**OPTIMIZING EEG-BASED CONTROL**

**TO**: Aaron Maus

**FROM**: Justin Haysbert, Gabriel Sagrera, Shayne Shelton, Ryan Stevens

**DATE**: 12/07/23

**GitHub**: <https://github.com/stevensryanw/BCI_Infinity>

**CODE UPDATES**

*Machine Learning*

Our machine learning scripts are still in a similar state from the last milestone report. We currently have working scripts for multiple SKLearn algorithms, PyTorch, and Tensorflow. The Tensorflow and PyTorch scripts have been tested on an M1 mac and a desktop PC with the latest generation of AMD processors and graphics cards. However, with the AMD route, Ubuntu is required as the instruction set is not available for Windows. The SKLearn algorithms are very compatible and have worked with zero change on every device we have tested, but they are much less intensive algorithms that do not require GPUs.

*Graphical User Interface*

Our Tkinter GUI now has pages for live streaming recorded data, user training prompting, model selection and training, and model output. Temporarily, while we wait for a headset, we have a keyboard output page to test if PyAutoGUI will work in our GUI to control our game inputs. Once we do receive the headset, we will be able to train, model, and test, so we can take away the page since it is just hard coded buttons for arrow key outputs. All of the pages are still in very basic states while we finish their individual python scripts for GUI functions. We will continue this strategy over winter break to develop each part of the GUI as we create each function we need.

*Python Video Game*

For our video game implementation, we are currently deciding which route to take. The options are Gather Town, a web or application based game/zoom/conference room combination, or building our own Python game in the GUI from scratch. We need to quickly decide which route to take, to do this we are actively working on if our GUI key outputs can control Gather Town or not.

*User Prompting*

We have started writing the script for our user training prompting separately from the GUI, still using Tkinter, for faster development. This script consists of calling the data recording function, but also modifying it to add our new labels based on norm or movement type that was prompted. We will be basing all of the prompting based on our EEG Planning and Procedure from Milestone 3.

*NeuroPype*

We are looking into using NeuroPype as our data streaming and preprocessing software. NeuroPype is specifically designed to aid in biosignal processing and could help us in simplifying our preprocessing methodology.

**EEG PLANNING & PROCEDURE**

While we still need to receive our Open BCI EEG headset, we have completed a procedural plan for our data collection sessions as stated in Milestone 3. We believe we can provide comfortable and efficient training sessions for our subjects using these guidelines and procedures. We also want to preface that these guidelines and procedures are subject to change and adaptation to ensure the best training experience and results. This procedure has also been registered in our IRB.

**IRB**

We have submitted our initial Humans Subjects Research determination worksheet. The sheet has been processed with several notes. Initially, we made the determination that we were not going to be completing a systematic investigation in any regard; however, our IRB manager has noted that we are “using machine learning algorithms to affect the use of a medical device.” While we do not entirely agree with this note, we may need to edit this portion of our worksheet to move towards approval. Another note we received was “Please answer ‘Yes’” to both of the following questions: “Will you obtain information or biospecimens through intervention with living individuals…?” and “Will you obtain information or biospecimens through interaction with living individuals”. We disagree with this point because we do not believe we are collecting true bio-specimens from individuals. The final note we received on our IRB application was “endeavoring to ‘optimize the training process’ for participants using such a device would appear to be an activity intended to affect the function of the body.” We also disagree with this function because EEG cannot impact the functioning of a person at all. Overall, we have decided to get into contact with our IRB manager to sort out these issues. There is an apparent disconnect on the idea of what our project actually is, which seems to stem from the misconception that we are using this EEG as a medical device. We hope that by meeting with our manager we can resolve these misinterpretations. Note that the form we are referring to is only a determination form; therefore, until we hash out these errors, we will not be able to move forward with our full IRB. This IRB is essential if we decide to include outside participants in our study.

**UPDATED PRIOR WORKS**

We have found some useful youtube videos that will aid us throughout our project. One in particular comes from youtube sentdex who uses the openBCI headset to play a directional game using a two class classification method. This video provides valuable insight into data preprocessing and streaming. While we do believe that sentdex’s methodology in classifying EEG data is subpar, we do believe that we can learn from his preprocessing techniques.

