

Shaping and Exploring Interactive Motion-Sound Mappings Using Online Clustering Techniques

Hugo Scurto
Ircam - Centre Pompidou
STMS IRCAM–CNRS–UPMC
Paris, France
Hugo.Scurto@ircam.fr

Frédéric Bevilacqua
Ircam - Centre Pompidou
STMS IRCAM–CNRS–UPMC
Paris, France
Frederic.Bevilacqua@ircam.fr

Jules Françoise
School of Interactive Arts and
Technologies
Simon Fraser University
Surrey, Canada
jfrancoi@sfu.ca

ABSTRACT

Machine learning tools for designing motion-sound relationships often rely on a two-phase iterative process, where users must alternate between designing gestures and performing mappings. We present a first prototype of a user adaptable tool that aims at merging these design and performance steps into one fully interactive experience. It is based on an online learning implementation of a Gaussian Mixture Model supporting real-time adaptation to user movement and generation of sound parameters. To allow both fine-tune modification tasks and open-ended improvisational practices, we designed two interaction modes that either let users shape, or guide interactive motion-sound mappings. Considering an improvisational use case, we propose two example musical applications to illustrate how our tool might support various forms of corporeal engagement with sound, and inspire further perspectives for machine learning-mediated embodied musical expression.

Author Keywords

Motion, Sound, Embodied Interaction, Machine Learning, Expressiveness, Max/MSP.

CCS Concepts

•Human-centered computing → Gestural input; Sound-based input / output; •Applied computing → Sound and music computing;

1. INTRODUCTION

Designing digital musical instruments that are adaptable to user-specific movement characteristics has become increasingly accessible through the use of interactive machine learning. With these technologies, users can build custom

motion-sound mappings [9] by physically demonstrating examples of gestures for given sounds — thus relying on corporeal knowledge instead of programming skills.

Most interactive approaches to machine learning for designing motion-sound mappings have relied on a two-step, iterative design process (see figure 1) [8]. In the first step, called training or *design* step, users perform gestures *along with* pre-defined sounds. In the second step, called *performance* step, users experiment with the newly-created mapping. For example, they can perform similar gestures to the ones they recorded during the design step in order to replay, or re-enact, previously-selected sounds; or, they can perform new gestures in order to explore, and discover, new sonic forms. Users must then alternate several times between these two steps in order to succeed in building a subjectively-rewarding mapping.

Several user studies have proven that this iterative design process can support corporeal engagement with sound [1, 7]. However, recent works have raised a number of points yet to be improved [17]. For example, some users may have difficulties in designing gestures and evermore to fine-tune mapping. Importantly, Scurto et al. found that users might appreciate machine learning-based mappings that surprise and challenge them through continuous physical interaction [17].

In this paper, we describe a novel user adaptable tool for designing motion-based interactive music systems. It is based on an online machine learning implementation that allows mappings to adapt to users in real-time while generating sound, thus merging design and performance steps into one fully interactive experience. We first define our system's workflow, which aims at tightening action-perception loops through continuous adaptation. Second, we describe a first model and implementation of our tool for supporting embodied exploration of motion-sound relationships. Finally, we propose an improvisation use case involving various gestural controllers and sonic environments, and discuss how our approach could support corporeal engagement with sound in real-world musical applications.



Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). Copyright remains with the author(s).

NIME'17, May 15-19, 2017, Aalborg University Copenhagen, Denmark.

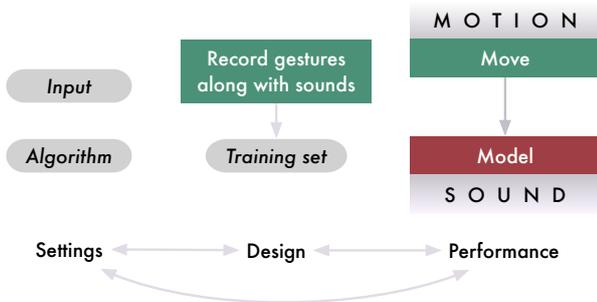


Figure 1: A Mapping-by-Demonstration workflow.

