Towards Playing in the 'Air': Modeling Motion–Sound Energy Relationships in Electric Guitar Performance Using Deep Neural Networks

Çağrı Erdem
RITMO, Department of Musicology
University of Oslo
cagri.erdem@imv.uio.no

Qichao Lan
RITMO, Department of Musicology
University of Oslo
qichao.lan@imv.uio.no

Julian Fuhrer
RITMO, Department of Informatics
University of Oslo
julianpf@ifi.uio.no

Charles Martin
Research School of Computer Science
The Australian National University
charles.martin@anu.edu.au

Jim Tørresen
RITMO, Department of Informatics
University of Oslo
jimtoer@ifi.uio.no

Alexander Refsum Jensenius
RITMO, Department of Musicology
University of Oslo
a.r.jensenius@imv.uio.no

ABSTRACT

In acoustic instruments, sound production relies on the interaction between physical objects. Digital musical instruments, on the other hand, are based on arbitrarily designed action–sound mappings. This paper describes the ongoing exploration of an empirically-based approach for simulating guitar playing technique when designing the mappings of 'air instruments'. We present results from an experiment in which 33 electric guitarists performed a set of basic sound-producing actions: impulsive, sustained, and iterative. The dataset consists of bioelectric muscle signals, motion capture, video, and audio recordings. This multimodal dataset was used to train a long short-term memory network (LSTM) with a few hidden layers and relatively short training duration. We show that the network is able to predict audio energy features of free improvisations on the guitar, relying on a dataset of three distinct motion types.

1. INTRODUCTION

Playing 'in the air' can be seen as a way of music appreciation [1], and also has potential for music expression [2]. But when you play an 'air instrument'—for example, the 'air guitar'—what are you actually playing? What kind of sound is supposed to be produced, and which strategies can be used in the design of such mappings? Our approach is based on the idea of letting the action–sound *couplings* found in acoustic instruments guide the design of action–sound *mappings* in a digital musical instrument [3]. The aim is not to recreate the action–sound couplings of (electro)acoustic guitar performance directly, but rather let them inspire the mappings in a new 'air instrument'.

There are several examples of air guitar instruments based on fairly coarse body movement, such as, the Virtual Air Guitar [4] and the Virtual Slide Guitar [5]. There are also more recent examples of using deep learning and computer vision to map between fingers and tones, for example, the 'deep air guitar' framework [6]. Such a camera-based approach is less useful in a performance scenario, since it is so dependent on the placement of the camera.

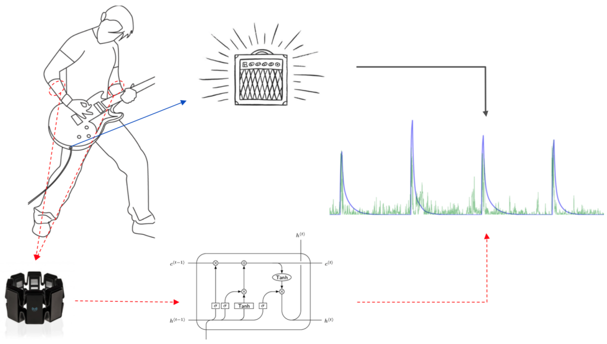

Figure 1. Overview of the data collection used in the study.

An alternative to using cameras to look at hands or fingers, is to use muscle information as the input of an air instrument. The muscle signals on the forearms are closely related to the finger movement, and the muscle signals can be measure by technologies such as electromyography (EMG) [7]. This approach is promising, and affordable muscle-sensing devices (such as the Myo) have been widely used in digital musical instrument designs [8]. Working with the muscle signals is not trivial, however, and often results in arbitrary mappings between action (captured as muscle signals) and the generated sound. In this paper, we therefore ask the question:

- How can we model relationships between action and sound in guitar playing, using muscle-sensing as the input?

This question has been broken down to two main challenges that will be presented in the following: (1) building a dataset that can be used for machine learning, and (2) developing a model based on the dataset. We first introduce the background of action–sound analysis and the application of machine learning in music performance. Then we elaborate on the data collection and the tools used for recording the dataset. Finally, we describe the model architecture and discuss the results.

2. BACKGROUND

2.1 Music-related body motion

Imagine a guitarist playing a song. For each chord to be strummed, the guitarist would lift a limb upwards, and move it back downwards to hit the strings. This process relies on *motion* and *force*. The former is defined as the continuous displacement of a limb or an object in space over time, while the latter refers to the push or pull experiences during the interaction. Force can set an object into motion, and motion can lead to the experience of force. While these can be objectively measured using a range of sensing devices (see, for example, [9] for an overview of different methods for sensing music-related body motion), we reserve the term *action* to what can be described as the goal-oriented *chunking* of such continuous physical phenomena, what Godøy and Leman refers to as 'cognitive units' [10].

