Virtual Testing Environment for Brain-Controlled Vehicles

James Cate, Jinxiao Yu, Soheil Borhani

Department of Mechanical, Aerospace, and Biomedical Engineering

Introduction

Background

Brain computer interface (BCI) systems are rapidly growing mechanisms set on controlling computer programs and robotics using the mind. Recently, non-invasive methods such as electroencephalography (EEG) have become a popular alternative to invasive techniques. EEG uses an electrode array placed on the scalp to monitor electric activity produced by brainwaves across multiple areas of the brain. Algorithms can be used to decode this data into intentions or instructions, but training time can last days, even months.

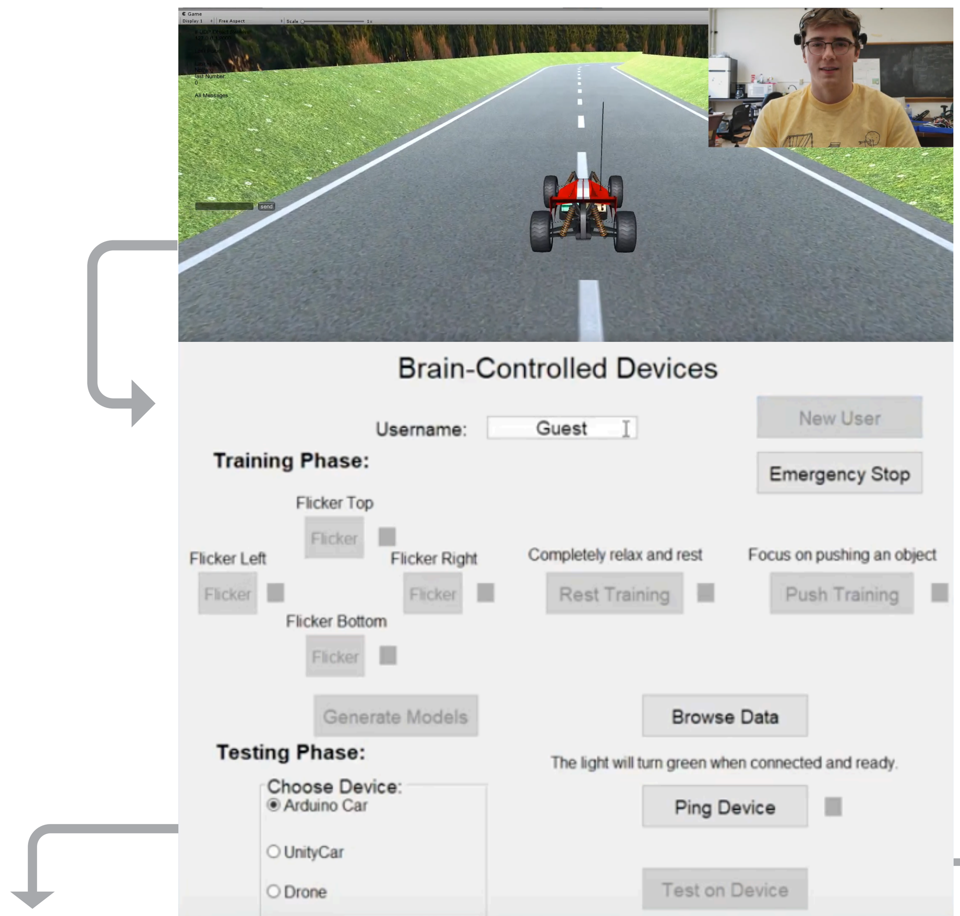
Our lab, through the use of the novel paradigm, imagined body kinematics (IBK), was able to drastically reduce training time required to control an RC car. IBK instructs the user to simply imagine moving an object using their mind without any physical movement necessary. With just 20 seconds of training, 10 for pushing, 10 for resting, our model could produce a feeling of control from the user. One problem was that our system required a physical testing environment with RC cars that were cumbersome to set up and prone to failure. Having a virtual testbed could solve our system's current issues while opening many doors for future work.

Objective

The goal was to use Unity to simplify and empower our BCI system, making it maximally portable, adjustable, and testable, allowing easier data collection and presentation. When developing this virtual testbed we recognized the opportunity to explore other paradigms in juxtaposition and combination with IBK, namely steady state visually evoked potentials (SSVEP). SSVEP recognizes that flashing lights leave artifacts that are echoed in EEG data, so by instructing a user to look at lights of a given frequency and codifying said frequency with meaning we can infer their intention based on what they are looking at. This leads us to the second goal, using SSVEP to incorporate higher degrees of freedom, such as having the car turn left and right, while maintaining the short training time that sets our platform apart.

Methodology

Our prior protocol required use of BCI2000 to distribute data, so our first step was to develop a GUI in MatLab to minimize the amount of software required to run our package. Before a user can control the car, a training phase must occur, as seen below. Based on this training data, classification models are quickly formed using multiple regression. Upon testing, data is read through the sensing headset, processed by our MatLab classifiers, and the result is forwarded over UDP to Unity, where it is received and translated into car movement real-time.

