GRAB-AND-PLAY MAPPING: CREATIVE MACHINE LEARNING APPROACHES FOR MUSICAL INCLUSION AND EXPLORATION

Hugo Scurto
Department of Computing
Goldsmiths, University of London
h.scurto@gold.ac.uk

Rebecca Fiebrink
Department of Computing
Goldsmiths, University of London
r.fiebrink@gold.ac.uk

ABSTRACT

We present the first implementation of a new tool for prototyping digital musical instruments, which allows a user to literally grab a controller and turn it into a new, playable musical instrument almost instantaneously. The tool briefly observes a user interacting with a controller or sensors (without making any sound), and then it automatically generates a mapping from this observed input space to the control of an arbitrary sound synthesis program. The sound is then immediately manipulable using the controller, and this newly-created instrument thus invites the user to begin an embodied exploration of the newly-created relationships between human movement and sound. We hypothesize that this approach offers a useful alternative to both the creation of mappings by programming and to existing supervised learning approaches that create mappings from labeled training data. We have explored the potential value and trade-offs of this approach in two preliminary studies. In a workshop with disadvantaged young people who are unlikely to learn instrumental music, we observed advantages to the rapid adaptation afforded by this tool. In three interviews with computer musicians, we learned about how this "grab-and-play" interaction paradigm might fit into professional compositional practices.

1. INTRODUCTION

Historically, computer programming has been a core technique used in the creation of new digital musical instruments. The "mapping" [1] that specifies how a musician's movements (sensed using a controller or sensors) relate to sound (e.g., the values of sound synthesis parameters) is often created by writing programming code. While programming allows a mapping to be specified precisely, the process of translating an intended mapping function to code can be frustrating and time consuming [2], even for expert programmers, and it is inaccessible to non-programmers.

Machine learning has been used as an alternative mechanism for generating mappings since the early work of [3]. Most work that has employed machine learning for mapping creation has employed supervised learning algorithms, which can create a mapping from input sensor values to sound synthesis control parameters using a set of "labeled" training examples. In this labeled training set, each example consists of one vector of input sensor values, plus the "labels"—the vector of sound synthesis parameter values the designer would like to be produced in response to those sensor values. Research has suggested that supervised learning offers a useful alternative to programming, for instance by making mapping creation faster, by enabling designers to encode an embodied understanding of the desired gesture/sound relationships in the training examples, and by making mapping accessible to non-programmers [2, 4].

However, existing supervised learning approaches to mapping creation do not directly address some of the most fundamental needs of instrument designers. For instance, an instrument designer often does not know a priori precisely what type of mapping she wants in a new instrument. It is only by prototyping—experimenting with alternative designs in a hands-on way—that she can more fully understand the potential offered by a set of sensors and synthesis tools, and understand how she might fit these together into an instrument or a performance. An instrument designer who wants to explore many different prototypes using machine learning must still create many different sets of training data, and explicitly choose the type of relationship between sensors and sounds that should be embedded within each set.

New approaches to mapping generation might accelerate the discovery and realisation of new design ideas, by taking advantage of the computer's ability to generate mapping functions under different types of constraints or with different types of goals. This could be useful when the user does not have a specific relationship between movement and sound already in mind, or when other properties of the instrument (e.g., playability, comfort) supersede any preference for a particular sensor/sound relationship.

In this paper, we describe first steps toward exploration of such alternative mapping strategies. We have implemented a fully-functioning tool capable of generating many alternative mappings from a single set of unlabeled training examples, which encode the range of motion of a performer using arbitrary sensors/controllers. In our first version of the system, alternative mappings are generated from this single training set using a computationally straightforward approach to transform the unlabeled training set into multiple alternative labeled training sets, which can each be used to build a mapping using supervised learning. Many other computational approaches to generating multiple alternative mappings from unsupervised learning are also possible.

We have worked with two sets of users to evaluate this approach and better understand its potential use. These users include youth with disabilities and difficult life circumstances, as well as three professional computer music composers. This work suggests that the rapid adaptation afforded by this approach could benefit the first category of users, while the predisposition to musical exploration and discoveries could benefit the second category.

