Elemental: a Gesturally Controlled System to Perform Meteorological Sounds

T. Brizolara da Rosa
IRISA
rue Yves Mainguy
Université Bretagne Sud
56000 Vannes, France
tiago.brizolara-da-rosa@irisa.fr

S. Gibet
IRISA
rue Yves Mainguy
Université Bretagne Sud
56000 Vannes, France
sylvie.gibet@irisa.fr

C. Larboulette
IRISA
rue Yves Mainguy
Université Bretagne Sud
56000 Vannes, France
caroline.larboulette@irisa.fr

ABSTRACT

In this paper, we present and evaluate *Elemental*, a NIME (New Interface for Musical Expression) based on audio synthesis of sounds of meteorological phenomena, namely rain, wind and thunder, intended for application in contemporary music/sound art, performing arts and entertainment. We first describe the system, controlled by the performer's arms through Inertial Measuring Units and Electromyography sensors. The produced data is analyzed and used through mapping strategies as input of the sound synthesis engine. We conducted user studies to refine the sound synthesis engine, the choice of gestures and the mappings between them, and to finally evaluate this proof of concept. Indeed, the users approached the system with their own awareness ranging from the manipulation of abstract sound to the direct simulation of atmospheric phenomena - in the latter case, it could even be to revive memories or to create novel situations. This suggests that the approach of instrumentalization of sounds of known source may be a fruitful strategy for constructing expressive interactive sonic systems.

Author Keywords

Gestural control, Audio environmental synthesis, Perceptual evaluation

CCS Concepts

•**Applied computing → Sound and music computing;** Performing arts;

1. INTRODUCTION

Traditionally, the audiovisual industry has relied on prerecorded sounds for effects/ambience. As computing power increased, procedural models have been introduced, allowing to fulfill also the needs imposed by nowaday's interactive media, where the triggering of actions - and hence, of sounds - is being replaced in considerable extent by continuous interactions, demanding then continuous sound changes. With this demand, came the issue of properly controlling these models. To achieve this goal, sound designers have been turning their attention to the strategies of control of sound synthesis of Digital Musical Instruments (DMIs) [10, 1].

As both procedural models for synthesis of environmental sounds and devices to capture human gestures get more efficient and accessible, we understand that there is an opportunity for the instrumentalization of such sounds (in the utilitarian sense of use as an instrument [9]) for artistic performance.

The case of meteorological sounds is particularly interesting, as while there exists a universally shared knowledge about these sounds, they are not produced by human movement in an obvious causal way. Even more, they have an intangible origin – the atmosphere –, which motivates us to investigate the performance of these sounds through free gestures.

In particular, by means of the development and evaluation of a proof of concept (*Elemental*), we seek to address the challenges in working towards a NIME (New Interface for Musical Expression), that takes advantage of the shared knowledge about the natural sounds involved. For that we start by applying to guidelines of DMI/NIME design, such as low latency and transparent mappings.

Our system being a new interface for musical expression, it necessarily involves human perception. To assess its quality and understand its affordances we developed experimentations with users, regarding ease of use, ease of learning, pleasure in using the system, intuitiveness of the mappings between gestures and sounds, efficiency exploring and refining sounds, suitability of the system for expressive applications and the quality of the sounds.

2. RELATED WORK

Real-time Synthesis of Environmental Sounds.

Regarding the sounds of interest in this work, Verron and Drettakis [20] developed an approach for the synthesis of textural sounds based on what they called perceptual *sound atoms*. These atoms were defined analytically in a time-frequency form, allowing for use in an efficient real-time sound spatialization method [19]. They demonstrated the technique, by implementing examples for rain, wind and fire integrated with graphics in the Unreal game engine[1]. Liu *et al.* [14] developed a hybrid physical and statistical model to synthesize rain sounds along with a method to retrieve the sound of the impact of the rain on a given material from audio recording, allowing to encode it in a material graphics shader, avoiding manual parameterization of the materials and facilitating spatialization.

Both Baldan *et al.* [3] and Farnell [5] developed extensive

NIME'20, July 21-25, 2020, Royal Birmingham Conservatoire, Birmingham City University, Birmingham, United Kingdom.

---

[1] `https://www.unrealengine.com`

sets of open source physics-inspired procedural models. The former includes a model of resonation of a cavity from wind interaction. The latter includes models of rain, wind and thunder, with a downside of the required manual work to achieve a unified rain model, since there are different signal-based models for different rain regimes.

Although we achieved less realistic rain sounds than those of [20] and [14], we chose to work with Farnell's implementations for the versatility of the wind model, the availability of the thunder model and the open source code. Although not used here, the wind model by Baldan et al. could be adapted to our system also.