We have also found documentation of an openBCI Mental Imagery project on the openBCI forum. This project details two class classification of motor imagery using openBCI’s EEG headset. We hope to apply some of the methodology used in this project into our 4 class classification project. This project also provides insight into preprocessing tools that we can use throughout our project. The most useful one being NeuroPype, a software that integrates with python allowing custom data filtering, real time visualizations, data streaming methods, data segmentation, and data annotation. We believe that this software could be extremely useful in simplifying our preprocessing pipeline.

**HEADSET PRINTING**

We have decided to use the Ultracortex Mark 4 found on OpenBCI for the EEG. For this headset, the control board and the electrodes are sent, while the frame to hold the electrodes to the person’s head is 3D printed by the buyer. Despite some issues that came up during the printing process, we have the frame 3D printed. The frame has also been completely assembled with just the electrodes missing. We currently have only created the large headset version as the person we will first collect data on using the EEG falls under this size. If we do perform human subject testing on a population outside the group, we will print and assemble the frame for the other two sizes (small and medium) if this occurs.

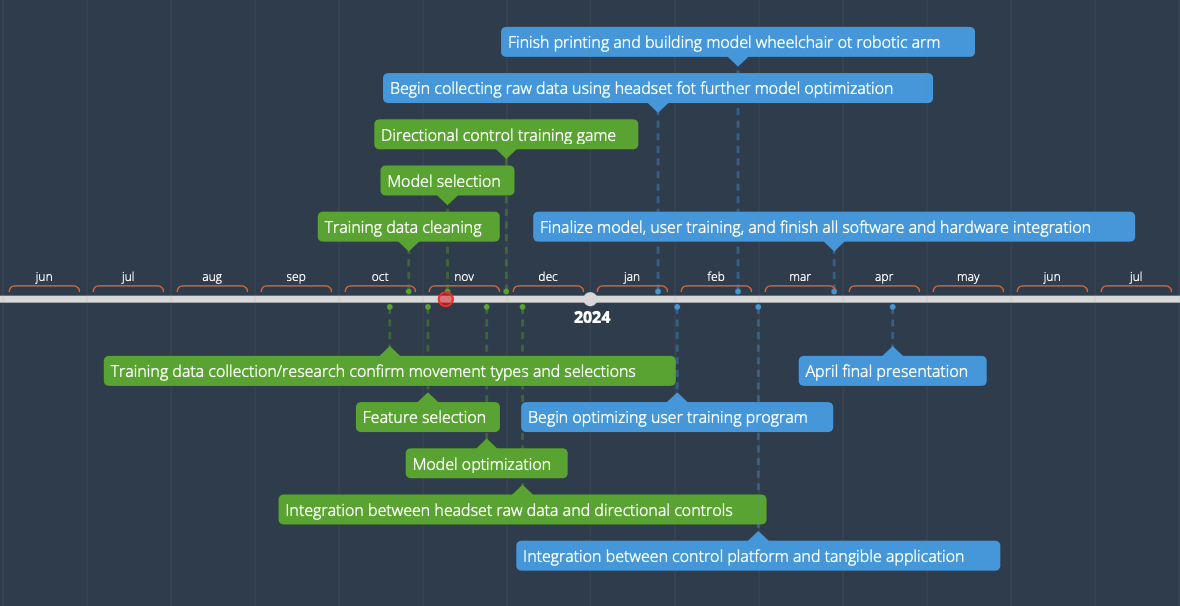
**WHEELCHAIR PROTOTYPE**

While we are developing a game for training purposes, we need a working robotic wheelchair for concept validation as our ultimate goal is to control a robotic device. We originally planned to use a RaspberryPi or Arduino based device. However, due to budget constraints, we decided to downsize our microcontroller. Additionally, RaspberryPi’s and Arduinos have more processing power than we would need for the tasks we are trying to complete. Our current workflow puts a majority of the processing and computational load on laptop while the microcontroller just receives and executes commands. We will be using an ESP-12F WiFi development board in our project as it allows for wireless development, and there exists some prior open source code for miniature car applications. We will order frames and connectors in the coming week to begin development.

