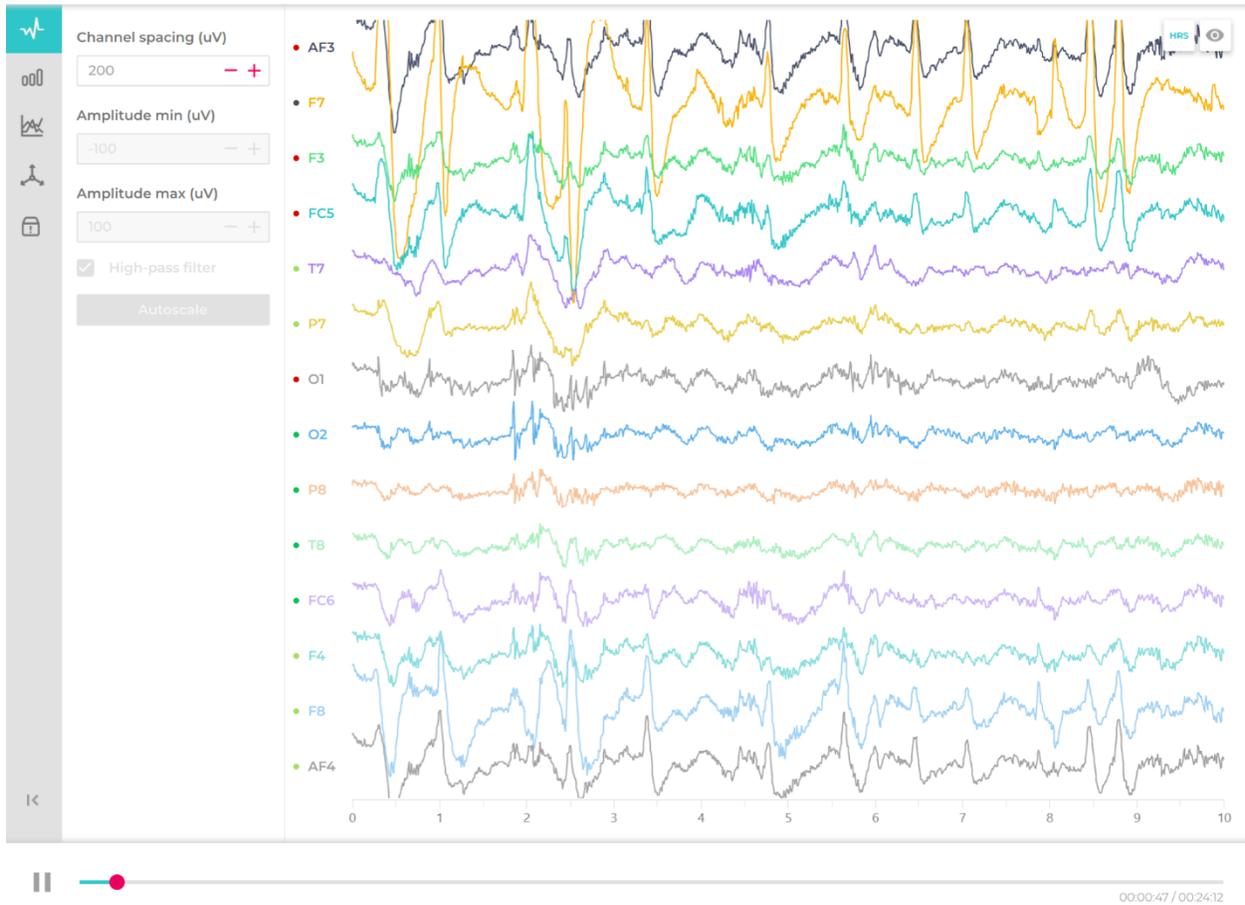
Figure 7: EmotivPRO program – data stream playback from Emotiv EPOC X.

3.3.2 EmotivBCI
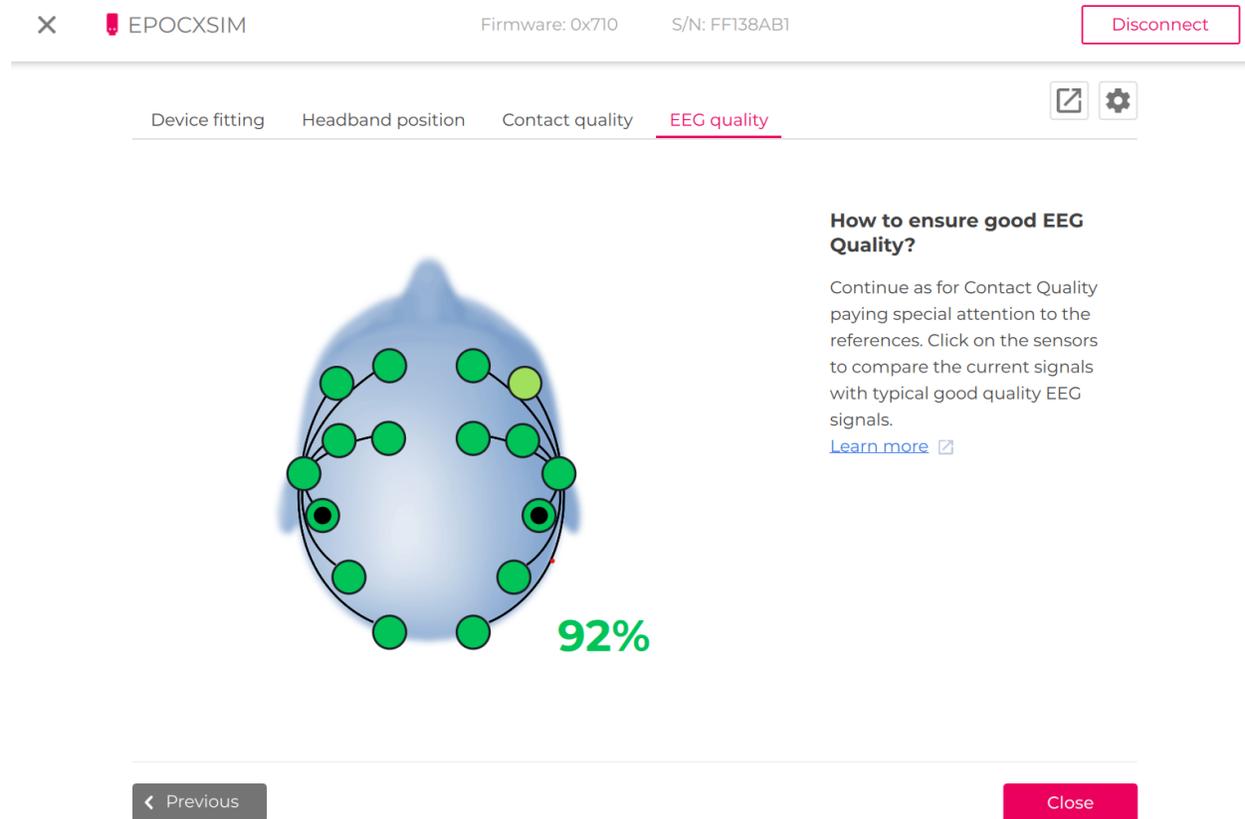
Figure 8: Emotiv BCI program – EEG quality user interface.

The EmotivBCI program is the main piece of software our Unity program interfaces. This program handles training profiles for each participant with a unique identifier (See *Figure 9c*) that we can use to load any training profile into our Unity program. Before training any of these profiles, we must check the contact and EEG Quality (EQ) currently streamed from the headset. The Contact Quality (CQ) and EEG Score (ES) are calculated individually for each node. They are ranked on the sensor map as follows: black means *very bad* or no contact or EEG signal is detected from this node, red means the contact or ES is *bad*, orange means the contact or ES is *average*, green indicates the contact or ES is *good* (*Figure 9b*). CQ is just the measurement of the impedance of the channel for that node or channel. The CQ measure and the EQ are not the same. However, the EQ can only be as good or less than the contact quality, so this

value should be 100% all the time to ensure the highest quality EEG stream. EQ is a bit fickler, and achieving a consistent 100% EQ like CQ is not likely. ES is a numeric score based on the following metrics:

1. Contact Quality (CQ): impedance measure indicating the quality of the electrical signal passing through the sensors and reference (0 or black – very bad, 1 or red – bad, 2 or orange – average, 4 or green – good)

2. ML Signal Quality (SQ): a machine learning algorithm trained on high-quality EEG recordings collected by the Emotiv Research team. Data is accessed in 2-second increments (0 or black—very bad, 1 or red—bad, 3 or light green—good, 4 or green—great).

3. Signal Magnitude Quality (SMQ): measures a single amplitude. Even if CQ and SQ are high, but the signal amplitude can still be low which would lead to an undetectable Fast Fourier Transform (FFT). (0 or black – very low, 1 or red – low, 4 or green – average).

The color corresponding to the minimum value out of these three metrics is displayed in the EmotivBCI program under the "EEG Quality" tab. (See *Figure 8*). EQ is a percentage calculated by taking the minimum three ES of all available nodes, summing them up, and dividing by the maximum ES of the same three nodes (value of 4 for CQ, SQ, or SMQ). The EQ can also be presented as an equation:

$$EQ = \left(\frac{Lowest\ ES_1 + Lowest\ ES_2 + Lowest\ ES_3}{\text{sum of max } possible\ ES\ for\ 3\ channels} \right) \times 100 = \left(\frac{Lowest\ ES_1 + Lowest\ ES_2 + Lowest\ ES_3}{12} \right) \times 100$$

3.3.2.1 Training Data Collection & Mental Commands

After confirming that our contact quality and overall EEG quality are high, we begin training a profile on mental commands. A mental command is a recognized pattern in brain activity. When triggered, a mental command allows for the setting of an event handler to respond; this forms the core of the NeuroGaze system. Although these mental commands have a set name in the EmotivBCI (*Neutral, Pull, Push,*

Lift, Drop, etc.), if the same thought is consistently producing a recognizable pattern in brain activity, it does not matter if you're thinking of something different than what the mental command describes (e.g., thinking of making an item shrink when training the *pull* mental command). Before you can train other mental commands, you must start by training the *neutral* mental command. The *neutral* mental command is the baseline state representing brain activity when the mind is relaxed, at ease, and not focusing on a particular thought.
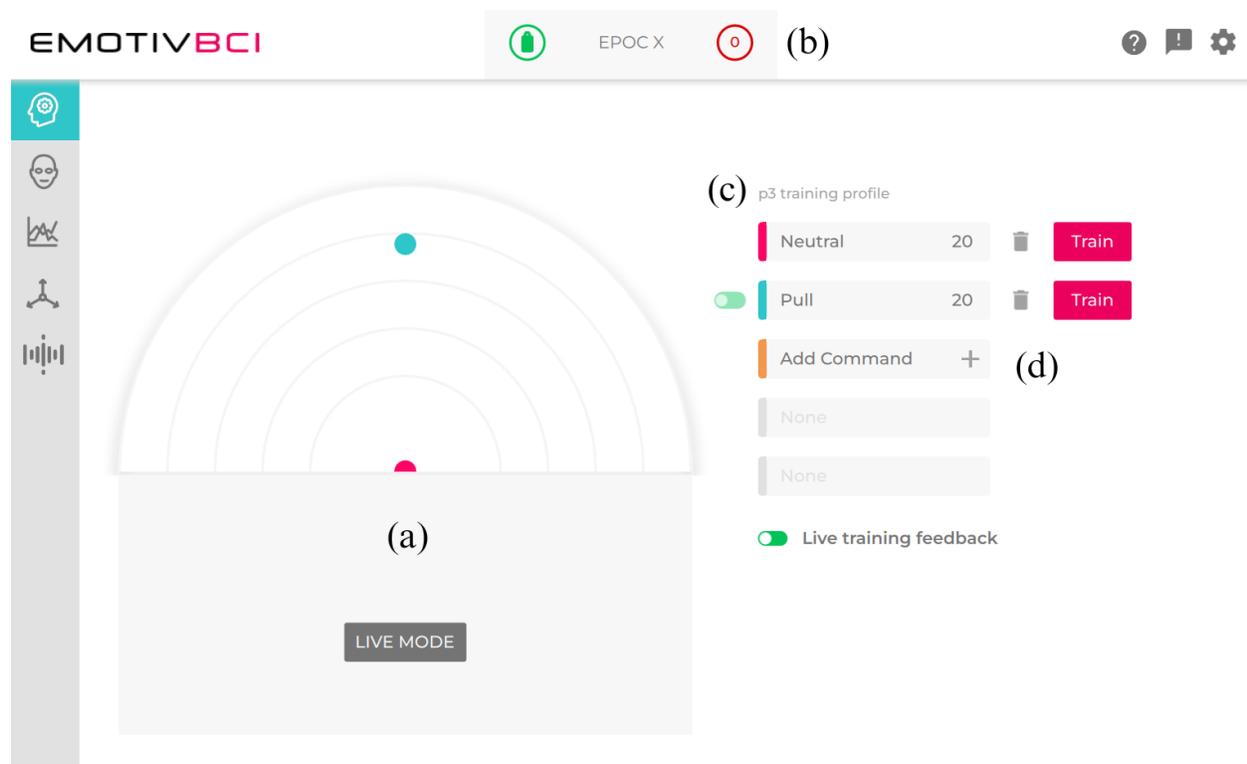
Figure 9: EmotivBCI program – training profile menu: (a) brain space diagram, (b) connected headset and EEG quality, (c) user training profile name, (d) training profile mental commands.

