Myo Mapper: a Myo armband to OSC mapper

Balandino Di Donato
Embodied Audiovisual Interaction Unit
Goldsmiths, University of London
SE14 6NW, London, UK
info@balandinodidonato.com

Jamie Bullock
Independent Researcher
jamie@jamiebullock.com

Atau Tanaka
Embodied Audiovisual Interaction Unit
Goldsmiths, University of London
SE14 6NW, London, UK
a.tanaka@gold.ac.uk

ABSTRACT

Myo Mapper is a free and open source cross-platform application to map data from the gestural device Myo armband into Open Sound Control (OSC) messages. It provides an easy to use tool for musicians to explore the Myo's potential for creating new gesture-based musical interfaces. Together with details of the software, this paper reports on projects realised with the Myo Mapper as well as a qualitative evaluation. We propose guidelines for using Myo data in interactive artworks based on insight gained from the works described and the evaluation. We show that Myo Mapper empowers artists and non-skilled developers to easily take advantage of raw data from the Myo data and work with high-level signal features for the realisation of interactive artistic and musical works. Myo Mapper: 1) Solves an IMU drift problem to allow multimodal interaction; 2) Facilitates an clear workflow for novice users; 3) Includes feature extraction of useful EMG features; and 4) Connects to popular machine learning software for bespoke gesture recognition.

Author Keywords

Myo armband, mapping, feature extraction, EMG, hand gestures recognition, interactive machine learning.

CCS Concepts

•**Human-centered computing** → **Gestural input;** •**Applied computing** → *Sound and music computing;* Performing arts;

1. INTRODUCTION

Over the last three decades, muscle sensing technology has been used as an interface for musical performance. Recently, it has been used for manipulating vocal sounds [16], for developing new concepts of multimodal expressive interactions [7], for mapping micro-interactions into audio parameters [15], for allowing dancers performing music [14], and exploring the physiological influence of external auditory stimuli [18].

Muscle activity can be analysed via electromyography. It enables us to monitor the electromyogram signal (EMG) originating from our somatic nervous system and transported to our muscles through efferent nerves [17] (Fig. 1).

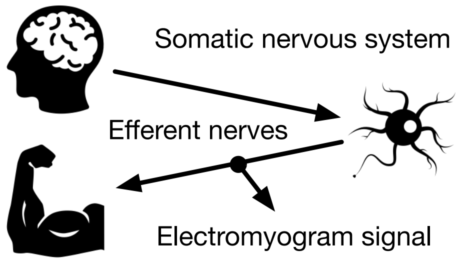

Figure 1: Electromyo biosignal flow.

With the commercialisation of EMG sensing technology and low-cost abrication, EMG-based interaction has become available to artists interested in creative works using physiological interaction. The Thalmic Labs Myo[1] is a multi-modal gestural input device that includes eight EMG sensors and an Inertial Measurement Unit (IMU) (gyroscope, accelerometer and a magnetometer). Factory applications provide us with a pose-recognition algorithm able to recognise five hand poses[2], and different applications to gesturally interact with different software such as Power Point, VLC, Spotify, Adobe Reader. The Myo has the potential of making biosignals accessible to artists; however, no easy-to-use application enables non-skilled developers to access raw physiological data.

We therefore designed and developed Myo Mapper (MM)[3] an application that enables users to extract and stream raw data to third-party audiovisual interactive software through a simple graphical user interface (GUI). Moreover, MM gives access to an easy Myo calibration process, data scaling, data feature extraction, and communication via OSC with machine learning (ML) software such as Gesture Recognition Toolkit (GRT) [13], ml.lib [6] or Wekinator [11].

This paper first explores related work. We give a detailed description of the software and its architecture. We then describe a number of creative projects in which MM has been utilised. We also report an informal qualitative evaluation. From these experiences, we derive a set of design guidelines for the use of EMG in musical applications.

