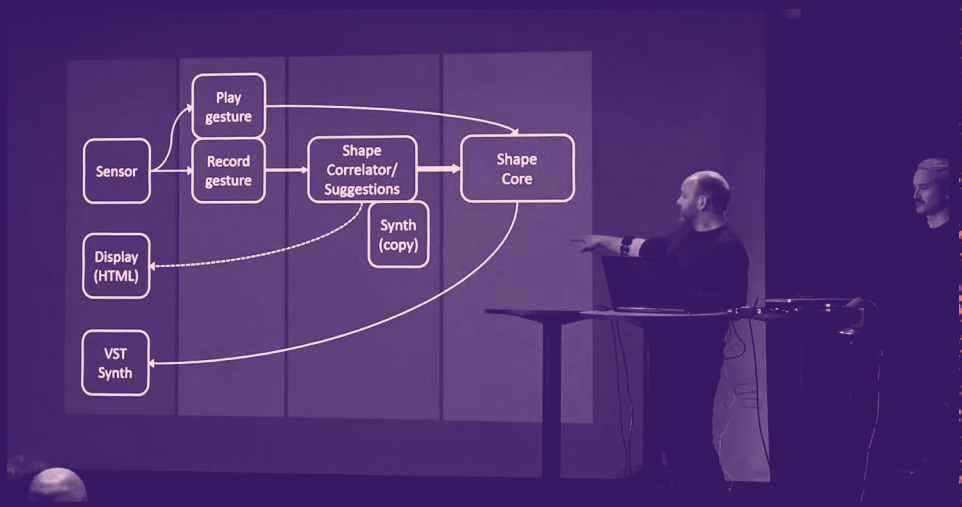
Øyvind Brandtsegg and Axel Tidemann

Shape: an adaptive musical interface that optimize the correlation between gesture and sound

Colloquial Paper

Video of presentation can be found here



{*Shape*: an adaptive musical interface that optimizes the correlation between gesture and sound

Øyvind Brandtsegg[1] and Axel Tidemann[2]

[1]Department of Music, Norwegian University of Science and Technology, Trondheim, Norway, oyvind.brandtsegg@ntnu.no
[2]AI and Analytics, Telenor Research, Trondheim, Norway, axel.tidemann@gmail.com

Abstract. The development of musical interfaces has moved from static to malleable, where the interaction mode can be designed by the user. However, the user still has to specify which input parameters to adjust, and inherently how it affects the sound generated. We propose a novel way to *learn* mappings from movements to sound generation parameters, based on inherent features in the control inputs. An assumption is that any correlation between input features and output characteristics is an indication of a meaningful mapping. The goal is to make the user interface evolve with the user, creating a unique, tailor made interaction mode with the instrument.

Keywords. Gesture sensing, HCI, Artificial intelligence, Machine learning, Machine aesthetics

Introduction

The problem of mapping realtime performance data from an input device to control the parameters of a synthesis engine is common in digital music instrument design (Hunt and Wanderley, 2002). Designing explicit mappings allows detailed and precise control over the relationship between input device and synthesis engine, but we also recognize that this is a complex process. Using generative mechanisms such as neural networks allow for ways of managing this complexity. This has been done e.g. by Lee and Wessel (1992), Modler (2000), Fiebrink and Cook (2010), Martin and Torresen (2019) and Visi and Tanaka (2020). In the current implementation, we have used a gestural input device (Myo armband), although the methods could easily be adapted to any input device.

Gesture to audio interfaces has a long history, with one seminal work being Michel Waisvisz's "The Hands" (Torre et al., 2016), with a plethora of variations and methods shown in Wanderley and Depalle (2004), and a more recent artistic example the Strophonion (Nowitz, 2019).

One way of managing mapping complexity is to aim for a strong connection between the gestural qualities of a movement and the resulting perceptual parameters of the generated sound. This has been researched e.g. by Metois (1996). The perceptual qualities *volume*, *pitch* and *timbre* are commonly used, where the first two are relatively easy to describe but the last is much more ambiguous. Different approaches to defining and parametrizing timbre has been made by e.g. Schaeffer (1966), Smalley (1986), Wessel (1979), Bernays and Traube (2011) and others. Part of the problem may lie in the fact that timbre is a complex combination of temporal and spectral characteristics of a sound, but it is also related to the perceptual aspect. Perception happens in the individual, and as such also involves both psychological and philosophical considerations. An interesting recent approach to tackling these challenges can be seen in Magda Mayas' "Orchestrating Timbre" (Mayas, 2019) where perceptual timbre maps are created based on the personal artistic practice of the musician herself. As Mayas demonstrate, timbre can be adequately approached by a personal experiental approach in how it relates to nuanced performative expression on a musical instrument. Thus, it seems to make sense to take an experimental and explorative approach in designing new musical instruments with correlated perceptual mappings. Such an approach has been explored also by Visi and Tanaka (2020), Dahlstedt (2001) and others. In the current work with the instrument "{*Shape*", we have attempted to combine this approach with a mechanism to optimize the perceptual correlation between gesture and sound by means of feedback from audio analysis of the synthesizer output. Mappings with a higher correlation between trajectories of gestural and audio analysis features are presented to the user for further exploration. A noteworthy difference between