2. BACKGROUND

Mapping, defined as the link between a gestural controller and a sound source, has been central in research on interfaces for musical expression. While most initial research focused on explicit motion-sound relationship programming [9], using machine learning algorithms for mapping design have proven very promising in a musical context where notions of expressivity and generativity are of prime interest [3]. Current interactive approaches to supervised machine learning have turned these algorithms as user-facing musical design tools [5], allowing users to physically demonstrate examples of motion-sound relationships to build a desired mapping function instead of writing code, and without needing to have knowledge on algorithms.

In this context, several supervised algorithms have been studied, depending on the musical task users would like to achieve. For example, Bevilacqua et al. [1] focused on gesture following tasks and implemented a Hidden Markov Model to perform continuous tracking on users’ gestural data. Fiebrink et al. investigated static mapping building using neural networks for regression tasks and several standard algorithms for classification tasks, such as k-nearest neighbors [6]. Françoise et al. proposed four static and/or dynamic models able to perform both classification and regression tasks [8]. Finally, Caramiaux et al. developed a system that recognizes gestures and adapts to performance variations [2].

Beyond such objective-oriented tasks, research by Fiebrink et al. have shown that machine learning can support creative discoveries in musical motion-sound mapping design [6]. For example, criteria such as unexpectedness and accessibility have been praised by computer musicians when composing an instrument [7]. In this spirit, Scurto and Fiebrink proposed new methods for rapid mapping prototyping which shift users’ focus from designing motion-sound relationships to the embodied exploration of relationships that have been generated partly by the computer [17].

However, to our knowledge, most of these approaches remained focused on a two-step design process (see figure 1), where users alternate between demonstrating gestures along pre-recorded sounds (movement acted from the experience of listening to a sound) and interacting with newly-created mappings (movement acted as having an effect on sound). This iterative process might interrupt musical intentionality encoding, which, as theorized by Leman, necessitates an active, action-oriented, corporeal engagement of humans with sound [12]. Interestingly, other computational approaches aimed at providing users with such continuous interactive flows, for example using dynamic mapping strategies [14] or physics-based mappings [15].

Inspired by such approaches and other interactive music systems [10], we propose to reconsider mapping creation to bridge the gap between design and performance steps.

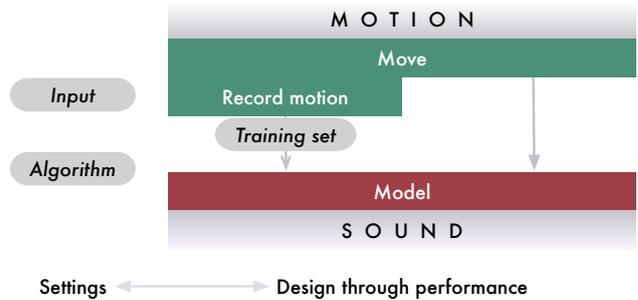


Figure 2: The interactive workflow of our system.

3. DESIGN THROUGH PERFORMANCE

In this section, we define the learning workflow of our system as well as its interaction modes.

3.1 Definition

Our wish is to enable users to design mappings in an online fashion, where design would be made possible through performance. We propose the following workflow, which is depicted in figure 2.

3.1.1 General workflow

Our system allows users to design machine learning-based motion-sound mappings while performing with them. More precisely, it enables online multidimensional adaptation to users input gestural space by continuously recording input data as the training set of a machine learning algorithm. Both design and performance steps are thus supported under the same motion flow. The modelling of the “internal structure” of users’ gestural space can then drive sound synthesis in several manners (as described in section 5), all of them being characterized by direct, corporeal interaction with sound and personalized exploration of motion in relation to sound. We designed our system with a particular focus on reducing GUI actions taken in-between performances. One level of interaction with machine learning still remains available to users: similarly to previous mapping-by-demonstration tools, the “setting” step allows for configuring a minimal set of learning parameters as well as input parameters (as described in section 4).

