Towards Assisted Interactive Machine Learning: Exploring Gesture-Sound Mappings Using Reinforcement Learning

Federico Ghelli Visi[1, 2, *] and Atau Tanaka[2, **]

[1]GEMM))) Gesture Embodiment and Machines in Music
School of Music in Piteå, Luleå University of Technology, Sweden

[2]EAVI – Embodied Audiovisual Interaction
Goldsmiths, University of London, UK

[*]mail@federicovisi.com
[**]a.tanaka@gold.ac.uk

Abstract.

We present a sonic interaction design approach that makes use of deep reinforcement learning to explore many mapping possibilities between input sensor data streams and sound synthesis parameters. The user can give feedback to an artificial agent about the mappings proposed by the latter while playing the synthesiser and trying the new mappings on the fly. The design approach we adopted is inspired by the ideas established by the interactive machine learning paradigm, as well as by the use of artificial agents in computer music for exploring complex parameter spaces. We refer to this interaction design approach as Assisted Interactive Machine Learning (AIML). We describe the architecture of an AIML system prototype, a typical workflow for interacting with the agent and obtain gesture-sound mappings. We then present feedback data collected during a demonstration and discuss perspectives for developing the AIML paradigm further, pointing out current limitations. In light of the feedback obtained and the considerations arisen following the use of the system in a multimedia performance piece, we suggest that the implementation and evaluation of new features should take into consideration established creative workflows and take place close to actual artistic practice.

Keywords. Gestural Interaction, Interactive Machine Learning, Reinforcement Learning, Artificial Agents, Sonic Interaction Design

Introduction

Gesture-sound Interaction Design and Interactive Machine Learning

Designing gestural interactions between body movement and sound synthesis is a multifaceted process. At its core, it takes place through the definition of mapping functions between input signals (usually obtained through some motion sensing device) and sound synthesis parameters (Hunt and Wanderley, 2003). Adopting an effective mapping strategy is one of the key factors affecting the expressive potential of a live interface, and as the spaces defined by input signals and synthesis parameters become more highly-dimensional and heterogeneous, designing mapping structures can be an increasingly complex task, with many possible solutions (Van Nort et al., 2014). In addition to the relatively abstract realm of designing mappings, researchers on gesture-sound interaction draw upon several fields of inquiry. The results of quantitative studies of music-related body motion based on sound-tracing experiments were indicated as a useful source for defining mappings in musical interfaces (Nymoen et al., 2013). Informed by environmental psychology, the notion of sonic affordance was introduced to look at how sound may invite action, and how this could potentially aid the design of gestural interfaces (Altavilla et al., 2013). Qualitative observations of the gestural aspects of traditional musical instruments have been looked at to inform mapping strategies (Visi et al., 2014), as well as to develop musical practices where bodily gestures are seen as fundamental compositional elements (Östersjö, 2016).

Machine learning frameworks for interacting with sound synthesis environments (Fiebrink et al., 2009) have brought about sonic interaction design approaches where mappings are not explicitly defined and manually coded, but are "shown" – or given by demonstration – to a system capable of "learning" them (Françoise et al., 2014). The widespread adoption of such interactive gesture-sound mapping approaches, facilitated by the accessibility of software tools such as the Wekinator (Fiebrink and Cook, 2010), lead to the establishment of the Interactive Machine Learning (IML) interaction design paradigm. The machine learning algorithm is thereby considered as an interface between humans and computers, a creative tool that, with its own affordances and constraints, supports the process of musicians and sonic artists (Fiebrink and Caramiaux, 2018).

The increasing detail offered by motion sensing technologies, and the complexity of sound synthesis engines creates the opportunity for exploring the numerous, non-obvious ways in which these domains can be interfaced. The user-centric interface design methods enabled by machine learning delineate a scenario in which exploring gestural mappings can be done interactively, intuitively, and with the assistance of algorithms that become part of the creative tool kit of musicians.