2. RELATED WORK

The commercial and research communities have released tools to facilitate the use of the Myo in musical applications and the data mapping of Myo data into Open Sound Control (OSC) and MIDI messages. One example is Leviathan[4],

[1] `https://www.myo.com/`
[2] `https://support.getmyo.com/hc/en-us/articles/202647853`
[3] `www.balandinodidonato.com/myomapper/`
[4] `http://precisionmusic.technology/`

NIME'18, June 3-6, 2018, Blacksburg, Virginia, USA.

an application to control chords and effects parameters in Digital Audio Workstations (DAWs) using the Myo's factory poses. Myo-Ableton[5], is a "connector" (an extension) to the Myo daemon to maps factory poses data to control the popular Ableton Live music software. MyOSC[6] and myo-osc[7], are solutions to map Myo raw data to OSC messages. In addition to making raw data available, Myo-maxpd[8] and Francoise's Myo for Max[9] are externals for Max giving access to connection settings and haptic feedback. Although these tools interface the Myo with music recording and programming environments, they do not represent turn-key solutions for musicians not experts in interactive technology [4]. Specifically, they require additional software for extracting high-level data features for data mapping and the use of machine learning for recognising non-factory hand poses and gestures.

The Myo armband provides EMG and IMU data, enabling musicians to work with multimodal interaction [23] and use data relative to the arm's orientation coupled with isometric and isotonic muscle activity to generate, control and transform sound. Making use of the orientation, however, can be difficult as the yaw value drifts. We will show in the next section how we solve the problem of drift to allow Myo Mapper to be a robust interface for multimodal musical interaction.

3. ARCHITECTURE AND IMPLEMENTATION

Myo Mapper is a cross-platform application developed in C++ using the JUCE framework[10] and Myo SDK[11].

The software architecture is comprised of five main blocks: Myo communications, feature extractors, the OSC ports, shared spaces for storing application settings and a separate space for storing sensor data and extracted features. The GUI is made of three different windows: 'Settings', 'Calibrating and Scaling' and 'Feature Selection'.

The Myo SDK allows the application to communicate with the Myo hardware through bindings to the `libmyo` C library. The entry point to the SDK is the Myo Connect application which functions as a 'hub', managing the connection between the computer and one or more Myos (Fig. 2a). The SDK provides access to accelerometer, gyroscope, orientation, and EMG data from the device and control over its vibrational motors (Fig. 2b).

To facilitate the Myo data mapping to audiovisual authoring environments and interactive machine learning software, Myo Mapper includes different feature extractors (Fig. 2c), that can be selected in the Feature Selection window (Fig. 2e). The GUI also allows users to set the OSC communication between MM and third-party applications and includes tools to visualise orientation data. These configuration settings (OSC ports, features, calibration and scaling parameters) are stored in a shared space to facilitate the communication between the GUI and the back-end (Fig. 2d). MM includes an OSC receiving port, to which the user can send OSC messages to control calibration and scaling features (Fig. 2g).

Gestural data is sent out through two independent OSC ports, *main* and *ml* (Fig. 2f). The *main* port is used to send Myo data to the main musical application able to receive OSC messages. While the *ml* port is dedicated to sending OSC on a side chain to machine learning software. This second port was implemented to keep the end-user from having to use additional software to organise data features into a single feature vector to stream to the interactive machine learning subsystem.

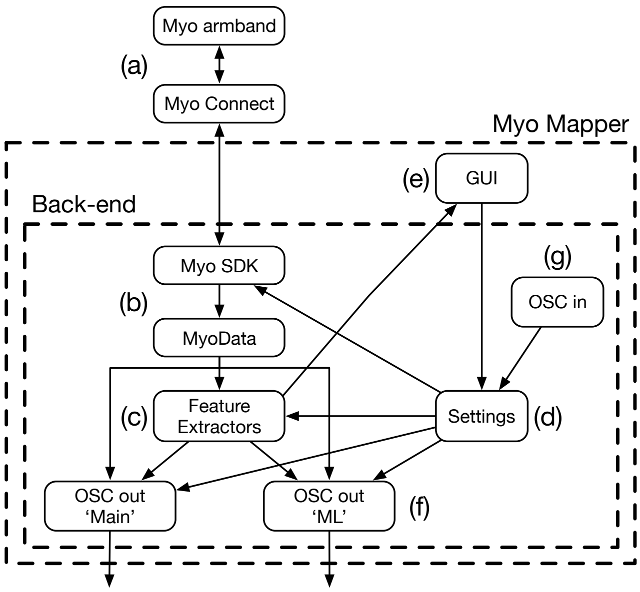

Figure 2: Myo Mapper's architecture.