2. BACKGROUND AND PREVIOUS WORK

Machine learning algorithms have been widely employed in musical interaction, both as a means to analyze musical gestures and to design gesturally-controlled digital musical instruments (see [5] for an overview of the field).

Research by Fiebrink and collaborators has focused on understanding the impact of using machine learning (as opposed to programming) on the instrument design process [2], and on designing user interfaces to allow instrument builders to use machine learning effectively and efficiently, without requiring prior machine learning expertise [4]. Fiebrink's Wekinator [1] toolkit allows instrument builders to create supervised learning training sets by demonstrating performer gestures alongside the instrument sounds the designer would like to be produced by those gestures. The Wekinator uses general-purpose algorithms for regression (e.g., multilayer perceptron neural networks, linear and polynomial regression) and classification (e.g., nearest-neighbor, support vector machines) to create mappings from this data.

Other recent research has explored the development of new modeling approaches that are tailored to building gestural musical interactions [6, 7], notably allowing for similarity estimations between a gesture being performed and recorded references. Such approaches are particularly successful when the task is to recognize and track given gestures.

There is a growing interest among music researchers in the importance of bodily experience in sound perception and cognition [8]. According to this theory, it is primarily through the body that performers convey inner information about their artistic intentions and emotions; this bodily information is encoded into and transmitted by sound to listeners who can in turn attune to the performer's inner intent. It is important to underscore that such body movements, or gestures, are not necessarily pre-defined for the performer, and can appear to be metaphorical [9] rather than descriptive [10, 11]. In this sense, mapping approaches that value exploration rather than explicit definition could be relevant to facilitate the use of metaphorical gestures in performance.

3. GRAB-AND-PLAY MAPPING

3.1 Definition

We propose a new paradigm for mapping creation, called "grab-and-play mapping", that enables the very rapid creation of new instruments from a very small amount of data communicating some minimal, soft design constraints—namely, the way the user might want to move while playing

[1] www.wekinator.org

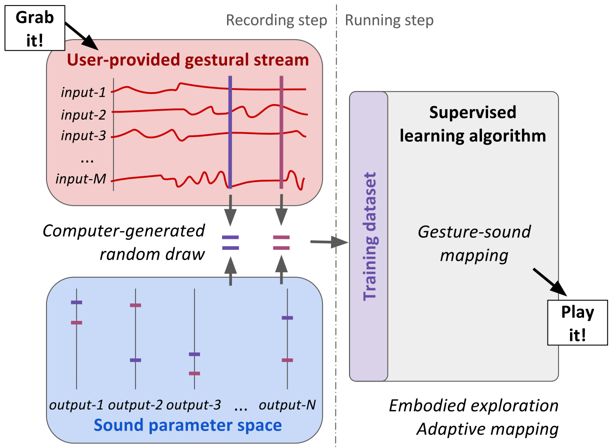

Figure 1. Our first implementation of the grab-and-play mapping paradigm. Inputs and outputs are respectively drawn from the user's recorded gestural stream and the sound parameter space. Outputs from 1 to N are sound synthesis parameters. In this schema, the training database contains two examples (i.e. two input-output pairs).

this new instrument. This minimal set of data allows the creation of mappings which are customised to a controller and/or to a player in a loose sense, by aiming for a mapping that is playable using whatever range of motion and types of variation are present in the examples provided by the designer. But this process does not require a designer to specify other information about the instrument, other than potentially the range of legal values for each sound synthesis parameter that will be controlled by the mapping. Our approach thus shifts the designer's focus from one of imagining and then implementing new gesture-sound relationships, to a focus on discovering new relationships that have been designed partly by the computer, and on embodied exploration of those relationships.

3.2 Implementation

Our vision of grab-and-play mapping could be implemented using a number of techniques for automatically generating a mapping. This paper reports on our first implementation, which is described in Figure 1.