2.2 Sound-producing actions

There are several types and categories of music-related body motion [11], but in this context we will primarily focus on the *sound-producing actions* that are responsible for note production. These can be further divided into *excitation* actions, such as the right hand that excites the strings, and *modification* actions, such as the left hand modifying the pitch. The excitation can be further divided into three main categories [12] (as sketched in Figure 2):

- *Impulsive*: fast attack, discontinuous energy transfer
- *Sustained*: gradual onset, continuous energy transfer
- *Iterative*: series of discontinuous energy transfer

Musical performances typically combine all these types in expressive ways. Drawing on such a conceptual apparatus, we can however assume the continuous music-related body motion/force as a series of goal points, which, when temporally close enough, can overlap and become *coarticulated* [13]. In other words, we can think of an entire instrumental performance in terms of such coarticulated combinations of the three aforementioned motion types.

2.3 Action–sound couplings and mappings

The relationships between action and sound in acoustic instruments are dictated by the laws of physics, and can be thought of as *action–sound couplings* [14]. However, digital musical instruments (DMIs) are based on the creation of *action–sound mappings*, in which the relationships between the physical energy of the input action may not necessarily correspond to that of the output sound. The creation of meaningful action–sound mappings in digital musical instruments is therefore critical for how they are perceived [15], and has been a central topic in the field of new interfaces for musical expression (NIME) over the last decades [16].

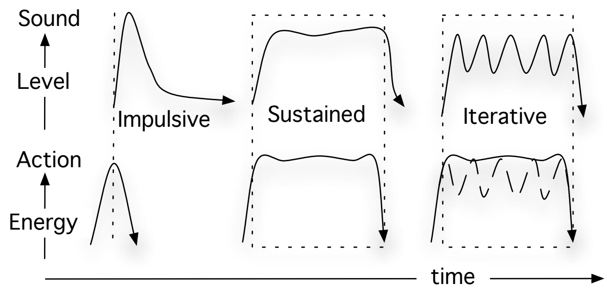

Figure 2. Illustration of the three basic action–sound types: impulsive, sustained, and iterative (from [14]).

2.4 Machine learning in mapping design

Machine learning has been a part of NIME design since the early 1990s [17]. Well-known examples include the *Wekinator* [18], Gesture Follower [19], ml.* library [20], Gesture Recognition Toolkit (GRT) [21], Gesture Variation Follower (GVF) [22], and ml.lib [23]. These (and other) tools allow for using machine learning algorithms through either a graphical user interface (GUI), or, in the form of external libraries for audio programming platforms, such as Max/MSP and Pure Data. A number of new musical interfaces have employed such systems, such as, Snyder's *The Birl* [24], Kiefer's use of Echo State Networks (ESNs) [25], and Schacher and colleagues' *Double Vortex* [26].

In recent years there have been an increasing interest in applications of deep neural networks (DNNs) for symbolic music generation or audio modelling. There are fewer musical examples of physical interaction (see, for example, [27] for an overview of deep predictive models in interactive music). A recent interactive music framework for deep learning is *IMPS*, which uses a mixture density network (MDN) over LSTM layers, and provides a low-entry-fee for musical exploration of DNNs [28]. Within instrument design, Gregorio's intelligent mapping structure [29], and McCormack et al's human–machine collaborative improvisation system [30] are some of the recent works.

2.5 Conceptual Idea

The central idea of this project is to investigate the action–sound couplings found in electric guitar performance, and use these for the creation of action–sound mappings in interfaces that do not rely on a physical controller. This can be thought of as the creation of technologies that allow for sonic interaction in the 'air' [2]. Previous research on the topic has primarily focused on capturing 'overt' motion, using optical or inertial motion tracking devices. The challenge is, then, how to exert effort while the haptic feedback of a physical interface is not available. To tackle this issue, we explore how the 'covert' muscle signals related to physical motion can be used for such interaction, in which the authors have been working on artistic-scientific projects in the recent years ([**?**, **?**]). Thus, we focus on electromyographic (EMG) signals that represent the electrical activity of muscles [31].

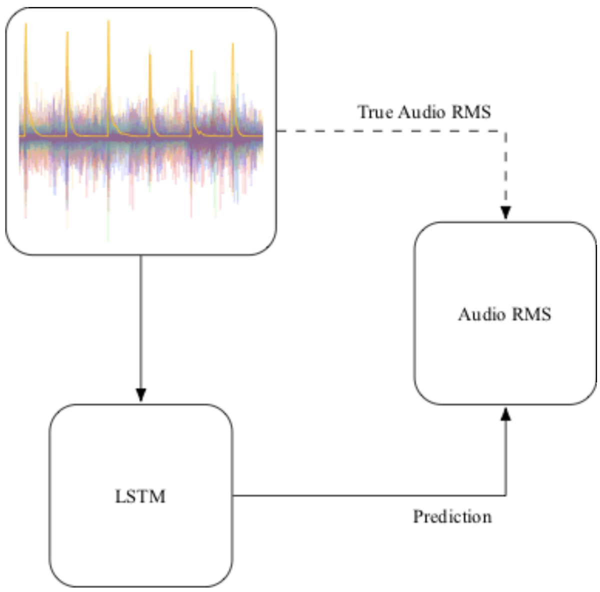

Figure 3. Sketch of the model: Raw EMG data and audio RMS are input to the neural network. The network (LSTM architecture) then outputs a predicted audio RMS.

