

Controlling Player Avatars in Game Worlds using Multi-Modal Input Systems

Charlie Lloyd-Buckingham

May 28, 2022

1 Acknowledgements

This is my acknowledgements.

2 Abstract

My project is about....

Contents

1	Acknowledgements	2
2	Abstract	2
3	Introduction	4
3.1	Context	4
3.2	Research Problem	4
3.3	Project Aims	4
3.4	Project Objectives	4
4	Research	5
4.1	Literature Review	5
5	Methodology and Process	7
5.1	System Inputs	9
5.1.1	EEG	9
5.1.2	EMG	9
5.1.3	VR	10
5.2	Cross Network Data Transfer	11
5.2.1	LSL	11
5.2.2	Integration with Unity	11
5.3	Machine Learning	13
5.3.1	Keras	13
5.3.2	Keras Tuner	14
5.3.3	Baracuda	15
5.4	The Implementation of the Game	16
6	Conclusion and Future Work	17
6.1	Conclusion	17
6.2	Future Work	17

3 Introduction

3.1 Context

Throughout the history of the games industry, game makers have been exploring new ways to deliver their players unique experiences. This has come in the forms of the narrative decisions within a game’s story, stylistic decisions through the game’s art and, as a focus for this paper, the interaction systems designed around the player’s influence within a game’s world.

On account of the digital nature of video games, as technology has evolved over the years, so too has the hardware and methods of interaction used within gaming consoles. Beginning in 1972 with the creation of the first home console, the Magnavox Odyssey, over the following years, video game console manufactures have continued to push the perceived boundaries of interactive entertainment. In 2006, Nintendo demonstrated the potential of a motion controlled input system with the Wii. Following this, other manufactures also began to expand into the technology, with Microsoft developing the Xbox Kinect and Sony the PlayStation Move. Nowadays, companies like Oculus and the HTC corporation are exploring the market of consumer virtual reality headsets with the Quest and Vive, while Nintendo is showing off new ways of using old technology with their Labo Toycon games. All of this has demonstrated that with the market’s continued interest in new experiences, console manufactures and game developers will continue innovating new methods of interaction for their players.

3.2 Research Problem

There has also been a growing interest in the usage of electroencephalographic (EEG) and electromyographic (EMG) techniques being coupled with game engines for the purpose of research. The use of games in research has been invaluable for decades, due to the control gained over how a subject is stimulated throughout an experiment’s lifetime. The act of reversing this dynamic and using these systems for the purpose of entertainment has also been explored (Marshall et al., 2013), however there is yet be any product viable for consumers to play.

3.3 Project Aims

It is with this in mind, that this paper proposes to continue with the investigation into the viability of using EEG, EMG, and other additional input modalities in a video game. By acquiring data from multiple devices and combining them, the aim is for the creation of a single system capable of attaining and extracting meaningful interactions from it’s users for control over aspects of game world. Along side this, two example games will also be constructed, these will be used to verify if the system is capable of performing it’s tasks correctly and potentially demonstrate a working example from within a use-case environment.

3.4 Project Objectives

To get this system working, data from each device will need to be accessed live and streamed into the Unity game engine where a pre-trained artificial neural network (ANN) will be used to decipher a meaning suitable for the given game. From this the two games will be developed: the first will have the player use motor imagery to control the limbs of a virtual avatar; the second game will use the system measure the users state of mind, allowing for adaptive difficulty depending on the focus of the player.

4 Research

4.1 Literature Review

The usage of multi-modal input systems is not unheard of, almost all modern first person console shooters use the input controller’s built in gyroscope, in addition to the right joystick to control the direction of the players camera. This multi-input system allows much more finesse when aiming, resulting in the players having a more enjoyable experience (Toktaş and Serif, 2019). In the work put forward by Silva and Amaral (2014), they demonstrate that the inclusion of a multi-modal input system compared to a uni-modal input system can make video games perceivably more enjoyable, “it can be used to increase the feeling of empowerment on the player when using certain abilities, or to intentionally make in-game actions more difficult by demanding more physical effort from the player”. Though gyroscopic-assisted aiming is a more recent phenomenon, the concept of incorporating multiple input systems for singular interactions within gaming has been around for a while. Even back in 2009 with the release of ‘The Legend of Zelda: Spirit Tracks’ (Nintendo, 2009), when using the flute, players would be required to both blow into the DS microphone and use the touch screen to play specific tones. This interaction could have very easily be performed by mapping tones to regions on the touch screen, or pairing each tone to a button on the Nintendo DS, much like how previous instruments in the franchise have been implemented, E.g. ‘The Legend of Zelda: Majoras Mask’ (Nintendo, 2000). However, as stated by Silva, players could have a “feeling of empowerment” overcoming the set-pieces Nintendo designed, through this added multi-modal challenge.