Artificial Agents for Parameter Space Exploration in Computer Music

Modern sound synthesis techniques are often characterised by a high number of parameters one can manipulate in order to make different sounds. Whilst these afford vast synthesis possibilities, exploring the resulting extensive parameter spaces may be a challenging task, which can be particularly difficult to accomplish by manipulating every parameter by hand. In computer music, mathematical models inspired by biological processes have long been used to explore the possibilities afforded by sound synthesis techniques. To provide a few examples, Miranda (1995) used cellular automata (a model of biological self reproduction) to generate a large amount of sonic particles that form complex sound events. Dahlstedt (2001) proposed a system based on genetic algorithms where the users listen to the sounds generated by the software and select those they find more interesting. Following an evolutionary model, the system then proposes new sounds by "mating", "mutating", and "evolving" the sounds that were selected by the user in previous generations. Genetic algorithms were also adopted later by Yee-King (2016) to explore timbre spaces in sound synthesis. The same author and collaborators later applied several machine learning and optimisation techniques to automatically programme a synthesiser to match a given target sound as accurately as possible (Yee-King et al., 2018). An approach based on sound matching and genetic algorithms informed by the work of Horner et al. (1993) was also adopted by David Griffiths and the FoAM network in a collaboration with the electronic music artist Aphex Twin.[1]

Reinforcement learning is an area of machine learning in which artificial agents are programmed to take actions in an environment defined by a set of parameters. Their goal is to maximise the positive feedback – or rewards – they are given by a human (or by another algorithm) observing the outcome of their actions. Deep reinforcement learning approaches – such as the Deep TAMER algorithm – leverage the power of deep neural networks and human-provided feedback to train agents able to perform complex tasks (Warnell et al., 2018). Recently, Scurto et al. (2019) implemented the Deep TAMER algorithm to design artificial agents that allow to interactively explore the parameter spaces of software synthesisers.

We present a system that makes use of deep reinforcement learning in the form an artificial agent to explore different mappings between an input device and a sound synthesis engine. The user can give positive or negative feedback to the agent about the proposed mapping while playing with the interface, and try new mappings on the fly. The design approach we adopted is inspired by the ideas established by the IML paradigm, as well as by the use of artificial agents in computer music for exploring complex parameter spaces. We call this interaction design approach Assisted Interactive Machine Learning (AIML).

Method

The system is designed to interactively explore the motion-sound mappings proposed by the artificial agent. This iterative collaboration can be summarised in four main steps:

1. Sound design: the user authors a number of sounds by editing a set of salient synthesis parameters;

[1] The result of the collaboration between FoAM and Aphex Twin is described here: https://fo.am/activities/midimutant/

2. Agent exploration: the agent proposes a new mapping between the signals of the input device and the synthesis parameters based on previous feedback given by the user[2];
3. Play: the user plays with the synthesiser using the input device and the mapping proposed by the agent;
4. Human feedback: the user gives feedback to the agent.

Steps 3 and 4 are repeated until the user has found as many interesting motion-sound mappings as they like. The following subsections will describe the system architecture and a typical workflow.

It is worth noting that, differently from most IML applications for gestural interaction, there is not a gesture design step during which the performer records some sample sensor data for training the system.[3] This is perhaps one of the most obvious differences between the IML and AIML paradigms. In an AIML workflow, the sample sensor data used for training the model is provided by the artificial agent, whereas the user gives feedback to the agent interactively while playing the resulting gesture-sound mappings.