3.1.2 Online learning

Such an interaction paradigm differs from previous interactive supervised learning approaches: instead of demonstrating gestural examples that have been designed and labeled in a separate step, users physically interact with an adaptive model that constantly generates sound, depending on both previous and current user movement. Importantly, our system thus switches from current mapping-by-demonstration supervised paradigms (where user-provided pairs of gestures and sounds constitute a training set) to an unsupervised learning paradigm (where the training set consists in unlabeled gestural data). However, as we will see, users still have the possibility to consciously influence the learning by performing and correcting the system. We will discuss such learning workflows in section 6.

3.2 Interaction modes

From this definition, we designed two interaction modes based on different memory processes. The main concept is to allow users to design parts of their input space through the metaphor of temporal persistence, where “occupation time” (as an “accumulation process”) is central to the creation of the training set. There are several other ways to

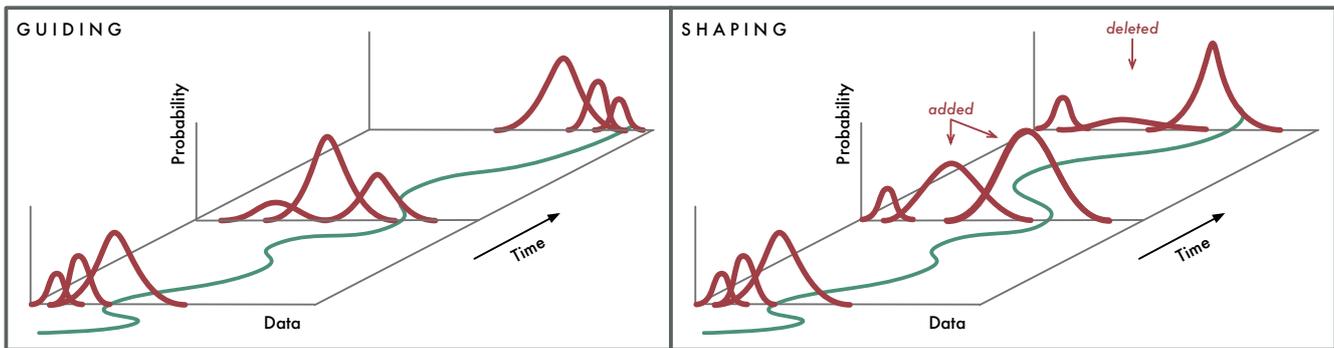


Figure 3: The two interaction modes. Probability clusters are sampled at 3 discrete times for 1-dimensional gestural data. On left, clusters continuously evolve as users’ gestural data is recorded to the training set with a sliding window. On right, users continuously modify clusters’ parameters as they successively add and delete gestural data to the training set.

interact online with a machine learning algorithm through its training set: we will discuss it in section 6.

3.2.1 Guiding

The guiding mode (figure 3, left) consists in having users adding gestural data with a sliding temporal window to the training set during the interaction. It can be seen as an interactive music system with a constant-size memory, where users could directly and physically explore sound spaces in order to foster creative discoveries. It allows mappings to evolve continuously, focusing in or out of some spaces in users’ gestural input space in real-time following abstract embodied specifications of users. A typical situation would involve the creation of clusters in a relatively small area of the input space by having users stay in this part of the input space, then its real-time evolution (or guiding) by moving in larger areas of the input space. This personalized interaction relies on an finite memory process where old data would be continuously replaced from the training set by new data.

3.2.2 Shaping

The shaping mode (figure 3, right) consists in having users interactively adding and/or deleting gestural data to the training set during the interaction. It can be seen as a continuous extension of previous interactive machine learning systems, where users could delete and re-add a previously-recorded example in a design step by clicking on a button in a design step, then see the effect in a performance step. Here, users can add new examples and delete old ones by (re-)demonstrating them, while hearing the sonic consequences in real-time. Like using a pencil with eraser, this would allow rapid, custom, and fine-tuned modification of mappings. A typical situation would involve the creation of a new cluster for a new gesture, then its modification (or shaping) by adding or deleting variations of this gesture in the recorded data. This personalized interaction relies on an (almost-)infinite memory process where the training set would grow as users successively supply the system with data.