In this implementation, the user must first demonstrate how she will physically interact with a musical controller; this results in a recorded, continuous stream of gestural input data. Next, the computer transforms this stream of unlabeled inputs into a labeled training set that can be passed to a supervised learning algorithm for mapping creation (e.g., a neural network). Specifically, a number of examples are chosen at random from the recorded inputs. Each example is assigned a randomly-generated value for each sound synthesis parameter. These random sound synthesis parameters could be chosen from user-selected "presets" (i.e., vectors of parameter values that, together, result in sounds the user might want to have present in the instruments). Or, each parameter could be randomly generated from a uniform distribution over the range of all legal parameter values (e.g., [0,1]).

Finally, this artificially-generated training set is fed into a supervised learning algorithm that builds a mapping function capable of computing a new sound synthesis parameter vector for any new control vector. The user can now play the newly-created instrument by interacting with the

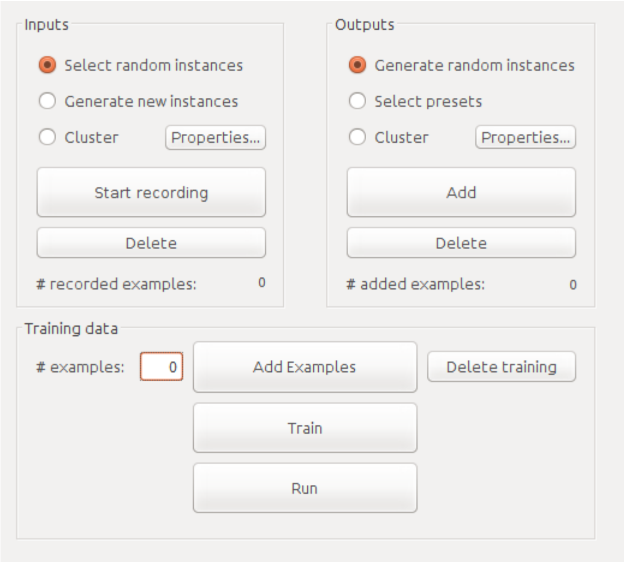

Figure 2. The current GUI of the tool. Observational studies reported in this paper only used the random implementation for input; random implementation and preset implementation for output. Other implementations of the grab-and-play approach are already implemented (see section 6), and will be studied in the near future.

input controller or sensors and discovering how the sound changes with her actions.

This new tool is implemented in Java as a branch of the Wekinator software. All code is available online [2]. The new tool adds the following additional functionality to Wekinator (see Figure 2):

- Grab-and-play mappings can be generated using the procedure above, requiring only that the user demonstrate a brief input sequence.
- New alternative mappings can be generated indefinitely from the same grab-and-play unlabeled training sequence.
- The user can interactively change the number of supervised training examples randomly generated from the grab-and-play training sequence.
- The user can switch between grab-and-play mapping and mappings generated using supervised learning.

The tool also takes advantage of the following existing capabilities of Wekinator:

- Any type of input controller or sensor system can be used to control sound, provided data about the input is sent as an OSC message [12].
- Any sound synthesis software can be used to play sound, provided it can receive synthesis parameter vectors as OSC messages.
- The GUI allows users to switch immediately and repeatedly between generating mappings and playing the generated instruments in real-time.
- The GUI allows users to easily change mappings by deleting and adding training examples.
- Advanced or curious users can customise aspects of the machine learning process, e.g., changing the learning algorithm or its parameters, changing the selected features, etc.
- Learning algorithms are set to default configurations that have been shown to work well for many mapping problems, so novice users never have to make an explicit choice of learning algorithm or algorithm parameters.

[2] http://github.com/hugoscurto/GrabAndPlayWeki

4. PRELIMINARY WORKSHOP WITH DISABLED YOUNG PEOPLE

We used this tool in a workshop with disabled young people to gain a preliminary understanding of how it might be useful for building new musical instruments for people with disabilities, and of how youth might respond to the customised yet unpredictable mappings built by this tool.