Before looking into EEG as a part of a larger multi-modal input system, a focus on how it has been used on its own within video gaming will be explored. Games using EEG most commonly occur in the form of serious games, defined by Alvarez and colleagues (2011) as a video game that is “intended to depart from the simple entertainment”. This refers to games built for research, education and rehabilitation. Whether this is just for providing an environment for the comparison between different electroencephalographs (Liarokapis, Debattista, et al., 2014), evaluating a participant’s emotions and satisfaction (Vourvopoulos, 2013), screening for early signs of mental illness (Tarnanas et al., 2015), or cognitive rehabilitation (Alchalcabi et al., 2017).

The data captured from EEG can be used in a multitude of ways, depending on what is considered important for a given experiment. By using games it is then possible to focus on these targeted aspects of EEG: allowing for the invocation of responses necessary for measuring event related potentials (Ahn and Lee, 2011), or to provide a real-time environment capable of demonstrating the changes in mental states (Liarokapis, Vourvopoulos, et al., 2015) and motor-imagery (Ndulue and Orji, 2019).

However, it is debated whether EEG technology is still in it’s infancy. When investigating the current state of brain computer interfaces (BCIs) usability in the context of information transfer rates (ITR), Rashid and colleagues (2020) write “In spite of the many outstanding breakthroughs that have been achieved in BCI research, some issues still need to be resolved... the existing BCIs offer somewhat poor ITR for any type of effectual BCI application”, while in Cattani’s (2021) comparison, they state “In practice, this means that BCIs are unusable in traditional inputs, such as in keyboards or mice”. To the contrary, EEG technology has already been used within for video games as an interactive medium: from modifying a game’s difficulty based on the focus of the player with Tetris (Liarokapis, Vourvopoulos, et al., 2015); walking around the streets of Ancient Rome using motor-imagery in Rome Reborn (Ndulue and Orji, 2019); to piloting a space ship in Rock Evaders (Ndulue and Orji, 2019). EEG has shown its potential as a possible use for video game control.

The usage of EMG within video games has also been quite extensive. EMG based serious games have permitted the research of various topics, from measuring arousal to stimuli using the muscles in the face (Schuurink et al., 2008), to the effectiveness of myoelectric prosthesis training (Bessa et al., 2020). While their use outside of research has aided in the rehabilitation of patients, suffering: post injury (Gutierrez et al., 2020) (Schönauer et al., 2011); strokes (Ghassemi et al., 2019) and other medical disorders (Labruyère et al., 2013). Though much more infrequent, entertainment focused EMG games do exist. In large part with they goal for allowing people with motor impairments to have the opportunities to play games, Kamau-Mugro (2020) states “focusing on neck EMG, would give more control to individuals with hand disabilities or SCI [Spinal Cord Injury] patient as a control scheme or an entertainment interface”.

Having investigated the viability of EEG and EMG as uni-modal input systems, an exploration into their combined usage alongside other additional input modalities can be performed. Though the appearance of entertainment based multi-modal EEG and EMG systems are infrequent, multi-modal serious games using EEG and EMG are a lot more common. An example of which is Sivanathans (2014) work on multi-modal EEG analysis, in which data from in-game events was coupled with the EEG input streams to allow for a greater understanding of the results. Though there isn’t many instances of multi-modal EEG and EMG based video game systems, that isn’t to say inspiration for these systems cannot come from elsewhere, the majority of research in which multi-modal EEG and EMG systems have been built for is in the development of consumer prosthetics (Shi et al., 2019) and wheelchair controllers (Carlson and Millan, 2013). These systems pull from the data streams of EEG and EMG and by using eye-tracking, computer vision and inverse kinematics (McMullen et al., 2013) are able to allow for finer control over hardware systems that on their own EEG and EMG wouldn’t be able. By adopting a similar approach, these real world solutions can be used with virtual controllers instead.

5 Methodology and Process

In this section, the design and implementation of both the system and example game will be discussed. The system, shown in Figure 1, will be devised to allow for the retrieval of real-time data from three input devices: the Emteq Pro, for sampling the EMG data; the eego sports, for sampling EEG data; and a compatible VR headset, for tracking player position and orientation. Following this, the input data would then be passed directly to the application or distributed across the local network where through the use of machine learning, a meaning would be construed and used in a way to affect the game.