{Shape and many machine learning based mapping approaches is that *{Shape* does not require any user specification of target synthesis parameters. Rather, the system suggests these parameter values based on an analysis of the audio output from the synthesizer, effectively bootstrapping the mapping into existence. Our approach is to learn the mappings without having to specify anything else than a demonstration of a few starting gestures. These mappings are not static, they evolve with the user. This can free up the whole interaction design process by making it an inherent part of playing with the instrument. The cost is the time spent learning and evolving with the instrument itself.

Related work

The system has some functional similarities to the Gesture Variation Follower (GVF) by Caramiaux et al. (2014) in that it allows scaling and time variations to be induced from the performed deviations from learned gestures, and that these variations are used to align the mapping from input to output. Both GVF and our current system can do early recognition, and also respond to expressive deviations from learned gestures. GVF has outputs for scaling, rotation and time deviation. These can be used as transformative vectors in the mapping. *{Shape* achieves this by enabling a more dynamic mapping based on deep learning (LeCun et al., 2015). This is different from GVF, which uses a statistical approach. In our system, the time dimension is implicitly handled by the use of convolutional neural networks, which will be elaborated in the next section.

The mappings from gestural input to sound synthesis parameters are done in an auto-adaptive and generative manner in our system, based on analysis of the gestural qualities in the input. The algorithm attempts to produce an output sound that preserves the gestural qualities of the input without prior knowledge of the synthesis engine. A combination of classification and regression is used, to enable smooth transitions between the different multidimensional mappings learned by the system.

To seamlessly morph between sounds using neural networks has received attention lately, in particular with the WaveNet-based approach by Engel et al. (2017). However, their work was on morphing between instruments by learning from raw waveforms, and not being linked to another modality. The introduction of generative adversarial networks (GANs) by Goodfellow et al. (2014) has also seen applications within the audio domain, namely by creating more sophisticated audio by generating a lot of the audio properties (e.g. log-magnitude spectrograms) through the GAN approach (Engel et al., 2019). This mixture of neural networks and raw audio generation is hampered by computing power, but nevertheless interesting for mimicking real sounds - however, in this work we focus more on the creative applications that arise through synthesis.

The system

The system has separate operational modes for learning and performing. While these modes can be operational simultaneously, we have opted to separate them with manual control of mode switching in this initial implementation of the system. An overview of how the system works is shown in Figure 1, and will be elaborated in the following sections. Source code is available at the author's GitHub[1].

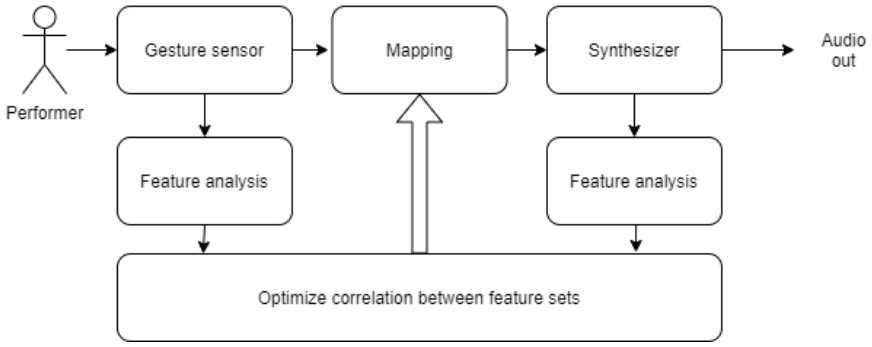


Figure 1: Simplified signal flow of the system

[1] github.com/Oeyvind/shape

Learn mode

The system starts in a "tabula rasa" state, and has to build up a library of movements and their mappings to synthesis parameters. The system is guided in this phase through a simple interaction mechanism with the user. Upon completion of a gesture, the system creates suggestions for parameter mappings that result in sounds that have similarities between the audio and gesture domain. The initialization of the mappings is done by randomly selecting one of the gesture input axes, multiplying it by a random number in the range $[0, 1]$ and then using the resulting vector as a synthesis parameter. This process is repeated for each of the synthesis parameters.