4.1 Using machine learning to build instruments with disabled people

Machine learning has been recently applied to build custom musical interfaces for disabled adults through several workshops [13]. Not only did the authors of that work find similarities between the musical goals and practices of disabled people and expert musicians, but they also noted some difficulties for participants to develop a memorable gestural vocabulary. In our workshops, we were therefore curious whether this grab-and-play approach might circumvent some user frustration, by explicitly inviting exploration of new instruments rather than suggesting that gesture design and memorisation are important.

4.2 Workshop setup

4.2.1 Participants

The workshop we led was the first workshop of a Musical Inclusion Programme [3], one of the aims of which is to help disadvantaged young people take part in musical activities. “Disadvantaged” stands for a broad variety of living conditions, ranging from health, behavior or learning disorders to looked-after children. Such young people may not have the opportunity to access high-quality musical activities, thus preventing them from the benefits music can provide in a social context. Bespoke digital musical instruments have the potential to make music-making easier and more fun for many of these youth. It is also possible that using personalised instruments may reduce social pressure, since the mapping function is unique to each user. By emphasising participation as a process of exploration of instruments and sound rather than performing a piece of music correctly, we also hoped to make the experience fun and inclusive for everyone.

The 15 youth we worked with all had physical and/or mental disabilities. They were accompanied by their parents or guardians, and their level of concentration was variable depending on their disabilities.

4.2.2 Workshop structure

The workshop was a one-hour session during which each of the two workshop leaders led a sequence of small-group sessions with one youth participant and their parent/guardian(s).

The input device used was a GameTrak Real World Golf controller, which senses 3D position of the user’s hands

[3] http://www.nmpat.co.uk/music-education-hub/Pages/musical-inclusion-programme.aspx

using two strings. Sound was generated by Max/MSP. The following setups were available to the participants:

- Grab-and-play classification for triggering pre-recorded sound samples, in a "funk" style.
- Grab-and-play regression for controlling audio effects (pitch shifting and reverb).
- The same sample triggering and effects control as above, but using Wekinator's existing supervised learning interfaces for classification and regression (i.e., requiring users to specify labeled training examples).

In each small group, the workshop leader controlled the computer (including the GUI for mapping creation), and the youth participant was given the input controller (sometimes with the help of parent/guardians). Participants therefore did not have to learn to use the GUI or other software. All participants tried at least one grab-and-play mapping, and participants who had time and expressed interest also tried supervised learning mapping.

4.3 Observational study

4.3.1 Grab-and-play setup

Our grab-and-play approach was very useful to build adapted interfaces. It allowed us to build instruments whose gestural range was wholly dependent on the participant: during the recording step, some people made wide movements, while others with strong motor disabilities were only able to make small movements. In this sense, the adaptivity of our tool prevented it from building non-playable instrument for a given person. Some participants also seemed to find the exploratory side of the running step very fun. They spent a lot of time trying to find where the different audio samples were in their gestural space: this activity seemed to capture participants' attention, as they usually seemed to engage in choosing which sample to trigger.

Grab-and-play classification seemed to elicit different types of interaction compared to regression. People using classification focused on triggering known sounds, whereas people using regression focused on exploration (alternating between searching for new sounds and playing them back). Both approaches thus have their own pros and cons, depending on which musical activity people and carers want to take part in.

4.3.2 Original Wekinator setup

Participants who had enough concentration also tried the supervised learning setup. They first recorded different GameTrak positions for each of the four classes of samples, and then tried their instrument. Several participants reported that they liked being able to choose where to place the audio samples in their gestural space, giving them even more control on what was going on. However, it was hard for some participants to concentrate on the process of choosing different gestures to trigger different samples. Even if the customization of the interface was enjoyed by some participants, it was not necessary to support meaningful musical experiences for most participants.

Both classification and regression were understood by participants, as they knew which audio effect to expect since they had chosen them during the recording step.

4.3.3 General discussion

This preliminary workshop has shown the utility of our grab-and-play approach to build custom musical interfaces. Our observations show this approach can be useful to build personalised devices, both for participants that were not able to concentrate for a long time, and for participants with specific motor disabilities. In any case, using the grab-and-play mapping could be a fun first musical experience for these young people. Supervised learning could later allow them to more deeply explore customisation.