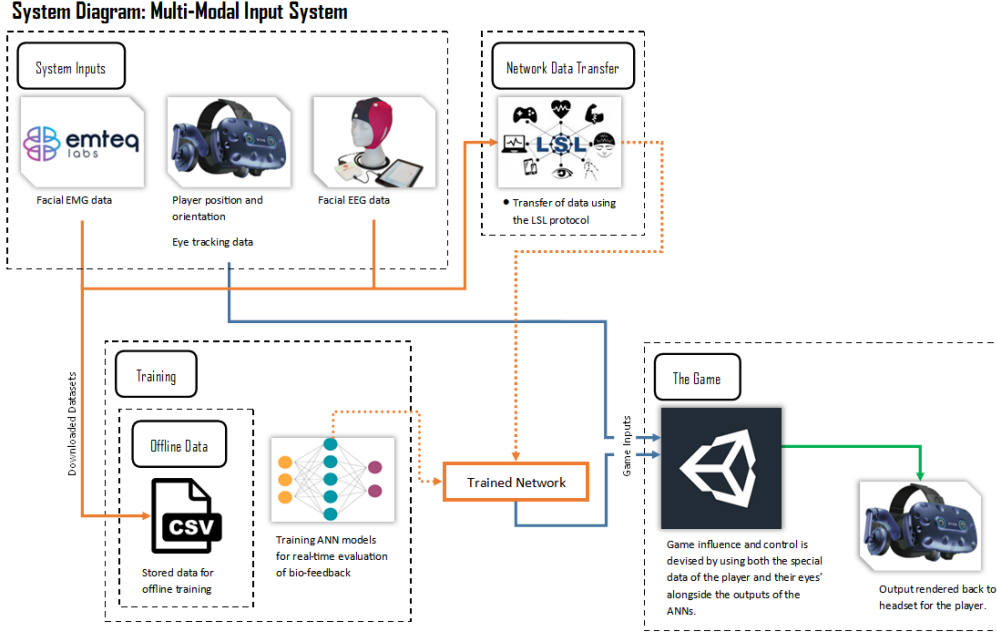


Figure 1: System diagram: How input data from the specified technologies will be accessed and used within the game.

The game was developed in the game engine Unity 3D. To use the data from the VR, EEG and EMG devices, each technology was integrated into the engine using C#, and a variety of packages provided by and for Unity. The game itself consisted of a pair of two mini-games, both focusing on potential aspects of how a system like this could reasonably be used in a real world example.



(a) Concentration based difficulty, Breakout mini-game. (b) Motor imagery controlled, sequence mini-game.

Figure 2: Screen shots of the game.

The first, Figure 2a, was a clone of the arcade game Breakout [Citation Need], in which the players aim is to break each block on the games board by bouncing a ball into them using a movable paddle, placed at the bottom of the play area. The intent with this part of the project was to allow the player to be able to effect the game world through an indirect influences, in this case, through the scaling of game difficulty. This was done by classifying the players state of relaxation and concentration, increasing and decreasing the balls movement speed based what state the neural networks within the system determines the focus level to be.

The second mini-game, Figure 2b, was about repeating a randomised sequence of colours. The player would be tasked with repeating this sequence back, by activating crystals of the same colours in the same order. This mini-game was used to explore the concepts of motor-imagery based avatar control. This was accomplished by requiring the player to activate a crystals by lifting one of the avatars arms up. To accomplish this the player was required to think left-wards or right-wards and in doing so the avatar would begin lifting the chosen hand forwards.

5.1 System Inputs

Research into the different input methodologies demonstrated



Figure 3: System diagram: The input devices for the entire system.

5.1.1 EEG

The recording of live EEG data is achieved through the sampling of voltage potentials across a subjects scalp. This is done through the use of a sensor array, the signals collected from these sensors are then collated within an amplifier, such as the EEGO Sport, and accessed in the form of a stream of float channels, usually one channel per sensors. Finally to record this data the amplifier is then connected back to a host computer, in which the saving and recording of the data is completed. The usages of this technology within the project will be specificity for motor imagery, and relaxation and concentration classification. Accessing this data live will be discussed in the networking part of this paper, however worth noting is that, due to the impractical nature of constant EEG tests, the large majority of this project was built around spoofing live EEG data using offline recordings. The specific ways in which this will be achieved will also be covered later in this paper.

