# Semi-Automated Mappings for Object-Manipulating Gestural Control of Electronic Music

Virginia de las Pozas
NYU Steinhardt School of Culture,
Education, and Human Development
82 Washington Square E,
New York, NY 10003
vdelaspozas@gmail.com

## ABSTRACT

This paper describes a system for automating the generation of mapping schemes between human interaction with extramusical objects and electronic dance music. These mappings are determined through the comparison of sensor input to a synthesized matrix of sequenced audio. The goal of the system is to facilitate live performances that feature quotidian objects in the place of traditional musical instruments. The practical and artistic applications of musical control with quotidian objects is discussed. The associated object-manipulating gesture vocabularies are mapped to musical output so that the objects themselves may be perceived as DMIs. This strategy is used in a performance to explore the liveness qualities of the system.

## Author Keywords

digital musical instrument, mapping strategies, gestural control of music, gesture vocabularies

## CCS Concepts

• **Applied computing → Sound and music computing**; Performing arts;

## 1. INTRODUCTION

### 1.1 Background

Participation in the creation and development of digital musical instruments (DMI) has surged in recent years due to an increase in computing power and the availability of real-time signal-processing applications [12]. These new capabilities have altered notions of what can be considered a musical instrument, as the totality of digital sound generation opens up new possibilities for controlling music [1].

The anatomy of a DMI can be described as the combination of a controller or interface, sound generation mechanism(s), and, between the two: a mapping strategy. As long as a sensor array is capable of adequately capturing a motion, that gesture may be mapped to a defined musical event. Usually, the gesture-sound relationships which comprise a mapping strategy are arbitrarily assigned by the instrument designer based on their aesthetic values or practical needs [12]. The collection of these actions is referred to in the literature as a gesture vocabulary, wherein each gesture word is mapped to the control of one or many musical parameters. Many DMIs have been created which incorporate extramusical objects and actions as control mechanisms. A unique mapping may even be created for a specific performance or composition [4]. Mappings employ varying levels of control, determinacy, expressivity, and complexity [12].

Usually, the gesture-sound relationships which comprise a mapping strategy are arbitrarily assigned by the instrument designer based on their aesthetic values or practical needs. In many cases, it is not desirable for a mapping to be “fully baked.” A DMIs control space may change; inconstancies in the human performer, faulty capture by the sensor array, or changes in artistic context can benefit from mapping strategies with adaptive capabilities. Given the tools available, it should be possible to automate the assignment of control space components to musical parameters. This would make the mapping process a viable task during rehearsal, for example. Rather than choosing arbitrary motions to form the vocabulary and designating map-points, this project aims to automatically determine a best-candidate mapping that changes dynamically throughout the performed activity according to the relationships between each gesture word in the variable control space.

### 1.2 Goals

The work described below is a prototypical system for the automation of mapping schemes in the context of an individual compositional practice. In this approach, repetitive, object-manipulating hand motions generate the data that determines output. The scope of musical output was limited to a style of experimental electronic dance music characterized by repetitive sequences and sparse instrumentation, making it a good candidate for testing the reproducibility and accessibility of the resulting mappings. Such a system would allow the composer to experiment with different gesture vocabularies and combinations thereof during the creative process, rather than devoting time and resources to the assignment of every action-sound relationship in the mapping scheme. Extramusical skills might be exploited as performative activities, and the nuances of sonifying those gesture vocabularies explored. The automated mapping of a quotidian gesture vocabulary has the potential to benefit artistic and compositional practices by expanding the conceptual implications of instrumental performance, visual themes in instrumentality, and enabling inclusive musical collaborations.

The goal of this project was to develop a MAX/MSP performance environment for the automated mapping of repetitious sensor input to MIDI messages, to be used in the control of loop-based electronic music [5][8]. The input scope was limited to object-manipulating gestures; as a basis for composition and performance but also to promote predictable interpretations of the sensor data. By recording and mapping the primary cycle of interaction with an object, that object may be perceived as a DMI. A successful DMI is defined in the literature as one possessing accessible, reproducible, and expressive gesture-sound relationships [12]. At least some of these relationships should be simple enough to grasp on first use yet

*NIME'20,* July 21-25, 2020, Royal Birmingham Conservatoire, Birmingham City University, Birmingham, United Kingdom.

complex enough to for the performer to develop skill and individuality [11]. Embodied instrumental practices increase the feeling of control and expression for mapping schema having more than a handful of linear relationships and highlight the unique characteristics of the performer's movement [1][9]. Furthermore, a system for the control of music should have a certain degree of indeterminacy [3]. These criteria guided the design of the decision-making process for the mapping scheme and the modes of control used in the patch.