4. SYSTEM IMPLEMENTATION

We present the first learning model implemented in our system, as well as our system’s current architecture.

4.1 Learning model

The current version of our tool implements an online, unsupervised version of Gaussian Mixture Model (GMM). GMMs

are very general and versatile probabilistic models for designing motion-sound relationships, providing with variables for both classification and regression at a relatively low computational cost [8].

A GMM is a learning model that can perform soft clustering, which is identifying groups of similarity in gestural data and computing for a new data point \mathbf{x} each probability that it belongs to each of these clusters. Here, clusters are modelled as Gaussian distributions \mathcal{N} , and the probability p of belonging to the overall model θ is given by:

$$p(\mathbf{x}|\theta) = \sum_{k=1}^K \pi_k \mathcal{N}(\mathbf{x}|\mu_k, \Sigma_k) \quad (1)$$

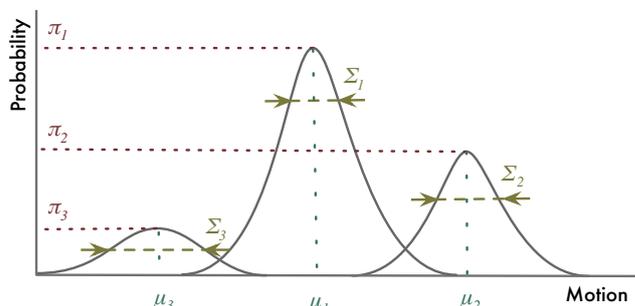


Figure 4: A Gaussian Mixture Model with $K = 3$ cluster components for 1-dimensional motion data.

There are four categories of parameters in GMM (see figure 4 and equation 1). The first one is the number of clusters K , which is the number of multivariate Gaussian distributions used in the mixture model. These clusters can be used for classification purposes. Then, each Gaussian distribution has its own mean vector μ_k and covariance matrix Σ_k , as well as its own weight π_k in the mixture. These parameters can be used for regression purposes. In a standard interactive supervised learning setup, such parameters are set and learnt offline from custom gesture-sound examples demonstrated by users. In our paradigm, the learning is online: Gaussian parameters would evolve in real-time as users supply the model with only gestural data, which support continuous action-perception workflow as specified in the previous section.

In such an online, unsupervised paradigm, we propose to add entropy $H = -\sum p(\mathbf{x}) \ln p(\mathbf{x})$ as a supplementary parameter for controlling sound synthesis. Our idea is to

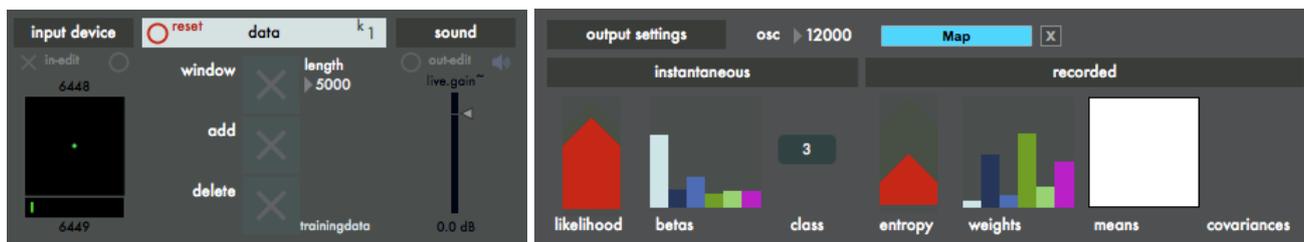


Figure 5: Current user interface. On left: Main window allowing recording gestural data following different interaction modes. On right: Output model parameters.

report the evolution of the model as users modify the training set, and use it as a modality for sound generation. Entropy can be seen as an abstract scalar quantity rendering the amount of order in data. For example, non-overlapping clusters with close-to-zero covariance values would give a low entropy value, whereas an almost-uniform distribution would give a much higher value.