System Architecture

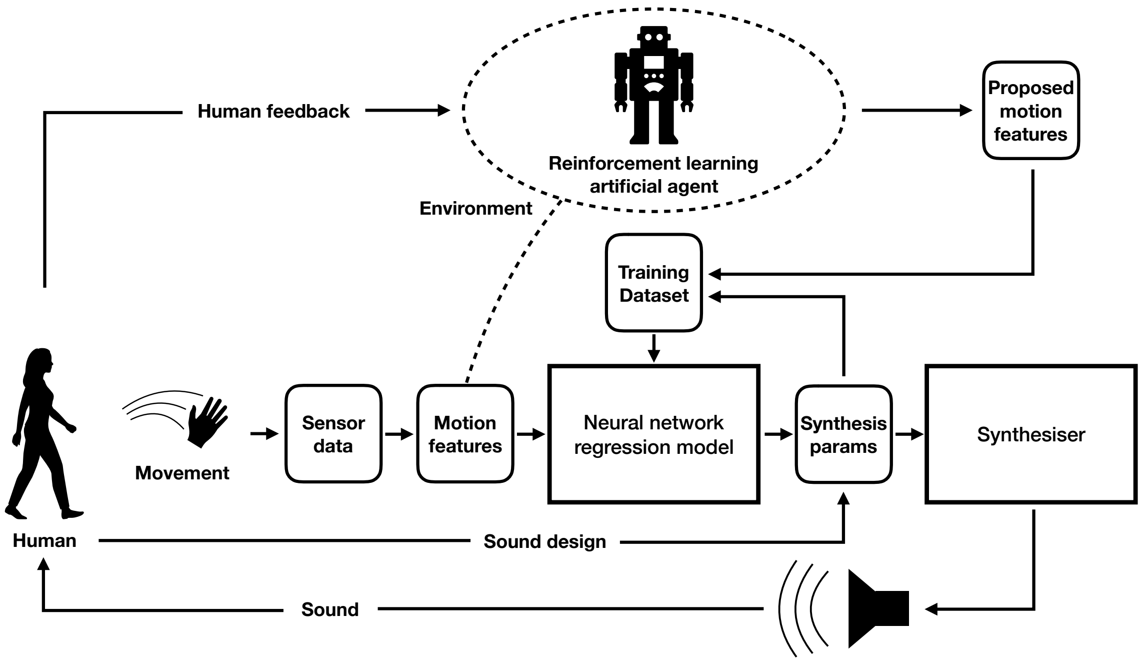

Figure 1: System architecture

The architecture of the system is schematised in Fig 1.

The movements of the human performer are captured by means of an input device. In the first prototype of the system we used the built-in accelerometers of a smartphone, while in a second version we captured gesture using a Myo sensor armband, which provides 8-channel electromyogram (EMG) muscle sensing and a 9-axis inertial measurement unit (IMU).

The motion features[4] extracted by the raw sensor data are stored in a vector and sent to a regression model created using a neural network. This was implemented in Max[5] using the `rapidmax` object (Parke-Wolfe et al., 2019), an external built using RapidLib[6] (Zbyszyński et al., 2017), a set of software libraries for interactive machine learning applications in the style of Wekinator (Fiebrink et al., 2009). These features

[2]If no feedback was previously given, the agent starts with a random mapping.
[3]For an example of gesture design in an IML workflow, see the study by Tanaka et al. (2019) carried out within the BioMusic project.
[4]The motion features may be a set of descriptors derived from the raw data and/or the raw data itself, depending on the hardware used and how the system architecture is implemented.
[5]http://cycling74.com/products/max/
[6]www.rapidmixapi.com

also represent the dimensions of the environment in which the artificial agent operates. By exploring this feature space following the user's feedback, the agent proposes a set of motion features to be paired with the synthesis parameters defined by the user during the sound design step. This becomes the dataset used to train the neural network. The resulting regression model maps the incoming sensor data to sound synthesis parameters.

For the agent, we used *Co-Explorer*[7], a Python deep reinforcement learning agent implementation by Scurto et al. (2019). Bidirectional communication between the agent and Max is done through Open Sound Control (Wright, 2005). Human feedback to the agent is given via a custom touch interface, which was designed in TouchOSC[8] and implemented on an iPhone (see Fig. 2).

Figure 2: The interface for playing the synthesiser and give feedback to the artificial agent.

For the first prototype of the system, we used a sample-based synthesiser to manipulate an audio file stored in a buffer. A second version of the system was instead built around the synthesiser used for the study by Zbyszynski et al. (2019), which implements corpus-based concatenative synthesis using MuBu[9] CataRT Max objects. Interaction with corpus-based concatenative synthesis was further refined by adopting the method based on self-organising maps by Margraf (2019). This method was then implemented in the piece "You have a new memory" (Visi, 2020).

Workflow

We will now describe more in detail the four main steps of the interactive collaboration between the human performer and the artificial agent. The whole workflow is schematised in Fig. 3.

1. Sound design

In this first step, the user defines a number of sounds by manipulating a set of synthesis parameters. This process may differ depending on the synthesiser chosen and which synthesis parameters are exposed to the user in this step. In the first version of the system using the sample-based synthesiser, the sounds are defined by manipulating six parameters (playback speed, pitch shift, start time, duration of the sample selection, filter cutoff frequency and resonance). Here, the user defines the parameters of four sounds that will be used to train a neural network in step 2 and perform regression in step 3. The sounds designed in the sound design step will thus act as timbral anchor points that define a space for interpolation and extrapolation of new sounds.