3.1 Feature extractors

Following the evaluation of EMG features presented in [2], we implemented the mean absolute value (MAV) feature. Works related to hand gesture-recognition through electromyographic analysis [21, 3], reported the importance of the zero-crossing rate (ZCR) feature to observe spectral qualities of the EMG signal. Thus, we implemented it using equations from [2]. We implemented additional data features: minimum (MIN), maximum (MAX), absolute value (ABS), moving average (MAVG), first and second order difference (FOD, SOD) for use in interactive machine learning based on Fiebrink's work with Wekinator and Wekinator Input Helper[12].

The MIN and MAX features were implemented for analysing the range of values through which a gesture or pose is represented. The ABS is useful when observing the EMG data that might be the negative component of a mirror image signal's positive component. FOD and SOD are useful for analysing the input data variation over time; for instance, the orientation FOD and SOD inform us of our arm gestures' velocity and acceleration. The MAVG can function as a filter to separate EMG data from background noise.

The use of feature extraction is useful to preprocess sensor data for pose or gesture classification. For instance, the moving average of EMG absolute values over a window of 40 data points (Fig. 3, red line) facilitates distinguishing the arm resting (Fig. 3, pink area) from when performing a fist pose (Fig. 3, green area) by testing if the input value is greater than a given threshold (0.05). With this approach, poses might be misclassified (9 times as seen long the red MAVG curve in Fig. 3) times as a consequence of the time needed to calculate the data feature. However, as EMG data are streamed at a frequency of 200Hz, this

[5]`https://github.com/GonzaloNV/Myo-Ableton`
[6]`https://github.com/benkuper/MyOSC`
[7]`https://github.com/samyk/myo-osc`
[8]`https://github.com/bcaramiaux/Myo-maxpd`
[9]`https://github.com/JulesFrancoise/myo-for-max`
[10]`https://juce.com/`
[11]`https://developer.thalmic.com/`
[12]`http://www.wekinator.org/input-helper/`

classification latency is minimised. Fig. 3 also shows that if we were to use the absolute values of EMG RAW data (Fig. 3, blue line), our algorithm would have misclassified the pose 19 times. An even less accurate result would have been obtained if using EMG raw data. Implementations of the above algorithm in Pure Data and Max are available in Section 8.

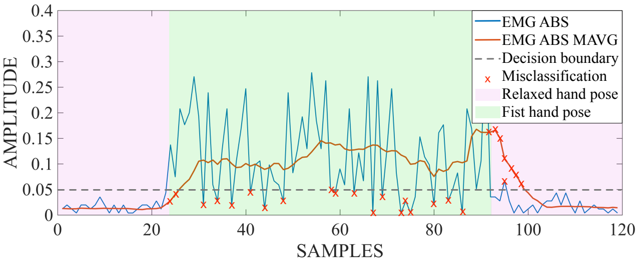

Figure 3: Comparison of EMG ABS and EMG ABS MAVG feature data for hand-pose recognition.

3.2 Orientation data scaling

Orientation data from the Myo is reported in a range $[-2\pi, 2\pi]$. Myo Mapper scales this data to the range $[0, 1]$ to ease their linear mapping into audiovisual processing parameters. However, the yaw value drifts 3.7 deg/s before reaching a stable value; with similar drift in the roll value [20]. In a real world situation, the data drift might be misinterpreted as a movement of the arm by a machine learning algorithm, which output triggers audio events every time that a movement occurs. To solve the orientation data drift issue, we included a *set origin* functionality which sets the current orientation data (yaw, pitch or roll) to a value of 0.5.

During tests, it emerged that the orientation data variation depends on the way the Myo is worn. (Flipping the bracelet on the arm changes which EMG channels correspond to which muscle groups, and inverts the IMU). Without a software solution, users would have to take off the Myo and turn it around. To avoid the user having to do this, we implemented a *flip* function, $y = 1 - x$, on yaw, pitch and roll. We also observed that in some cases, arm movements produce a too small a variation in the data to control audio parameters. To address this, we implemented range functions to limit the values in input (*in min*, *in max*) and rescale the values in output (*out min*, *out max*).