4.2 Software

We implemented a prototypical version of our system as a Max/MSP patch¹ that makes an extensive use of *XMM* library for learning GMMs [8] and *MuBu* objects for storing and editing data [16]. The tool’s GUI provides users with different capabilities (see figure 5):

- Connect any kind of gestural input device, provided its data is sent as an OSC message.
- Experiment with different kinds of sound synthesis module, provided they receive OSC messages.
- Modify the training set physically either by adding, deleting, or window streaming gestural data.
- Define the length of the recording window.
- Define the number of Gaussian components in the GMM.

Currently, our tool supports online learning by training and running a GMM at a sufficiently high rate (every 100 ms) so that it remains perceptually convincing in an action-perception workflow [10]. Gestural data is either stored incrementally or replaced dynamically by making use of overdub and append messages of the MuBu container. The “delete” action is made possible by identifying and deleting the first nearest neighbour of user live input in the training database. Ultimately, we will implement an online learning algorithm in a more global framework whose outline is defined in section 6, and work on interactive visualizations of Gaussians distributions to enable users to interact with our tool in a multimodal, audiovisual environment (as discussed in section 6).

5. EXAMPLE MUSICAL USE CASES

We propose to illustrate the possibilities of our tool in two use cases focusing on exploratory improvisational processes, using various gestural controllers and sonic environments. We implemented them as Max/MSP patches¹; mappings are described in figure 8.

5.1 Meta-sound synthesis engine

This first application aims at facilitating exploration of a subtractive synthesis engine through human motion. For the sake of clarity, we focused on two-dimensional static position features using a simple mouse position tracking patch.

¹<http://github.com/hugoscurto/OnlineGMM>

As shown in figure 6, we chose to pair each Gaussian to a resonant filter.

Users are first allowed to set the number of Gaussian components of the GMM. Here, it sets the number of resonant filters at stake in their synthesis engine: it is thus closely related to the complexity of the mapping. Then, they can experiment with the two interaction modes provided by our system. On the one hand, using the guiding mode allows them to control and explore the resonant filters through various movement strategies (for example, first focusing on a small spatial area where all resonant filters would be concentrated, then expanding the control area to larger spatial bounds, changing both control modalities and qualities of resonant filters as previously-computed Gaussians would evolve). On the other hand, using the shaping mode allows them to edit each resonant filter parameters step by step in an abstract manner (for example, first creating one resonant filter at a given position, then exploring different resonance parameters by either adding or deleting gestural data to modify the covariance of the paired Gaussian, then iterating this process with other resonant filters).

This use case illustrates how our system supports open-ended exploration of a sound synthesis engine, where embodiment could drive the composition of complex sound parameter combinations in real-time.

5.2 Expressive shaker

This second application aims at sonifying movement expressiveness through concatenative synthesis. We used an embedded module² to sense hand acceleration and extracted from it a 36-dimensional wavelet spectrum to render movement quality. We chose to map entropy with random pitch variation of each played grain, and variance values to random duration variation of each played grain. As shown in figure 7, we paired each Gaussian mean to a “reader head” that allows for navigating through different sound grains in a given audio descriptor space.

Again, users first set the number of Gaussian components of the GMM. Here, it sets the number of reader heads of the concatenative synthesis engine, as well as the number of movement qualities modelled by the GMM. Then, they can experiment with the two interaction modes provided by our system. On the one hand, the guiding mode would allow them to focus in turns on different movement qualities: when a given movement quality is stable, all Gaussians gather together on a precise localization in the input space, and play the same sample at a constant pitch and duration. On the other hand, the shaping mode should allow them to successively specify given movement qualities.