Gestural Control of Environmental Sounds.

O'Modhrain and Essl [16] controlled simulated environmental sounds like water splashes, or dropping, or shuffling of objects, by means of a modified granular synthesis, parameterized at the grain level ($\sim 10 - 100\ ms$) from the live audio capture of pebbles in a box or different objects in a bag. Playback of grains was triggered by onset detection and the playback rate was parameterized by the zero crossing rate of the audio input at the instant of the onset.

Françoise[7] approached the control of environmental sounds inside a methodology called *mapping by demonstration*. First, a set of short sound examples was created within a space of acoustical descriptors. Then, for each sound, the user designed a hand movement which was captured using the Leap Motion technology[2]. A Gaussian Mixture Regression (GMR) model was then used to learn the mapping of each pair of corresponding hand motion / sound descriptor trajectories. Finally, when performing, the sound parameters associated with the hand movements were predicted with the GMR model and entered into a Corpus-Based Concatenative Synthesis.

In context of continuous interaction, such as games or virtual reality, Heinrichs and McPherson [10] emphasized the need for real time convincing audio synthesis. To this end, they built a *performable model* to synthesize the sound of an opening door, with phenomenologically relevant parameters and appropriate mappings. We consider that our approach is very close to the one of *performable models*: we need to work with phenomenologically relevant parameters (instead of physical parameters) to achieve controlability, and keep the sound synthesis believable. It is worth noting that a complex model may sound wrong if hard to control, even being physically accurate. To this end, physically inspired models tend to be more readily suitable than pure physics-based models, even if the latter also implement a phenomenologically relevant layer.

3. ELEMENTAL SYSTEM

Elemental is a system to perform meteorological sounds by means of free gestures. Controls of rain and wind are continuous, using orientation and angular speed information. A thunder can also be triggered upon gesture recognition. In our current implementation, we are using two Myo armband devices[3] that capture inertial and electromyographic (EMG) data. The architecture, depicted in Figure 1, is that of a Digital Musical Instrument (DMI), without primary feedback [21]:

- the Myo Armband sensors along with the *gestural control server* constitute the Input Device. It preprocesses gestural data and streams it to the Mapping Unit;

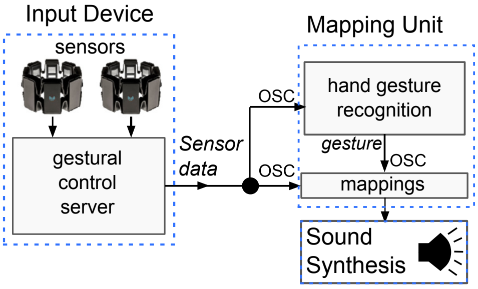

Figure 1: *Elemental* follows a DMI architecture: the Input Device sends gestural data to the Mapping Unit that makes use of this data to control the Sound Synthesis Engine.

- the *hand gesture recognition* - which in our implementation runs in a separate application, along with the *mappings* module, constitute the Mapping Unit, which receives the gestural data from the Input Device and outputs the synthesis control parameters for the Sound Synthesis Engine;
- and finally we have the Sound Synthesis Engine, consisting of procedural audio models, whose high level control parameters are the ones output from the Mapping Unit.

The *gestural control server* is based on the C++ application myo-osc[4], the mappings and the *Sound Synthesis* are written in Pure Data[17] and the *hand gesture recognition* is implemented within Wekinator[5], a software designed for applying Machine Learning in interactive applications[6]. These data transfers are executed via network using the Open Sound Control (OSC) protocol.

3.1 Inputs

By default, the rain is controlled by the Myo worn on the performer's right arm, while wind and thunder are controlled by the one worn on the left arm.

Each Myo armband device streams via Bluetooth to the *gestural control server*: gyroscope data ($3D$ angular speed); accelerometer data ($3D$ acceleration); orientation (unit quaternion, calculated by the firmware via sensor fusion of gyroscope, accelerometer and magnetometer data); 8 streams of EMG data (one for each electromyography sensor). Inertial data (gyroscope, accelerometer, orientation) is sent at 50 Hz, while EMG data is sent at 200 Hz.

The *gestural control server* sends to the *Mapping Unit*: angular speed; orientation converted to Euler angles; gyroscope data; and an array of EMG features (the latter are sent specificaly to the hand gesture recognition unit, explained in Section 3.3.3).

3.2 Sound Synthesis Engine

The input of the sound synthesis engine is composed of 7 controllable parameters, which we dubbed *phenomenological*, all normalized in the range $[0, 1]$. These 7 parameters are then mapped to the internal parameters of the procedural models that ultimately perform the synthesis. We present herafter a very brief overview of these models and indicate our changes within this work with respect to Farnell's implementations[5], which the reader will refer for a thorough description.

