Multimodal body machine interface based on finite state machine principles

THE UNIVERSITY OF RHODE ISLAND

Wearable Biosensing Lab

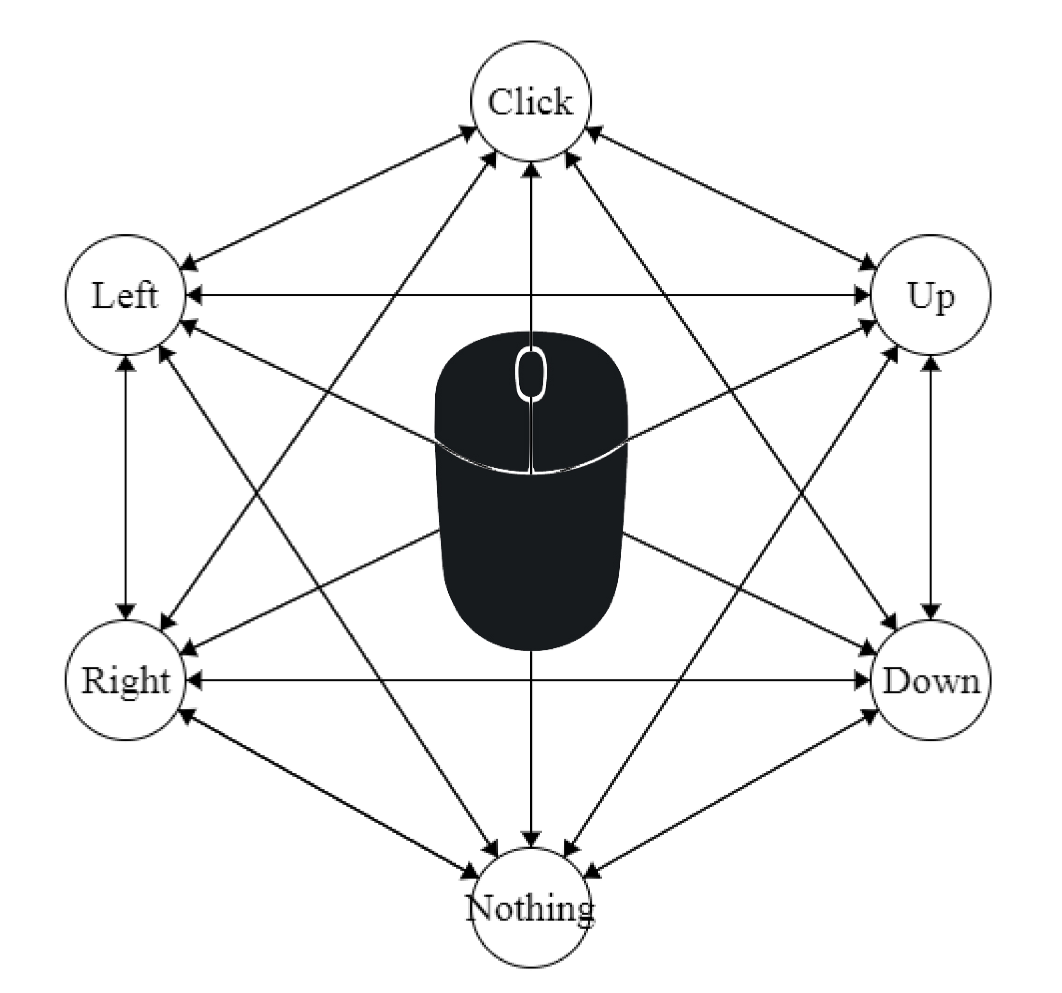
Dylan Kennedy, Kunal Mankodiya

Department of Electrical, Computer and Biomedical Engineering,
University of Rhode Island, Kingston, RI 02881

Rhode Island INBRE

Introduction

Patients suffering from various forms of paralysis find their quality of life to be greatly impaired. A body machine interface (BMI) acts as an augmented and alternative communication system which uses biological sensors to provide assistance to individuals whose capabilities are limited by some form of disability. The four components of a BMI are signal acquisition, feature extraction, feature translation, and device outputs. Broken down, this is: obtaining a signal, identifying characteristics of that signal, mapping those characteristics to some type of command, and using that command to generate an output. Due to the complexity of the process, the greatest impediment to the development of BMIs is feature extraction and translation. Finite state machine principles were used to create a BMI whose inputs are mapped to its outputs using a truth table (shown below in Figure 1). In this way, feature extraction and translation are drastically simplified.



Input					Output				
Left EMG Out	Left EMG In	Right EMG Out	Right EMG In	EEG Attention	Up	Down	Right	Left	Click
1	0	0	0	0	1	0	0	0	0
0	1	0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0

Figure 1. Finite state diagram and truth table

Materials and Methods

- 1x Arduino Uno is a microcontroller for receiving the data from the sensors and also for computing.
- 1x Neurosky Mindflex is a small headset with single channel electroencephalography (EEG).
- 4x MyoWare Muscle Sensors are electromyography detectors to measure muscle activity.
- 1x Computer was used for serial communication with the Arduino Uno.

A combination of EMG (MyoWare Muscle Sensors) and EEG (Neurosky Mindflex) were combined to form the biological sensors used in the proposed BMI. These sensors were then connected to a microcontroller (Arduino Uno) which was used both to process the incoming data and generate outputs. Using a computer attached to the Arduino via USB, a code was written in the Arduino IDE (microcontroller software) to implement the shown states using an embedded if-else loop. The sensors gather data which functions as inputs to the Arduino. The Arduino then determines whether these inputs meet a certain threshold and labels them as a binary 0 or 1. Depending on the combination of inputs, a specific output is generated. This process is described in Figure 1 above.