This use case illustrates how our system supports open-ended exploration of motion qualities, where sound could drive the embodiment of different kind of expressive motions in real-time.

²<http://ismm.ircam.fr/riot/>

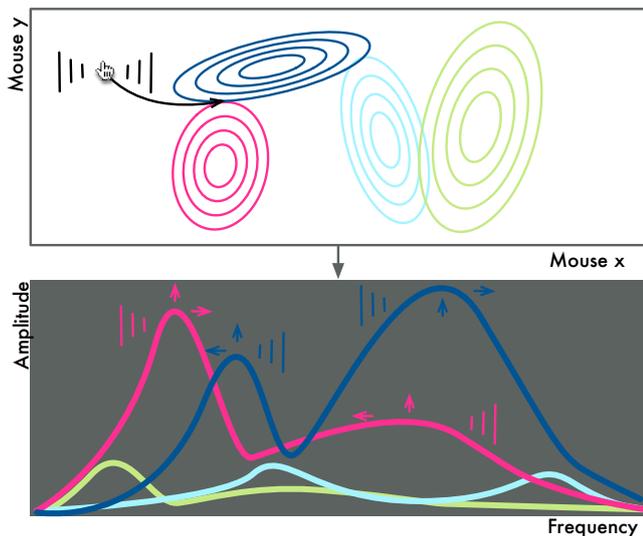


Figure 6: Schematic representation of the meta-sound synthesis engine. Here, it contains 4 resonators, each of them paired with one cluster.

6. DISCUSSION

In the next section we discuss several perspectives related to future evaluation and implementation of our system.

6.1 Exploration through interaction

When first interacting with a musical interface, users can take many paths and adopt various styles to explore the instrument’s possibilities and constraints, thus learning how (or renouncing) to use it. By enabling design through performance, our system aims at supporting exploration of novel paths in the creation and use of musical interfaces. We believe these paths may differ from those suggested by interactive supervised learning tools, where users have to alternate between design and performance steps to come up with a rewarding mapping.

Moreover, we designed the “guiding” and “shaping” interaction modes with a particular attention to support both fine-tune modification tasks and open-ended improvisational practices. However, as all user actions are implemented under the same experiential workflow (action-perception loops emerging from physical interaction with sound), other alternative uses may be achieved. For example, one could add data to the training set indefinitely to create a mapping that would progressively “freeze” once having recorded enough data. Also and perhaps surprisingly, the “Delete” action actually produces sound: one could imagine a performance where “Delete” gestures would act as control mechanisms for sonic events. Several new interaction styles could thus be explored with our tool, each of them placing corporeal engagement with sound as the main point of focus.

6.2 Evaluation and human learning

Our tool workflow heavily relies on listening abilities in relation to motion, and on a metaphor of temporal persistence and “occupation time”. In this sense, it is arguable that our tool may present novice users with a low threshold for taking part in musical activities. Meanwhile, its interaction modes potentially present a certain level of sophistication that might also support expert performing. To assess these points, we will lead user evaluation with both user groups in order to study how novice users could gain expertise in music making through our workflow, and to better understand what kind of creative paths composers and performers

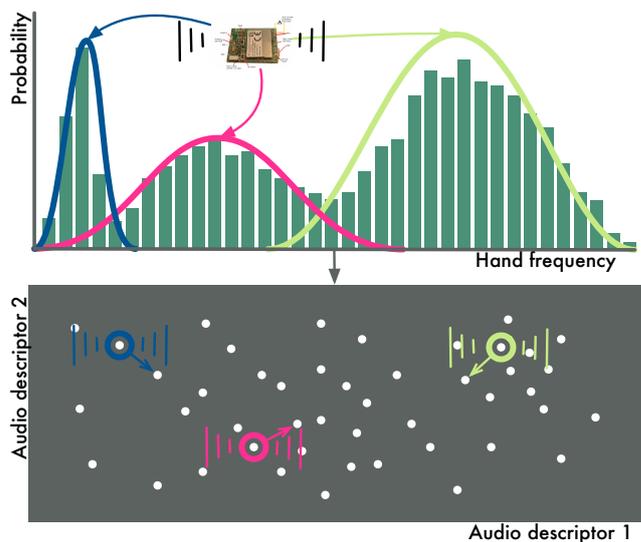


Figure 7: Schematic representation of the expressive shaker. Here, it contains 3 reader heads, each of them paired with one cluster.

would discover using our tool.