3.3 OSC message streaming

The *main* port is to send each selected feature in a unique OSC message. While the *ml* port is to send OSC messages to outboard machine learning software. When a feature, to be sent through the *ml* port, is selected in the Feature Selection window (Section 3.6), MM adds the selected feature's data to an OSC message with tag `/myoX`, where `X` is the number of the selected Myo (i.e. `/myo1` for the Myo number one). The data embedded in the messages are organised in the same order as they are selected. For instance, if streaming the raw orientation data and the EMG raw data, the OSC message will contain 11 floating point values (i.e. `/myo1 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.9 0.11 0.12`), where the first three are the RAW orientation data and the last eight the raw EMG data.

An OSC receiver port can be set through the Settings window (see Section 3.4). Through this last port, MM can receive messages, allowing the user to remote configure the set-origin, *flip*, *in min*, *in max*, *out min* and *out max* settings. All OSC messages' tag and type are specified in the MM Wiki page[13] and through a tooltip display.

3.4 Settings window

The Settings window includes controls to set the OSC communication (port number and IP address). In case more than one Myo is connected to the Myo Connect, here it is possible to select which of the Myos' data series have to be streamed.

3.5 Calibrating and Scaling window

The Calibration and Scaling window (Fig. 4) embeds controls to recall the *set origin*, *flip*, *in min*, *out min* and *out max* functions. When dragging the cursor over each button, a tooltip message containing instructions for having control of the button via OSC will appear.

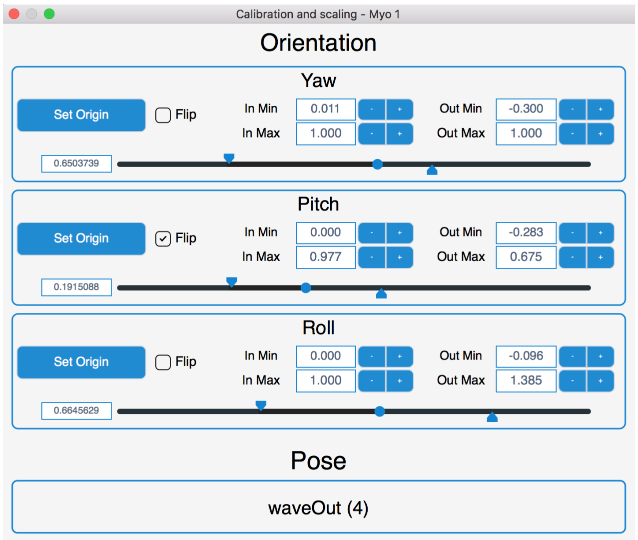

Figure 4: Myo Mapper's Calibration and scaling window.

3.6 Feature Selection window

The Feature Selection window (Fig. 5) enables the user to select one or more features to stream through the *main* and *ml* OSC ports, respectively, through the *To Main* and *To ML* toggles. The organisation of the features in a tree structure represents the data processing chain of each feature. For instance, the moving average of the raw EMG crossing rate can be streamed by selecting the fourth feature from the top in the EMG panel (Fig. 5). To facilitate the comprehension of the data processing chain, a tooltip shows the data chain processing. In the above example, the tooltip would contain the text: `EMG raw -> zero crossing rate`. Similarly, information relative to the OSC message (tag, number of values and type, sender's port number and IP address) of each feature are shown in a tooltip upon gliding the cursor over a feature (Fig. 5).

4. CASE STUDIES

Myo Mapper has been used for realising interactive music and dance performances, Virtual Reality (VR), robotics applications and for gesture recognition. It has been downloaded more than 2,300 times, and received the JUCE Award 2017. We are aware of 16 interactive projects built using Myo Mapper, and report on some of them here.