When training a mental command, the goal is performing a thought that is reproducible and separable from other mental commands. When training multiple mental commands, it becomes harder to distinguish between them. Emotiv recommends linking each mental command to a single and unique modality to alleviate this. An example of this could be visually seeing an object move up during a lift command training, hearing an object move towards you during a *pull* command training, and gripping your fists

during a drop command training. Additionally, if you have a disposition towards a single modality, picking a unique thought within that modality can also be very effective (e.g., a musician linking a mental command to thinking about what sounds different notes make). After clicking the Train button, mental command training sessions last eight seconds, and a training screen will appear. At this point, interactable objects will appear and begin performing the mental command you chose, and a timer decreases until the training time expires. After two training sessions, the EmotivBCI application provides feedback on how good the training was on a scale from 1-100 based on previous training sessions. After a session, you will see a number representing how good the session was, with a goal of 75 out of 100. You can accept or reject the training and try again (See *Figure 10*).
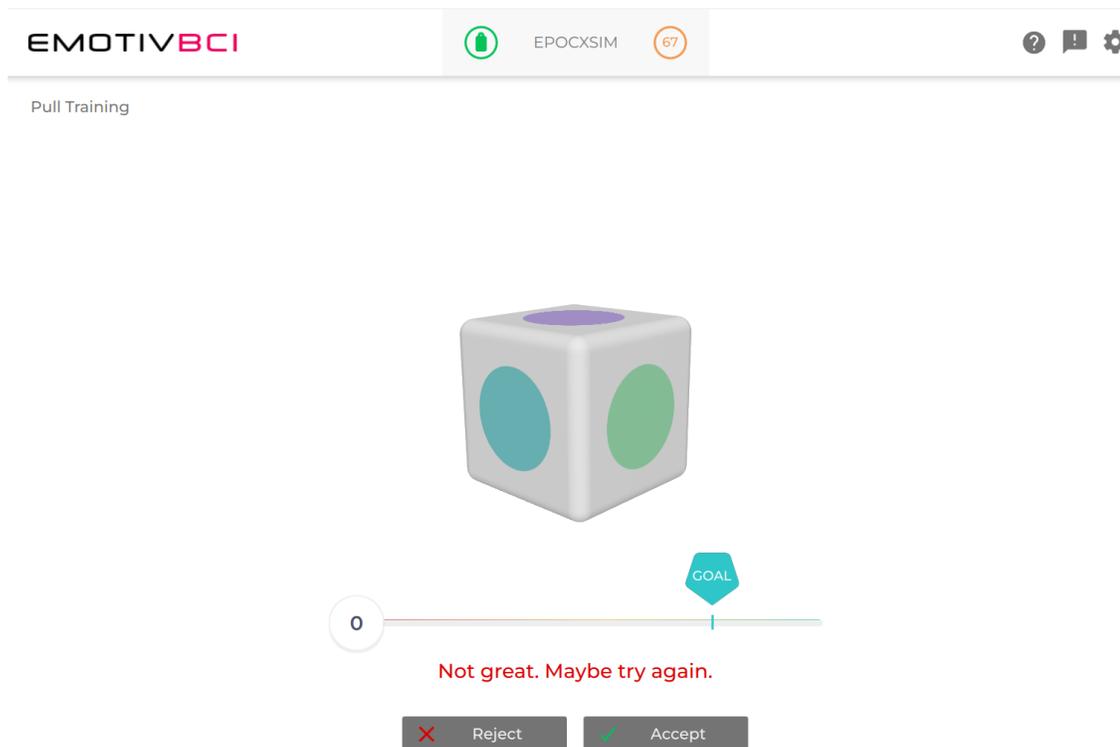
Figure 10: EmotivBCI Training user interface.

During the NeuroGaze evaluation, only training sessions valued at 75 or higher were accepted. Additionally, each mental command was trained for 20 high-quality training sessions. Due to the nature of

the evaluation, we only needed two mental commands: *neutral* and *pull*. The *neutral* state is represented by the user's mind when it is calm and wandering; the user does not want to select an item during this mental state. The *pull* mental command selects items of interest (objects that have met their eye gaze) in the scene. Users were prompted to think about a particular action and thought of their choosing. Based on the research done by the Emotiv Research team, we believed it would be unwise to attempt to standardize these thoughts and instead leverage the unique experiences of each participant and let them choose a thought that felt most comfortable and familiar to them. With mental commands trained, we can now provide the Unity game engine with the user's unique training profile to trigger events in the scene depending on the command sent to Unity via the Cortex API.

3.3.3 Cortex API

To access the training profile and mental commands from the EmotivBCI software, we needed to use the Cortex API from Emotiv to integrate with Unity. An EmotivID and paid License is required to create Cortex Apps. Cortex Apps are linked to your EmotivID, and it's how you and Emotiv can keep track of projects with access to your data. Once you have created a new Cortex App, you will be presented with a *Client ID* and *Secret ID,* which you will need to provide to the EmotivSDK and approve the third-party application for EEG data to start flowing to Unity.

3.4 NeuroGaze Design Architecture

The architecture of the NeuroGaze software design is split into two main sections: the Training Loop for creating and training profile for the EmotivBCI program and the Selection Loop for segmenting and classifying mental commands from the EEG stream. At a high level, the NeuroGaze input device pipeline is as follows:

1. Users create a new training profile and train the *neutral* and *pull* commands.

2. Using an EmotivID, a new Cortex App is created, and a new Client ID and Secret ID are generated.

3. The Client ID, Secret ID, Cortex App name, and unique training profile name are set in the MentalCommands and Config C# scripts.

4. The requestAccess API is called from Unity to Cortex and the Unity application is granted EEG data access via the Emotiv Launcher.

5. Once access is granted, the EPOC X connects to the PC running Unity via Bluetooth.

6. The controlDevice API with the "refresh" command is called from Unity to start scanning for headsets every frame.

7. The queryHeadsets API displays available Emotiv headsets to connect to.

8. The controlDevice API with the "connect" command specifies which headset the Unity application wants to connect to from the queried list.

9. The Authorize API is called to get a Cortex token.

10. The createSession API is called to open a new session for BCI streaming.

11. Call queryProfile API to get a list of training profiles associated with Emotiv ID

12. Load the user's training profile by calling LoadProfile with the user's unique profile name.

13. Subscribe to the EEG data stream from the EPOC X using SubscribeData.

14. Every frame, call GetMentalCommand from created instance of MentalCommands.

15. Every frame, check if the eye gaze is intersecting with a selectable object and mental command is *pull*.

    a. If both are true, select the object.

    b. If either is false, do nothing.

3.4.1 Training Loop
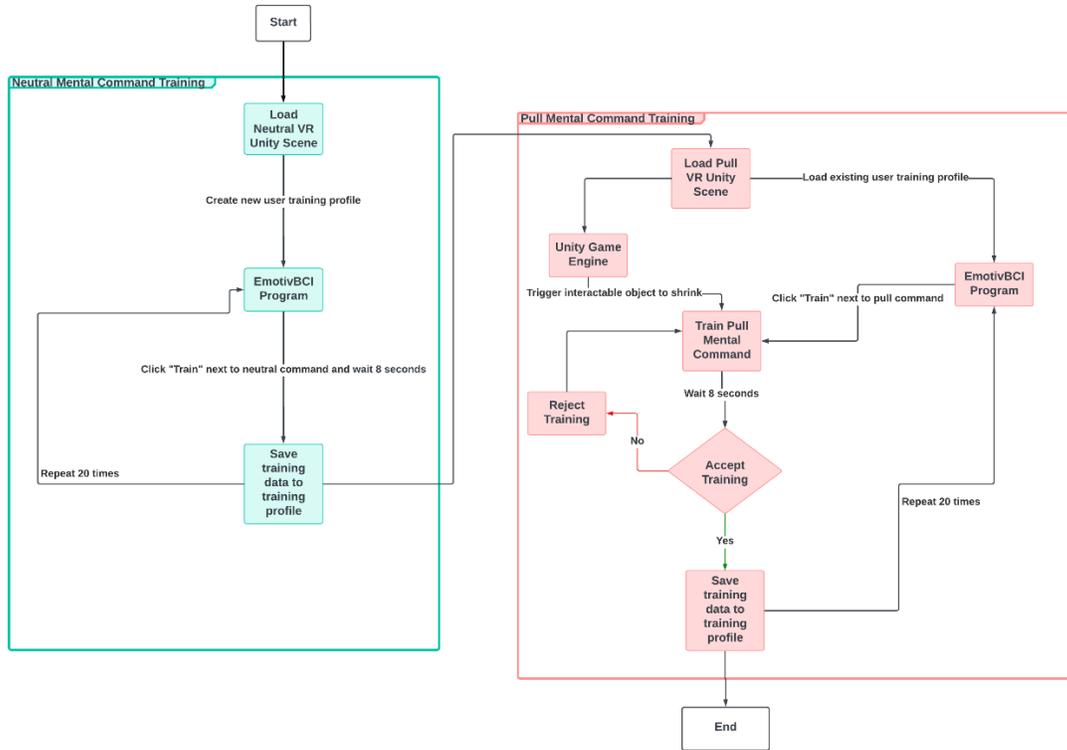
Figure 11: Training Loop System Architecture Diagram.

To train a new training profile, we first create a new training profile in the EmotivBCI program. Then, we load users into the Neutral Command Unity Scene, and they are prompted to look around the scene. We now click the *Train* button next to the *neutral* mental command. This repeats 20 times to establish a solid baseline for the training profile. Next, we train the *pull* mental command used in the evaluation scene to select an interactable object. Users are loaded into the Pull Command Unity Scene. This scene contains a single red interactable object that users will attempt to select. Using the training profile already trained on the *neutral* mental command, we click the *Train* button next to the *pull* mental command. Immediately after, we click the *Start Shrinking* button in the Unity program to give users the illusion that they are selecting the interactable object. This Wizard of Oz approach is done to get the most accurate EEG stream possible for a user to select an object. When a training round has been completed, the

EmotivBCI program presents us with the quality of the training, and we can either accept it or reject the training, and it will not be added to the user's training profile. This training is also done for 20 successful training rounds to ensure a distinct mental command for *pull*.

3.4.2 Selection Loop
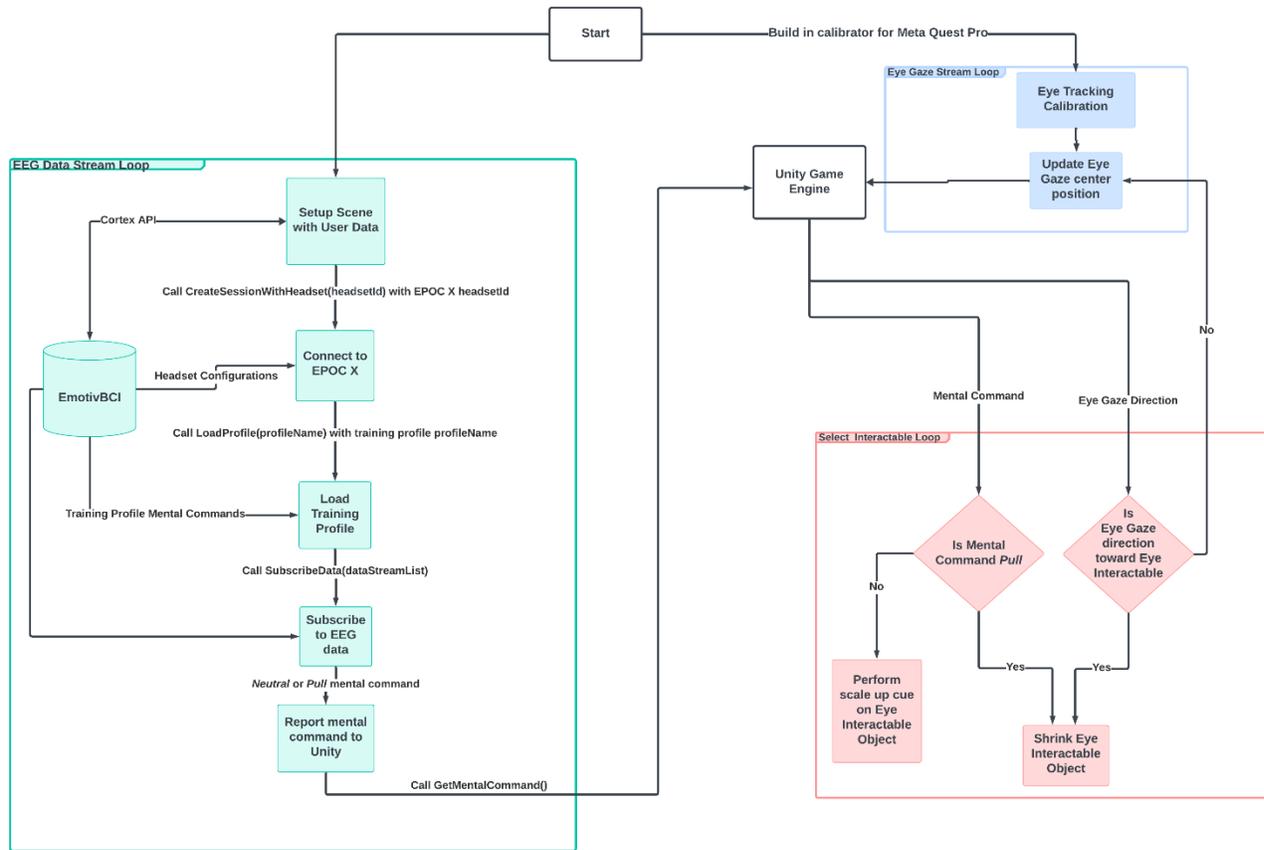
Figure 12: Selection Loop System Architecture Diagram.

The selection loop (See *Figure 12*) handles the data exchange between the EPOC X headset, the EmotivBCI program, and the Unity Game Engine. Once the Meta Quest Pro eye trackers have been calibrated and the Unity Game Engine has been configured (See *Mental Commands Manager*), the program will create a connection between the EmotivBCI program and the EPOC X, load the training profile (See *Training Profile Loop*) from EmotivBCI, and subscribe to the EEG data stream from EmotivBCI. For