Alternatively, we believe such direct interaction with sound could be applied to other musical contexts in order to arouse motivation and novel forms of expression. For example, using our system in a shared and distributed setup where several users would edit a unique training set could be of interest to support corporeal synchronization through musical activities. In future work we may use such data-driven scenarios with various kinds of users, from novice to expert performers as well as people with disabilities working with music therapists. We hope this would help us gain an understanding of how a machine learning entity could steer social and collective interaction in embodied musical activities.

6.3 Further implementation

Our current prototype provides users with one learning model (GMM) and a slider-based GUI. In future work we will implement an online expectation-maximization algorithm for continuous, optimized learning and inferring, and investigate interactive visualizations of Gaussian distributions to let users interact in an audiovisual augmented reality setup. Also and importantly, we would like to let users experiment with other learning algorithms, allowing for even more diverse musical uses. For example, a current limitation of the Gaussian Mixture Model is that it considers each new input as independent from previously-observed data points. Such a property might not be suitable to human movement, as dynamics are deemed of prime importance when dealing with qualities of corporeal expressiveness [13]. Therefore, modelling dynamic patterns in gestural data could be a promising approach for generating sequential musical output that would be stylistically coherent with users’ bodily expression. We plan to study adaptive dynamical systems to both model user-specific movement qualities and to generate continuous navigation trajectories [11]. Another approach would be to study a reactive factor oracle [4] to let users either shape a training set of movement patterns, or guide a discrete navigation through this training set.

Finally, our current implementation does not provide users with a completely continuous way to interact with machine learning. If the number of GUI actions has been reduced from previous interactive supervised learning systems, users still have to specify whether they would like to record, delete,

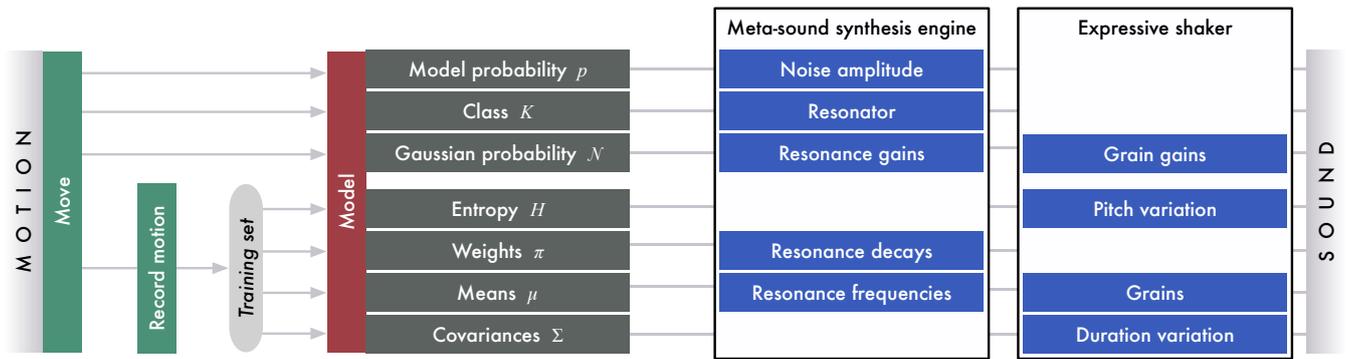


Figure 8: Example musical mappings implemented for our improvisational use case.

or window stream data during their performance. Other memory processes may be investigated to allow automatic recognition of physical actions taken by users [11], thus mediating embodied musical interactions more fluidly.