The idea is to create a model of relationships between extracted muscle activity and sound features. The model is trained on raw EMG signals and the RMS of the resultant sound. Finally, the system is tested with the EMG input from freely improvised recordings.

3. DATA COLLECTION

3.1 Participants

A total of 36 music students and semi-professional musicians took part in the study, three of which were excluded due to incomplete data. Thus, our dataset consists of data from 33 participants (32 male, 1 female, mean age and standard deviation is 27 ± 7 years). All the participants had some formal training in playing the guitar, ranging from private lessons to university-level education. The recruitment was done through an online form published on the web site of the University of Oslo, which was announced in various communication channels targeting music students. Participation was rewarded with a gift card (valued approx. €30). The study obtained ethical approval from the Norwegian Centre for Research Data (NSD), with project number 872789.

3.2 Apparatus

Recordings took place in the fourMs motion capture lab at the University of Oslo. We recorded the audio at 16-bit 48 kHz using an Universal Audio Apollo Twin audio interface. All participants used the same performance setup: A Sadowsky Semihollow guitar with 11-49 gauge roundwound strings, a 1.5mm Jim Dunlop Tortex plectrum, a Roland AC-40 acoustic guitar amplifier (clean tone with all-flat equalizer settings) connected into the audio interface through the line output. The sound level was set to be comfortably loud for the participant.

We recorded the participants' muscle activity as surface EMG with two systems: consumer-grade Myo armbands and medical-grade Delsys Trigno. The former has a sample rate of 200 Hz, while the latter has a sample rate of 2000 Hz. Overt body motion was captured with a twelve-camera Qualisys Oqus infrared, optical motion capture system at a frame rate of 200 Hz. This system tracked the three-dimensional positions of reflective markers attached to each participant's upper-body and instrument. A trigger unit was used to synchronise the Qualisys and Delsys Trigno systems. We have also developed our own software for recording data from the Myo armband in synchrony with the audio (see Section 3.4). Regular video was recorded with a Canon XF105, synchronised with the Qualisys motion capture system.

For the current paper, only EMG data from the Myo will be considered, since the aim is to use the trained model in performance. Data from the Delsys system, as well as the motion capture and video recordings, have been used for reference only.

3.3 Procedure

The participants were recorded individually and were asked to perform warm-up, four specific performance tasks, and a final improvisation:

0. A warm-up improvisation with metronome at 70 bpm
1. Task 1
   (a) Softly played *impulsive* notes
   (b) Strongly played *impulsive* notes
2. Task 2
   (a) Softly played *iterative* 16$^{\text{th}}$ notes
   (b) Strongly played *iterative* 16$^{\text{th}}$ notes
3. Task 3
   (a) Softly played hammer-ons and pull-offs
   (b) Strongly played hammer-ons and pull-offs
4. Task 4
   (a) Softly played *sustained* semi-tone bending
      i. 'As fast as possible'
      ii. 'As slow as possible'
   (b) Strongly played *sustained* semi-tone bending
      i. 'As fast as possible'
      ii. 'As slow as possible'
5. A free improvisation (the tone features and the use of metronome are at the participant's discretion)