After the calculation of parameters and generation of sounds, the sounds are analyzed and sorted by which sounds are most similar to the gestural input axes. This is achieved by calculating the mean squared error between the different axes of the gesture and the various audio features that are analyzed after the sound has been created. For each gesture input axis, the most similar audio feature contributes to the mapping's overall similarity score. All the generated sounds are sorted based on the similarity score.

The user is subsequently presented the sorted list, and selects a preferred mapping. The system then learns the correlation between gesture input and synthesis parameters by using a neural network. There is a 1:1 relation between the sampling rate of the gesture input and the update frequency of the mapping, while the synthesis parameters are interpolated in between updates. The neural network is also trained to classify the gesture at each time step. Both the regression (i.e. output of synthesis parameters) and classification (of the gesture) are achieved by performing a 1D convolution over all the axes of the input signal with a specified history length. A recurrent neural network could also be used, like a bidirectional LSTM (Hochreiter and Schmidhuber, 1997). However, convolution neural networks are faster to train and more robust to noisy sequences, since they are translation invariant with respect to the input signal.

Play mode

When the system is in play mode, it continuously predicts the synthesis parameters and classifies the gesture. Note that the history length directly specifies the "memory" of the network, and therefore the stability of the predictions. This enables a fluid mapping from gestures into synthesis parameters and finally, sound.

Independence from application environment

The methods and the system proposed in this paper are independent of the type of gesture sensor, and also independent from the choice of synthesizer or sound producing engine. The system adapts to the set of input parameters that it is trained on, and as such can be used with a variety of gesture sensing technologies or other input devices. Similarly, it adapts to the output parameter set available in the specific synthesizer used in training. Due to the fact that optimization takes place by feature analysis of the sound produced by the synthesizer, the specifications and implementation details of the synthesizer becomes irrelevant to the mapping system. There are however two requirements to the synthesizer: it must have parametric control that can shape the resulting timbre, and for practical reasons it must be able to render sound in an offline fashion. Due to the large number of suggested sounds, the process would be very slow with realtime-only sound production devices. When rendering sound offline, this process can be sped up and automated with parallelization such that the learning process can be accomplished as quickly as computing resources permits. Our current implementation uses the Csound audio programming library for synthesis (csound.com, see also Lazzarini et al. (2016)). In addition to the freely programmable synthesis algorithms, it will also allow running off-the-shelf software synthesizers in plugin format so that the mapping system can be explored with the user's favourite audio generators.

Discussion

The system is able to create interpolations between different trajectories in the high-dimensional space represented by the neural network that embeds the gestures. This enables the system to suggest novel parameter mappings to the user when presented with novel gestures. These will sensibly combine with previously learned gestures and their corresponding synthesis parameters, and it is capable to do so because a deep learning model is used to learn this concept. Interpolations between combinations are most likely not linear, further suggesting the need to have a computationally powerful model that deep learning neural networks provide.

The synthesis parameter controlling perceived pitch is of special relevance in a musical instrument. In the cases where a clear pitch can be perceived, this will often take precedence over other timbral variations. This might mean that synthesis parameters affecting the perceived pitch might need to be given special consideration in generated mappings. In the current implementation, we have opted to make a manual switch to activate or de-activate mapping to these parameters.

The generative aspect of the work contributes to the field of computational creativity. It starts out in a random fashion, without any a priori knowledge of what the user likes. A continuing challenge is to enable a process of gradual development of a mutual vocabulary between the system and the user.

Future work

Since this paper presents a first version of the system, the operating modes are explicitly set by the user. However, a future goal is to make the system automatically select between play or learn modes. This can be achieved by a system that continuously predicts gestures, and is able to to discern when a new gesture is encountered. For instance, this can be achieved by using a deviance tolerance. The deviation tolerance can be set as a numeric value, or we can allow the system to use an adaptive strategy based on distance between the gestures already learned. The deviation tolerance can also be measured by the amount of prediction errors done by the already trained neural network. A combination of these might also be used, where the user also sets an absolute minimum deviation needed. This could help prevent the system from entering learn mode when a large number of gestures have been learned and their representations might start to overlap.

Another idea worth exploring is the evaluation of the suitability of generated mappings. Currently, this is done by the user through an interface. However, the system would operate more seamlessly if this process could be performed by another neural network, which has the role of learning the aesthetic preferences of the user. This neural network can then be trained alongside the selection of generated sounds, and the system learns predictions of what the user likes and dislikes. This could evolve into a crude form of aesthetic artificial intelligence.