[7]https://github.com/Ircam-RnD/coexplorer
[8]https://hexler.net/products/touchosc
[9]https://forumnet.ircam.fr/product/mubu-en/

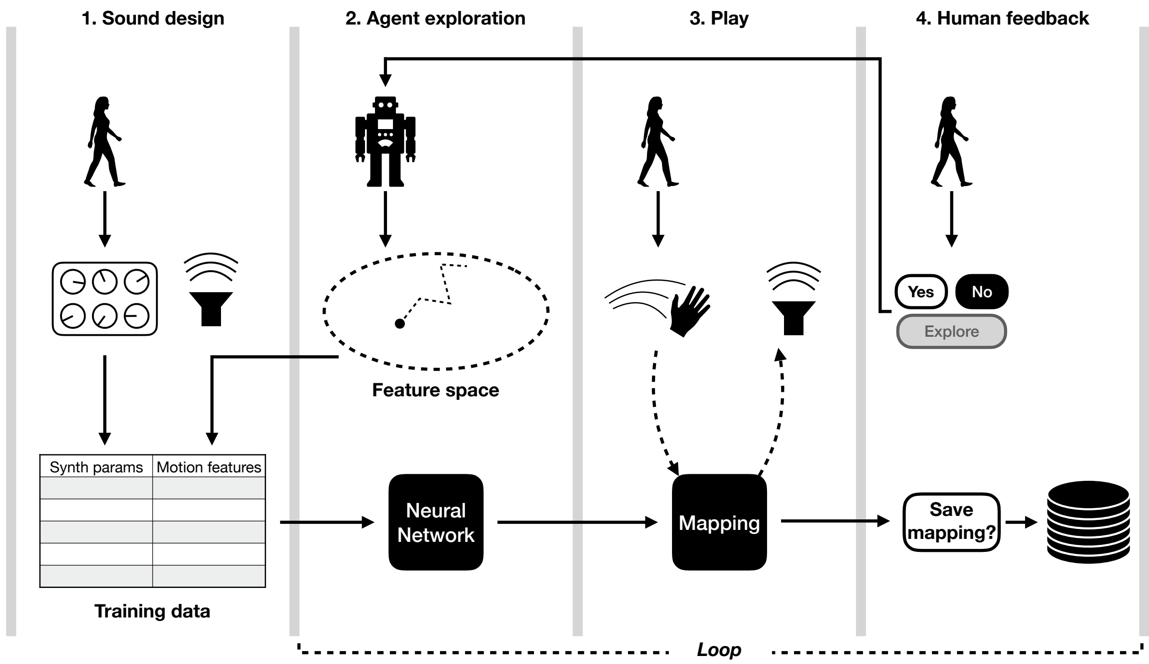

Figure 3: The prototypical assisted interactive machine learning workflow.

2. Agent exploration

The dimensions of the environment explored by the agent are defined by the motion features extracted from the raw sensor data for each of the sound presets. Thus, at the end of the exploration step, the agent returns a vector with a set of input features for each of the sound synthesis parameters sets defined in the sound design step. This means that in the case of the version of the system using a 2D accelerometer, the agent will return four 2D vectors. These will be automatically paired with the synthesis parameters to train a neural network and create a regression model, which will be used in the following step to map live incoming sensor data to sound synthesis.

3. Play

In this step, the user is free to play with and explore the resulting gesture-sound mapping for however long they like. Given that the regression models allow both interpolation and extrapolation of the input sound synthesis data, this step also allows to explore the timbral possibilities of the synthesiser while playing the mapping.

In our prototype, the big button in the touch interface (shown in Fig. 2) triggers the amplitude envelope of the synth, while the movements tracked by the accelerometer axes are used to modulate the synthesis parameters through the mapping implemented in the regression model.