5.1.2 EMG

Similarly with EEG, EMG is the recording of the ambient voltage of the body, however specifically for EMG this comes from the activation of motor neurons. By placing these sensors across the certain regions of the body, the activity of muscular movements can be recorded and understood. The use of this technology within this project was focused around the eye, where it was used to detect when the player would blink. The purpose of this was to be able to detect and cease the effects of any EEG data on the game during the times a player blinked, this was seen as a requirement to the large amount of interference the movement of facial muscle can cause upon EEG recordings. In parallel to that of the EEG, pre-record sessions of normal and blinking EMG data was also used in the stead of requiring access to live data each time the project need to be tested.

5.1.3 VR

Though this system was initially intended for the use of the HTC Vive Pro Eye, with Unity's XR Interaction Tool-kit package the specific headset used within the project is a bit less consequential, due to the new input system adapting interactions to action bindings, rather than using the original method for managing game inputs using if statements. Using the this package, tracking data accessed from almost any headsets can be very easily bound as an action in the interaction asset. Headsets do however require that a project to be built for a specific platform, such as Android or Windows. With the intent of building this project for the HTC Vive however, the build platform can remain as Windows. In addition, this allowed for a mouse and keyboard solution to also be integrated into the project, to replace the inputs given via the headsets. This was important to do as it allowed for faster and less cumbersome testing of the solution. This replaced the headsets orientation tracking with a parametric based look function controlled via the mouses movement delta, as well as control over the paddle in breakout using the 'A' and 'D' keys, instead of inputs from the headsets controller.

5.2 Cross Network Data Transfer

Unlike accessing data from both the VR headset and the Emteq Pro, where both maintain a direct physical connection to the games host device, accessing EEG data is more complicated. As stated previously, sampling EEG data is done through connecting the sensors to an amplifier and then the amplifier to a separate computer for recording, however to maintain an undisturbed and interference free sample collection this system must be separate from that of the system the game is running on. This creates a new problem, how to access the data generated on a separate machine across a local network, and more specifically how to access this data in as close to real-time as possible.



Figure 4: Data transfer across the local network using the LSL protocol.

5.2.1 LSL

For this to be done the solution of the LSL protocol (Kothe et al., 2019) was explored. LSL offers time-synchronous multi-type data transfer across local networks. Fortunately systems like the EEGO Sport come with software that supports this protocol by default, this has allowed for the distribution of all live data recorded from the head-wear to be accessed by any device on the network.

5.2.2 Integration with Unity

However, though the broadcasting of EEG data via LSL, using software like EEGO, is inconsequential, accessing these streams within Unity requires a bit more work. To do this the 'liblsl-Csharp' package was used. This package allows for the integration of the LSL protocol into Unity through C# scripting, requiring only the inclusion of the provided LSL.cs file, a C# scripting file, and a specific .dll, library file, matching the games build platform. The library file used in the project is determined by the VR headset that a game is being built for, in the case that this projects aims to build on a device like the HTC Vive, the required library was LSL's Windows .dll. However for headsets like the Pico G2 4KS and the Oculus Quest, an Android based library will need to be built for the specific version of Android these headsets are running.

Maybe talk about the actual C# scripts in as an extra paragraph

Also talk about the spoofing of the EEG and EMG data using LSL

5.3 Machine Learning

Even with the data accessible within Unity, understand a meaning from the data is more complicated, and requires a more involved solution. Taking these inputs at face value, both the EEG and EMG data returns a series of float values each sample, in the form of either a list or array. Manually testing these numbers directly within Unity wasn't a viable solution, due to how individually these values are somewhat meaningless. Instead the projected opted for the use of pre-trained FFNNs (Feed Forward Neural Networks) within the game, this would allow for an analysis of the patterns forming within the data set as a whole. From this, three networks were developed for the project, each was tasked with the classification of both the EEG and EMG streams, into the given outputs: for the motor-imagery portion of the project the networks were trained to classify the likelihood between, an imagined left, right and neutral state; as for the concentration based network, EEG data was once again used to determine between a state of relaxation and of concentration; while lastly, the final network used EMG data to classify between the opening and closing of a players eyes.



Figure 5: Artificial neural networks, there inputs and outputs.

5.3.1 Keras

To create these networks the Python library Keras (Chollet, 2015) was used. Keras is an interface library for Google Brains TensorFlow (2015), another python library used for the creation and training neural networks. Using the offline motor-imagery, relaxation and focus, and eye movement data sets mentioned in the System Inputs section of this paper, a model using 'Binary Cross-Entropy' was generated for each. These models where originally designed with a hidden layer 20% the size of the input layer, as shown in Figure 6a. However the results of this network after several hours of training didn't appear promising



(a) Example of the original FFNN used in the project. (b) Example of the updated FFNN using more hidden layers, where the size of each is also larger.