every frame (approximately 16 milliseconds), the GetMentalCommand method is running and provides the Unity Scene with either the *neutral* or *pull* mental command. The direction of the user's eye gaze is also recorded in every frame to check if their gaze is met with any interactable object. If it is, the interactable object will grow to a fixed scale to provide users with visual feedback. Once their gaze leaves the object, it will return to its original scale. If the user's eye gaze meets an interactable object and the mental command being classified and returned from the EmotivBCI program is *pull,* the interactable object shrinks until and disappears (similarly to how the object shrinks during the training loop) to show the user they have selected that object.

3.4.3 Unity Game Engine
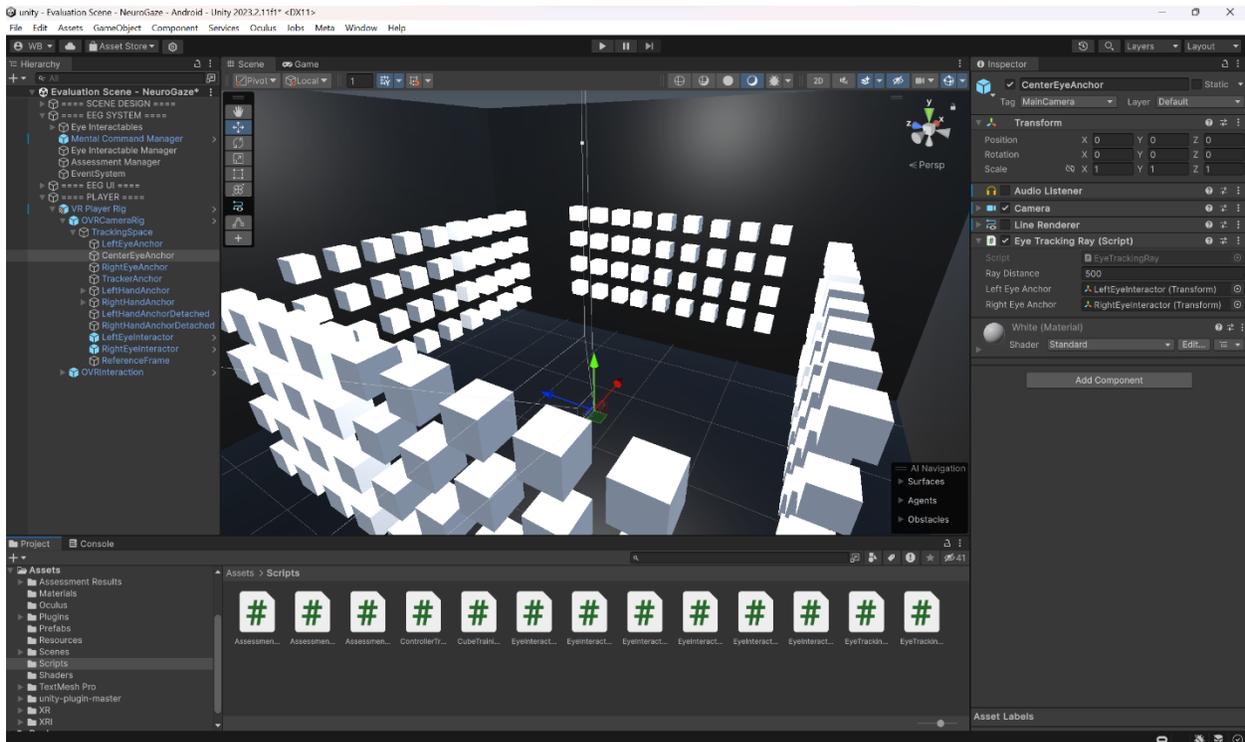
Figure 13: Example Unity Scene – NeuroGaze evaluation scene.

The Unity Game Engine played a pivotal role in developing the NeuroGaze system. Unity provided us a platform to build our VR application and test the NeuroGaze system in a VE. Unity's Game

Object and Prefab components allowed us to configure structures to handle replicable logic for interaction and assessment management. Integrating NeuroGaze with Unity allowed us to use the Meta XR All-in-One SDK to create a VR player rig and interactable objects that users can select while in VR. Using C# scripts, we were able to program logic for handling eye tracking, hand gestures, selection, and a gameplay loop.

3.4.3.1 Scenes

Scenes in Unity are where the evaluation gameplay content lives. In the NeuroGaze Unity project, scenes were used to separate parts of the training and evaluation. We had five scenes: Training Scene – Neutral Mental Command, Training Scene – Pull Mental Command, Evaluation – NeuroGaze, Evaluation – Eye Gaze combined with Hand Gesture, Evaluation – VR Controllers (See *Figure 14*). All these scenes consist of a bare black room, lit with point lights, to ensure no unnecessary artifacts were produced in the EEG data from an overstimulating design. The *neutral* command training scene consists of a wall with white interactable objects that the users can hover on using their eye gaze.
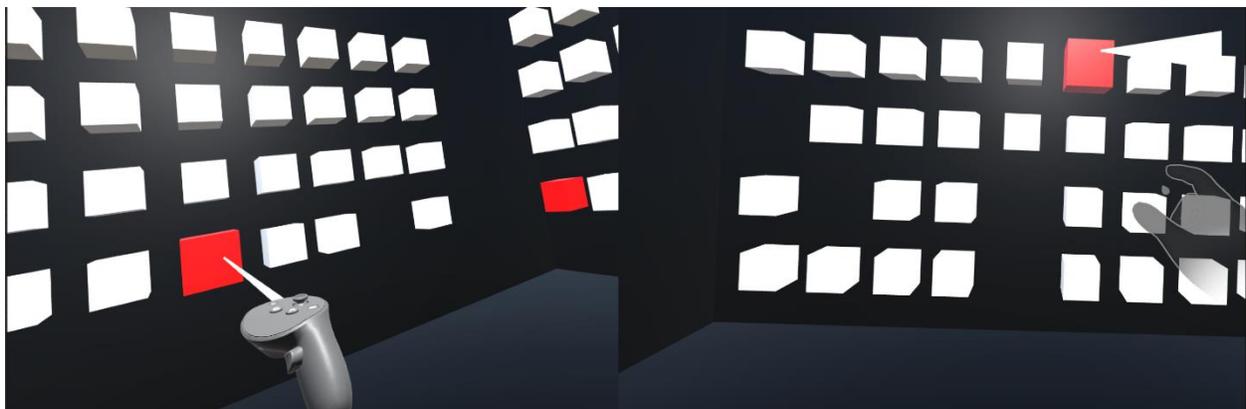
Figure 14: Evaluation Scenes – VR Controllers (left), Eye Gaze combined with Hand Gestures (right).

The *pull* command training scene is identical in appearance and functionality to the *neutral* command training scene, except it has a single red interactable in the middle that acts as a focus point for users to train their brain activity for the *pull* command. The evaluation scene for all three input devices is

the same. The only notable difference in the scene is what represents the player's hand (VR controllers or synthetic hands) and where the ray cast originates from (VR controller or midpoint between the users' eyes). Unity scenes provided us with a VE in which to test input devices against each other in an intuitive and useful way.

3.4.3.2 VR Player Rig

Using the Meta XR All-in-One SDK imported into the Unity project, we could configure a VR rig that handles tracking and represents the user's eyes, hands, and controllers in the VE. The VR rig consists of an *OVRCameraRig* game object that contains the main VR camera and *Tracking Space* game object. This Tracking Space contains the *LeftEyeInteractor* and *RightEyeInteractor* game objects, which represent the position of the user's eyes in the scene (World Space) using the *OVREyeGaze* C# script. The *OVREyeGaze* script works by requesting permission from the Meta Quest Pro HMD for its eye-tracking data; then every frame eye-tracking data is retrieved and processed using *GetEyeGazeState* and a predefined *ConfidenceThreshold*. The transform component of the LeftEyeInteractor and RightEyeInteractor are changed to match the appropriate position, and rotation users should expect their eyes to be in the VE.

```
private void Update()
{
    Vector3 eyesCenter = (leftEyeAnchor.position + rightEyeAnchor.position) * 0.5f; // Average the left and right eye anchors to find the midpoint
    Vector3 forwardDirection = (leftEyeAnchor.forward + rightEyeAnchor.forward) * 0.5f; // Average the forward direction of both eyes
    Ray ray = new Ray(transform.position, transform.forward);
    RaycastHit hit;
    bool isHit = Physics.Raycast(ray, out hit, rayDistance);

    // Check the users eye gaze has hit an object
    if (isHit)
    {
        EyeInteractable eyeInteractable = hit.collider.GetComponent<EyeInteractable>();

        // Check if the users eye gaze hit an item we can interact with (select)
        if (eyeInteractable)
        {
            Debug.Log($"The distance between the users eyes and the cube is: {hit.distance}");

            if (lastEyeInteractable != eyeInteractable)
            {
                if (lastEyeInteractable != null)
                {
                    lastEyeInteractable.Hover(false); // If previous interactable object was in a hover state, set this state to false
                }
                lastEyeInteractable = eyeInteractable;
            }
            eyeInteractable.Hover(true);  // Enlargen or set this cubes "hover state" to true to let users know that can select it
        }
        else if (lastEyeInteractable != null)
        {
            lastEyeInteractable.Hover(false); // If previous interactable object was in a hover state, set this state to false
            lastEyeInteractable = null;
        }
    }
    else
    {
        if (lastEyeInteractable != null)
        {
            lastEyeInteractable.Hover(false); // If previous interactable object was in a hover state, set this state to false
            lastEyeInteractable = null;
        }
    }

    if (isHit)
    {
        if (hit.collider.gameObject.tag == "Wall" && MentalCommands.Instance.GetMentalCommand() == "pull")
            AssessmentManager.Instance.errorCountForMisclicks++;
    }

    // Update the LineRenderer to represent the ray deriving from the users eye
    lineRenderer.SetPosition(0, transform.position);
    lineRenderer.SetPosition(1, transform.position + transform.forward * rayDistance);
}
```

Figure 15: Eye Tracking Ray C# script source code for NeuroGaze input device.

Using these Eye Interactors, we have written a script called *EyeTrackingRay* (See *Figure 15*) with

its logic update every frame (updates at approximately 60 frames per second). This script calculates the

midpoint between two eye interactors and the forward vector. Using this position vector, a RayCast is pro-

jected from the midpoint of the user's eyes, and a check is performed to see what object the user is look-

ing at. If it was an object the user can interact with, the EyeInteractable slowly grows to a fixed scale to

signify to users that it is in a *hover state*. If the user makes eye contact with another *EyeInteractable* or non-*EyeInteractable*, the previous object the user looked at will return to its original scale and be out of its *hover state*. A Line Renderer component is also updated every frame from the user's eyes to 500 meters in the forward vector direction to represent where the user is looking. The *EyeTrackingRay* script triggers the hover state on *EyeInteractable* in the NeuroGaze and Eye + Hand Tracking input methods.

3.4.3.3 Eye Interactable

Interactable objects in the Unity scene represent the Eye Interactable game objects. They are the only objects users can select in the scene. Each Eye Interactable game object in the scene has the *EyeInteractable* script (See *Figure 16*) attached to it, which interfaces with the *EyeTrackingRay* and *MentalCommandsManager* scripts. These objects have two states: a hover state set to true and a hover state set to false. When the object's state is set to true, its scale will slowly grow to a maximum size of 1.28x the original scale (*minimum scale*: $0.18m^3$, *maximum scale*: $0.2304m^3$). Similarly, if the hover state is false, the object will slowly shrink from its current scale to its original scale. When an Eye Interactor's hover state is set to true and the mental command received from is *pull,* further interaction with this object is disabled, and the object is shrunk and destroyed from the scene. The Eye Interactable object knows what mental command is currently being classified from the EEG data through a reference to the *MentalCommandsManager* Instance, which has a public method called *CurrentMentalComand* that scripts in our Unity project can access. During the assessment of each input device, 12 of these Eye Interactable objects will turn red, and users must select them as quickly and accurately as possible.