7. CONCLUSION

We presented a user adaptable tool for designing motion-sound mappings that merges design and performance steps into one fully interactive experience. It implements an on-line learning workflow that allows mappings to adapt in real-time to users' movement while generating new sonic parameters. We designed two main interaction modes allowing different degrees of modification and exploration in designing mappings, as well as two example musical applications in an improvisational use case to show how our tool might support corporeal engagement with sound in an innovative manner. In future work we will implement a computational framework supporting continuous corporeal interaction with both static and dynamic learning models along with interactive visualizations of the system's internal state. This should allow us to conduct novel artistic and educational real-world applications where human motion would be at the center of musical expression.

8. ACKNOWLEDGMENTS

This work was partly supported by the RAPID-MIX Innovation Action funded by the European Commission (H2020-ICT-2014-1 Project ID 644862).

9. REFERENCES

- [1] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, and N. Rasamimanana. Continuous realtime gesture following and recognition. In *International Gesture Workshop*, pages 73–84. Springer, 2009.
- [2] B. Caramiaux, N. Montecchio, A. Tanaka, and F. Bevilacqua. Adaptive gesture recognition with variation estimation for interactive systems. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 4(4):18, 2015.
- [3] B. Caramiaux and A. Tanaka. Machine learning of musical gestures. In *NIME*, pages 513–518, 2013.
- [4] A. Chemla. Guidages de l'improvisation musicale homme-machine. Master's thesis, UPMC, 2015.
- [5] R. Fiebrink and B. Caramiaux. The machine learning algorithm as creative musical tool. In *Oxford Handbook of Algorithmic Music*. Roger Dean and Alex McLean (Eds.). Oxford University Press., 2016.
- [6] R. Fiebrink, P. R. Cook, and D. Trueman. Human model evaluation in interactive supervised learning. In *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, pages 147–156. ACM, 2011.
- [7] R. Fiebrink, D. Trueman, C. Britt, M. Nagai, K. Kaczmarek, M. Early, M. Daniel, A. Hege, and P. R. Cook. Toward understanding human-computer interaction in composing the instrument, 2010.
- [8] J. Françoise, N. Schnell, R. Borghesi, and F. Bevilacqua. Probabilistic models for designing motion and sound relationships. In *Proceedings of the 2014 International Conference on New Interfaces for Musical Expression*, pages 287–292, 2014.
- [9] A. Hunt and M. M. Wanderley. Mapping performer parameters to synthesis engines. *Organised sound*, 7(02):97–108, 2002.
- [10] S. Jorda. Digital lutherie crafting musical computers for new musics' performance and improvisation. *Department of Information and Communication Technologies*, 2005.
- [11] D. Kulic, W. Takano, and Y. Nakamura. Incremental learning of full body motions via adaptive factorial hidden markov models. In *Proc. of the International Conference on Epigenetic Robotics*, 2007.
- [12] M. Leman. *Embodied music cognition and mediation technology*. Mit Press, 2008.
- [13] M. Leman. *The expressive moment: How interaction (with music) shapes human empowerment*. 2016.
- [14] A. Momeni and C. Henry. Dynamic independent mapping layers for concurrent control of audio and video synthesis. *Computer Music Journal*, 30(1):49–66, 2006.
- [15] J. C. Schacher, D. Bisig, and P. Kocher. The map and the flock: Emergence in mapping with swarm algorithms. *Computer Music Journal*, 2014.
- [16] N. Schnell, A. Röbel, D. Schwarz, G. Peeters, R. Borghesi, et al. *MuBu and friends—Assembling tools for content based real-time interactive audio processing in Max/MSP*. Ann Arbor, MI: Michigan Publishing, University of Michigan Library, 2009.
- [17] H. Scurto and R. Fiebrink. Grab-and-play mapping: Creative machien learning approaches for musical inclusion and exploration. In *Proceedings of the International Conference of Computer Music*, 2016.