Figure 6: Manually designed networks.

Instead, the trialling of network architectures designed with more varied shapes and larger epoch counts began. Quickly, better accuracy's started to appear, the most promising of which were shown with networks that contained at least 2 or more hidden layers, and where the first layers node count exceeded 200.

This again was not much of a success, even with the higher accuracy, sitting at around 90%, when compared against unseen (validation) data the accuracy became equal to that of random chance.

5.3.2 Keras Tuner

Due to the lack of accuracy given by the manually designed FFNN's: the unlikely chance that with the continued testing against human designed networks resulting in any reasonably accurate networks; and the slow speed in which fairly testing these networks required; it was clear that another solutions was again needed. This solution emerged as hyper-parameter optimisation, specifically the library Keras-Tuner (O'Malley et al., 2019). Keras-Tuner allows for the automation of neural networks by generating and training through the use searching algorithms. There are 3 types of tuner algorithms supported by Keras-Tuner, 'Random Search', 'Hyperband' and 'Basyesian Optimisation'. Tests were performed over each of these tuners to test the creation of each of the three FFNN's needed for the project. After a while the choice was made to keep basyesian optimisation tuner for the networks as it showed the most success in predicting a high validation accuracy.

The inputs used for the network tuners are as shown in Table 1. Having ran this for a while for each of the networks....

Talk about how the networks performed, ANN for motor imagery is rubbish, but the eye one is decent

Parameter	Value
Objective	'val_accuracy'
Layers	1 to 6
Nodes	4 to 512, in steps of 4
Hidden layer activation function	'relu', 'sigmoid', 'softmax'
Output layer activation function	'relu', 'sigmoid', 'softmax'
Learning rate	0.01, 0.001, 0.0001
Early stopping callback rate	'val_loss'
Early stopping patience	3

Table 1: Input parameters used for tuning network search generation.

5.3.3 Baracuda

How where the networks put into Unity

What was my initial test case, the box moving?

5.4 The Implementation of the Game

Though this project is designed to work on a VR headset with real-time EEG and EMG data provided, testing resulted in building the project for Windows PC's with the a mouse / keyboard input system, alongside running the offline EEG and EMG data to supply a valid prototyping environment without the need for the access to both equipment and participants.

Player State Machine?

Moving controls, per state?

parametric look function?

How the networks where used?

Blocking of network inputs for blinking?

Visual indecation of player ANN states, blink idecator and the focus bar?

Lerping of everything?

growth rate?

Spawning of levels?

level interaction code?

6 Conclusion and Future Work

6.1 Conclusion

I conclude something...

6.2 Future Work

I want to do this in the future....