38

```
private void Update()
{
    string mentalCommand = EyeInteractableManager.Instance.CurrentMentalCommand; // Reference to EyeInteractableManager Instance

    // Check hover state and current mental command
    if (isHovered && mentalCommand == "pull")
    {

      // Set flags to select and destroy cube, stop cube from being interacted with
        shouldShrink = true;
        isBeingPulled = true;

    }
    else
    {
        if (isBeingPulled)
        {
            transform.localScale = originalScale;
            isBeingPulled = false;
        }
    }

    // Check mental command and hover state triggers
    if (shouldShrink)
    {
        ShrinkAndDestroy();
    }
    else if (isHovered)
    {
        ScaleUp(); // Change interactable scale to show user they are looking at it
    }
    else
    {
        ScaleDown(); // Change interactable scale to show user they are not looking at it
    }
}
```

Figure 16: Eye Interactable C# source code for NeuroGaze input device.

3.4.3.4 Mental Commands Manager

```csharp
void Start()
{
    try
    {
        _emotivUnityItf.Init(clientId, clientSecret, appName);
        _emotivUnityItf.Start();
        DataStreamManager.Instance.ScanHeadsets();
        try
        {
            _emotivUnityItf.CreateSessionWithHeadset(headsetId);
            _emotivUnityItf.LoadProfile(profileName);
            _emotivUnityItf.SubscribeData(dataStreamList);
        } catch (Exception e)
        {
            Debug.LogException(e);
        }
    } catch (Exception e)
    {
        Debug.LogException(e);
    }
}


private string lastMentalCommand = "";

@ Unity Message | 0 references | Wanyea Barbel, 24 days ago | 2 authors, 2 changes
void Update()
{
    if (mentalCmdRcvd)
    {
        string currentCommand = _emotivUnityItf.mentalCmdIs();

        Debug.Log($"The current mental command is: {currentCommand}");
        if (currentCommand != lastMentalCommand)
        {

            mentalCommand = currentCommand;
            lastMentalCommand = currentCommand;

            // Trigger the event
            MentalCommandChanged?.Invoke(mentalCommand);
        }
    }
}

2 references | Wanyea Barbel, 43 days ago | 1 author, 1 change
public string GetMentalCommand()
{
    return mentalCommand;
}

@ Unity Message | 0 references | 0 changes | 0 authors, 0 changes
private void OnApplicationQuit()
{
    _emotivUnityItf.UnSubscribeData(dataStreamList);
    _emotivUnityItf.UnLoadProfile(headsetId);
}
```

Figure 17: Mental Commands C# script - Connecting to the headset and subscribing to the data stream

The Mental Commands Manager is the core component to accessing the training profile and EEG data collected in the EmotivBCI program. This game object is configured with the *MentalCommands* script, which utilizes the Cortex API to retrieve EEG data from the Emotiv EPOC X (See *Figure 17*). *MentalCommands* is configured with the App Name, Client ID, and Client Secret generated when you create a new Cortex App. Additionally, the unique name of the training profile created in the EmotivBCI program and the headset ID found on the EPOC X are configured in this script. The NeuroGaze system is set up to start all the Emotiv logic, and we can immediately start reading the EEG stream from the headset. When the program begins, a new instance of *EmotivUnityItf* is initialized with the user credentials. If the user's profile has a valid Emotiv license and the Unity application has been approved to access Emotiv data (via the Emotiv Launcher), the program will begin scanning for Emotiv headsets. Then, if the *headsetId* matches a headset found during the scan, a connection to the headset will be established with the *CreateSessionWithHeadset* method. Then, the *LoadProfile* method will be called with the *profileName*. If a training profile associated with the *clientId* and *clientSecret* is found, an EEG data stream will start, and mental commands can be extracted from the EmotivBCI program. Accessing these mental commands is achieved by calling the *GetMentalCommand* method from the instance of *EmotivUnityIft,* and a string representing this mental command will be returned. The Mental Commands Manager is referenced by each Eye Interactable every frame to check the user's current mental command. By creating one instance of this object, we can send data about the EPOC X to any part of our Unity program without configuring an EEG stream per component.

3.4.3.5 Interactable Manager

The Interactable Manager checks the state of each Eye Interactable in the scene and reports their status to the Assessment Manager. Additionally, this component is responsible for the assessment handoff to the Assessment Manager to collect and store evaluation data. Each Eye Interactable is a child of one of

four parent game objects: leftWall, rightWall, frontWall, or backWall. When the Interactable Manager starts the assessment, four of the child game objects on each wall will randomly turn red (See *Figure 18*).

```csharp
IEnumerator ChangeRandomCubesColor(GameObject parentObject)
{
    if (parentObject == null)
    {
        Debug.LogError("Parent object is not assigned!");
        yield break;
    }

    // Get all child objects from parent
    Transform[] children = parentObject.GetComponentsInChildren<Transform>();

    // Shuffle the array
    System.Random rng = new System.Random();
    int n = children.Length;
    while (n > 1)
    {
        n--;
        int k = rng.Next(n + 1);
        Transform value = children[k];
        children[k] = children[n];
        children[n] = value;
    }

    // Change the color of 3 cubes per wall to red
    int cubesColored = 0;
    foreach (Transform child in children)
    {
        if (child != parentObject.transform && cubesColored < targetsPerWall)
        {
            Renderer renderer = child.GetComponent<Renderer>();
            if (renderer != null)
            {
                renderer.material.color = Color.red;
                child.gameObject.tag = "RedCube"; // Tag the cube as RedCube
                cubesColored++;
            }
        }
    }
}
```

Figure 18: Interactable Manager C# script – Changing interactable colors when the assessment has started.

42

At this moment, the Assessment Manager will begin monitoring the Eye Interactable object to see how many red interactable objects have been deleted, how many white interactable objects have been deleted, and how many complete misses have been made.

3.4.3.6 Assessment Manager

The Assessment Manager oversees all the Eye Interactable objects when the Unity scene is in *assessment mode*. During this mode, twelve Eye Interactable objects will turn red as a cue to inform which objects they should be selecting. The Assessment Manager keeps track of how many times the user attempted to select an object and missed it, how many times they selected the incorrect object, and how long it took for the user to select all the correct objects in the scene successfully. These values are stored until the assessment is over, and then exported as a CSV. A unique participant ID is set by the evaluator, and this ID is included in the exported CSV file. This logic is applied to all three select devices to give evaluators an easy and intuitive way to run the same assessment on any input device across users.

CHAPTER FOUR: USER STUDY

4.1 Experimental Design

A user study was conducted using the NeuroGaze, Eye Gaze + Hand Gesture, and VR controller input devices. When users are presented with an efficiency-accuracy selection task, we want to determine which input device yields the lowest task completion time and error count. User sentiments are also recorded using NASA-TLX surveys and a post-evaluation survey to understand each input device's cognitive load and preference. The study was performed with a sample size of $n = 20$ participants gathered through local university students and staff. Participants were only eligible to participate in the study if they were 18+, had 20/20 vision or corrected to normal vision and hearing, were proficient in English, had no previous history of simulator sickness, and could extend both arms and use their hands. The sample size consisted of 12 participants who identified as male and 8 participants who identified as female. Age ranges consisted of 8 people between 18-23, 9 people between 24-29, and 3 people between 30-35. Participants were asked to rank their experience with VR, AR, and VR video games (see *Appendix B.1*) on a Likert point scale of 1 (no experience) to 5 (very experienced). Participants were also asked to complete a NASA-TLX survey for each input device following the evaluation of each device (See *Appendix B.2*) and a Post Evaluation Survey (See *Appendix B.3*) to holistically gain overall user sentiment about the experiment and input devices.

The NeuroGaze input device's design was crafted according to the type of evaluation we wanted to perform. We wanted to leverage the closed-loop control system from the NeuroGaze input device to increase usability and decrease cognitive load during use. We used a grow and shrink scale cue to denote when users can select an interactable object to achieve this. Additionally, the eye ray casting metaphor was used to explicitly show users where their eye gaze is so they can adjust their motor control appropriately. When the evaluation begins, the interactable objects the user should be selecting turns red, and the

rest should stay white. The interactable objects do not move position in space and only change in scale when a user's eye gaze meets the object. This decision was made to limit the amount of noise users would create in their EEG data due to needing to make quick and jarring motions to keep track of moving objects. If we could classify these artifacts more accurately, future work could involve evaluating NeuroGaze in dense scenes with moving selectable objects.

The metrics we chose to evaluate are completion task time and number of errors. To give every input device a chance at low movement and competition times, the distance between the user and each interactable object is approximately 2 meters, and the size of each interactable object is 0.18m^3 when it is not in a hover state and 0.23m^3 when it is in hover state. We understand that when evaluating both speed and accuracy, it is essential to consider stimulus-response compatibility [10, 17], so using the ray-casting metaphor and size-based cues ensures interaction with the scene is natural, intuitive, and intentional. All three input devices are evaluated identically, and we predicted a learning curve may be present, especially with selection in VR. To mitigate this, each input is assessed over three rounds to give users time to become familiar with the input device. Additionally, improved performance across input devices due to the learning curve is addressed by randomizing the order in which a single participant is evaluated on the input devices.

The NeuroGaze design limitations described previously lead us to believe that users will have the highest task completion time with this input device (*Hypothesis I*). However, we also predict that users will make the least amount of system error with NeuroGaze compared to the other two input devices due to the specific motor imagery required to trigger a selection described in the system design (*Hypothesis II*). Besides comfort, we predict the cognitive load experienced by users will be the lowest with the NeuroGaze system, besides the Temporal or Mental Demand subscales (*Hypothesis III*). Ultimately, we are looking to evaluate the usability of NeuroGaze against the other input devices, given the limitations and scope of this Thesis.

4.1.1 Materials & Apparatus

The evaluation scenes were built in Unity and ran on the Meta Quest Pro connected to a Windows 11 laptop (via Meta Link Cable) equipped with an Intel Core i9-13950HX CPU, an NVIDIA GeForce RTX 4070 GPU, 32 gigabytes (GB) of RAM. The EmotivBCI program was used on this computer to collect EEG training profile data used by the Unity evaluation scene. The user was equipped with the Emotiv EPOC X HMD, the Meta Quest Pro HMD, and two Meta Quest controllers.
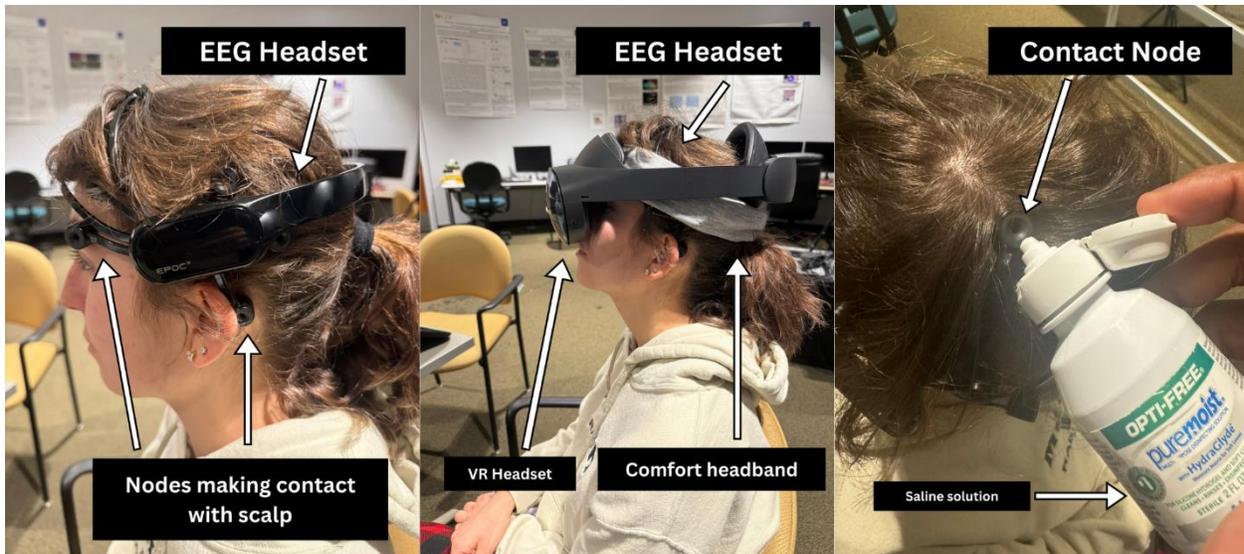
Figure 19: EPOC X on the user (left), NeuroGaze input device: EPOC X, comfort headband, and Meta Quest Pro on the user (middle), applying the saline solution to EPOC X electrodes (right).