These observations suggest improvements for our future workshops, in which we plan to experiment with other musical activities, and to test future grab-and-play implementations. Other input devices (such as joysticks, Wiimotes, or dance maps) as well as other output programs (such as instrument-specific samples or visual outputs) could be used to design instruments that are even better customised to each participant. Further, the social aspect of collective musical practices could be investigated through grab-and-play mapping, for example by having different young people exchanging their newly-created models, or more simply by having teachers sing with young people's sonic outputs.

5. INTERVIEWS WITH COMPOSERS AND PERFORMERS

We report on interviews held with three professional computer musicians to analyze how our grab-and-play approach could influence their music practice and/or composition processes.

5.1 Interview setup

5.1.1 Participants

We held individual interviews with three professional computer musicians. All three were composers and performers, as well as active in teaching computer music at university level. One reported previous experience with the Wekinator. We hoped to gather feedback to better understand how our grab-and-play approach could support embodied exploration processes and rapid mapping generation. We also aimed to collect information on ways to improve our first implementation. For instance, we wondered how much control the random generation method would leave to composers.

5.1.2 Structure

Each interview was a 30-minute exchange in which experimentation alternated with semi-structured interview questions. The musician was presented with a one-stringed GameTrak which allows the sensing of a user's 3D hand position, while the first author controlled the computer GUI and led the interview. Experimentation started with our grab-and-play paradigm, spanning regression and classification algorithms; it ended with the original supervised learning setup, using the same regression and classification algorithms. When first trying the grab-and-play setup, composers were not told about its implementation: they thus had no presuppositions when experimenting with it. They were asked about their playing strategies and how they thought it was working. Then, they were asked about ways they could imagine improving this grab-and-play approach. Finally, they used the original Wekinator supervised learning setup, allowing them to experiment and compare the two approaches. For regression, we used a digital synthesis instrument based on similarities between physical models of the flute and electric guitar [14], potentially allowing for vast sound space exploration. Experimentation with classification relied on the sample trigger we used in the previous user study.

5.2 Observational study

5.2.1 Grab-and-play setup

The exploratory aspect of our grab-and-play approach was praised by the three composers. One of them described the system as *"kind of an enigma to solve"*, and was interested in the fact that *"it kind of challenges you: you have to explore a bit, and try to understand the criteria, or how to deal with these criteria"* to perform with it. Also, the possibility of rapidly prototyping a new instrument allowed them to experiment with very different gestural and sonic interactions. Using the same recorded gestural stream to build two instruments, one composer reported when comparing their playing that *"[he doesn't] feel any consistency between them in terms of gesture and sound: they felt like completely different mappings"*, saying he could explore them *"endlessly"*.

Different strategies were adopted to exploit the system's capabilities. One composer first spent time exploring the sonic parameter space, then tried to regain control and to replicate certain sounds. He then decided to reduce the space he was exploring by moving the controller in a given plane rather than in 3D, allowing him to learn certain gesture-to-sound relationships in a *"pleasant"* way. In this sense, one composer reported he could eventually learn how to play such an instrument. After having been told gestural data was randomly selected, one composer tried to exploit this aspect by spending more time in certain locations in his gestural space to increase the likelihood of their inclusion in the mapping. He indicated he was interested in *"playing more"* with this exploit.

The random selection also had some weaknesses: for example, a composer reported he had too little gestural space to explore between two interesting sounds in a given mapping. Another composer said he would require more control over the selection of sound parameters while agreeing that randomly selecting could *"definitely"* go with his vision of composing (*"the embodiment of being able to control the sound with enough level of control, regardless of what the movement is"*). Ways to modify a given mapping would be required as an improvement (this is discussed in section 5.2.3).

5.2.2 Original Wekinator setup

When testing the original Wekinator setup, one composer underlined its effect on his expectation of how a given instrument would work: *"it sets up all the run expectations, and it also affects the way I play it, because now I start to remember these poses, rather than just exploring in an open-ended way"*. Choosing gestures when building a mapping can thus be a responsibility composers want to avoid when creating meaning through sound. In this sense, a composer even mentioned that he *"never care[s] about gesture"* in composition, rather seeing these gestures as movements that are related to his own instrument practice: *"actually, what I care about is the exploration process afterwards"*.