4. Human feedback

After playing with the mapping, the user can give feedback to the artificial agent through the touch interface (Fig. 2). We adopted the concepts of *guiding feedback* and *zone feedback* implemented in the agent designed by Scurto et al. (2019). Guiding feedback is a binary evaluation of the actions performed by the agent, or the direction of its exploration of the feature space. Zone feedback is instead an evaluation of the area of the feature space the agent is currently exploring. For example, a negative guiding feedback would change the direction of the agent's trajectory in the feature space, whereas a negative zone feedback would immediately transfer the agent to a different region of the space.

In our system, the user can give positive or negative guiding feedback to the agent about the proposed mapping. This feedback guides the direction of the next explorations of the feature space, and thus affects the next mappings proposed by the agent. In addition, the user can tell the agent to move to a different area of the feature space by using the *EXPLORE* button. This corresponds to a negative zone feedback, and will likely result in a new mapping that is considerably different from the previous one. In practice, this could be useful for trying something new once one is satisfied with the mappings proposed by the agent after a few guiding feedback iterations. In fact, whereas negative guiding feedback results in adjustments to the mappings currently being proposed by the agent, negative zone feedback triggers the exploration of a new area of the feature space, thus exploring new mapping possibilities. Users can save mappings to JSON files. Mappings can then be retrieved later for performance or as mapping material to be further refined using other approaches.

Early User Feedback

We showed the first prototype of the system during an informal demo at the Human Data Interaction (HDI) workshop *Art, AI-created content, & industrial/cultural effects.*[10] There, attendees were explained the purpose of the system and showed how to interact with the artificial agent. The audio loaded on the synthesiser was constituted by six samples, each lasting one second and taken from a different sound source: speech, a field recording in a train station, a drum beat, a string ensemble, a whispering voice, and the sound of glass breaking. This was done in order to have some timbre variety when playing with the synth. The samples were the same for all the participants, and so were the synthesis parameters sets, so the demo focused on steps 2 to 4 of the workflow.

We collected feedback from 8 workshop attendees that tried the system in the form of a questionnaire. All the participants reported that they are active in one or more artistic discipline among music, visual arts, and performance. Additionally, three of them reported that they are developers and two of them academics. The questionnaire included five questions that the participants could answer using a five-level Likert scale where 1 corresponded to “not at all/strongly disagree” and 5 “yes very much/strongly agree”. The questions were:

- Q1: Did the artificial assistant help you discover the sounds the synthesiser can make?
- Q2: Did the feedback you gave to the artificial assistant help obtain sound interactions you liked better?
- Q3: Did the artificial assistant surprise you with sound interactions that you weren’t expecting but that you found interesting?
- Q4: If you’re a practicing musician, do you see yourself using similar AI-based procedures to explore the possibility of your musical tools? (optional)
- Q5: Was it fun to play with the artificial assistant?

We report the results in the bar chart in Fig.4.

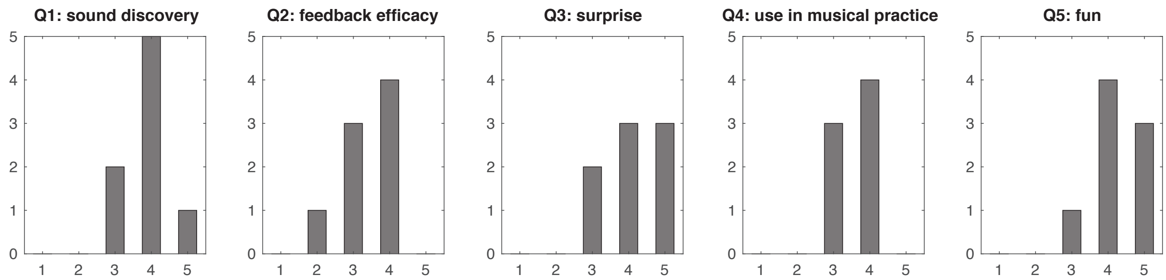

Figure 4: Responses to the feedback questionnaire. Answers were given using a ve-level Likert scale where 1 corresponded to “not at all/strongly disagree” and 5 “yes very much/strongly agree.”

In addition, participants were allowed to leave comments about their experience with the system. Four participant included some comments, two of them reported that they would have liked to have had access to a wider palette of sounds, one that the system was very responsive, and another one reported that they weren’t sure about what the system was doing.