[13] `https://github.com/balandinodidonato/MyoMapper`

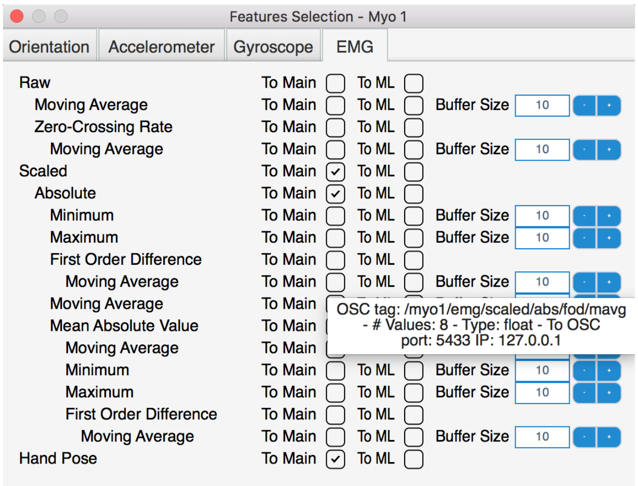

Figure 5: Myo Mapper's Feature window.

4.1 Musical performances

In *Fantasie pour violoncelle* by Gonzalo Villegas Curulla (2017)[14], *X/Centris Delirium Machine: Les bruits de l'esprit* by Lagacè and Dagher (2017)[15] and *Haptic Vertex* by Michailidis (2017), the instrumental and vocal sound produced by the performers were transformed by mapping Myo data.

An early version of MyoSpat, a gesturally controlled interactive system for manipulating audiovisual effects using the Myo armband and Myo Mapper [9], was used in Grace Savage's *Music Gesture Beatbox* (2016)[16] at Music Tech Fest 2016 in Berlin. MyoSpat was later used also by Eleanor Turner for performing *The wood and the water* (2017)[17] at Shanghai Symphony Hall for the Electronic Music Week 2017 and *Star Cluser* (2016) by Kirsty Devaney[18]. In these last two performances, MM facilitated the classification of arm poses and gestures that are different from those recognised by the device's factory default. Specifically, three different arm poses (frontwards, outwards, downwards) and two different gestures (plucking and throwing) were classified using the ml.lib Support Vector Machine (SVM) classifier [9]. The orientation scaling functions were very important in the classification success for those works. Specifically, when performing in different spaces, MM allowed the adjustment of the gestural data mapping into audiovisual processing parameters to obtain the optimal audiovisual result in relation to the venue's sound and video projection system set-up without making changes to the composition.

In [8] MM has been used for classifying gestural data to trigger different panning settings. Here the authors used MM and ml.lib for training a SVM classifier system. The ML algorithm was fed using moving average of the of the EMG data's absolute value over 50 samples. During the mapping phase, the feature extractor buffer size was reduced to 10 samples to minimise the classification latency.

4.2 Dance performance

MM has been adopted for realising interactive dance performances. In Sonia Sabri's *Nu Body* (2017), orientation data from the Myo's IMU were mapped to (i) parameters for controlling the virtual sound source's position of a spatialiser, (ii) lighting colours changing cues and (iii) video processing parameters. Gyroscope ABS data were used to relate the sound intensity of audio events and the light's brightness with arm movement acceleration. A Neural-Network Multilayer Perceptron (MLP) classifier was implemented using Wekinator for the mapping process and Pd as the audiovisual engine[19]. Michailidis and Di Donato (2016) presented an interactive system that aimed to translate a performer's gestures into haptic feedback sensed by a different performer using vibrational motors. This system aimed to empower dancers to communicate and inform each other of their artistic intentions during performance[20] [19]. A later work based on the same system enabled a dancer to communicate to pianists the music to be played[21]. In both applications, yaw and pitch's FOD and EMG ABS data of the Myo worn by a performer were mapped to parameters for controlling the intensity of vibrational motors of other Myos worn by the second performer. At TaikaBox's DigiDance workshop[22], dancers controlled sound and video projections. MM was used to stream yaw, pitch and EMG ABS data to Isadora[23] and Live[24].

4.3 Virtual Reality

MM has been used to conduct experiments in VR and Mixed Reality. Specifically, it has been used for the realisation of System 2 [5], which spatialised sound files represented as objects through an avatar using gestural control[25]. The avatar position on the screen was driven by direct mapping of yaw, pitch, and roll values. Myo factory poses (fist, finger spread, wave in and wave out) triggered different System 2 functionalities. In [8], direct mapping of yaw, pitch and EMG ABS data were used to construct an audio-based mixed reality system where the user can crumple a non-existent piece of paper through a natural hand movement and then 'throw' it into an imaginary 'cave' represented by a bin.