By constraining the scope of gestural input to repetitive interaction with physical artifacts, the movement of the performer is limited by the shape and size of the artifact as well as the time it takes to complete one cycle of interaction. This made the task of sonifying a gestural event more predictable, and the resulting music more likely to be perceived as being generated by that movement. Although the sensor array measures only the performer's movements, not the artifact itself, given the project constraints the system can be understood as an augmented extramusical instrument controller. The gesture vocabulary is limited to the scope of movement in the object-manipulating gesture that is performed within a musical phrase.

## 2. METHOD

### 2.1 Sensors

To study the mapping procedure for various object-manipulating gestures, a sensor array was chosen which would be suitable for the capture of as many object-manipulating gestures as possible. An ideal sensor array would not impede nor be impeded by any fine motor movements involving objects. Early experimentation with the system incorporated a Leap Motion controller.[1] The Leap Motion captured movement reliably but required multiple sensors in the case that an object obfuscated any portion of the performers hands. The Myo armband was found to be a good alternative. Although it is prone to data fluctuations between uses, it has the benefit of not cutting into the performer's mental bandwidth [4].

The Myo armband is an off-the-shelf device that measures both bioelectric and inertial data. The armband contains 8 electromyographic (EMG) sensors as well as an inertial measurement unit (IMU) with 9 degrees of freedom [11]. The sensor array has been shown to be useful for the control of prosthetic devices. Although the sampling rate of 200 Hz is less than the medical standard, it meets the suggested rate for control of musical parameters [12].

**Figure 2. The Myo Armband**

IMUs and electromyography are most frequently used in open-air control systems and augmented instruments to drive continuous musical parameters [12]. This project utilized the Myo Armband and Myo for MAX external library to control loop-based electronic music, which commonly consists of a sequence of discrete events. Defining the conditions for generating those events was based on a comparison between the sensor input and a user-defined sequence.

### 2.2 Automated Mapping Performance Environment

Through a combination of pre-processing, feature extraction, and concepts from machine learning, a working prototype was created. The system requires that the user program a sequence of *n* instrument patterns, designate global musical variables, load examples of audio samples of instruments matching the chosen instrumentation and run an analysis on a matrix of the sequenced audio. A phrase-length *n*-dimensional stream of input from the sensor is then analyzed and grouped into *n* streams, matching the number of voices chosen for the piece. The grouped sensor inputs are again analyzed and assigned to the voice type which they are best suited to control based on comparison to the matrix of sequenced audio. Individual stream groups drive MIDI messages to their assigned voice/MIDI channel.

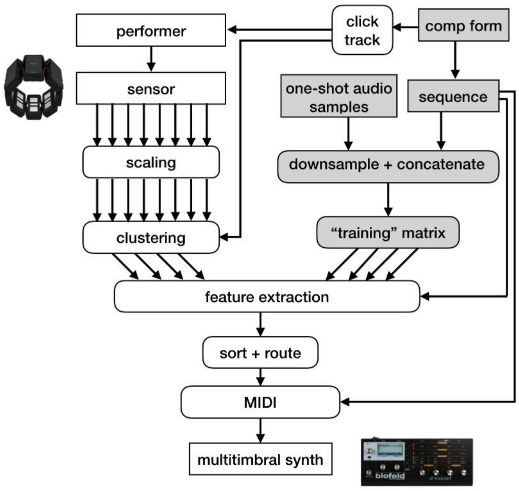
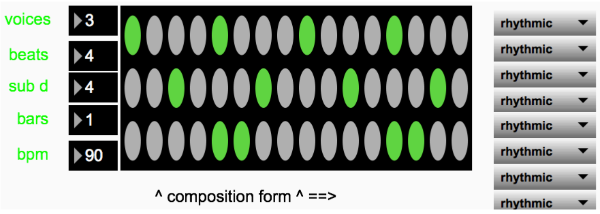


**Figure 1. Diagram of the patch functionality.**

Preparing for a composition requires that a number of settings be defined before operation. This is accomplished in the "composition form" module, which requires that the user program a sequence of desired output in addition to defining number of voices, beats per measure, rhythmic subdivision per beat, measures per phrase, and tempo. These values control the behavior of every module of the patch. The range of possible values is limited by available processing power.



voices 3
beats 4
sub d 4
bars 1
bpm 90

**Figure 1. The composition form.**

[1] https://www.ultraleap.com

## 2.3 Synthesized Training Data, Feature Extraction, and Voice Assignments

To automate the stream-to-voice determination of the mapping scheme, a phrase-length section of the cluster-streams is compared to a set of examples describing the ideal control data for that voice. These examples are generated by sequencing downsampled audio of the desired instrument according to the patterns in the composition form. Audio samples were collected from a multi-genre consumer sample pack of conventionally used one-shot drum machine samples. The resulting lists fill a Jitter matrix, one voice per plane, which is used for storage and visualization.