3.2.1 Rain Amount R_a

R_a controls the superimposition of four models, enabling the production of various rain qualities, ranging continu-

---

[2] `http://leapmotion.com/`

[3] from former Thalmic Labs, now North company `https://www.bynorth.com/`

[4] `https://github.com/Sindel/myo-osc`

[5] `http://www.wekinator.org/`

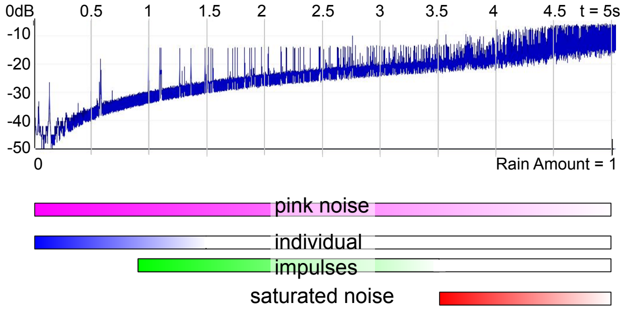

Figure 2: The rain is synthesized thanks to a superposition of 4 rain models: pink noise, two models of individual impulses and saturated noise. As *Rain Amount* R_a increases, models are added to the synthesis. The top curve shows a dB time profile of the rain sound synthesized for a 5 seconds ramp of R_a, from 0 (minimum) to 1 (maximum). *Rain Color* parameter R_c is fixed at 0.5 (corresponding to a one pole filter with 4000 Hz cutoff).

ously from small amounts of rain (few drops) to torrential rain. We manually adjusted all models to yield a consistent relation $dB \times$ *rain amount* and a smooth transition between the models. This is illustrated on Figure 2.

The four models that constitute the overall rain model are explained below.

To simulate a **small amount of rain**, a random distribution of up to $\sim$ 20 impulses/second of varying durations and amplitudes is obtained by extracting the upper portion of a cosine signal, whose frequency is modulated by filtered noise. This model corresponds to the blue bar in Figure 2.

The **moderated amount of rain** model is comprised of instances of parabolic impulse signals obeying a Poisson time distribution. We use 10 instances, resulting in $\sim$ 20 impulses per second at $R_a = 0.18$, up to $\sim$ 100 impulses per second, at $R_a = 0.8$.

Each instance i has a different base pulse duration d_0^i and a range of random variations Δd. This diversity of durations renders variation of perceived pitches.

This model corresponds to the green bar in Figure 2.

Note that the models for *Small amount of rain* and *Moderated amount of rain*, instead of working with real impact sounds, rely on the the fact that signals shorter than $\sim$ 20 ms are perceived as pitched impulses, despite the so-called Gabor's limit [8]. It has been shown that pulse signals in this duration regime are assigned a perceptual frequency that is inversely proportional to their duration [15][13].

The model for **moderated and intense rain** uses white noise that is consecutively band-passed, thresholded and saturated. This model corresponds to the red bar in Figure 2.

The model for **all amounts of rain** consists in a pink noise signal, present in all ranges, with gain proportional to the rain amount, to account for the rain noise background (pink bar in Figure 2).

3.2.2 Rain Color R_c

The Rain Color R_c corresponds to the cut-off frequency of a high-pass filter applied to the total resulting rain signal defined by R_a. This parameter simulates altogether the size of the drops and the type of object or ground that the rain is hitting.

The initial idea was to use a band pass filter to allow execution of noise melodies or noise sweeps. Results were not realistic and this was due to the physics of rain: drops scatter small fragments after initial collision, leading to high frequency sounds from subsequent impacts. So we always have to preserve the trebble part of the spectrum to take it into account - hence the choice of a high pass filter.

For the *Moderate amount of rain* model, as the pitch is a perceptual result of the pulse duration, filtering is ineffective. So, to apply our Rain Color, we modulate the base pulse durations d_0^i of the instances of the pulse model.

3.2.3 Rain Throw

This parameter controls the rain amount of a copy of the *Moderated and intense rain* model, tuned for higher pitched sounds. This results in individual rain bursts and is independent of the the underlying rainfall controlled by R_a.

3.2.4 Wind Speed

The *Wind Speed* W_s parameter controls the level of a background noise and the excitement of virtual objects in the stereo scene, namely two whistlings (as those generated by wires or poles) and two howls (such as those from slits of doors or windows). Each object has an excitation response lying in a different range of wind speeds.