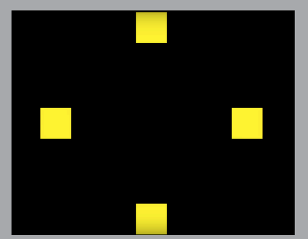


Visual feedback loop

User interfaces with MatLab GUI

Training

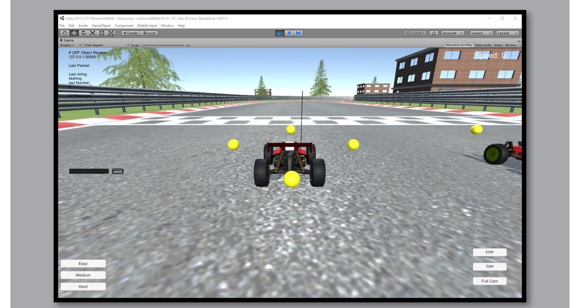
Data is processed and models formed using Simulink and MatLab.



Training sessions involve either mentally pushing an object (IBK) or observing flickering lights (SSVEP at 7, 9, 11, and 13 Hz).

Testing

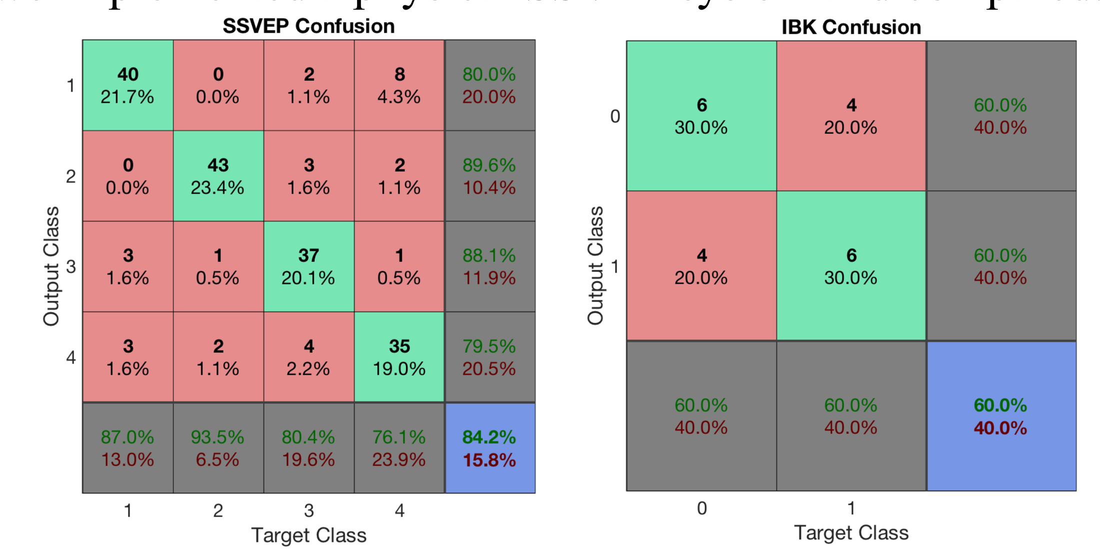
Testing data is processed by user's models and classification is sent to Unity over UDP

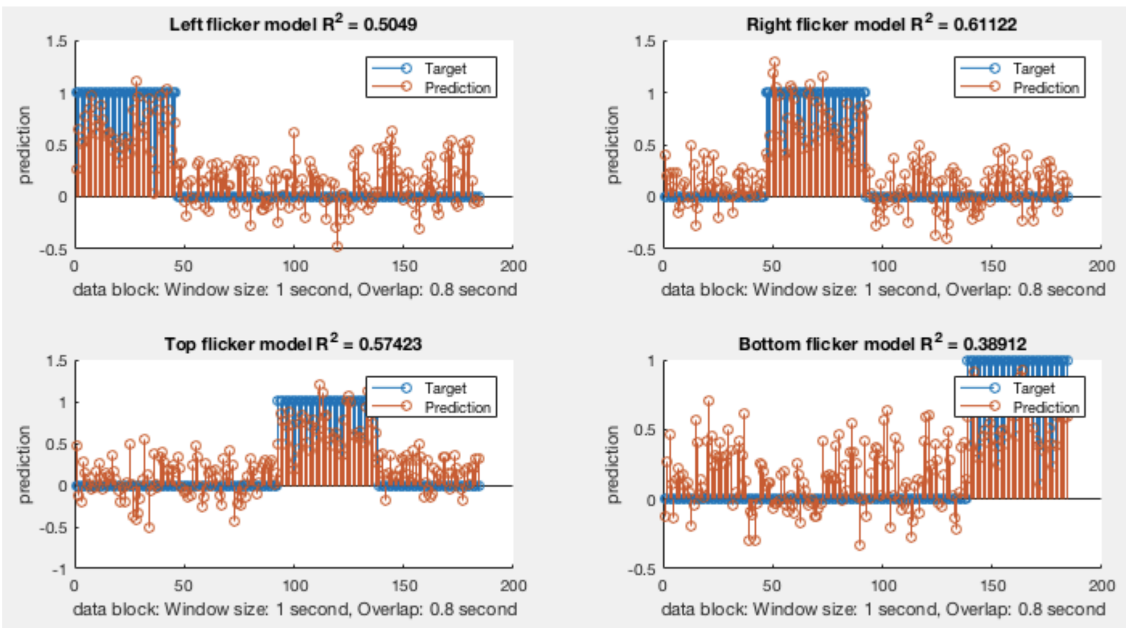


Testing sessions involve trying to manipulate the car toward the goal.