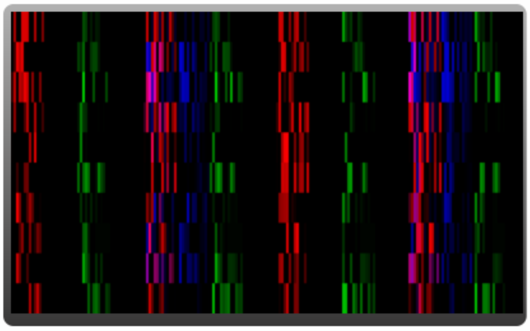


**Figure 3. A synthesized "training" matrix for 3 voices: kick, high hat, and snare.**

Before comparing a phrase of the input stream to the data matrix, feature representations of the individual patterns were created. Time-domain features were extracted from each row in each plane of the matrix, which contains a list of the downsampled sequence's amplitude values. These include mean absolute value, zero-crossing rate, maximum, and first-order difference [6]. In addition, the number of peaks is measured by the `[ml.peak]` object from ml.lib [2], and the number of onsets is measured by the same process used to drive note-on messages in the MIDI output module.

Spectral features were extracted through use of the zsa.descriptors external Max objects. These include Mel-Frequency Cepstral Coefficients from `[zsa.mfcc~]`, which describe the spectral envelope of the audio and are often used for instrument classification, as well as Spectral Centroid from `[zsa.centroid~]`, which provides information about the center of mass of a spectrum and is used as a measure of brightness [7]. These features were chosen for the current implementation because of the priority of differentiating between drum sounds.

Determining the best fit mapping for a specific cluster-stream is accomplished by comparison of the relationship between clusters to the relationship between the planes in the synthesized matrix. Early experiments were done using supervised classification models using K Decision Trees [10] and K Nearest Neighbors [2], but this relies on the assumption that each cluster will be best suited to a different voice type. It is desirable for the automated scheme to result in the same instrumentation as was intended by the composer when populating the composition sequence. Rather than classifying the clustered streams of sensor input, a hierarchical method was implemented in which feature sets were extracted from the "training" data and used to sort the voice labels in order of the mean of their values. The same extraction and sorting are then performed on the input streams. The resulting lists are then paired by order and sent to a routing module, which forwards the cluster-stream to its assigned voice.

It is assumed that the updated voice assignments are accessible because of the conventional patterns and instrumentation shared by the sub-genres of electronic dance music: 4 on the floor kick drums, off-beat high hats, etc. This accessibility is dependent on the sequencing of the composition form and polyphonic synthesizer timbres following those conventions.

## 2.5 MIDI Output

Once the composition form has been populated and sensor input has been clustered into the desired number of voices, MIDI output begins. MIDI information is transmitted to an external synthesizer as CC messages, which differ between hardware synthesizers. Changing the CC messages which correspond to synthesis parameters makes the Max patch adaptable to different external synths, as long as they are multi-timbral. For this implementation and experiment, the Waldorf Blofeld[2] was used, which is a 16-voice polyphonic synthesizer with internal effects. The timbres on each channel were set to match the voice types defined in the composition form.

For rhythmic voices, the input cluster-stream which was assigned to that voice is run through an onset detection algorithm[3] with an adaptive threshold calculated by a moving average over the last 20 samples. The value of the stream at that onset is paired with the note-on MIDI message and controls the velocity of that note. The minimum time between onsets and the resulting note-on messages is determined by the millisecond value of the smallest rhythmic subdivision, which is defined by the user in the composition form.

Melodic voices are controlled by two streams: the first controls note-on messages in the same way as a rhythmic voice and the second is used as an input to key prediction. Tonal scale is defined in the composition form, and the key is predicted by a Hidden Markov Model using `[ml.hmm]`. Examples of melodic sequences in the user-defined scale are used to build emission and transition matrices, the model is trained, and new values from the cluster stream are analyzed to update the predicted key. This is based on a key-prediction method found in the ml.* external Max package [10]. Currently, melodic output functions for a single voice.

# 3. DISCUSSION & CONCLUSION

## 3.1 Live Performance

A live performance was given midway through the development of this implementation. Through collaboration with an extra-musical instrumentalist, a piece was constructed around an AI-generated narrative.[4] During the performance, a speech synthesis recording of the generated text was played back in sections, and each section of the narrative cued the instrumentalist to interact with one of a set of physical props. To keep in time with the patch's internal clock, the instrumentalist wore Bluetooth headphones to which the metronome clicks were routed. Each time they began a new gesture a phrase-length section of sensor data was recorded for clustering. This process waits for the first beat of a new phrase to start recording, so that the cluster assignment is aligned with one gestural cycle.