4.4 Robotics

In two different applications, MM has been deployed for controlling digitally controlled motors. At K-Array's laboratories, MM has been used to map Myo data into parameters for controlling orientation and the audio signal amplitude of the KW8 (aka Owl) moving head loudspeaker[26]. A different approach is taken in TaikaBox's work which looks at ways to create choreographies for humans and robots, MM has been used for controlling a robotic arm with a light mounted on its extremity[27]. In both works, robotic components were driven by mapping and converting yaw, pitch and roll into signals for controlling the KW8 and a robotic light.

5. EVALUATION

We collected user feedback from Myo Mapper's web presence on: our own developer website, GitHub, and SourceForge. We also carried out a workshop at Goldsmiths,

[14]https://www.youtube.com/watch?v=EioMZD9LbF0
[15]https://youtu.be/R90_UCJOVcQ
[16]http://bit.ly/musicGestureBeatbox
[17]http://bit.ly/TheWoodAndTheWater
[18]https://youtu.be/9ToP33Ki2SE
[19]http://bit.ly/MyoMapperDance
[20]https://youtu.be/n1x0fVHA2iw
[21]https://youtu.be/oxxiF0y7hFY
[22]https://youtu.be/EuQZSNm6Ut4
[23]https://troikatronix.com/
[24]https://www.ableton.com/en/live/
[25]https://vimeo.com/174099457
[26]https://vimeo.com/131770240
[27]https://youtu.be/Un2rP4ZyYNM

University of London as a means to conduct an informal evaluation of Myo Mapper.

5.1 Online feedback

Feedback collected from online users reported different issues in installing and using early iterations of the software. For this reason, the JUCE framework was used to guarantee the software portability across different operating systems. Three users requested the possibility to extract data from multiple Myos. Although this is not supported through the GUI, it is possible to do so by launching a second MM instance and selecting a different Myo in the Settings Window. This method was successfully implemented in Lagacè's musical work and Sabri's dance performance. John Collingswood, TaikaBox director, commented:

> *'This is amazing. I've been looking for this exact thing for a couple of years. I use Isadora to create interactive environments for dancers and other performers, and it [MM] instantly started kicking out useful and stable OSC that I can monitor in Isadora.'*

5.2 Workshop

Six students and two researchers from the Goldsmiths' Computing, Music, and Psychology departments took part in a half-day workshop. All participants were aware of the Myo, and 50% of them had used it prior the workshop. Problematic aspects of the GUI were highlighted. The organisation of feature labels in the Features Selection window was not clear for some participants. Most of the participants asked different times about the OSC message's tag, type and number of values of each feature. It was based on this feedback that the GUI tooltip was implemented.

The feature extractors were shown to be easy to use for all participants. In particular, they found the moving average of EMG absolute value very useful to filter out background noise. Through this feature, an undergraduate music student (guitarist) built a Neural-Network Multilayer Perceptron (MLP) classifier to recognise three different plucking gestures, using Myo Mapper in conjunction with Wekinator fed with the moving average of EMG absolute value and the absolute value of gyroscope data. In 15 minutes the student obtained a model outcome that achieved 90% accuracy in direct evaluation[12]. A second participant (Computing PhD student) aimed to recognise sign language gestures using MM and Wekinator. However, due to the complexity and number of gestures she sought to classify, MM did not allow the participant to quickly built a robust machine learning model. At the end of the workshop, participants commented on the software as being very useful and requested the implementation of additional features such as Root Mean Square (RMS) and Bayesian filters.

6. INTERACTION DESIGN GUIDELINES

After reviewing the works realised using Myo Mapper and considering the outcomes of the qualitative evaluation, we propose the following guidelines to aid musicians and artists in using EMG and IMU data from the Myo for creative interactive projects.

We observed that having all data within the same range $[0, 1]$ facilitates user workflow. Scaled data within the same range also helps to build more robust machine learning models. The MAVG feature is useful for filtering background noise from EMG data. Raw EMG is a noisy, stochastic, information and artefact rich signal, and effective use of feature extraction is fundamental in the use of EMG and machine learning.

