

Myo Armband and 3D objects

Electromyography or EMG is a medical technique used for evaluating and recording the electrical activity produced by muscles. Thalmic Labs adopted its technology in its armband Myo, which tracks a user's muscle movements to interact with displays and computer programs.

Purpose

The main purpose of this work is to explore the Myo armband technology in gesture control, and assess its potential application in the University.

The main objective of the project is to create a prototype, in order to use the armband to control 3D objects by rotating, zooming, moving and navigating, providing an immersive experience. In the end, the prototype should be able to load 3D objects found in the Lapworth Museum.

Another objective is to evaluate Myo and see if it fits in the University environment, by providing teachers and/or students a new way to improve the learning environment.

This report also contains an evaluation of the Armband.

History of Myo Armband

Myo armband was first announced by Thalmic labs on the 25th of February 2013, with a video on YouTube linking to a pre-order page. Two days later they announced that 10 thousand units had been pre-ordered. In December, the alpha version of the bracelet was shipped to some developers and partners, so they could start making programs, or connectors (as the company calls them), to make it easy to control your computer while listening to music, watching movies, presentations, and even Facebook.

On June 2014, Thalmic Labs revealed the final version of their product, thinner and lighter than the alpha version.

Today you can order one from their: site <https://store.myo.com>.

How it works

Myo contains 8 pods with a proprietary EMG sensor on each, in order to detect muscle movements around your



Figure 1: Myo armband

forearm and hand gestures. It also contains a gyroscope, an accelerometer and a magnetometer to track arm movements, like spinning or moving.

The Myo vibrates to send information to the user. (E.g. When it is synced with a pc or when a gesture is detected.)

The battery should support a full day of use, as said on the company site, and it can be easily recharged by through a micro USB port.

Myo Armband and 3D Objects (04/09/2015)

Bruno Mouta

By detecting muscle movements when a gesture is made, Myo is trained to guess which gesture has been made and respond accordingly. The gestures that it currently supports are:

- **Double-tap:** You tap your middle finger with your thumb twice. It is normally used to unlock Myo, so it can begin detecting other gestures.
- **Fist:** To use this gesture the user must close the hand in a fist.
- **Spread Fingers:** To use this gesture the user must spread all five fingers.
- **Wave in:** To use this gesture the user must wave his hand towards his body.
- **Wave out:** To use this gesture the user must wave his hand contrary to his body.

How to use it

1. Install Myo Connect, the program that makes the interface between computer and bracelet.
2. Make sure that Myo is charged
3. Check that the Bluetooth adapter and USB cable are connected to a computer, so Myo Connect can initialize your armband for use.
 - To use it later in a different computer you only need to touch the bracelet with the Bluetooth adapter.
4. After the initial configuration steps, remove the cable and Myo will be ready to be used.
5. Wear it on your forearm, with the USB port towards your wrist and the Myo logo led facing up in your arm, to ensure maximum accuracy with the factory profile.
 - If the user wishes, he can create a custom profile, so Myo can better detect the movements.
6. Perform the sync gesture (the current version syncs by waving out and holding the gesture for a while) until Myo vibrates to warn the user that it is synced.
 - Myo sometimes need some time to warm up with the user body temperature, and it may ask to do the sync gesture one more time.



Figure 2: Myo being synced

Included connectors and modes

Myo already comes with some connectors (like VLC), in which the user can control the media player using the armband to pause and start a movie, rewind, fast-forward and control volume. It comes with Spotify, iTunes and Netflix connectors also.

Myo also include a keyboard mapper, so user can map the gestures to a specific keyboard letter or mouse click, and with it create a connector easily, so it can be used with any program the user desires.

Myo has also has a presentation mode that can be used to control Keynote, Power Point, Prezi, Google Slides or Adobe Reader. The movement controls are simple, simply having to double-tap to advance a slide, wave in and hold to go back one slide, make a fist and rotate your hand counter clockwise to enable the pointer and even zoom in the application when you make a fist and rotate your hand clockwise. To get out of pointer mode or zoom, you just have to double-tap.

The feature that is used in the prototype is the Mouse Control. It enables you to control the mouse of your computer, so you can move it normally with your arm. To activate it, when using Myo, raise your arm, make a fist, lower your arm and then release the fist. You will see a menu. Select Myo Mouse and confirm by spreading your fingers. The controls are simple:

- **Double-tap:** Enable or disable mouse movements
- **Fist:** Left click

- **Spread Fingers:** Right click
- **Wave in:** Scroll down
- **Wave out:** Scroll up

With these controls in mind, the prototype was made to show 3D objects and controlling their movement.

Implementation

- The first task of the implementation would be to read the 3D objects and render them in the screen, allowing the user to move it freely with the armband, mouse or keyboard. I knew nothing of how to render a 3D object on a screen, so I opted for OpenGL because of the amount of material available to help developers, being the most know API for high performance graphics.
- After a week studying OpenGL, I decided to choose a language to implement in the program. Myo SDK uses C++, but as I did not know the language, I decided to look for alternatives. I chose python because it is an easy language to write and understand, so the project could be easily maintained if necessary.
- Python supports OpenGL, and it has Pygame, a known Framework used to make games and interactive software.



Figure 3: Example of 3d objects rendered by the prototype

- After numerous attempts at implementing gesture control built-in in the program, I was not capable of properly detecting arm movement using Python, so I decided to use the provided functions that comes in Myo Connect; Mouse Mover.
- The program loads all the 3D images contained in its folder. The user then has 3 options to control the objects:
 1. **Keyboard:** Use the arrows to control the rotation, the letters “z” and “x” to zoom in and out, and letter “n” to skip to the next object.
 2. **Mouse:** Click once with left click to start the rotate mode, and click again to turn it off. Use the wheel to zoom in or out and right click to go to the next object.
 3. **Myo armband:** Since you have to be in the Myo Mouse mode, the controls are the same as the mouse, but with the corresponding gestures:
 - **Fist:** Enable or disable rotate mode, rotate by moving your arm freely
 - **Spread fingers:** Go to the next object
 - **Wave In:** Zoom in the object
 - **Wave Out:** Zoom out the object

Future Work

- Design a custom interface so the user can freely choose which image to see.
- Optimizations in the loading of the objects. Currently only the 64-bit version of the program works, because of a Memory Error while loading many images.

- Capacity of seeing 2D images.

Conclusion

Being easy to use, easy to program, and relatively inexpensive (\$199) Myo has the potential to be the most used product in the gesture control department. With proper guidance, even a child can be able to use it, provided that the armband will fit in his forearm. Currently it can be used for presentation in lectures to easily control slides, giving teachers the liberty to walk around the room, while being able to be in control of the presentation. However, a simple clicker can achieve the same objective, whilst being significantly cheaper and more intuitive.

After making the prototype, I have learnt that if I wanted to fully implement Myo in any program, the best option would be to learn C++ and fully study Myo SDK. Connectors can be made with LUA, a known script programming

Figure 4: Myo used during a presentation

language, or with Myo connect itself via the keyboard mapper option, exporting it as a connector. Myo effectively adopts EMG in its technology, giving it great potential for future use. Currently, however, its functions are still limited by the lack of a wide range of user friendly applications/programs specific to Myo.