Users should first put on the EPOC X by moving hair out of the way and making direct scalp contact with each electrode. To maintain a stable connection between the user's head and the EPOC X electrodes, OPTI-FREE saline solution was applied to all 14 contact electrode access points. Then, a comfort headband keeps the EPOC X headset in place. Finally, the Meta Quest Pro is worn on the EPOC X (See *Figure 19*). Before tightening the Meta Quest Pro, users verify that they can see the VE.

46

4.1.2 Procedure

The experiment is designed as a 1 (task complexity; within-subject) x 3 (available input devices; within-subject) mixed design study in which participants will be evaluated on all three input devices in virtual reality (NeuroGaze: eye gaze ray for hovering, EEG for selection to interact with the object. VR controller: ray from controller to hover, trigger button to select. Eye Gaze + Hand Tracking: eye gaze ray for hovering, pinch gesture to interact with object). Users will have to select all the red interactable objects out of the white ones in the scene, where task performance will be evaluated by determining the time taken to select all the red interactable objects and how many were selected correctly. The studies took place on the UCF main campus in the same room with the Principal Investigator (PI) and consisted of the following procedure:

1. The participant first completed a demographics survey (See *Appendix B.1*) recording age, gender, familiarity with using VR systems, familiarity with playing video games, listing any video games they play, listing any VR video games they play and if the participant is allergic to any of ingredients found in our saline solution.

2. We provided an overview of the electroencephalogram (EEG) headset the participant will be wearing (the Emotiv EPOC X) and the VR headset (the Meta Quest Pro) and demonstrated to participants how to put both headsets on.

3. The participant put the EEG headset on, and we assisted in moving the nodes around to ensure the best contact quality and comfort for the participants. A saline solution was applied to each node on the EEG headset to help establish a higher contact quality between the headset and the participant's scalp. Some participants were asked to put their hair in a higher position to accommodate the EEG node placement.

   a. At this time, the EmotivBCI software used to collect user EEG data should be running and connected to the Emotiv EPOC X EEG headset.

b. The EmotivBCI program displays the contact quality of each node on the participant's head, and this is the time to ensure that every node has a high contact quality.

c. More saline solution may be applied to a node on the EEG headset if the EmotivBCI program shows that the contact quality of that node is low.

4. We put the VR headset on the participant's head and helped the participant adjust it for their comfort.

5. Using the built-in calibration software, We calibrated the eye trackers in the Meta Quest VR headset.

6. Each participant is randomly assigned an input device order to be evaluated. Steps 7 – 14 describe each input device's evaluation setup and actions.

7. We load the VR NeuroGaze training game and explain what the participant needs to do.

a. At this time, the EmotivBCI software should be ready to collect data.

8. The participant performed tasks in VR within the training game we used to gather their EEG data.

a. The training game within the VE consists of a black room with singular red interactable objects in the middle of the room.

b. Participants were told to look around the room at 8-second intervals, repeated 20 times (160 seconds in total).

c. c. Over these 8-second intervals, the PI started collecting EEG data within the EmotivBCI program for the neutral state, the baseline state for the participants' EEG data.

d. Participants were then told to look at the red interactable objects in the black room and, over 8-second intervals, attempt to shrink the interactable by thinking of a repeatable action they can easily recall. This 8-second interval was repeated 20 times (160 seconds in total).

e. During these 8-second intervals, the PI started collecting EEG data within the EmotivBCI program for the *shrink state*. Additionally, the PI forced the red interactable object to shrink over this 8-second interval to give the illusion that the participant is shrinking the interactable object for the EEG training data to be as accurate as possible. This data is used to classify when the participant wants to shrink an object.

9. Once the training sessions were over, we loaded the evaluation game in VR, and participants were told what to do.

   a. The evaluation game puts participants in the same black box within the VE as they were during the training game. Instead of one red interactable object, each wall has a 4x9 array of interactable objects (36 per wall, 144 in total).

   b. When participants look at the interactable objects, they will grow to a set size to let the participants know they are interacting with the interactable objects.

   c. When the participant is ready, the PI will start the evaluation game and start the link between the evaluation game and the EmotivBCI program.

   d. Once the evaluation game has started, three interactable objects on each wall will be randomly chosen to turn red (there will be 12 red interactable objects in total).

   e. Participants must look at these red interactable objects and imagine they are shrinking them. Once these conditions have been met, the red interactable objects in the game will disappear.

   f. Participants are tasked with making all the red interactable objects in the game disappear as quickly and accurately as possible to complete the evaluation game.

   g. This evaluation game is run for three rounds, and performance data is recorded.

10. We removed the EEG and VR heads from the participant's headsets.

11. Participants completed one Qualtrics NASA-TLX survey on a laptop, which gave feedback on the NeuroGaze interaction technique. After this survey is completed, the NeuroGaze evaluation section of the experiment will conclude.

12. We put the VR headset back on the participant and helped adjust it for their comfort. We will load the evaluation game in VR, and the participant will perform a task using their eyes and hands.

    a. Like the NeuroGaze evaluation, participants are placed in the same black box with the same 4x9 array of interactable objects that change size when the users look at them.

    b. When the PI starts the assessment, four interactable objects will randomly turn red, and participants must, as quickly and accurately as possible, look at a red interactable and perform a "pinch gesture" with their hands to make all the red interactable objects disappear.

    c. This evaluation game is run for three rounds, and performance data is recorded.

13. We removed the VR headset from the participant's head, and participants will complete one Qualtrics NASA-TLX survey on a laptop that gives feedback on the Eye Tracking combined with Hand Gesture interaction technique. After completing this survey, the Eye Tracking combined with Hand Tracking evaluation section of the experiment concludes.

14. We put the VR headset back on the participant and helped adjust the headset for their comfort. We put the participant in the VR evaluation game, and the participant will perform a task using VR controllers.

    a. Participants are placed in the same black box with the same 4x9 array of interactable objects.

    b. These interactable objects will change size when the user points their controller at them.

c. When the PI starts the assessment, four interactable objects will randomly turn red, and participants must, as quickly and accurately as possible, point their controller at a red interactable object and click the trigger to make all the red interactable objects disappear.

d. This evaluation game is run for three rounds, and performance data will be recorded.

15. We removed the VR headset from the participant's head and gathered the VR controllers from the participant. Participants will complete one Qualtrics NASA-TLX survey on a laptop that gives feedback on the VR controller interaction technique. After completion of this survey, the VR controller evaluation section of the experiment is concluded.

16. The VR headset was removed, and we gathered the VR controllers.

17. Participants completed one post-evaluation survey, giving feedback about the entire experiment and all three interaction techniques evaluated.

4.2 Results

4.2.1 Demographics Survey

Participants were asked to rank their familiarity with VR, AR, and video games. Specifically, participants were asked: "Please rate your level of experience with each of the following (1 means no experience, five means very experienced)." The results found in Table 1 show that for VR, 3 participants reported no experience, 2 participants reported little experience, 7 participants reported moderate experience, 5 participants reported above moderate experience, and 3 participants reported that they were very experienced. For AR experience: 7 participants reported no experience, 5 participants reported little experience, 5 participants reported moderate experience, 2 participants reported above moderate experience, and 1 participant reported that they were very experienced. For VR Video Games, 3 participants reported no experience, 3 reported little experience, 3 reported moderate experience, 5 reported above moderate experience, and 2 reported very experienced.

Table 1: Experience Level Results (1 – no experience, 5 – very experienced).

	1	2	3	4	5
Virtual Reality	3	2	7	5	3
Augmented Reality	7	5	5	2	1
VR Video Games	3	3	7	5	2

4.2.2 Task Completion Time & Error Count

As stated in the procedure, participants were tasked with selecting all the red interactable objects in the scene as quickly and accurately as possible. Error in our evaluation is defined as missing a selectable object or selecting the wrong object in the scene. The task completion time for each round was defined as how long each user took to select every red interactable object in the scene. The averaged results of how each selection technique performed after this evaluation can be found in Table 2, where Average Task Completion Time is measured in seconds. The data is visualized in Figure 20, which has 95% confidence interval error bars.

Table 2: Evaluation result means for each input device.

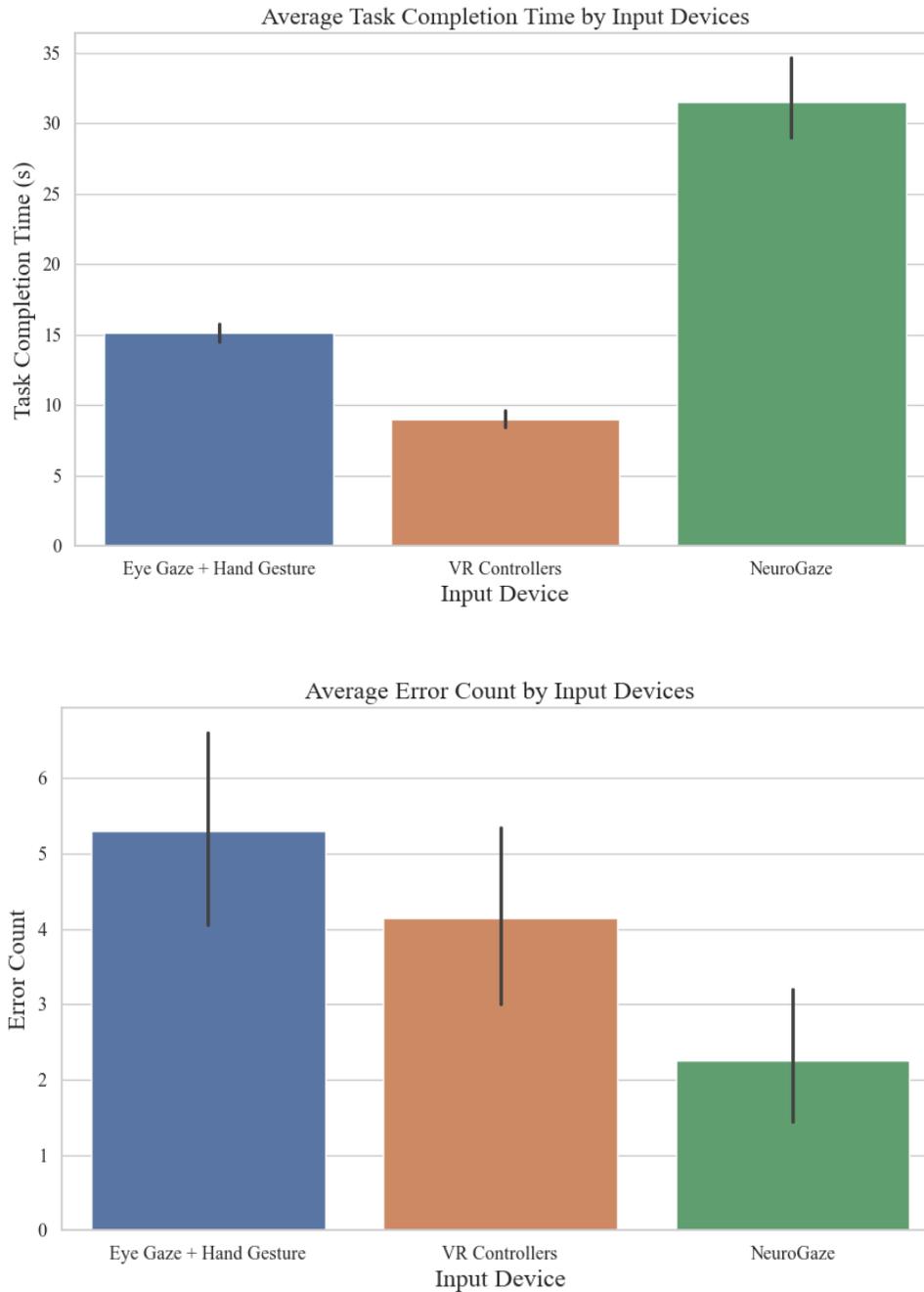| Selection Technique	$\mu_{Task\ Completion\ Time}$ (s)	$\mu_{Average\ Error}$
NeuroGaze	29.23	2.25
Eye Tracking combined with Hand Gesture	15.02	5.3
VR Controllers	9.25	4.15

Figure 20: Average Task Completion Time (top) and Average Error Count (bottom) for each input device, with 95% confidence intervals.

Before applying significance tests, we performed Mauchly's Test for Sphericity on the Task Completion Time ($\chi^2(20) = 29.222, p < 0.001$) and Error Count ($\chi^2(20) = 4.933, p = 0.85$ evaluation

data. The results for Task Completion Time violated sphericity, so we applied the Greenhouse-Geisser correction. A repeated measures ANOVA test with Greenhouse-Geisser was performed to measure the significance between selection devices for task completion time and average error count.

Table 3: RM-ANOVA statistical effects, Task Completion and Error Count (* = p < .05; ** = p < .01; *** = p < .001).