A natural immediate prospect will be the exploration of a variety of input devices and different audio synthesis models. The use of a selection of off-the-shelf and well known synthesizers could be enlightening in comparing this mapping approach with other alternatives.

References

Bernays, M. and C. Traube (2011). Verbal expression of piano timbre: Multidimensional semantic space of adjectival descriptors. In *Proceedings of the International Symposium on Performance Science*. 1

Caramiaux, B., N. Montecchio, A. Tanaka, and F. Bevilacqua (2014). Adaptive gesture recognition with variation estimation for interactive systems. *ACM Trans. Interact. Intell. Syst. 4*(4). 2

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, ICMC 2001, Havana, Cuba, September 17-22, 2001*. Michigan Publishing. 1

Engel, J., C. Resnick, A. Roberts, S. Dieleman, D. Eck, K. Simonyan, and M. Norouzi (2017). Neural audio synthesis of musical notes with wavenet autoencoders. In D. Precup and Y. W. Teh (Eds.), *Proceedings of the 34th International Conference on Machine Learning*. 2

Engel, J. H., K. K. Agrawal, S. Chen, I. Gulrajani, C. Donahue, and A. Roberts (2019). Gansynth: Adversarial neural audio synthesis. *CoRR abs/1902.08710*. 2

Fiebrink, R. and P. Cook (2010). The wekinator: A system for real-time, interactive machine learning in music. *Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010)*. 1

Goodfellow, I. J., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio (2014). Generative adversarial nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, Cambridge, MA, USA, pp. 2672–2680. MIT Press. 2

Hochreiter, S. and J. Schmidhuber (1997). Long short-term memory. *Neural Computation 9*(8), 1735–1780. 3

Hunt, A. and M. M. Wanderley (2002, August). Mapping performer parameters to synthesis engines. *Organised Sound 7*(2), 97–108. 1

Lazzarini, V., S. Yi, J. Ffitch, J. Heintz, O. Brandtsegg, and I. McCurdy (2016). *Csound: A Sound and Music Computing System* (1st ed.). Springer Publishing Company, Incorporated. 3

LeCun, Y., Y. Bengio, and G. E. Hinton (2015). Deep learning. *Nature 521*(7553), 436–444. 2

Lee, M. A. and D. Wessel (1992). Connectionist models for real-time control of synthesis and compositional algorithms. In *Proceedings of the 1992 International Computer Music Conference, ICMC 1992, San Jose, California, USA, October 14-18, 1992*. Michigan Publishing. 1

Martin, C. P. and J. Torresen (2019, June). An interactive musical prediction system with mixture density recurrent neural networks. In M. Queiroz and A. X. Sedó (Eds.), *Proceedings of the International Conference on New Interfaces for Musical Expression*, Porto Alegre, Brazil, pp. 260–265. UFRGS. 1

Mayas, M. (2019). *Orchestrating timbre – Unfolding processes of timbre and memory in improvisational piano performance*. Ph. D. thesis, University of Gothenburg. Faculty of Fine, Applied and Performing Arts. 1

Metois, E. (1996). *Musical Sound Information – Musical Gestures and Embedding Systems*. Ph. D. thesis, Massachusetts Institute of Technology. 1

Modler, P. (2000). Neural networks for mapping gestures to sound synthesis. In M. Wanderley and M. Battier (Eds.), *Trends in Gestural Control of Music*. Ircam – Centre Pompidou. 1

Nowitz, A. (2019). Monsters i love: On multivocal arts. Available at: https://www.researchcatalogue.net/view/492687/559938. 1

Schaeffer, P. (1966). *Traité des objets musicaux*. Paris: Seuil. 1

Smalley, D. (1986). Spectromorphology and structuring processes. In S. Emmerson (Ed.), *The Language of Electroacoustic Music*, pp. 61–93. Basingstoke: Macmillan Press. 1

Torre, G., K. Andersen, and F. Baldé (2016). The hands: The making of a digital musical instrument. *Computer Music Journal 40*(2), 22–34. 1

Visi, F. and A. Tanaka (2020). Towards assisted interactive machine learning: Exploring gesture-sound mappings using reinforcement learning. In Ø. Brandtsegg and D. Formo (Eds.), *Proceedings of the International Conference on Live Interfaces*, Trondheim, Norway. NTNU. 1

Wanderley, M. M. and P. Depalle (2004). Gestural control of sound synthesis. *Proceedings of the IEEE 92*(4), 632–644. 1

Wessel, D. L. (1979). Timbre space as a musical control structure. *Computer Music Journal 3*(2), 45–52. 1