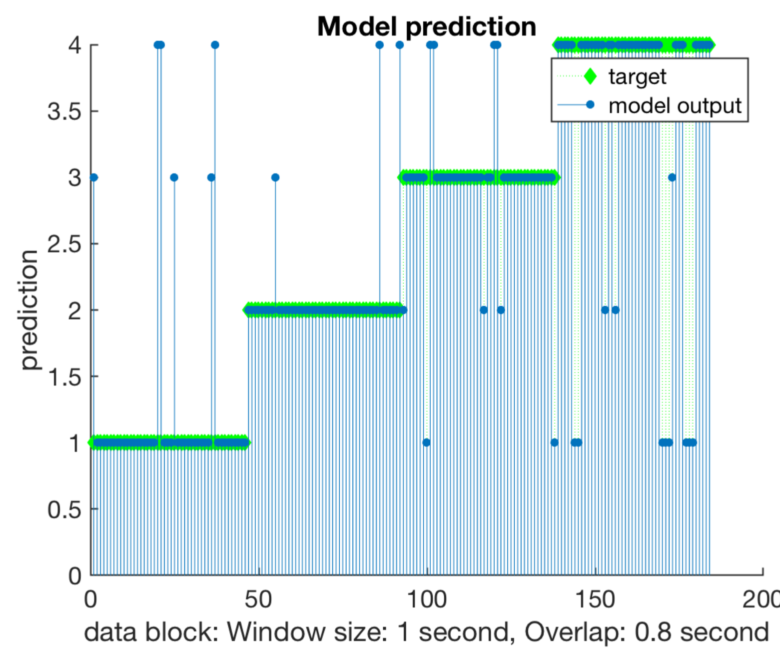
Results

Classification results and a demonstration video are seen below. By introducing SSVEP to the platform we were able to add two degrees of freedom without sacrificing accuracy or greatly lengthening training time, a feat nearly impossible using solely IBK. Total training time is now one minute. By integrating both techniques our platform can not only decipher direction, but by using IBK can tell whether movement is desired at all, however there is some room for improvement here. To test accuracy of the virtual platform, we implemented a physical SSVEP system and compared.





Scan here for a video of the the training and testing process



Conclusion

The virtual testbed was successfully implemented and confirmed to behave analogously to physical systems. The entire platform is now contained within MatLab and Unity, greatly increasing portability and making data collection very simple. Furthermore, this new platform has already been used to add two degrees of freedom through the addition of the SSVEP paradigm. The high information transfer rate of the paradigm allows for high classification accuracy, and when used in tandem with IBK we can remove most "fidget" motions. SSVEP adds 10 seconds of training time per degree of freedom, keeping our platform at an impressive one minute required training. There is still work to be done in perfecting the IBK classifier, optimizing the light positions around the car, adding more degrees of freedom (perhaps a virtual drone), and experimenting with additional paradigms using this tool. This tool could be used in neuro-rehabilitation for amputees, stroke victims and those with ALS, greatly helping to improve their way of life and augmenting their reality with virtual reality.

Bibliography

1. Bakardjian, H., Tanaka, T. and Cichocki, A. (2010). Optimization of SSVEP brain responses with application to eight-command Brain–Computer Interface. *Neuroscience Letters*, 469(1), pp.34-38.

2. Guneysu, A. and Akin, H. (2013). An SSVEP based BCI to control a humanoid robot by using portable EEG device. *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*.

3. Prueckl, R. and Guger, C. (2009). A Brain-Computer Interface Based on Steady State Visual Evoked Potentials for Controlling a Robot. *Lecture Notes in Computer Science*, pp.690-697.

4. Wright, J., Macefield, V., van Schaik, A. and Tapson, J. (2016). A Review of Control Strategies in Closed-Loop Neuroprosthetic Systems. *Frontiers in Neuroscience*, 10.

5. Wu, C., Chen, Y., Zaeni, I. and Chen, S. (2016). A new SSVEP based BCI application on the mobile robot in a maze game. *2016 International Conference on Advanced Materials for Science and Engineering (ICAMSE)*.

THE UNIVERSITY OF TENNESSEE KNOXVILLE

Acknowledgements:

This work was in part supported by NEURONET and the Office of Undergraduate Research at the University of Tennessee, Knoxville. The authors are grateful to Justin Kilmarx and Reza Abiri for useful discussions. The work was conducted under the supervision of Prof. Xiaopeng Zhao at the Department of Mechanical, Aerospace, and Biomedical Engineering.