On the other hand, one composer liked the fact that he could immediately replicate a given sound as he *"kind of see[s] what's being mapped there"*. He enjoyed the idea of spending less time on exploration and having more control, as *"in some kind of performance, you want to be very meticulous"*. Comparing the grab-and-play and original Wekinator setups, composers seemed to agree that *"both are useful"*, depending on what they would want to achieve. *"If you set up the mapping yourself, and the system yourself, you have more control, but then again maybe it's too predictable"*, one composer summed up.

5.2.3 Suggestions for improvement

Talking about ways to improve such setups, one composer evoked the idea of *"a hybrid approach"*, where one could record specific gesture-sound relationships and add some randomness in between: *"some points could be manually controlled, and some points automatically"*. This would be a way to address the previously-mentioned trade-off between control and exploration: one could then explore and discover the control space during performance, while having access to predetermined gesture-sound relationships in the mapping.

The random selection was praised for its rapidity in prototyping and experimenting, as for *"most trainings, actually, you're not really so concerned about the specific thing that's done: you just want stuff mapped out"*. However, composers would like to have a bit more control over both gesture and sound when building such a mapping. In this sense, one could imagine clever ways to select gestural and sound parameters that would still enable rapid instrument prototyping. Going further, one composer suggested incorporating the design process within the performance. Instead of being *"a static thing"*, the design process would become a real-time evolution of one's control space (*"me creating the control space in real-time"*). For example, such a performance could entail repeating the same gesture to tell the machine to add new sounds to this gesture. This idea is reminiscent of Fiebrink's play-along mapping approach [15].

Finally, one composer noticed the difficulty in editing a newly-generated mapping: *"It's really frustrating when you're working musically because you just want to tweak that thing, and then the whole thing blows up"*. One could edit the training data, or, as the composer suggested, *"regression is just a geometry, so why can't we just start stretching things and manipulate them?"* Designing a user interface that allows the intuitive modification of an N-dimensional geometry would be necessary; however, this goes beyond the scope of our grab-and-play mapping paradigm.

5.2.4 General discussion

These individual interviews have clarified what kind of compositional processes could be allowed with our grab-and-play approach. Composers' opinions globally corresponded to our intuitions about the discovery and exploration processes encouraged by our first implementation of the tool.

As mentioned by one composer, such a random process may be used when starting a piece, as a way to let new ideas emerge, then opening up a reflection on how to use them: quoting him, *"all these mapping processes are about making decisions that are rational: it's just building blocks. Then, musical decisions come as you actually walk through them..."*

Other implementations of our grab-and-play paradigm may also support composers' needs (see Figure 2). For example, clustering gestural data could meet composers' need for control over their gestural space in relation to sound, while allowing rapid prototyping. This setup is already implemented but not yet tested. Also, most composers wanted to have more control over the choice of sounds: in future work, we would like to allow a user to choose output labels by selecting high-level perceptual characteristics of a synthesis engine's sound space. Finally, hybridizing grab-and-plau mapping with the original supervised learning setup could be a way to encourage discovery while allowing customization. We plan to experiment with each of these implementations in the near future.

6. CONCLUSIONS AND FUTURE WORK

We presented a first implementation of our "grab-and-play" approach to mapping that allows the prototyping of digital musical instruments. We reported on a first workshop with disabled young people, suggesting that the tool could be useful in the context of musical inclusion. The rapid prototyping of adapted musical interfaces allowed youth with less concentration to instantaneously take part in musical activities, while those with more concentration were curious about both grab-and-play and supervised learning setups, notably enjoying the customization of the latter. We also reported on interviews with three composers and performers, suggesting that the tool could encourage the realisation of new musical outcomes. Each of them valued the grab-and-play approach for embodied musical exploration, and underlined the balance between discovery and control that such a paradigm could support. Their feedback allowed us to imagine future improvements to the current implementation. More generally, the grab-and-play's simple yet expressive framework reflects our wish to get more people progressively included in modern musical activities, and in a broader sense, to have them create new technologies more easily.