[10]The workshop took place on 20th September 2019 at Music Hackspace – Somerset House Studios in London, UK: https://hdi-network.org/workshop-art-ai-created-content-industrial-cultural-effects/

Discussion

The feedback we collected from this small group of experienced participants (all of them reported being involved in the arts while a few were also developers and academics) led us to make useful considerations regarding the design of the first prototype, as well as on the development of AIML as a general interaction design paradigm.

Firstly, despite scoring high on all the other questions, users entered relatively low scores for the question regarding the efficacy of the feedback given to the agent. This may indicate that the mappings proposed by the agent were fun to play and allowed the participants to explore the different ways of interacting with the synth, yet the immediate effects of giving positive or negative feedback to the agent were not always clear. As also noted by Scurto et al. (2019), for the agent to learn quickly from a small amount of feedback data provided by the users is a challenge. Despite the improved performance of recent reinforcement learning algorithms making use of deep neural networks, the system requires multiple iterations to learn from the user's feedback, especially when dealing with high-dimensional feature spaces. This is particularly challenging to the context of a short demo where participants spent a short amount of time playing with the system. The guiding feedback given to the agent is binary: yes or no. The proposed gestural interactions with the synth are often characterised by complex timbral articulations. Several participants suggested that they would have liked to be able to tell the agent *what* they liked about the mapping. This suggests that a more sophisticated way of giving feedback to the agent might be a useful feature of an AIML system. However, other participants appreciated the immediacy and simplicity of a binary feedback design, which allowed to quickly try several mappings and save those considered useful for later use. This brings to mind the design dilemma often discussed in music technology and other creative domains: sophistication and access to a high number of features vs simplicity and designed limits. The topic has been recently addressed also by the multimedia artist and music technologist Robert Henke, who discussed how the different mindsets and priorities of composers, performers, and software developers affect the design of musical tools, and described how he sees constraints and limitations as something useful in his own creative practice (Henke, 2016). This is indeed a wider, highly subjective topic of discussion, and addressing it is beyond the scope of this paper. It is however important to keep such issues in mind when developing the AIML paradigm further, given that what might be useful to improve the capabilities of an AIML system might not be what makes it more valuable in the context of actual music creation. This became more evident when the system was used by the first author for the development of the multimedia performance piece "You have a new memory" (Visi, 2020). Even though – compared to system shown in the demo – the sound synthesis engine used in this case was more sophisticated, the mappings obtained by interacting with the agent through the workflow described above resulted in a rewarding creative process. It is felt that more ways of giving feedback to the agent might have shifted the focus away from attentive listening and exploration of the sonic interactions. This is, once again, highly subjective and depends very much on one's creative goals.

Higher feedback on Q1 and Q3 suggest that the system has been perceived as a useful tool for exploration and discovery. A defining characteristic of the architecture we propose is that the sound synthesis space is shaped by the presets defined by the user, while the agent provides mappings between that and the input motion features. Yet, even though the synthesis anchor points are defined explicitly, the articulations between them can be very diverse and complex. Exploring such articulations through the mappings proposed by the agent allows for the discovery of sonic gestures within the synthesis space defined by the anchor points. In other words, it is like discovering different ways of performing the same sound synthesis material. We argue this has considerable musical usefulness. Varying the same source material – whether notes or synthesis parameters – is a well-established process in music making, and it is also at the centre of several other implementations of machine learning in music production, such as for example the Magenta Studio[11] plugins.

The approach we described differs from the typical IML workflow as it does not include a gesture design phase, and also differs from the study by Scurto et al. (2019) since the agent does not explore the sound synthesis parameter space. This does not mean that the approaches cannot be combined. Automated sound synthesis parameter exploration techniques can potentially be employed in the sound design phase, while mappings saved while interacting with the agent can be recalled and improved by providing additional input sample data as it is typical in the IML workflow. We therefore see AIML as a way of extending and aiding established gestural interaction design practices.

[11] https://magenta.tensorflow.org/studio

Conclusions and Future Work

We presented an interaction design approach that uses artificial agents and machine learning to interactively explore mappings between gestural input and sound synthesis. We refer to this model as Assisted Interactive Machine Learning. We implemented this paradigm in a prototype system that uses reinforcement learning and linear regression to obtain mappings between accelerometer data and a sample-based synthesiser while playing the instrument. The feedback given to the artificial agent allows to guide its exploration, thus affecting the mappings it proposes after each iteration.