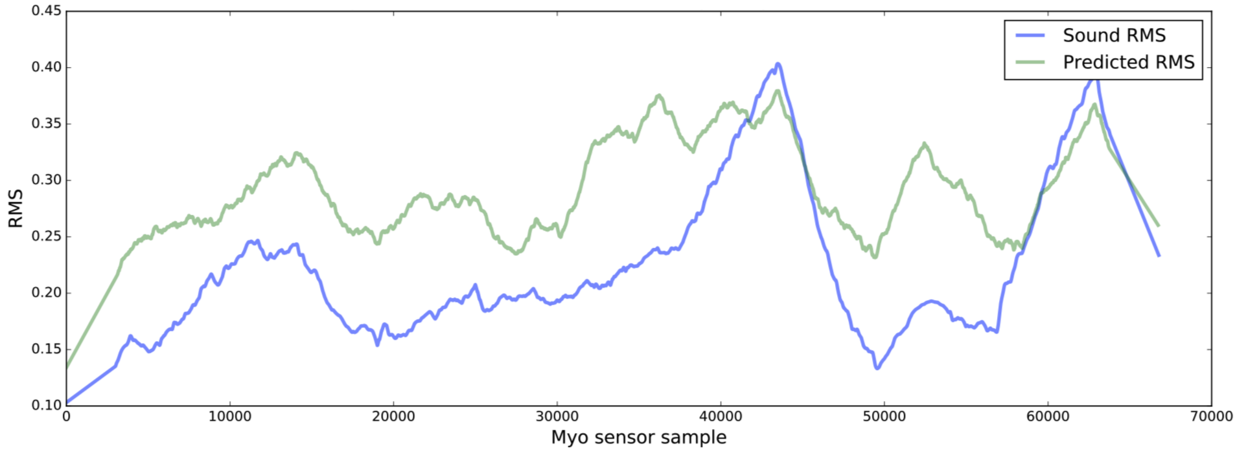

Figure 4. The RMS of the recorded sound and the model prediction. Both curves are processed with a Savitzky-Golay filter to reflect the general shape of the RMS comparison.

Figure 5. A participant during the recording session. Motion capture cameras can be seen hanging in the ceiling rig behind, and on stands in front of, the performer. The monitor with instructions can be seen below the front left motion capture camera.

Figure 6. Placement of the EMG sensors on the arms of the guitarists. Two delsys EMG sensors were placed on each side of the arm, right below the Myo armbands.

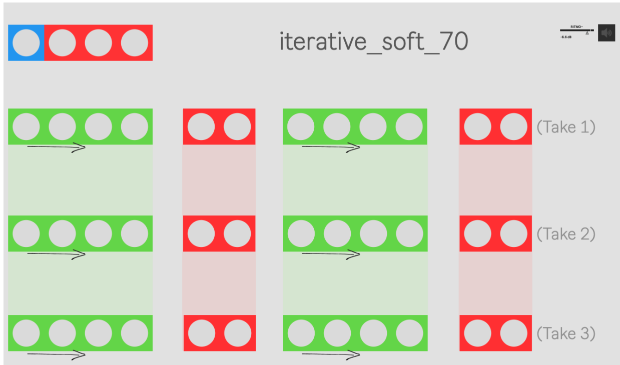

Figure 7. A screenshot of the 'prompter' that the participants would see on the screen in front of them during the experiment.

All the given tasks (1–5) focused on the notes B3 and C4 on the 4^{th} (D) string played by index and middle fingers. Each task was recorded as a fixed-form track of duration 2'16", where participants were instructed to play for 4 bars, rest for 2 bars, and repeat the same pattern for 5 more times. All tasks are prompted through a Max/MSP patch on a screen (Figure 7), which allowed for a consistent and efficient experiment process.

3.4 Data Acquisition

We built a custom Python interface to record synchronised sensor data and audio. This was using our previously developed *myo-to-osc* bridge [32], which implements low-latency support for multiple Myo armbands connected via individual Bluetooth Low Energy (BLE) adapters. This is necessary to overcome possible bandwidth limitations. The latency can also be documented and eliminated after the recording. [1]

The data acquisition interface contains three parts: (1) data collection from the two Myo armbands, (2) generation of a metronome sound for the performers, and (3) au-

[1] `https://github.com/chaosprint/dual-myo-recorder`

dio recording using *PyAudio*. Audio and metronome timeline information was captured alongside the EMG data to simplify the segmentation and organisation of the training dataset.

3.5 Post-processing the data

To prepare data for our model, we first aligned EMG and audio arrays based on the recorded metronome timeline, then we applied interpolation on the EMG data and calculated the root mean square (RMS) from the audio signal.

3.5.1 Interpolation of the EMG data

The data recorded from Myo armbands needs to be preprocessed before it can be used for further analysis. This is to compensate for noise and possible data loss during recording. Here we solved this by performing a linear interpolation on the data. Since the data was recorded at a frequency of 200 Hz, the data loss is usually not more than a few samples. Thus, this additional step to account for the lost data should not create much of an error.

3.5.2 Root Mean Square of the audio signal

The root mean square (RMS) was calculated to reduce the dimension of the discrete signals and to characterise the signal. The RMS of a discrete signal $x = (x_1, x_2, \ldots, x_n)^\intercal$ with n components is defined by

$$\mathrm{RMS} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2} = \sqrt{\frac{x_1^2 + x_2^2 + \cdots + x_n^2}{n}} \,. \qquad (1)$$

Even though it is a simple measure, the RMS can be argued to have both physical and perceptual significance. Its physical significance is related to the proportionality to the effective power of the signal. On average, one could argue that RMS is also correlated to perceptual loudness. The brain can judge whether a signal is loud, soft or in between, but it cannot infer where a periodic signal is peaking or is at a zero-crossing [33, 34]. Thus, for our purposes, RMS is a better feature than simply taking just the peak value within a given time interval.