****

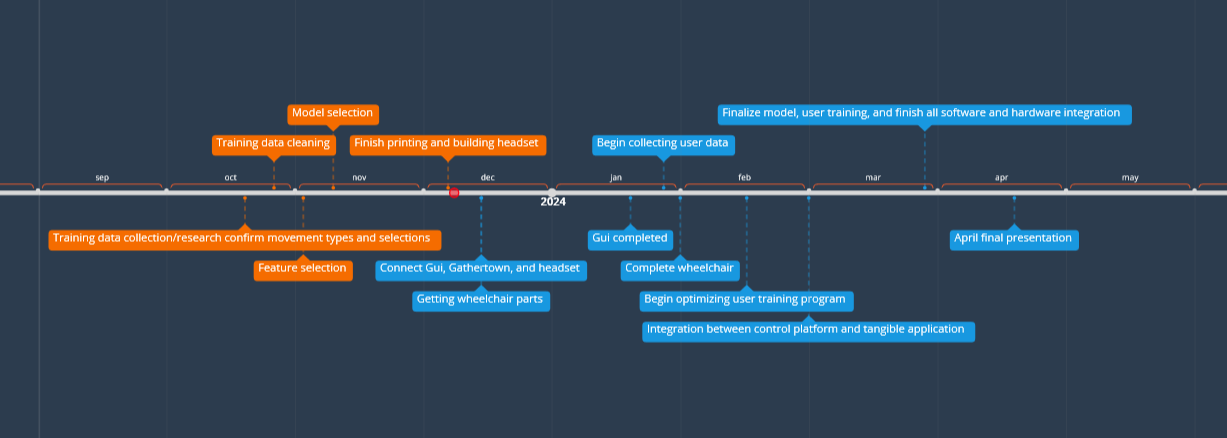
*Figure 1*- Picture of proposed prototype to develop

**TIMELINE UPDATE & NEXT STEPS**

****

*Figure 2* - Original project timeline

Our mentor, Professor Hassan, agreed to purchase an OpenBCI headset for our group to have more data collection channels. The headset will take a few months to order and deliver, so we made a few changes to our timeline. The new timeline pushes our data acquisition to the Spring as we wait for the headset.



*Figure 3* - Updated project timeline

In *Figure 3*, all completed steps of our timeline are indicated in orange. By the end of this semester we plan to have the GUI and Gathertown implemented to allow control of the character from the GUI. We will also implement the headset into the GUI if it arrives before the end of the semester and if not it will be implemented at the beginning of next semester. Once we implement the headset we will start collecting user data in order to begin model refinement.

**TEAM MEMBER RESPONSIBILITIES**

*Justin Haysbert*

Completed a procedural design for data collection and am working to contact our IRB manager so that we can complete the IRB. I am currently researching NeuroPype to plan data streaming and preprocessing methodology for our application.

*Gabriel Sagrera*

I am currently assisting Shayne with designing the wheelchair model as well as the mechanism for moving it. I will also work on integrating the wheelchair with the headset and on implementing Gathertown into the GUI.

*Shayne Shelton*

I researched small vehicle models for the model wheelchair concept validation testing in the Spring. I am responsible for planning a cheap and feasible plan for manufacturing a single working wheelchair prototype that takes 4 inputs and can integrate with our GUI.

*Ryan Stevens*

Recently achieved a way of outputting system keystrokes through our Python GUI. Next I will be working on a way to send those keys to Gather Town, or we will quickly need to decide whether to change routes and create a very simple game inside the GUI. Additionally, I have started outlining the user prompting page and creating scripts for the data modification. Once we receive the headset we believe we can hit the ground running and start working to train an initial model to test the output functionality.

**MENTOR MEETINGS**

We have met weekly with Professor Hassan every Friday at 2 pm. In these meetings, we have discussed the week’s progress, the timeline overall, and what to do for the upcoming week. We hope to have the headset soon for testing, in which we can work on the code during the meetings instead of only going over deadlines.