References

- Ahn, J.-S., & Lee, W.-H. (2011). Using eeg pattern analysis for implementation of game interface. *2011 IEEE 15th international symposium on consumer electronics (ISCE)*, 348–351.
- Alchalcabi, A. E., Eddin, A. N., & Shirmohammadi, S. (2017). More attention, less deficit: Wearable eeg-based serious game for focus improvement. *2017 IEEE 5th international conference on serious games and applications for health (SeGAH)*, 1–8.
- Alvarez, J., Djaouti, D., et al. (2011). An introduction to serious game definitions and concepts. *Serious Games & Simulation for Risks Management*, 11(1), 11–15.
- Bessa, D., Rodrigues, N. F., Oliveira, E., Kolbenschang, J., & Prahm, C. (2020). Designing a serious game for myoelectric prosthesis control. *2020 IEEE 8th International Conference on Serious Games and Applications for Health (SeGAH)*, 1–5.
- Carlson, T., & Millan, J. d. R. (2013). Brain-controlled wheelchairs: A robotic architecture. *IEEE Robotics & Automation Magazine*, 20(1), 65–73.
- Cattan, G. (2021). The use of brain-computer interfaces in games is not ready for the general public. *Frontiers in Computer Science*, 3.
- da Silva, G. A. (2014). Multimodal vs. unimodal physiological control in videogames for enhanced realism and depth. *arXiv preprint arXiv:1406.0532*.
- Ghassemi, M., Triandafilou, K., Barry, A., Stoykov, M. E., Roth, E., Mussa-Ivaldi, F. A., Kamper, D. G., & Ranganathan, R. (2019). Development of an emg-controlled serious game for rehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 27(2), 283–292.
- Gutierrez, A., Sepulveda-Munoz, D., Gil-Agudo, A., & de los Reyes Guzman, A. (2020). Serious game platform with haptic feedback and emg monitoring for upper limb rehabilitation and smoothness quantification on spinal cord injury patients. *Applied Sciences*, 10(3), 963.
- Labruyère, R., Gerber, C. N., Birrer-Brütsch, K., Meyer-Heim, A., & van Hedel, H. J. (2013). Requirements for and impact of a serious game for neuro-pediatric robot-assisted gait training. *Research in developmental disabilities*, 34(11), 3906–3915.
- Liarokapis, F., Debattista, K., Vourvopoulos, A., Petridis, P., & Ene, A. (2014). Comparing interaction techniques for serious games through brain-computer interfaces: A user perception evaluation study. *Entertainment Computing*, 5(4), 391–399.
- Liarokapis, F., Vourvopoulos, A., & Ene, A. (2015). Examining user experiences through a multimodal bci puzzle game. *2015 19th International Conference on Information Visualisation*, 488–493.
- Marshall, D., Coyle, D., Wilson, S., & Callaghan, M. (2013). Games, gameplay, and bci: The state of the art. *IEEE Transactions on Computational Intelligence and AI in Games*, 5(2), 82–99. <https://doi.org/10.1109/TCIAIG.2013.2263555>
- McMullen, D. P., Hotson, G., Katyal, K. D., Wester, B. A., Fifer, M. S., McGee, T. G., Harris, A., Johannes, M. S., Vogelstein, R. J., Ravitz, A. D., et al. (2013). Demonstration of a semi-autonomous hybrid brain-machine interface using human intracranial eeg, eye tracking, and computer vision to control a robotic upper limb prosthetic. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 22(4), 784–796.
- Muguro, J. K., Sasaki, M., Matsushita, K., Njeri, W., Laksono, P. W., & Suhaimi, M. S. A. b. (2020). Development of neck surface electromyography gaming control interface for application in tetraplegic patients’ entertainment. *AIP Conference Proceedings*, 2217, 030039.

- Ndulue, C., & Orji, R. (2019). Driving persuasive games with personal eeg devices: Strengths and weaknesses. *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization*, 173–177.
- Rashid, M., Sulaiman, N., PP Abdul Majeed, A., Musa, R. M., Bari, B. S., Khatun, S., et al. (2020). Current status, challenges, and possible solutions of eeg-based brain-computer interface: A comprehensive review. *Frontiers in neurorobotics*, 25.
- Schönauer, C., Pintaric, T., & Kaufmann, H. (2011). Full body interaction for serious games in motor rehabilitation. *Proceedings of the 2nd Augmented Human International Conference*, 1–8.
- Schuurink, E. L., Houtkamp, J., & Toet, A. (2008). Engagement and emg in serious gaming: Experimenting with sound and dynamics in the levee patroller training game. *International Conference on Fun and Games*, 139–149.
- Shi, C., Qi, L., Yang, D., Zhao, J., & Liu, H. (2019). A novel method of combining computer vision, eye-tracking, emg, and imu to control dexterous prosthetic hand. *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2614–2618.
- Sivanathan, A., Lim, T., Louchart, S., & Ritchie, J. (2014). *Temporal multimodal data synchronisation for the analysis of a game driving task using eeg* (Vol. 5). Elsevier.
- Tarnanas, I., Laskaris, N., Tsolaki, M., Muri, R., Nef, T., & Mosimann, U. P. (2015). On the comparison of a novel serious game and electroencephalography biomarkers for early dementia screening. In *Genedis 2014* (pp. 63–77). Springer.
- Toktaş, A. O., & Serif, T. (2019). Evaluation of crosshair-aided gyroscope gamepad controller. In *International conference on mobile web and intelligent information systems* (pp. 294–307).
- Vourvopoulos, A. (2013). *Brain-controlled virtual environments-an evaluation study of brain-computer interfaces for serious game interaction*.

Software

Chollet, F. (2015). Keras.

Google Brain Team, G. (2015). Tensorflow.

Kothe, C., Medine, D., Boulay, C., Grivich, M., & Stenner, T. (2019). Lab streaming layer.

Nintendo. (2000). The legend of zelda: Majora's mask.

Nintendo. (2009). The legend of zelda: Spirit tracks.

O'Malley, T., Bursztein, E., Long, J., Chollet, F., Jin, H., Invernizzi, L., et al. (2019). Kerastuner.