4. DEVELOPING A MODEL

The aim of the model is to map the EMG data (raw muscle signals) to the RMS of the instrument's audio signal. Concretely, the input to the neural network is every 50 samples of the EMG recorded from all 16 channels of the two Myo devices (e.g. sample N 0 to 49, sample N 1 to 50, etc.). As we use the data from both hands, and each Myo has 8 analogue channels, there are 16 channels for each sample. The output of the neural network is the predicted sound RMS energy on the guitar.

The Long Short-Term Memory (LSTM) recurrent neural network (RNN) model was built in PyTorch [35], a popular model for time-series prediction. [2] As depicted in Figure 3, the LSTM network receives the raw EMG data and audio RMS, which were aligned during the pre-processing, and produces a predicted audio RMS. The training loss function was defined as

$$\begin{aligned} \mathcal{L}(x_{\mathrm{RMS}}, \hat{x}_{\mathrm{RMS}}) &= \frac{1}{n}\|x - \hat{x}_{\mathrm{RMS}}\|_2^2 \\ &= \frac{1}{n}\sum_{i=1}^{n}(x_{\mathrm{RMS,i}} - \hat{x}_{\mathrm{RMS,i}})^2 \,, \end{aligned} \qquad (2)$$

where x_{RMS} are the recorded values, and $\hat{x}_{\mathrm{RMS}}$ are the values to be predicted. The sliding window has size n. The predicted RMS is computed according to Equation 1.

4.1 Training

A relatively small RNN was used for the training, consisting of five hidden layers and with 32 LSTM units in each layer. The window size of the input is 50, which is in line with the size of the input layer that is 50. For training, we used the data (excluding the improvisations) of 15 subjects out of 20 and validated it on the remaining subjects. We chose a batch size of 100 for determining the gradient of the cost function. Typically, at the first 5 epochs, the loss drops quickly and becomes stable after 10 epochs, which takes around 3 hours. Overall, we managed to finalize the training within the 12-hour limit of *Google Colab*'s graphics processing unit (GPU) resources.

4.2 Training result

The model is generally capable of predicting RMS. This can be seen in the figures of the recorded versus predicted RMS of the tasks of playing impulsive notes (Figure 9) and iterative 16$^{\text{th}}$ notes (Figure 10). For the latter, the model can generate a similar consecutive energy shape as series of attacks.

We were also positively surprised to see that the model could predict the general trend of the sound energy in free improvisation tasks (Figure 4). This is the task that is most relevant for our ultimate goal of creating an 'air instrument' to perform with. So we are particularly pleased that the model can, indeed, account for this, at least on a level of the sound envelope.

An interesting result can be seen for the prediction of the bending task (Figure 11). Although we describe three distinct motion types in Section 2.2 (impulsive, sustained, iterative), regular performance on the guitar does not afford sustained motion during the excitation phase (it could be done with a bow on the strings, but not with a plectrum). In other words, one can hit on a string either once (impulsive), or as series of impulses (iterative). However, sustained motion is available for the modification action, such as, bending the string with a finger on the left hand. Therefore in the prediction, we observe a longer decay when compared to impulsive, single attack of the right arm. We think that this is an interesting in-between result, which can be further interpreted as an augmentation of the guitar for creative purposes. In other words, this also shows that the model is promising for expanding the player's control space.

[2] `https://github.com/cerdemo/air_model`

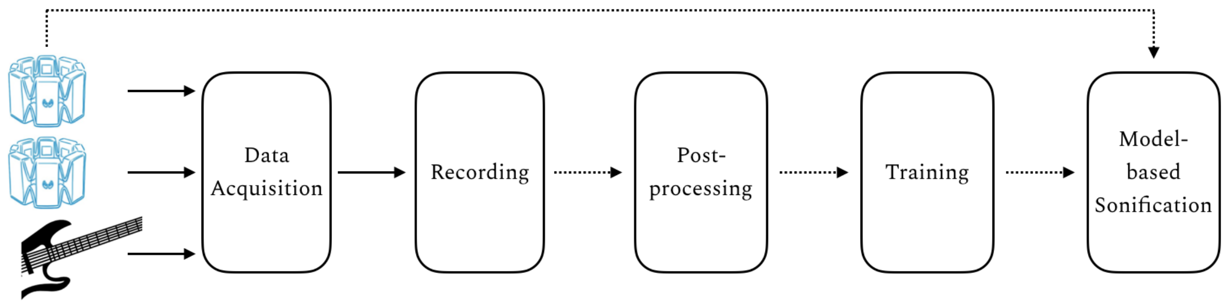

Figure 8. Simplified signal flow diagram of the system.