Results

The four components of a BMI were successfully implemented. It is able to acquire data from multiple sources, sort this data into either a 0 or 1, map the combination of 0s and 1s to a certain output, and subsequently generate this output. Thus, a robust multimodal body machine interface based on finite state machine principles was created. It serves as a proof-of-concept device to demonstrate how this process may be used to advance the development of other body machine interfaces. A diagram of the system is shown below in Figure 2.

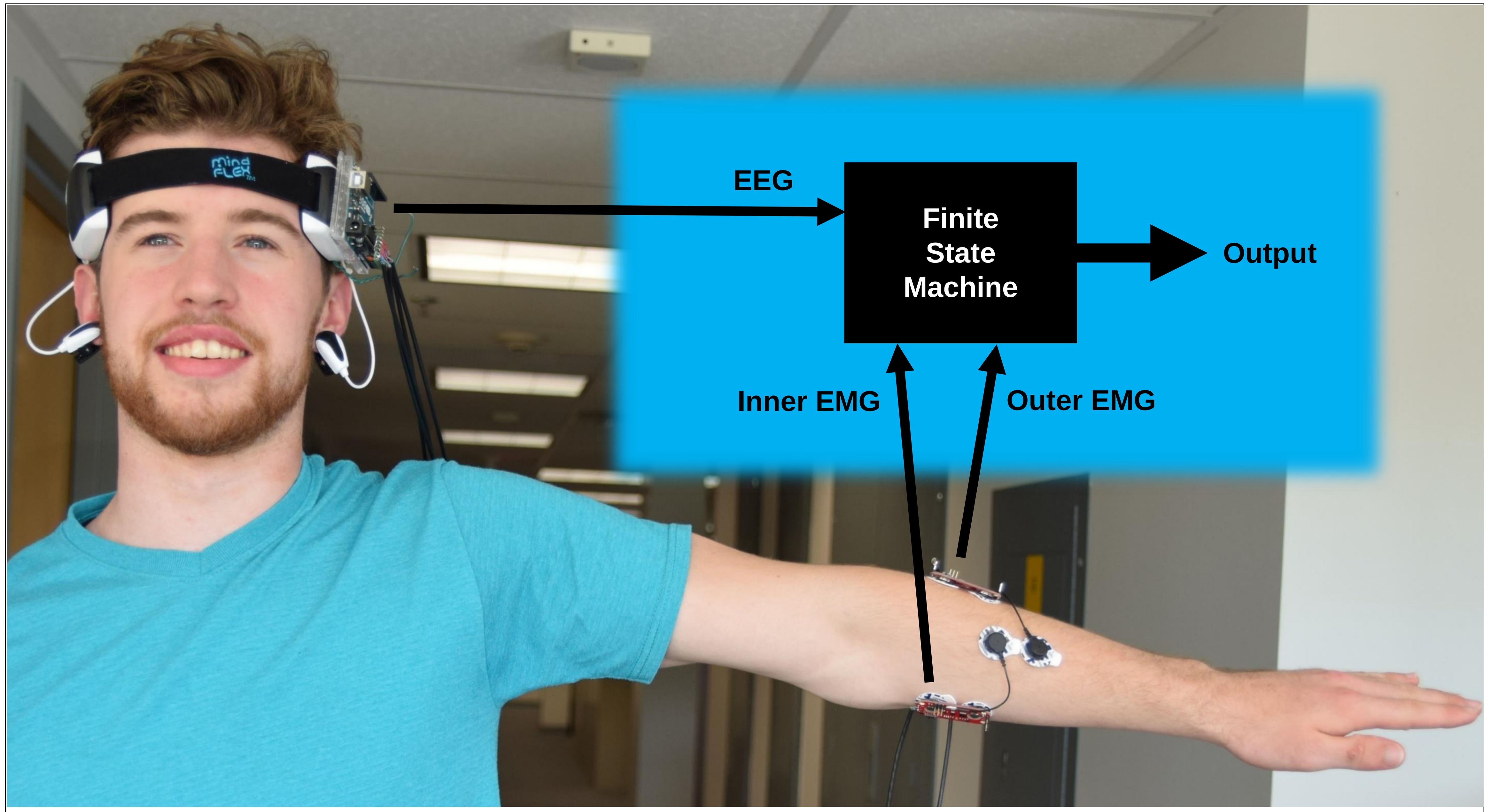


Figure 2. Completed body machine interface

Conclusions

Finite state machine principles have the capability to be used in creating robust body machine interfaces. The use of magnitude thresholds is a simple and effective method of feature extraction when used with multiple inputs. Thus, the need for complex signal processing is eliminated. This concept provides an effective framework from which more advanced BMI can easily be implemented. The system can be enhanced by adding new modalities, bluetooth, and hardware

References

1. Casadio, M., Ranganathan, R., & Mussa-Ivaldi, F. A. (2012). The body-machine interface: a new perspective on an old theme. *Journal of Motor behavior*, *44*(6), 419-433.
2. Lin & John (2006) Lin TA, John LR. Quantifying mental relaxation with EEG for use in computer games. International conference on internet computing; Atlanta: CSREA Press; 2006. pp. 409–415.

Acknowledgements

The authors would like to thank the RI-INBRE SURF program for funding this work. Printing services provided by the RI-INBRE Centralized Research Core Facility supported by Grant # GM103430 from NIGMS, NIH.