Measure	F	df_{effect}	df_{error}	p	η_p^2	Sig
Task Completion Time	97.624	2	19.77	< 0.001	0.844	***
Error Count	5.385	2	36	p = 0.009	0.23	**

The results show a significant difference in both the Task Completion Time ($F_{(2,19.772)}$ = 97.642), $p < 0.001, \eta_p^2 = 0.844$) and Error Count ($F_{(2,36)} = 5.385$), $p = 0.009, \eta_p^2 = 0.23$) between the input devices. A summary is in Table 3.

Table 4: Post-Hoc Analysis Results for Task Completion Time (M_1 refers to first input device condition in the pair, M_2 refers to second input device condition in the pair) (NG – NeuroGaze, EG+HG – Hand Gesture and Hand Gesture, VRC – VR Controllers) (* = p < .05; ** = p < .01; *** = p < .001).

Input Device Pairs	M_1 Mean	M_2 Mean	p	Sig
NG vs. EG+HG	33.01	15.24	< 0.001	***
NG vs. VRC	33.01	9.06	< 0.001	***
EG+HG vs. VRC	15.24	9.06	< 0.001	***

Table 5: Post-Hoc Analysis Results for Error Count (M_1 refers to first input device condition in the pair, M_2 refers to second input device condition in the pair) (NG – NeuroGaze, EG+HG – Hand Gesture and Hand Gesture, VRC – VR Controllers) (* = p < .05; ** = p < .01; *** = p < .001).

Input Device Pairs	M_1 Mean	M_2 Mean	p	Sig
NG vs. EG+HG	1.08	2.25	p = 0.041	*
NG vs. VRC	1.08	1.56	p = 0.441	no
EG+HG vs. VRC	2.25	1.56	p = 0.105	no

Post-hoc analysis was performed using pairwise t-tests for Task Completion Time (See *Table 4*) and Error Count (See *Table 5*). For Task Completion Time, the pairwise t-test yielded high significance for NG vs. EG+HG ($p < 0.001$), NG vs. VRC ($p < 0.001$), and EG+HG vs. VRC ($p < 0.001$), the

other pairs yielded no significance. The pairwise t-test for Error Count only showed significance for NG vs. EG+HG.

4.2.3 NASA-TLX Surveys
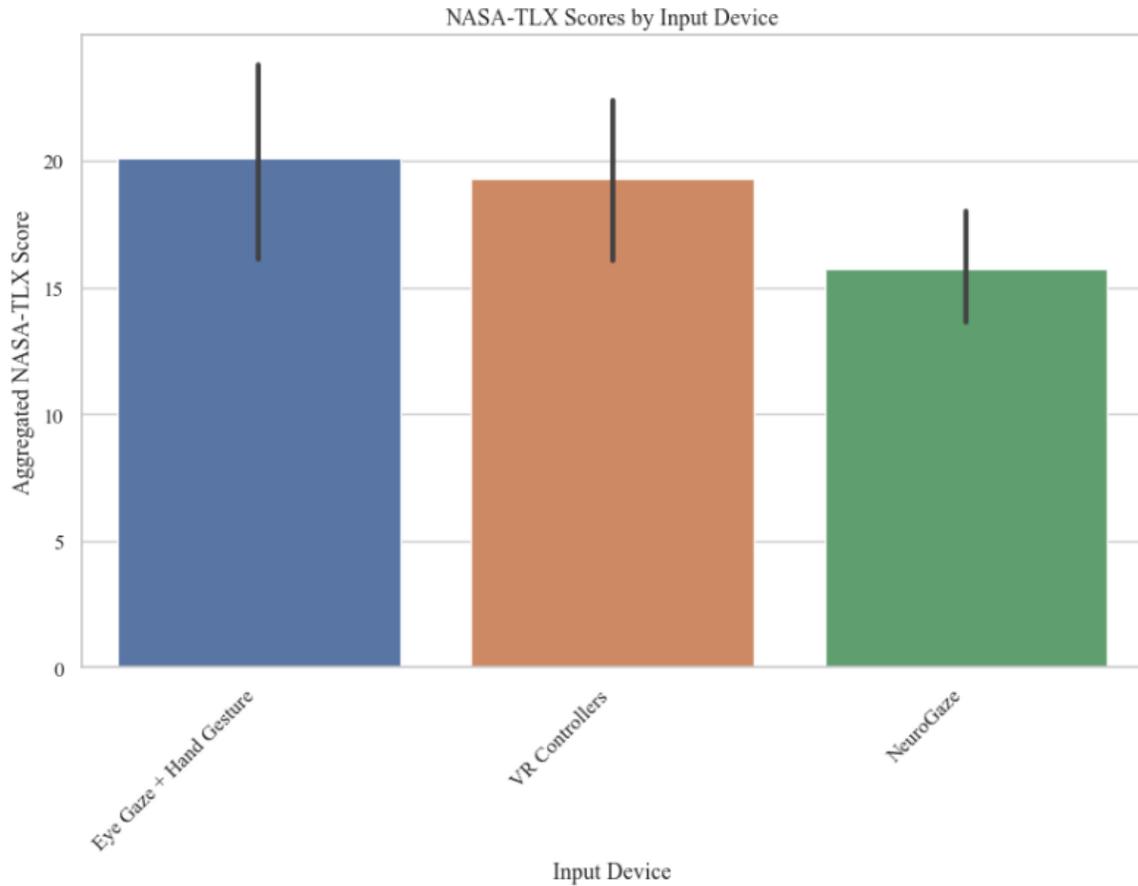
Figure 21: Aggregated NASA-TLX score for each input device, with 95% confidence intervals.

After each selection device, participants were asked to fill out a 7-point scale NASA-TLX to evaluate cognitive load. The NASA-TLX survey consisted of the following questions:

1. How mentally demanding was the task? (Mental Demand)

2. How physically demanding was the task? (Physical Demand)

3. How hurried or rushed was the pace of the task? (Temporal Demand)

4. How successful were you in accomplishing what you were asked to do? (Performance)

5. How hard did you have to work to accomplish your level of performance? (Effort)

6. How insecure, discouraged, irritated, stressed, and annoyed were you? (Frustration)

The total NASA-TLX score was calculated per input device as follows:

$$NASA - TLX\ Score\ =\ Mental\ Demand\ +\ Physical\ Demand\ +\ Temporal\ Demand\ +$$
$$(7\ -\ Performance)\ +\ Effort +\ Frustration$$

The average aggregated NASA-TLX score for each input device showing an Eye Gaze + Hand Gesture yielded a score of 20.1, VR controllers a score of 19.30, and NeuroGaze a score of 15.75. To determine the significance of repeated measures, a Friedman Test was conducted across the aggregated NASA-TLX scores across input devices. The test yielded $p =\ 0.29;\ p > 0.05$, yielding no statistical significance.

Since the calculated NASA-TLX score did not show significance, we evaluated significance across each NASA-TLX subscale score across input devices by running a Wilcoxon Signed-Rank Test (See *Table 6*). The question and input devices that showed significance are as follows: Mental question for VR Controllers vs. Eye Gaze+ Hand Gesture with $Z\ =\ 14$ and $p = 0.026$, Physical question for VR Controllers vs. NeuroGaze with $Z = 19$ and $p =\ 0.002$, Physical question for Eye Gaze + Hand Gesture vs. NeuroGaze with $Z = 12$ and $p =\ 0.006$, Physical question for VR Controllers vs. Eye Gaze + Hand Gesture with $Z\ =\ 11$ and $p = 0.03$, Temporal question for VR Controllers vs. NeuroGaze with $Z\ =\ 6.5$ and $p =\ 0.001$, Temporal question for Eye Gaze + Hand Gesture vs. NeuroGaze with $Z =\ 7$ and $p =\ 0.002$, Performance question for VR Controllers vs. Eye Gaze + Hand Gesture with $Z\ =\ 16$ and $p =\ 0.04$. The significance level has been adjusted for Bonferroni correction.

Table 6: Wilcoxon Signed Rank Test for Individual Subscale Scores for each input device (** = p < 0.003125; ns = p > 0.003125).

NASA-TLX Question	Input Device Comparison	Z	p, α^b	Significant?
Mental	VR Controllers vs. Eye Gaze + Hand Gesture	14	p = 0.026; α = 0.003125	ns
Mental	VR Controllers vs. NeuroGaze	89.5	p = 0.82; α = 0.003125	ns
Mental	Eye Gaze + Hand Gesture vs. NeuroGaze	57	p = 0.12; α = 0.003125	ns
Physical	VR Controllers vs. Eye Gaze + Hand Gesture	17	p = 0.89; α = 0.003125	ns
Physical	VR Controllers vs. NeuroGaze	19	p = 0.002; α = 0.003125	**
Physical	Eye Gaze + Hand Gesture vs. NeuroGaze	12	p = 0.006; α = 0.003125	ns
Temporal	VR Controllers vs. Eye Gaze + Hand Gesture	11	p = 0.03; α = 0.003125	ns
Temporal	VR Controllers vs. NeuroGaze	6.5	p = 0.001; α = 0.003125	**
Temporal	Eye Gaze + Hand Gesture vs. NeuroGaze	7	p = 0.002; α = 0.003125	**
Performance	VR Controllers vs. Eye Gaze + Hand Gesture	16	p = 0.04; α = 0.003125	ns
Performance	VR Controllers vs. NeuroGaze	43	p = 0.1; α = 0.003125	ns
Performance	Eye Gaze + Hand Gesture vs. NeuroGaze	61.5	p = 0.47; α = 0.003125	ns
Effort	VR Controllers vs. Eye Gaze + Hand Gesture	23	p = 0.35; α = 0.003125	ns
Effort	VR Controllers vs. NeuroGaze	35.5	p = 0.16; α = 0.003125	ns
Effort	Eye Gaze + Hand Gesture vs. NeuroGaze	55	p = 0.49; α = 0.003125	ns
Frustration	VR Controllers vs. Eye Gaze + Hand Gesture	22	p = 0.57; α = 0.003125	ns
Frustration	VR Controllers vs. NeuroGaze	38.5	p = 0.97; α = 0.003125	ns
Frustration	Eye Gaze + Hand Gesture vs. NeuroGaze	54	p = 0.72; α = 0.003125	ns

---

[b] Adjustment for multiple comparisons: Bonferroni
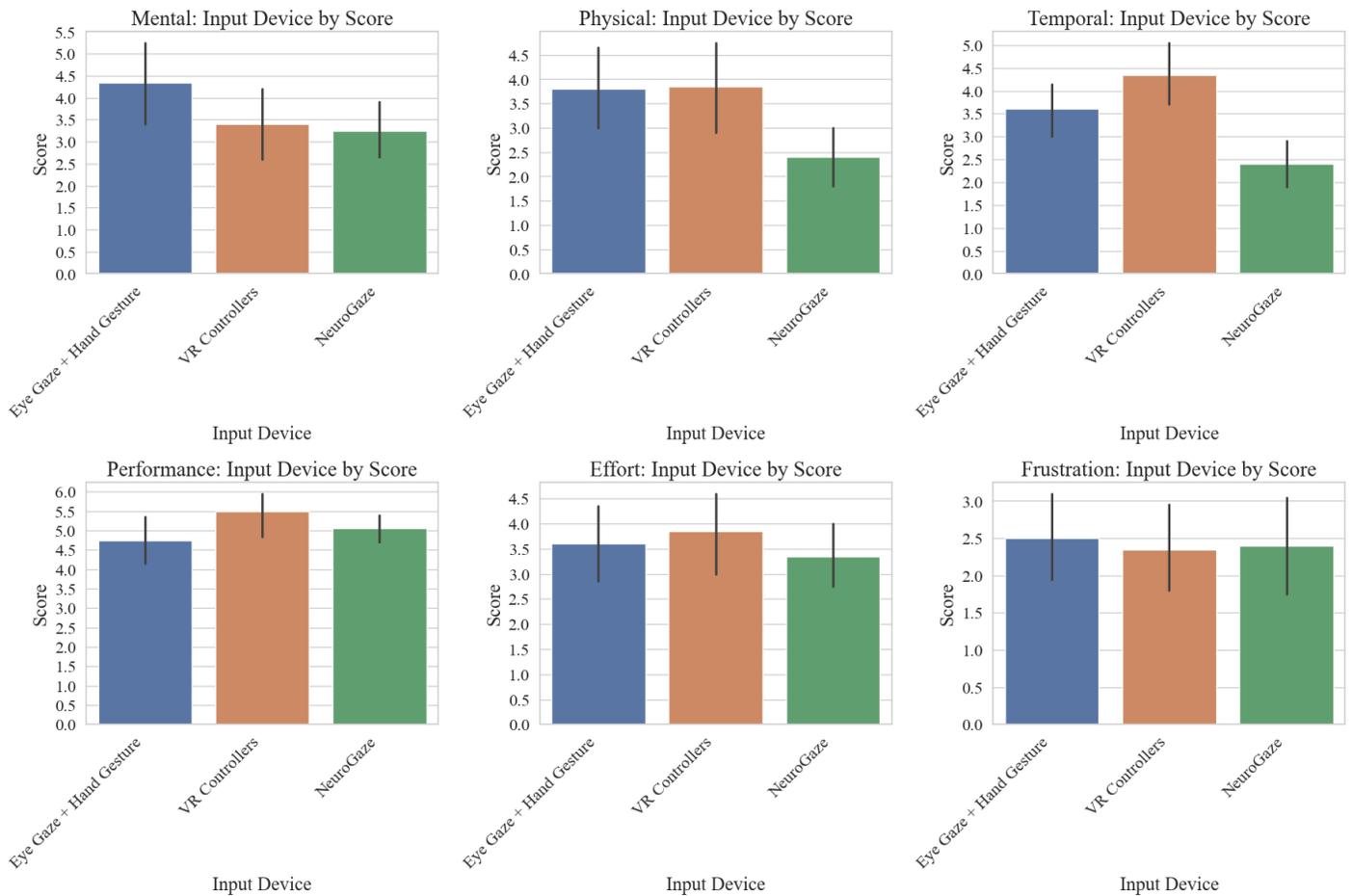
Figure 22: NASA TLX Individual Subscale Mean Score for each input device, with 95% confidence intervals.