4.3 Testing the model

The predicted features were tested using a preliminary sonification strategy. Here the trained model was fed with 16 channels of raw EMG test data to generate predicted RMS values. These values were then imported (as CSV files) into a Max/MSP patch that runs through the values at the same rate as the Myo armband (200 Hz). The sonification is built around a simple Karplus-Strong algorithm programmed in the Gen environment within Max/MSP, where the RMS value is mapped to the *decay* and *damping* parameters of the physical model. This effectively 'shapes' the white noise to simulate a (guitar-like) plucked string sound. For further sound design purposes, we use a sine-wave-based low frequency oscillator (LFO), and a fair amount of pitch shifting. This creates a lower octave of the sound that 'feels' similar to sub-frequencies of naturally resonating bodies of acoustic instruments and speakers.

The demonstration video [3] is structured as an alternation between the originally recorded sounds, and an offline sonification that relies on the predicted RMS values mapped to the temporal envelope of the sound synthesis. The onsets are extracted from the predicted values within the Python script, and stored in the CSV file along with the RMS values. In the video, it is easily noticeable that when playing a series of fast attacks during the *iterative* task, onsets of the 'air instrument' often loose the action–sound synchrony. This reveals an important weakness of the strategy. As such, it motivated us towards modelling the entire spectrum of the recorded audio, which will allow more reliable onset detection algorithms based on the spectral flux.

5. CONCLUSIONS

The paper has presented a method for building a neural-network model based on recordings of action–sound couplings from (electro)acoustic guitar performance. We show that the model can predict the overall trend of the sound energy (measured as RMS) of a freely improvised performance, solely based on a training dataset of particular motion types.

As part of the data collection, we had to develop a solution for low-latency recording of multiple Myo armbands, synchronised with audio and metronome. We also developed tools for post-processing the data including an interpolation algorithm to compensate the sample loss happened in Bluetooth transmission. This framework can be applied to the analysis and modelling of action–sound relationships in playing a variety of acoustic and digital instruments. As such, we will openly share our dataset and tools in service of further scientific and artistic studies.

Although no systematic evaluation has taken place, our sonification experiment shows that the trained model can be used reliably to control the 'feel' of an 'air instrument', using only muscle sensing as input. As such, we believe that creating models trained on recorded action–sound couplings from acoustic instruments is a promising strategy for the design of action–sound mappings in DMIs.

Of course, the prediction of a single temporal feature is insufficient for capturing the complexity of musical sound. The next step is therefore to expand the model with spectral, temporal and spatial features. This will allow for a wider and more flexible sound palette in real-time musical settings. Furthermore, how to use the space, how to structure the time, and how to interact with the audience and/or ensemble members while playing muscle-based 'air' instruments, introduce a number of conceptual, practical, and technological challenges. Thus, the relationship between different design considerations and the spatiotemporality of the performance will be explored. Future work should also focus on conducting a thorough user study of the model's use in real-time. We will also conduct a series of analysis on the 'muscle–sound' relationships, in order to improve the model and diversify its potential output. Finally, we also see the potential for conducting other types of analyses on the gathered dataset. It would be particularly interesting to perform a more systematic analysis of the different types of action–sound couplings, and how they were captured by the different recording devices (EMG, video, motion capture, sound). One can also envisage between-participant comparisons, to reveal individual differences. With an interdisciplinary approach that draws on sound theory and embodied music cognition, we can design more 'economical' deep learning models for music interaction. The results of such analyses could also prove valuable when improving the modelling framework and further sonification strategies.

Acknowledgments

This work was partially supported by the Research Council of Norway (# 262762) and NordForsk (# 86892).

[3] Video is available at `http://bit.ly/air_guitar_smc`

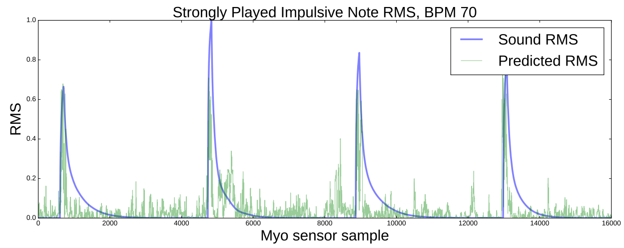

Figure 9. The RMS of the recorded sound and the model prediction for the impulsive note playing task.

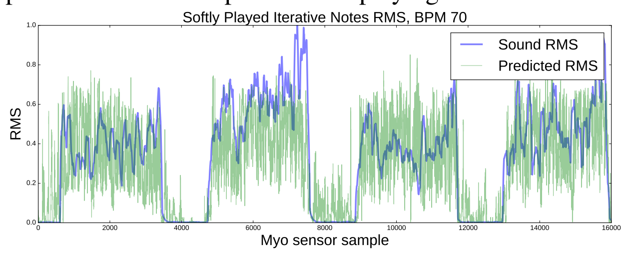

Figure 10. The RMS of the recorded sound and the model prediction for the iterative notes playing task.