In Section 4.1, we observed that by training a model using EMG data filtered with the MAVG function, the machine learning algorithm's output result was more accurate. After having built such model, the machine learning algorithm responsiveness could be optimized if MAVG analysis buffer size in the 'mapping' or 'performance' phase is lower than in the 'training' phase.

MIN and MAX features were shown to be useful to acknowledge the data range in which a pose or gesture occurs, to then adjust the data mapping accordingly. Direct mapping of orientation data resulted in useful ways to control position of virtual and robotics objects (Sections 4.3 and 4.4).

We also had interesting findings from [19], where the electromyography's lack of possibility to monitor involuntary movements can be exploited as an element of creativity in dance performance. In particular, dancers can control an interactive system differently when they are interacting or guiding each other's bodies, for instance, when a dancer moves the another dancer's arm. In Section 4.1, orientation data scaling controls gave the possibility to adapt the mapping strategy to the audio and lighting system. The same controls could also be beneficial in adjusting the data mapping process during live performance.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we presented Myo Mapper, a free and open source cross-platform application to process and map EMG and IMU data from the Myo device into OSC messages for creative applications that use machine learning. We gave descriptions of creative projects where it has been successfully used. We reported on online and workshop feedback that aided in design improvements of the software. We distil the insights taken from these projects and feedback in the form of design guidelines for effectively using the Myo Mapper to manage EMG and IMU data from the Myo in conjunction with machine learning software like Wekinator.

Myo Mapper has been used in different applications and is recognised by the research community. Works cited in Section 4 could have been realised using other software such as Max for Myo, Myo-maxpd or MyOSC, but they would have required the artist to implement their own data feature extractors, calibration and scaling functions and communications to interactive machine learning software. For instance, to implement a zero-crossing rate feature extraction algorithm that considers the background noise in the EMG signal using Pure Data or Max, would require a longer time than using Myo mapper. A comparison of the implementation of these two solutions with the use of Myo Mapper in the same environments is available in Section 8.

Future development of Myo Mapper will include GUI improvements, including native support for multiple Myos, and a more intuitive organisation of items in the Features Selection window. Future MM releases will implement additional feature extractors such as EMG RMS [10], Bayesian filter [22] and EMG Maximum Voluntary Contraction (MVC) [1]. Improvements will include MIDI and MPE (MIDI Polyphonic Expression) output, allowing the streaming of gestural data to non-OSC audio applications that accept only MIDI messages for external control. Future application areas, we have begun to explore, include music therapy and the use of interactive sonification in physical therapy and rehabilitation.

8. EXAMPLES

Examples mentioned in Sections 3.1 and 7 can be found at: `https://github.com/balandinodidonato/MyoMapper/tree/master/examples`.

9. ACKNOWLEDGMENTS

We would like to thank Jefferson Bledsoe (Integra Lab, Royal Birmingham Conservatoire) for his work as developer on Myo Mapper v3. Part of the research leading to these results has received funding from the European Union's Horizon2020 programme, H2020-ICT-2014-1 Project ID 644862, *Rapid-Mix*.

10. REFERENCES

[1] S. Al-Qaisi and F. Aghazadeh. Electromyography Analysis: Comparison of Maximum Voluntary Contraction Methods for Anterior Deltoid and Trapezius Muscles. *Procedia Manufacturing*, 3(C):4578–4583, July 2015.

[2] Z. Arief, I. A. Sulistijono, and R. A. Ardiansyah. Comparison of five time series EMG features extractions using Myo Armband. In *Proceedings of the 2015 International Electronics Symposium (IES)*, pages 11–14, Surabaya, Indonesia, Sept. 2015. IEEE.

[3] A. Boyali and N. Hashimoto. Spectral Collaborative Representation based Classification for hand gestures recognition on electromyography signals. *Biomedical Signal Processing and Control*, 24(C):11–18, Feb. 2016.

[4] J. Bullock and L. Coccioli. Towards a humane graphical user interface for live electronic music. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, NIME '09, pages 266–267, Pittsburgh, PA, United States, 2009.