After testing the prototype system internally, presenting it to an expert audience to collect feedback, and employing for the development of a multimedia performance piece, we argue that this model constitutes a useful creative tool for discovering musical interactions while actually playing the instrument. Moreover, AIML can in principle be combined with established interaction design and parameter exploration techniques, and thus be included easily in the workflow of practicing musicians and multimedia artists.

As discussed in the previous section, currently the guiding feedback given by the user to the audience is a simple binary response on the last proposed mapping. Even though the simplicity of the current version allows for a quick and intuitive interaction with the agent, a more sophisticated way of giving feedback may lead to a more rewarding experience with the system. For instance, the agent is currently agnostic of the sounds designed by the user and the output sound made while performing. By implementing machine listening and audio feature extraction techniques, the user could potentially give feedback to the agent regarding some basic timbral features of the output sound, similarly to systems that use audio analysis and genetic algorithms to define a target sound and program a synthesiser automatically (Dahlstedt, 2001; Yee-King et al., 2018). This way, the feedback given to the agent would include different weights for different timbral features. Additionally, the architecture of the current AIML prototype allows for static regression (Tanaka et al., 2019) and not for temporal modelling, which would allow interactions with more diverse dynamics.

Despite its limitations, it is worth keeping in mind that the simplicity of the current version affords a quick, fun workflow that was perceived as useful from the start, and that led to a rewarding creative process during the development *"You have a new memory"* (Visi, 2020). For the purpose of gathering further user feedback data, and to study how musicians would use an AIML system in their actual practice, we are aiming at carrying out a longitudinal study with a small group of professional musicians. This would allow us to situate the development of AIML systems in broader musical contexts, and thus gain insight that would be very difficult to obtain otherwise.

Grounding the research in music practice will also help with studying different ways of giving feedback to the agent and address other open design questions. For this reason, we are aiming at designing an AIML instrument using a specific input device and synthesis engine. A full system can be consistently studied and iteratively improved, by, for example, select the synthesis parameters that are exposed to the agent, try different ways of giving feedback and include other interaction design paradigms in the workflow to refine the mappings proposed by the agent. As with the longitudinal study, this will allow to spend more time with a consistent system and expose its affordances and constraints more clearly, and thus lead to a better understanding of the *ergodynamics* Magnusson (2019) of AIML as a sonic interaction design paradigm situated in music practice.

Acknowledgements.

The *GEMM))) Gesture Embodiment and Machines in Music* research cluster is funded by Luleå University of Technology, Sweden.

The research carried out at Goldsmiths, University of London leading to these results has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 789825 – Project name: BioMusic).

References

Altavilla, A., B. Caramiaux, and A. Tanaka (2013). Towards Gestural Sonic Affordances. In W. Yeo, K. Lee, A. Sigman, J. H., and G. Wakefield (Eds.), *Proceedings of the International Conference on New Interfaces for Musical Expression*, Daejeon, Republic of Korea, pp. 61–64. Graduate School of Culture Technology, KAIST. 1

Dahlstedt, P. (2001). Creating and exploring huge parameter spaces: Interactive evolution as a tool for sound generation. In *Proceedings of the 2001 International Computer Music Conference*, pp. 235–242. 2, 8

Fiebrink, R. and P. R. Cook (2010). The Wekinator: a system for real-time, interactive machine learning in music. In *Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010)*, Volume 4, pp. 2005. 2

Fiebrink, R., D. Trueman, and P. Cook (2009). A metainstrument for interactive, on-the-fly machine learning. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, Pittsburgh, PA, USA, pp. 280–285. 2, 3

Fiebrink, R. A. and B. Caramiaux (2018, 2). *The Machine Learning Algorithm as Creative Musical Tool*, Volume 1. Oxford University Press. 2

Françoise, J., N. Schnell, and F. Bevilacqua (2014). MaD: mapping by demonstration for continuous sonification. In *ACM SIGGRAPH 2014 Emerging Technologies on - SIGGRAPH '14*, New York, New York, USA, pp. 1–1. ACM Press. 2