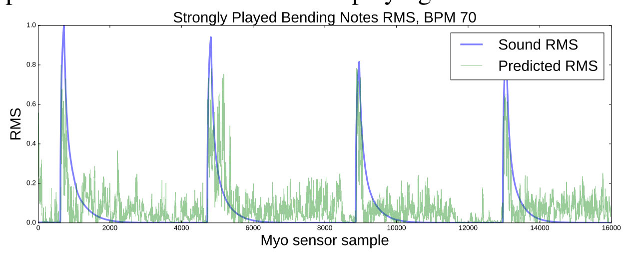

Figure 11. The RMS of the recorded sound and the model prediction for the bending (*sustained*) note playing task.

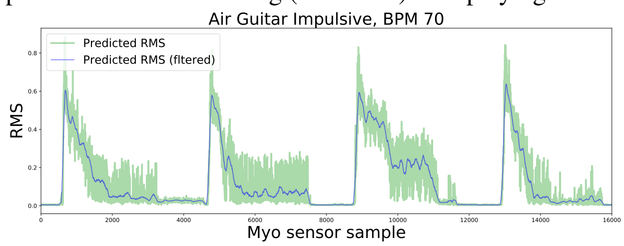

Figure 12. The predicted sound RMS of impulsive playing in the 'air' (as demonstrated in the video excerpts).

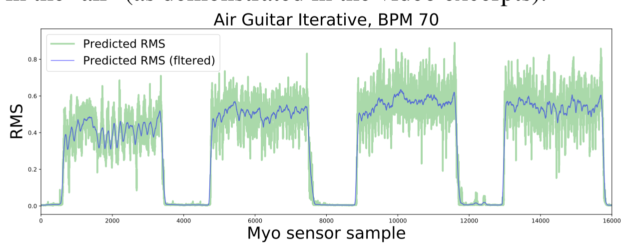

Figure 13. The predicted sound RMS of iterative playing in the 'air' (as demonstrated in the video excerpts).

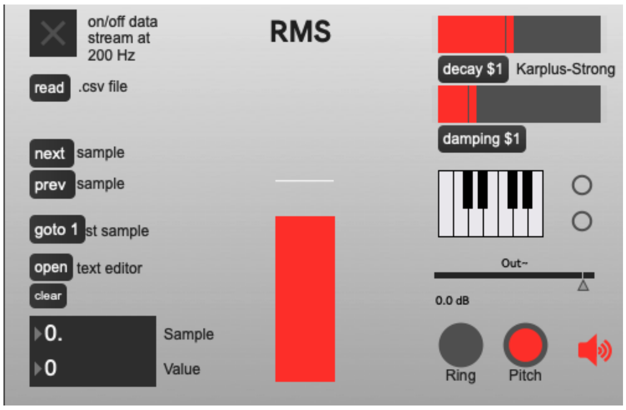

Figure 14. The user interface of the sonification patch.

6. REFERENCES

[1] R. I. Godøy, E. Haga, and A. R. Jensenius, "Playing "air instruments": Mimicry of sound-producing gestures by novices and experts," in *Gesture in Human-Computer Interaction and Simulation*, S. Gibet, N. Courty, and J.-F. Kamp, Eds. Berlin, Heidelberg: Springer, 2006, pp. 256–267.

[2] A. R. Jensenius, "Sonic microinteraction in "'the air"'," in *The Routledge Companion to Embodied Music Interaction*, M. Lesaffre, P.-J. Maes, and M. Leman, Eds. New York, NY: Routledge, 2017, ch. 46, pp. 431–439.

[3] ——, "An Action-Sound Approach to Teaching Interactive Music," *Organised Sound*, vol. 18, no. 2, pp. 178–189, 2013.

[4] M. Karjalainen, T. Mäki-Patola, A. Kanerva, and A. Huovilainen, "Virtual air guitar," *Journal of the Audio Engineering Society*, vol. 54, no. 10, pp. 964–980, 2006.

[5] J. Pakarinen, T. Puputti, and V. Välimäki, "Virtual slide guitar," *Computer Music Journal*, vol. 32, no. 3, pp. 42–54, 2008.

[6] Google, *Teachable Machine*. https://teachablemachine.withgoogle.com/, 2020.

[7] G. KaMen and E. Kinesiology, "Research methods in biomechanics," *Champaign, IL, Human Kinetics Publ*, 2004.

[8] A. Tanaka, "Embodied musical interaction," in *New Directions in Music and Human-Computer Interaction*. Springer, 2019, pp. 135–154.

[9] A. R. Jensenius, "Methods for studying music-related body motion," in *Springer Handbook of Systematic Musicology*. Springer, 2018, pp. 805–818.