In the next two years we will develop our contribution to musical inclusion through workshops and prototypes that will implement more engaging musical activities that are specifically adapted to a participant's abilities. We are also currently implementing more sophisticated ways to select gestural inputs and sound outputs. Using unsupervised learning algorithms to extract relevant clusters from the recorded gestural stream could be a possibility. Another possibility would be to generate input data that are more equally spread through the space delimited by user's gestural extrema. The choice of output labels could also be informed by the relationship between synthesis parameters and higher-level perceptual characteristics, enabling the creation of instruments capable of accessing a desired perceptual sound space. Hybrid approaches mixing grab-and-play mapping with user-provided pairs of inputs and outputs could also be a way to encourage exploration while allowing customization. More generally, we believe that having digital musical instruments generate their own gestural interactions just as they generate sounds could be an engaging conceptual framework, both scientifically and artistically, as it remains mostly unexplored in the context of computer music.

Acknowledgments

We thank Simon Steptoe and the Northampton Music and Performing Arts Trust for inviting us to join their Musical Inclusion Programme (funded by Youth Music and the Paul Hamlyn Foundation). We also thank Simon Katan, Mick Grierson and Jérémie Garcia for useful discussions and suggestions.

7. REFERENCES

[1] A. Hunt and M. M. Wanderley, "Mapping performer parameters to synthesis engines," *Organised sound*, vol. 7, no. 02, pp. 97–108, 2002.

[2] R. Fiebrink, D. Trueman, C. Britt, M. Nagai, K. Kaczmarek, M. Early, M. Daniel, A. Hege, and P. Cook, "Toward understanding human-computer interaction in composing the instrument," in *Proc. International Computer Music Conference*, 2010.

[3] M. Lee, A. Freed, and D. Wessel, "Real time neural network processing of gestural and acoustic signals," in *Proc. International Computer Music Conference*, 1991.

[4] R. A. Fiebrink, "Real-time human interaction with supervised learning algorithms for music composition and performance," Ph.D. dissertation, Princeton University, 2011.

[5] B. Caramiaux and A. Tanaka, "Machine learning of musical gestures," in *Proc. International Conference on New Interfaces for Musical Expression*, 2013.

[6] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, and N. Rasamimanana, "Continuous realtime gesture following and recognition," in *Gesture in embodied communication and human-computer interaction*. Springer, 2010, pp. 73–84.

[7] J. Francoise, "Motion-sound mapping by demonstration," Ph.D. dissertation, Université Pierre et Marie Curie, 2015.

[8] M. Leman, *Embodied music cognition and mediation technology*. MIT Press, 2008.

[9] R. I. Godøy and M. Leman, *Musical gestures: Sound, movement, and meaning*. Routledge, 2010.

[10] B. Caramiaux, F. Bevilacqua, T. Bianco, N. Schnell, O. Houix, and P. Susini, "The role of sound source perception in gestural sound description," *ACM Transactions on Applied Perception (TAP)*, vol. 11, no. 1, p. 1, 2014.

[11] H. Scurto, G. Lemaitre, J. Françoise, F. Voisin, F. Bevilacqua, and P. Susini, "Combining gestures and vocalizations to imitate sounds," *Journal of the Acoustical Society of America*, vol. 138, no. 3, pp. 1780–1780, 2015.

[12] M. Wright and A. Freed, "Open sound control: A new protocol for communicating with sound synthesizers," in *Proc. International Computer Music Conference*, 1997.

[13] S. Katan, M. Grierson, and R. Fiebrink, "Using interactive machine learning to support interface development through workshops with disabled people," in *Proc. SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2015, pp. 251–254.

[14] D. Trueman and R. L. DuBois, "Percolate," *URL: http://music.columbia.edu/PeRColate*, 2002.

[15] R. Fiebrink, P. R. Cook, and D. Trueman, "Play-along mapping of musical controllers," in *Proc. International Computer Music Conference*, 2009.