Henke, R. (2016). *Give me limits! Two perspectives on computer based music production in a time of exponentially growing possibilities.* Montreal, QC, Canada: CIRMMT - McGill University. 7

Horner, A., J. Beauchamp, and L. Haken (1993). Machine Tongues XVI: Genetic Algorithms and Their Application to FM Matching Synthesis. *Computer Music Journal 17*(4), 17. 2

Hunt, A. and M. M. Wanderley (2003, 1). Mapping performer parameters to synthesis engines. *Organised Sound 7*(02), 97–108. 1

Magnusson, T. (2019). Introduction: On Objects, Humans, and Machines. In *Sonic Writing.* Bloomsbury Academic. 8

Margraf, J. (2019). *Self-Organizing Maps for Sound Corpus Organization.* Master's Thesis. Audiokommunikation – Technische Universität Berlin. 4

Miranda, E. R. (1995). Granular Synthesis of Sounds by Means of a Cellular Automaton. *Leonardo 28*(4), 297. 2

Nymoen, K., R. I. Godøy, A. R. Jensenius, and J. Torresen (2013, 5). Analyzing correspondence between sound objects and body motion. *ACM Transactions on Applied Perception 10*(2), 1–22. 1

Östersjö, S. (2016, 9). Go To Hell : Towards a Gesture-Based Compositional Practice. *Contemporary Music Review 35*(4-5), 475–499. 1

Parke-Wolfe, S. T., H. Scurto, and R. Fiebrink (2019, 6). Sound Control: Supporting Custom Musical Interface Design for Children with Disabilities. In M. Queiroz and A. X. Sedó (Eds.), *Proceedings of the International Conference on New Interfaces for Musical Expression*, Porto Alegre, Brazil, pp. 192–197. UFRGS. 3

Scurto, H., B. Van Kerrebroeck, B. Caramiaux, and F. Bevilacqua (2019, 7). Designing Deep Reinforcement Learning for Human Parameter Exploration. *ArXiv Preprint*. 2, 4, 5, 7

Tanaka, A., B. Di Donato, M. Zbyszynski, and G. Roks (2019, 6). Designing Gestures for Continuous Sonic Interaction. In M. Queiroz and A. X. Sedó (Eds.), *Proceedings of the International Conference on New Interfaces for Musical Expression*, Porto Alegre, Brazil, pp. 180–185. UFRGS. 3, 8

Van Nort, D., M. M. Wanderley, and P. Depalle (2014, 9). Mapping Control Structures for Sound Synthesis: Functional and Topological Perspectives. *Computer Music Journal 38*(3), 6–22. 1

Visi, F., R. Schramm, and E. Miranda (2014). Gesture in performance with traditional musical instruments and electronics. In *Proceedings of the 2014 International Workshop on Movement and Computing - MOCO '14*, New York, NY, USA, pp. 100–105. ACM Press. 1

Visi, F. G. (2020). "You have a new memory". In *International Conference on Live Interfaces.* 4, 7, 8

Warnell, G., N. Waytowich, V. Lawhern, and P. Stone (2018, 9). Deep TAMER: Interactive agent shaping in high-dimensional state spaces. In *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pp. 1545–1553. 2

Wright, M. (2005). Open Sound Control: An enabling technology for musical networking. *Organised Sound 10*(3), 193–200. 4

Yee-King, M. J. (2016). The Use of Interactive Genetic Algorithms in Sound Design : A Comparison Study. *ACM Computers in Entertainment 14*(3). 2

Yee-King, M. J., L. Fedden, and M. D'Inverno (2018, 4). Automatic Programming of VST Sound Synthesizers Using Deep Networks and Other Techniques. *IEEE Transactions on Emerging Topics in Computational Intelligence 2*(2), 150–159. 2, 8

Zbyszynski, M., B. Di Donato, and A. Tanaka (2019). Gesture-Timbre Space: Multidimensional Feature Mapping Using Machine Learning & Concatenative Synthesis. In *14th International Symposium on Computer Music Multidisciplinary Research (CMMR)*, Marseille, France. 4

Zbyszyński, M., M. Grierson, and M. Yee-King (2017). Rapid Prototyping of New Instruments with CodeCircle. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, Copenhagen, Denmark, pp. 227–230. Aalborg University Copenhagen. 3