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

[11] A. R. Jensenius and M. M. Wanderley, "Musical gestures: Concepts and methods in research," in *Musical Gestures*. Abingdon, UK: Routledge, 2010, pp. 24–47.

[12] R. I. Godøy, "Gestural-sonorous objects: embodied extensions of schaeffer's conceptual apparatus," *Organised sound*, vol. 11, no. 2, pp. 149–157, 2006.

[13] ——, "Understanding coarticulation in musical experience," in *International Symposium on Computer Music Multidisciplinary Research*. Marseille, France: Springer, 2013, pp. 535–547.

[14] A. Jensenius, "Action – sound - developing methods and tools to study music-related body movement," Ph.D. dissertation, University of Oslo, 01 2007.

[15] A. Hunt and M. M. Wanderley, "Mapping performer parameters to synthesis engines," *Organised sound*, vol. 7, no. 2, pp. 97–108, 2002.

[16] A. R. Jensenius and M. J. Lyons, *A NIME Reader: Fifteen Years of New Interfaces for Musical Expression*. New York, NY: Springer, 2017, vol. 3.

[17] M. Lee, A. Freed, and D. Wessel, "Real-time neural network processing of gestural and acoustic signals," in *Proc. of the Int. Computer Music Conf.*, Montreal, Quebec, Canada, 1991, pp. 277–277.

[18] R. A. Fiebrink, *Real-time human interaction with supervised learning algorithms for music composition and performance*. Princeton, NJ: Citeseer, 2011.

[19] F. Bevilacqua, B. Zamborlin, A. Sypniewski, N. Schnell, F. Guédy, and N. Rasamimanana, "Continuous realtime gesture following and recognition," in *International gesture workshop*, Bielefeld, Germany, 2009, pp. 73–84.

[20] B. D. Smith and G. E. Garnett, "Unsupervised play: Machine learning toolkit for max." in *Proc. Int. Conf. on New Interfaces for Musical Expression*, Ann Arbor, MI, 2012.

[21] N. Gillian and J. A. Paradiso, "The gesture recognition toolkit," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3483–3487, 2014.

[22] 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)*, vol. 4, no. 4, p. 18, 2015.

[23] J. Bullock and A. Momeni, "Ml. lib: robust, cross-platform, open-source machine learning for max and pure data." in *Proc. Int. Conf. on New Interfaces for Musical Expression*, Baton Rouge, LA, 2015.

[24] J. Snyder and D. Ryan, "The birl: An electronic wind instrument based on an artificial neural network parameter mapping structure." in *Proc. Int. Conf. on New Interfaces for Musical Expression*, London, UK, 2014.

[25] C. Kiefer, "Musical instrument mapping design with echo state networks," in *Proc. Int. Conf. on New Interfaces for Musical Expression*, London, UK, 2014.

[26] J. C. Schacher, C. Miyama, and D. Bisig, "Gestural electronic music using machine learning as generative device," in *Proc. Int. Conf. on New Interfaces for Musical Expression*, Baton Rouge, LA, 2015.

[27] C. P. Martin, K. O. Ellefsen, and J. Torresen, "Deep predictive models in interactive music," *arXiv preprint arXiv:1801.10492*, 2018.

[28] C. P. Martin and J. Torresen, "An interactive musical prediction system with mixture density recurrent neural networks," *arXiv preprint arXiv:1904.05009*, 2019.

[29] J. Gregorio and Y. Kim, "Augmenting parametric synthesis with learned timbral controllers," in *Proc. Int. Conf. on New Interfaces for Musical Expression*, Porto Alegre, Brazil, 2019.

[30] J. McCormack, T. Gifford, P. Hutchings, M. T. Llano Rodriguez, M. Yee-King, and M. d'Inverno, "In a silent way: Communication between ai and improvising musicians beyond sound," in *Proc. of the Conf. on Human Factors in Computing Systems*. Glasgow, UK: ACM, 2019, p. 38.

[31] A. Phinyomark, E. Campbell, and E. Scheme, "Surface electromyography (emg) signal processing, classification, and practical considerations," in *Biomedical Signal Processing*. Springer, 2020.

[32] C. P. Martin, A. R. Jensenius, and J. Torresen, "Composing an ensemble standstill work for myo and bela," in *Proc. Int. Conf. on New Interfaces for Musical Expression*, Blacksburg, VA, 2018.

[33] L. L. Beranek and T. J. Mellow, "Chapter 1 - introduction and terminology," in *Acoustics: Sound Fields and Transducers*, L. L. Beranek and T. J. Mellow, Eds. Academic Press, 2012, pp. 1 – 19.

[34] M. R. Ward, *Electrical engineering science*. New York, NY: McGraw-Hill, 1971.

[35] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NIPS Autodiff Workshop*, Long Beach, CA, 2017.