4.2.4 Post-Evaluation Survey

Participants were given a Post Evaluation Survey (See Appendix B.3) to gather qualitative data about how about motor imagery users thought of to select objects and how they would rank each input device overall against one another. Common complaints about the NeuroGaze included not being comfortable to wear for longer durations, which we predicted due to poor ergonomic compatibility between the EEG headset and the Meta Quest Pro HMD. Users also complained about some lethargic response times from NeuroGaze at times when they wanted to select. Users were frustrated with VR controllers in

moments when they would mistake the grip button for the trigger, resulting in no selection when they wanted to. Even though we explicitly showed users the trigger was for selection, they repeatedly clicked the grip button and would complain. Users also included sentiments about their arms being tired by the time of the last round. The Eye Gaze + Hand Gesture input device seemed to frustrate users the most because they would not complete the entire pinching gesture required to select an object. At times, they wouldn't open their fingers on one hand before trying to select with another, increasing their task completion time and resulting in annoyance.

Participants were asked to rank the input devices evaluated overall, with 1 being most preferred, 2 being intermediate preference, and 3 being least preferred (See *Table 7*).

Table 7: Raw tally of user preference for input devices evaluated (1—Most Preferred, 2—Intermediate Preference, 3—Least Preferred).

Input Device	1	2	3
NeuroGaze	10	6	4
VR Controllers	5	8	7
Eye Gaze + Hand Gesture	5	6	9

A Chi-Squared Test of Independence was performed for the most preferred input device to gauge significance ($\chi_2^2(N = 20) = 4.8, p = 0.31$). The p-value is $p < 0.05$; thus, we can reject the null hypothesis and state there is not enough evidence to support a significant association between input devices and their preference rankings.

4.3 Discussion

The evaluation of NeuroGaze compared to traditional input devices, such as VR Controllers and the combination of Eye Gaze with Hand Gesture, offers insights into user performance and preferences within virtual environments. NeuroGaze showed a higher average task completion time, indicating a significant delay from when users intended to select an object. This points to a notable usability issue despite

recording the lowest error count. This finding is underscored by a very low p-value ($p < 0.001$) in the statistical comparison of input devices using RM-ANOVA and Post-Hoc analysis, validating our hypothesis that NeuroGaze would result in the longest Task Completion Time (*Hypothesis I*). In contrast, VR Controllers achieved a shorter average task completion time with a relatively high error rate, demonstrating this device's capability to effectively balance accuracy and speed Eye Gaze + Hand Gesture had the second longest task completion time and the highest error rate. Although NeuroGaze had the lengthiest average task completion time, it boasted the lowest average error rate, indicating users could select objects with greater precision. The statistical significance of the differences in error rates among the input devices was confirmed by RM-ANOVA ($p = 0.009$) and pairwise t-tests ($NG\ vs.\ EG + HG;\ p = 0.041$), particularly between NeuroGaze and Eye Gaze + Hand Gesture. NeuroGaze's strength lies in enabling accurate user selections, albeit with increased task completion times, supporting our second hypothesis that users would commit the fewest errors with the NeuroGaze system (*Hypothesis II*). However, we found no significant correlation between users' motor imagery and their performance, so we cannot link the lower error rate to specific imagery types.

Upon analyzing the statistics, we initially thought VR Controllers would emerge as the clear favorite, as they enabled users to select objects approximately 3.16 times faster than NeuroGaze and nearly twice as quickly as Eye Gaze + Hand Gesture, despite some errors. However, user-reported data painted a different picture. Although the average NASA-TLX scores for each input device showed no significant differences, individual subscale scores revealed meaningful distinctions among the devices. NeuroGaze scored lower on Physical Demand than VR Controllers and Eye Gaze + Hand Gestures, with significant differences ($p = 0.02\ and\ p = 0.006$ respectively). Still, no significant difference was found between VR Controllers and Eye Gaze + Hand Gestures ($p = 0.89$). Despite NeuroGaze's requirement for users to recall specific thoughts for object selection, no significant differences were found in Mental Demand when compared to VR Controllers ($p = 0.82$) or Eye Gaze + Hand Gesture ($p = 0.12$), with the means

for VR Controllers and NeuroGaze being closely matched ($\mu_{VR\ Controllers} = 3.3$; $\mu_{NeuroGaze} = 3.2$). Significant differences were observed in Temporal Demand scores across all device pairs, with NeuroGaze having the lowest mean ($\mu_{NeuroGaze} = 3.2$). This suggests users felt varied temporal pressures across devices, feeling the least rushed with NeuroGaze. Given its highest mean task completion time, its unsurprising users felt more at ease with the pace. Performance results were unexpected; the only significant difference was between VR Controllers and Eye Gaze + Hand Gesture ($p = 0.04$). With Eye Gaze + Hand Gesture having the highest error rate, users feeling least efficient with this device makes sense. Yet, the absence of significant differences between NeuroGaze and other devices, despite NeuroGaze's longer selection times, suggests users valued accuracy over speed, as evidenced by the lack of significant frustration differences. Our hypothesis that NeuroGaze would have the lowest cognitive load (*Hypothesis III*) was not statistically supported. However, significant differences in Temporal Demand showed NeuroGaze had the lowest cognitive load in this area, contrary to our expectations. Mental Demand also did not differ significantly between NeuroGaze and the other devices, indicating NeuroGaze was less mentally demanding than anticipated.

The post-evaluation survey allowed users to express their preferences explicitly. Results indicated a preference for NeuroGaze over VR Controllers and Eye Gaze + Hand Gesture. Although the Chi-Squared Test didn't confirm statistical significance, the data shows users did not strongly dislike NeuroGaze compared to more traditional selection methods, as seen by NeuroGaze being selected as the most preferred input device over the other two. Many requested improvements to the comfort of the EPOC X and Meta Quest Pro. When discussing object selection, every user mentioned kinematic actions, like "punching" or "crushing" the object, with some personalizing the action, such as "strumming my guitar."

Anecdotal evidence provided further insights. Users often expressed amazement at being able to control objects with their minds through NeuroGaze, describing the experience as unique and expressing a desire to try it again. A learning curve was quickly overcome, with one user comparing it to adapting to

the Apple Vision Pro HMD. In contrast, users frequently struggled with the pinch gesture required by the Eye Gaze + Hand Tracking input device and with the VR controller grip buttons, leading to frustration. This highlights both our input devices' potential and limitations, offering valuable lessons for future development.

CHAPTER FIVE: FUTURE WORK

This Thesis provided our research team with a proof of concept to build upon in future interactions. The NeuroGaze system has shown that it is possible to accurately select objects in a VE with eye gaze and non-invasive BCIs. This was a learning experience, and as such, we will be taking the lessons learned to iterate on the NeuroGaze design. We plan to try new EEG headsets as the EPOC X is a commercial EEG headset, and a medical-grade headset would give us access to more channels and higher-quality EEG data for classification. Additionally, users expressed general discomfort when wearing the NeuroGaze system for a prolonged period. To mitigate this in the future, we would try an EEG cap, which might reduce the pressure put on the front of the user's head since EEG caps usually have more evenly distributed channels. It might also be interesting to develop our own EEG headset using the electrode nodes, board, and firmware offered by OpenBCI[1]. This would allow us to strategically choose which channels we would want to access and design the headset in a way that's more compatible with the Meta Quest Pro HMD. If funding allows, we would like to research using functional near-infrared spectroscopy (fNIRS) instead of EEG. Local university advisors have suggested that EEG is better for the classification of emotional state, but fNIRS performs better in the classification of motor imagery. We believe this would increase our ability to more accurately classify selection artifacts from the EEG stream and hopefully decrease task completion time. We are also interested in potentially spending the time to train our own machine learning denoising and classification machine learning algorithm. Personalizing this model to our needs would allow us to classify specific artifacts we want to filter out, such as repeated kinematic movement. This would remove physical restrictions we had to put in place for the NeuroGaze system and would allow users to move around in the Unity scene without the fear of unnecessary noise in the EEG

---

[1] Company that develops open-source tools for neuroscience and biosensing (https://openbci.com/).

data. Evaluation of a user's ability to track moving objects with eye gaze and select using this version of NeuroGaze could also bring about some interesting research into the ability of this system to utilize the natural eye movement of *saccade* to *smooth pursuit*.

We would design the experiment to adhere to 3D UI evaluation criteria and best practices more strictly for future evaluations. For example, our performance metrics were completion time and accuracy, but we told users to attempt to perform their best in both categories each round. Instead, we will tell users to be as accurate as possible for one round, as quick as possible for one round, and then as quick and accurate for the final round. This would give us a more holistic overview of the tradeoffs between speed and accuracy for the NeuroGaze system and the other two input devices.

Overall, we are very interested in continuing to research the applications of BCI interfacing with VEs and 3DUIs. Although our task completion time was not optimal, the positive user sentiments seen in the NASA-TLX surveys and Post Evaluation survey suggest users still believe this input device has usability.

CHAPTER SIX: CONCLUSION

This Thesis presented NeuroGaze, a novel VR input device that combines eye tracking with EEG data to facilitate object selection in VEs. The system's design, development, and evaluation underscore the potential of integrating noninvasive biometric inputs to create more intuitive and accessible user interfaces. Despite facing challenges such as user adaptation and cognitive load, NeuroGaze demonstrates a promising direction for future VR interaction technologies. By pushing the boundaries of traditional VR input devices, NeuroGaze offers a possible enrichment to the interaction paradigm within VEs and could serve as an alternative for users with physical disabilities who have little or no autonomy to interact with VEs. As VR continues to evolve, we believe continued research into systems like NeuroGaze will play a pivotal role in shaping immersive experiences that are both inclusive and engaging. NeuroGaze represents a significant step forward in the quest for seamless human-computer interaction within virtual worlds. We hope its development and study contribute valuable insights to the field of HCI and BCI research, laying the groundwork for future innovations that will further dissolve the barriers between the human mind and interactions in virtual environments.

APPENDIX A: IRB APPROVAL FORM

UNIVERSITY OF CENTRAL FLORIDA

APPROVAL

March 14, 2024

Dear Wanyea Barbel:

On 3/14/2024, the IRB reviewed the following submission:

Type of Review:	Initial Study, Category
Title:	NeuroGaze in Virtual Reality: Assessing EEG and Eye Tracking Interfaces Against Conventional VR Selection Methods
Investigator:	Wanyea Barbel
IRB ID:	STUDY00006401
Funding:	None
IND, IDE, or HDE:	None
Documents Reviewed:	• HRP-251 - FORM - Faculty Advisor - NeuroGaze.pdf, Category: Faculty Research Approval; • 502.pdf, Category: Consent Form; • 503.docx, Category: IRB Protocol; • A&FCommunicationsRequestForm.pdf, Category: Recruitment Materials; • GoogleFormExplained.pdf, Category: Survey / Questionnaire; • Qualtrics_NeuroGaze_Post_Evaluation_Survey.docx, Category: Survey / Questionnaire; • Qualtrics_NeuroGaze_VR_Demographics_Survey.docx, Category: Survey / Questionnaire; • Qualtrics_NeuroGaze_VR_NASA_TLX.docx, Category: Survey / Questionnaire; • RecruitmentEmail.pdf, Category: Recruitment Materials; • ReferenceImages.docx, Category: Other;

The IRB approved the protocol on 3/14/2024.

In conducting this protocol, you are required to follow the requirements listed in the Investigator Manual (HRP-103), which can be found by navigating to the IRB Library within the IRB system. Guidance on submitting Modifications and a Continuing Review or Administrative Check-in is detailed in the manual. If continuing review is required and approval is not granted before the expiration date, approval of this protocol expires on that date.

Page 1 of 2

67

If this protocol includes a consent process, use of the time-stamped version of the consent form is required. You can find the time-stamped version of the consent form in the "**Documents**" tab under the "**Final**" column.

To document consent, use the consent documents that were approved and stamped by the IRB. Go to the Documents tab to download them.

When you have completed your research, please submit a Study Closure request so that IRB records will be accurate.

If you have any questions, please contact the UCF IRB at 407-823-2901 or irb@ucf.edu. Please include your project title and IRB number in all correspondence with this office.