[5] J. Bullock and B. Di Donato. Approaches to visualising the spatial position of 'sound-objects'. In *Proceedings of the Conference on Electronic Visualisation and the Arts*, EVA '16, pages 15–22, London, United Kingdom, July 2016.

[6] J. Bullock and A. Momeni. ml.lib: Robust, cross-platform, open-source machine learning for max and pure data. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, NIME '15, pages 265–270, Baton Rouge, Louisiana, USA, June 2015.

[7] B. Caramiaux, M. Donnarumma, and A. Tanaka. Understanding gesture expressivity through muscle sensing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(6):31:1–31:26, Jan. 2015.

[8] B. Di Donato and J. Bullock. gSpat: Live sound spatialisation using gestural control. In *Proceedings of the 21st International Conference on Auditory Display Student ThinkThank (ICAD 2015 STT)*, pages 7–8, Graz, Austria, July 2015.

[9] B. Di Donato and J. Dooley. MyoSpat: A system for manipulating sound and light through hand gestures. In *Proceedings of Workshop on Intelligent Music Production (WIMP)*, Salford, United Kingdom, Sept. 2017.

[10] G. Dubost and A. Tanaka. A Wireless, Network-based Biosensor Interface for Music. In *Proceedings of the International Computer Music Conference*, ICMC, pages 92–98, Göteborg, Sweden, Mar. 2002.

[11] R. Fiebrink. Wekinator. `http://www.wekinator.org/`, 2009. Accessed: 24/01/2017.

[12] R. Fiebrink, P. R. Cook, and D. Trueman. Human model evaluation in interactive supervised learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 147–156, New York, NY, USA, 2011. ACM.

[13] N. Gillian and J. A. Paradiso. Gesture recognition toolkit (grt). `http://www.nickgillian.com/grt`, 2016. Accessed: 8/01/2018.

[14] J. Jaimovich. Emovere: Designing Sound Interactions for Biosignals and Dancers. In *Proceedings of the Interantional Coference on New Interfaces for Musical Expression*, NIME '16, pages 316–320, Brisbane, Australia, July 2016.

[15] A. R. Jensenius, V. G. Sanchez, A. Zelechowska, and K. A. V. Bjerkestrand. Exploring the Myo Controller for Sonic Microinteraction. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, NIME '17, pages 442–445, Copenhagen, Denmark, June 2017.

[16] E. Jessop. The Vocal Augmentation and Manipulation Prosthesis (VAMP): A Conducting-Based Gestural Controller for Vocal Performance. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, NIME '09, pages 256–259, Pittsburgh, PA, June 2009.

[17] E. Kaniusas. *Biomedical Signals and Sensors I: Linking Physiological Phenomena and Biosignals.* Biological and Medical Physics, Biomedical Engineering. Springer-Verlag Berlin Heidelberg, Berlin, Germany, 2012.

[18] S. Kara and P. Özel. Determination of Effects of Different Music on Frontal Muscle by Using EMG Signal. In *2007 Frontiers in the Convergence of Bioscience and Information Technologies*, pages 449–452, Jeju City, South Korea, Aug. 2007. IEEE.

[19] T. Michailidis and B. Di Donato. Vibrotactile feedback for interaction in dance performances. In *Abstracts of the InDialogue Symposium*, Nottingham, United Kingdom, 12 2016.

[20] K. Nymoen, M. R. Haugen, and A. R. Jensenius. Mumyo - evaluating and exploring the myo armband for musical interaction. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, NIME '15, pages 215–218, Baton Rouge, Louisiana, USA, 2015.

[21] S. M. M. Roomi, R. J. Priya, and H. Jayalakshmi. Hand Gesture Recognition for Human-Computer Interaction. *Journal of Computer Science*, 9(6):1002–1007, Aug. 2010.

[22] T. D. Sanger. Bayesian Filtering of Myoelectric Signals. *Journal of Neurophysiology*, 97:1839–1845, Feb. 2007.

[23] A. Tanaka and R. B. Knapp. Multimodal Interaction in Music Using the Electromyogram and Relative Position Sensing. In *A NIME Reader*, pages 45–58. Springer, 2016.