It was clear from this experience that a sense of rhythm and musicality are crucial to the performer's ability to repeat musical output, and that practicing precise repetition of movements was necessary to achieve the desired result. The instrumentalist reported feeling as if they were a member of an orchestra being

[2] https://www.waldorfmusic.com/en/blofeld-overview

[3] adaptive threshold onset detection by Rodrigo Constanzo: https://cycling74.com/forums/fastest-onset-detection-native-or-external

[4] https://talktotransformer.com

conducted, and that although they did not feel completely in control of the output, the music did respond to their expressive modulations of the gestures.

Voice assignments and spectral features were not yet implemented at this time, but the experience of live performance proved that real-time clustering of sensor data was feasible in a live setting. General responses to the performance suggested an awareness that the performer's interaction with props had control over the music, as well as positive engagement with the visual elements of that control.

## 3.2 Reproducibility

The automated mappings were found to be reproducible through repeated use of the patch in testing and performance. However, musical output was not found to be reproducible or varied enough to be recognizable as being controlled by a familiar gesture. This is due in part to the current implementation using only many-to-one mappings and rhythmic voices. This may be improved upon by expanding the control space to include additional musical parameters.

# 6. CONCLUSIONS & FUTURE WORK

This prototype sets the foundation for continued research of automated best-candidate mapping strategies and provides a framework for the exploration of object-manipulating gestures in musical performance. The approach generates mappings with little to no effort from the user. However, all the generated mappings are comprised of one-to-many or one-to-one relationships. The system is playable but lacks expressivity, which is partially due to the simplicity of the mappings. Short examples can be viewed at the following links:

https://www.youtube.com/watch?v=wEds97EUWb0
https://www.youtube.com/watch?v=PzyiV6zgFJo

Improvement of best-candidate mapping would likely be achieved by the addition of phases and/or layers of input analysis, as well as the use of additional musical parameters. The author hopes to develop this approach alongside compositional experiments which serve to inform the work.

# 7. ACKNOWLEDGMENTS

This work was completed as part of a master's thesis in Music Technology at NYU Steinhardt. Thank you, Dr. Morwaread Farbood, Dr. Tae Hong Park, Brian McFee, and Dafna Naphtali for the invaluable advice and support.

# 8. REFERENCES

[1] P. Alperson. "The Instrumentality of Music." The Journal of Aesthetics and Art Criticism, Winter, 2008, 66:1.

[2] J. Bullock and A. Momeni. ml.lib: Robust, Cross-platform, Open-source Machine Learning for Max and Pure Data. 2015. ml.lib: Robust, Cross-platform, Open-source Machine Learning for Max and Pure Data. *Proceedings of the International Conference on New Interfaces for Musical Expression*, Louisiana State University, pp. 265–270.

[3] J. Chadabe. 24AD. The Limitations of Mapping as a Structural Descriptive in Electronic Instruments. *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 38–42.

[4] P.R. Cook. 1AD. Principles for Designing Computer Music Controllers. *Proceedings of the International Conference on New Interfaces for Musical Expression*, pp. 3–6.

[5] Cycling '74 (n.d.) Max/MSP. https://cycling74.com

[6] B. Di Donato, J. Bullock, and A. Tanaka. 2018. Myo Mapper: a Myo armband to OSC mapper. *Proceedings of the International Conference on New Interfaces for Musical Expression*, Virginia Tech, pp. 138–143.

[7] M. Malt and E. Jourdan. Zsa.Descriptors: A Library for Real-Time Descriptors Analysis. *Proceedings of the Sound and Music Computing Conference.* Berlin, 2008, pp. 134-137.

[8] MIDI Association (n.d.) MIDI Specifications. Retrieved from https://www.midi.org/specifications

[9] D. Overholt. "The Musical Interface Technology Design Space," Organised Sound. Cambridge University Press. pp. 217-226, 2009.

[10] B.D. Smith and G.E. Garnett. 2012. Unsupervised Play: Machine Learning Toolkit for Max. *Proceedings of the International Conference on New Interfaces for Musical Expression*, University of Michigan.

[11] A. Tanaka and B.R. Knapp. 2002: Multimodal Interaction in Music Using the Electromyogram and Relative Position Sensing. A NIME Reader, Current Research. Springer International Publishing, pp. 45-58, 2017.

[12] M. Wanderley and M. Depalle. Gestural Control of Sound Synthesis. *Proceedings of the Conference for the Institute of Electrical and Electronics Engineers*, 2004, Vol. 92, No. 4, pp. 632-644.