Sincerely,
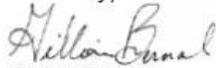
Gillian Bernal
Designated Reviewer

APPENDIX B: SAMPLE SURVEYS

B.1 Qualtrics Demographic Survey

NeuroGaze VR Demographics Survey

Welcome! This is the Demographics Survey for the Study on Assessing EEG and Eye Tracking Interfaces Against Conventional VR Selection Methods. Click the arrow to begin.

Q1 Participant Number

_____

Q2 Age

_____

Q3 Gender Identity

_____

Q4 Please rate your level of expertise with each of the following (1 means no experience, 5 means very experienced)

	1 (1)	2 (2)	3 (3)	4 (4)	5 (5)
Virtual Reality (1)	○	○	○	○	○
Augmented Reality (2)	○	○	○	○	○
Video Games (3)	○	○	○	○	○

Q5 List any virtual reality games you have played:

_____

_____

_____

_____

_____

Q6 Are you allergic to any of the following ingredients found in the saline solution () we will be using?

Ingredients: Sodium Citrate, Sodium Chloride, Boric Acid, Sorbitol, Aminomethyl Propanol, Disodium EDTA, Two Wetting Agents (Tetronic 1304 and HydraGlyde Moisture Matrix [EOBO-41- polyoxyethylenepolyoxybutylene]) with Polyquad (Polyquaternium-1) 0.001% and Aldox (Myristamidopropyl Dimethylamine) 0.0006% Preservatives. HydraGlyde Moisture Matrix is a Proprietary Multi-functional Block Copolymer that is Primarily Designed for Wetting and Lubricating Silicone Hydrogel Lenses.

◯ Yes (1)

◯ No (2)

Q7 Do you have any previous history of simulator sickness?

◯ Yes (1)

◯ No (2)

End of Block: Default Question Block

B.2 Qualtrics NASA-TLX Survey

NeuroGaze VR NASA TLX

Welcome! This is the NASA TLX Survey for the Study on Assessing EEG and Eye Tracking Interfaces Against Conventional VR Selection Methods. Click the arrow to begin.

Participant Number

_____

Technique being Evaluated

○ NeuroGaze: EEG + Eye Tracking

○ Eye Tracking + Hand Tracking

○ VR Controllers

Please answer each prompt based on your experience:

	1 (Very Low)	2	3	4	5	6	7 (Very High)
How mentally demanding was the task?	○	○	○	○	○	○	○
How physically demanding was the task?	○	○	○	○	○	○	○
How hurried or rushed was the pace of the task?	○	○	○	○	○	○	○
How successful were you in accomplishing what you were asked to do?	○	○	○	○	○	○	○
How hard did you have to work to accomplish your level of performance?	○	○	○	○	○	○	○
How insecure, discouraged, irritated, stressed, and annoyed were you?	○	○	○	○	○	○	○

End of Block: Default Question Block

B.3 Qualtrics Post-Evaluation Survey

NeuroGaze Post Evaluation Survey

Welcome! This is the Post Evaluation Survey for the Study on Assessing EEG and Eye Tracking Interfaces Against Conventional VR Selection Methods. Click the arrow to begin.

Participant Number

_____

Did you feel like you were making the cube shrink during the training scene? (Include any additional comments under your answer choice)<

○ Yes _____

○ No _____

How intuitive was each technique to to use? (Include any additional comments the under technique)

	1 (least intuitive)	2	3	4	5 (most intuitive)
NeuroGaze (EEG combined with Eye Tracking)	○	○	○	○	○
Eye Tracking combined with Hand Tracking	○	○	○	○	○
VR Controllers	○	○	○	○	○

Page 1 of 2

77

What would you change about any of the interaction techniques or the experiment? If nothing, answer "Nothing"

_____

Which interaction technique did you prefer overall?\
(Include any additional comments under your answer choice)

◯ NeuroGaze (EEG combined with Eye Tracking)
_____

◯ Eye Tracking combined with Hand Tracking
_____

◯ VR Controllers _____

LIST OF REFERENCES

[1] Agarwal, M., & Sivakumar, R. (2019). Blink: A Fully Automated Unsupervised Algorithm for Eye-Blink Detection in EEG Signals. In *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)* (pp. 1113-1121). Monticello, IL, USA: IEEE. https://doi.org/10.1109/ALLERTON.2019.8919795

[2] Argelaguet, F., & Andujar, C. (2013). A Survey of 3D Object Selection Techniques for Virtual Environments. *Computers & Graphics, 37*(3), 121-136. https://doi.org/10.1016/j.cag.2012.12.003

[3] Baceviciute, S., Lucas, G., Terkildsen, T., & Makransky, G. (2022). Investigating the redundancy principle in immersive virtual reality environments: An eye-tracking and EEG study. *Journal of Computer Assisted Learning, 38*(1), 120-136. https://doi.org/10.1111/jcal.12595

[4] Blattgerste, J., Renner, P., & Pfeiffer, T. (2018). Advantages of eye-gaze over head-gaze-based selection in virtual and augmented reality under varying field of views. In *Proceedings of the Workshop on Communication by Gaze Interaction*. https://doi.org/10.1145/3206343.3206349

[5] Clark, L. D., Bhagat, A. B., & Riggs, S. L. (2020). Extending Fitts' law in three-dimensional virtual environments with current low-cost virtual reality technology. *International Journal of Human-Computer Studies*, 139, 102413. https://doi.org/10.1016/j.ijhcs.2020.102413

[6] Decety, J. (1996). The neurophysiological basis of motor imagery. *Behavioural Brain Research, 77*(1-2), 45-52. https://doi.org/10.1016/0166-4328(95)00225-1

[7] Fakhruzzaman, M. N., Riksakomara, E., & Suryotrisongko, H. (2015). EEG Wave Identification in Human Brain with Emotiv EPOC for Motor Imagery. *Procedia Computer Science, 72*, 269–276. https://doi.org/10.1016/j.procs.2015.12.140

[8] Fahmi, F., Tanjung, K., Nainggolan, F., Siregar, B., Mubarakah, N., & Zarlis, M. (2020). Comparison study of user experience between virtual reality controllers leap motion controllers and senso

glove for anatomy learning systems in a virtual reality environment. *IOP Conference Series: Materials Science and Engineering, 851*(1), 012024. https://doi.org/10.1088/1757-899X/851/1/012024

[9] Fitts, P. M., & Seeger, C. M. (1953). S-R compatibility: Spatial characteristics of stimulus and response codes. *Journal of Experimental Psychology, 46*(3), 199-210. https://doi.org/10.1037/h0062827

[10] Lopez Luro, F., & Sundstedt, V. (2019). A Comparative Study of Eye Tracking and Hand Controller for Aiming Tasks in Virtual Reality. In Communication by Gaze Interaction (*COGAIN @ ETRA'19*), June 25–28, 2019, Denver, CO, USA. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3317956.3318153

[11] Hosp, B., Eivazi, S., Maurer, M., Fuhl, W., Geisler, D., & Kasneci, E. (2020). RemoteEye: An open-source high-speed remote eye tracker. *Behavior Research Methods, 52*, 1387–1401. https://doi.org/10.3758/s13428-019-01305-2

[12] Imaoka, Y., Flury, A., & de Bruin, E. D. (2020). Assessing Saccadic Eye Movements With Head-Mounted Display Virtual Reality Technology. *Frontiers in Psychiatry, 11*, 572938. https://doi.org/10.3389/fpsyt.2020.572938

[13] Jurcak, V., Tsuzuki, D., & Dan, I. (2007). 10/20, 10/10, and 10/5 systems revisited: Their validity as relative head-surface-based positioning systems. *NeuroImage, 34*(4), 1600–1611. https://doi.org/10.1016/j.neuroimage.2006.09.024

[14] Khan, G. M. (2015). A new electrode placement method for obtaining 12-lead ECGs. *Open Heart, 2*(1), e000226. https://doi.org/10.1136/openhrt-2014-000226

[15] Khazi, M., Kumar, A., & J, V. M. (2012). Analysis of EEG using 10:20 electrode system. *International Journal of Innovative Research in Science, Engineering and Technology, 1*(2).

[16] Khundam, C., Vorachart, V., Preeyawongsakul, P., Hosap, W., & Noël, F. (2021). A Comparative Study of Interaction Time and Usability of Using Controllers and Hand Tracking in Virtual Reality Training. *Informatics, 8*(3), 60. https://doi.org/10.3390/informatics8030060

[17] LaViola Jr., J. J., Kruijff, E., Bowman, D. A., McMahan, R. P., & Poupyrev, I. (2017). *3D User Interfaces: Theory and Practice*. Addison-Wesley Professional.

[18] Larsen, O. F. P., Tresselt, W. G., Lorenz, E. A., Holt, T., Sandstrak, G., Hansen, T. I., Su, X., & Holt, A. (2024). A method for synchronized use of EEG and eye tracking in fully immersive VR. *Frontiers in Human Neuroscience, 18*, 1347974. https://doi.org/10.3389/fnhum.2024.1347974

[19] Lin, W., Du, L., Harris-Adamson, C., Barr, A., & Rempel, D. (2017). Design of hand gestures for manipulating objects in virtual reality. In M. Kurosu (Ed.), *Human-Computer Interaction. Theories, Methods, and Human Issues: 19th International Conference, HCI International 2017, Vancouver, BC, Canada, July 9-14, 2017, Proceedings, Part I* (pp. 584–592). Springer International Publishing. https://doi.org/10.1007/978-3-319-58071-5_44

[20] Lopez Luro, F., & Sundstedt, V. (2019). A comparative study of eye tracking and hand controller for aiming tasks in virtual reality. *In ETRA '19: Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications* (Article No. 68, pp. 1–9). ACM. https://doi.org/10.1145/3317956.3318153

[21] Luong, T., Cheng, Y. F., Mobus, M., Fender, A., & Holz, C. (2023). Controllers or Bare Hands? A Controlled Evaluation of Input Techniques on Interaction Performance and Exertion in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics, 29*(11), 4633-4642.

[22] Oviatt, S. (1999). Mutual disambiguation of recognition errors in a multimodal architecture. In *CHI '99 - Proceedings of the SIGCHI conference on Human Factors in Computing Systems* (pp. 576-583). Association for Computing Machinery (ACM). https://doi.org/10.1145/302979.303163

[23] Perret, J., & Vander Poorten, E. B. (2018). Touching Virtual Reality: A Review of Haptic Gloves. Conference Paper, June 2018. Haption GmbH, Aachen, Germany; Department of Mechanical Engineering, KU Leuven, Belgium. Retrieved from https://www.researchgate.net/publication/324562855

[24] Piumsomboon, T., Lee, G. A., Lindeman, R. W., & Billinghurst, M. (2017). Exploring natural eye-gaze-based interaction for immersive virtual reality. In Proceedings of the 2017 IEEE Symposium on 3D User Interfaces (3DUI) (pp. 1-10). IEEE. https://doi.org/10.1109/3DUI.2017.7893315

[25] Plöchl, M., Ossandón, J. P., & König, P. (2012). Combining EEG and eye tracking: Identification, characterization, and correction of eye movement artifacts in electroencephalographic data. *Frontiers in Human Neuroscience, 6*, Article 278. https://doi.org/10.3389/fnhum.2012.00278

[26] Putze, F., Popp, J., Hild, J., Beyerer, J., & Schultz, T. (2016). Intervention-Free Selection using EEG and Eye Tracking. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction* (ICMI '16), Tokyo, Japan. ACM. https://doi.org/10.1145/2993148.2993199

[27] Rayner, K. (1998). Eye movements in reading and information processing: 20 years of research. *Psychological Bulletin, 124*(3), 372–422. https://doi.org/10.1037/0033-2909.124.3.372

[28] Riccio, A., Mattia, D., Simione, L., Olivetti, M., & Cincotti, F. (2012). Eye-gaze independent EEG-based brain–computer interfaces for communication. Journal of Neural Engineering, 9(4), 045001. https://doi.org/10.1088/1741-2560/9/4/045001

[29] Rosenfeld, R., D. Olsen, and A. Rudnicky (2001). "Universal Speech Interface." *Interactions 8(6): 33-44.*

[30] Whitmire, E. (2018). High-Fidelity Interaction for Virtual and Augmented Reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces* (pp. 796-798). Reutlingen, Germany: IEEE. https://doi.org/10.1109/VR.2018.8446520

[31] Wolpaw, J. R., Birbaumer, N., McFarland, D. J., Pfurtscheller, G., & Vaughan, T. M. (2002). Brain-computer interfaces for communication and control. *Clinical Neurophysiology, 113*(6), 767–791. https://doi.org/10.1016/S1388-2457(02)00057-3

[32] Zander, T. O., Gärtner, M., Kothe, C., & Vilimek, R. (2011). Combining Eye Gaze Input With a Brain–Computer Interface for Touchless Human–Computer Interaction. *International Journal of Human-Computer Interaction, 27*(1), 38-51. https://doi.org/10.1080/10447318.2011.535752

[33] Zhai, S. (1995). "Human Performance in Six Degree of Freedom Input Control." *PhD Dissertation*, Department of Computer Science, University of Toronto