3.2.5 Wind Oscillation

Wind Oscillation W_o controls the range of the natural wind speed oscillations. In Farnell's implementation the Wind Speed, W_s was perturbed by adding two random bipolar signals that were proportional to W_s and low pass filtered at 0.5 Hz and 3 Hz, respectively, to simulate two ranges of oscillation (”gust” and ”squall”). The result was very realistic, but represented also a clear lack of control. We decided then to control this parameter through gesture, similarly to a low-frequency oscillator in electronic music. The two disturbing signals have thus been replaced by a single signal, whose amplitude is controlled by W_o. In our current implementation, the low-pass frequency is set to 3 Hz.

3.2.6 Thunder (trigger)

The performer can also trigger a thunder. Except for the change of overall amplitude decay from linear to exponential and the addition of an explosion sound sample when thunder is triggered, no changes have been made to the model, which includes the sound from the shock waves coming directly from the thunder and estimated reflections in the environment. Except for the *distance* parameter, the user has little control over the thunder sound.

3.2.7 Thunder Distance

The *Thunder Distance* parameter models the distance from the thunder to the performer. It affects the volume of the initial explosion, the reverberation and softens the high frequencies of the N-shaped waves coming from the strike.

3.3 Mappings

The use of multiple layers of mapping allows for the definition of meaningful parameters and favours simpler mappings, facilitating both the design and performance of the system [22, 12, 2]. As seen in Section 3.2, we have 7 synthesis control parameters (that we are calling phenomenological parameters. In[22] these would be the *abstract parameters*). Our system has two layers of mapping: one that maps user gestural data to these 7 synthesis control parameters, and a subsequent, internal layer, that maps these parameters to

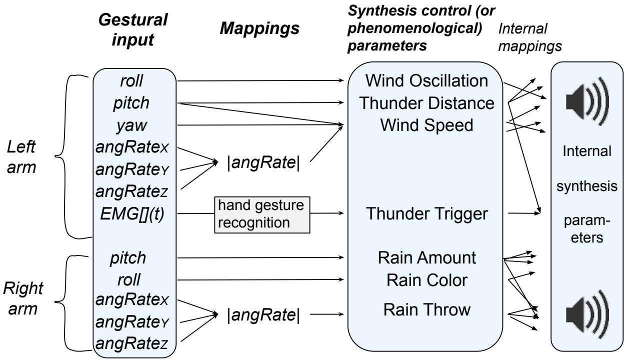

Figure 3: *Elemental* has two mapping layers: one maps gestural input data to the synthesis control parameters (called here *phenomenological parameters*) and the other maps these synthesis control parameters to the internal synthesis parameters - i.e., the parameters of the actual signal processing components of the procedural audio models involved.

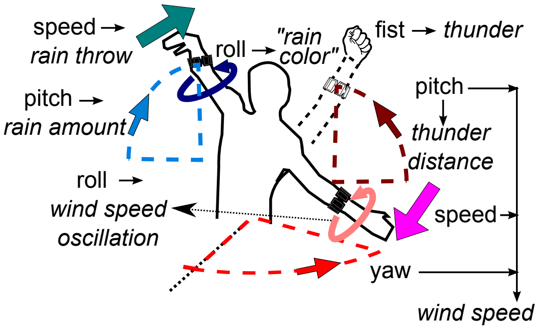

Figure 4: An overview of the mappings. Right arm controls models related to rain, while left arm controls thunder and wind.

the actual signal processing parameters of the algorithms involved in the synthesis, and that, thanks to this multilayer mapping are hidden from the performer (Figure 3). The signal conditionings of the gestural input parameters (for example, Figure 6), are not considered mappings here.

Considerations on the gestural parameters

The disconnection between the energy introduced by the performer and the energy of sound output is inherent in DMIs and can be a concern. Ruviaro[18], pointing laptop orchestras as an extreme case of this disconnection, proposes a definition of musical instrument that includes the aspect of *movement* (and *presence*). Hunt *et al* [12] conducted comparative tests where users reported better responses from instruments that required continuous input of energy. Nevertheless, as mid-air interaction is known to be prone to cause arm fatigue (or "gorilla arms") [11], we decided to base our mappings mainly on angular positions instead of speeds, accelerations or jerks (in fact, we tried to map arm angular speed directly to wind speed in informal tests. The resulting embodiment was impressive, but also tiring) and to use potential energy: both rain amount and wind speed descend to 0 with arms down.

We describe hereafter each mapping in details.

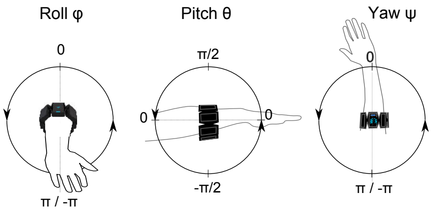

Figure 5: Myo orientation converted to Euler angles.

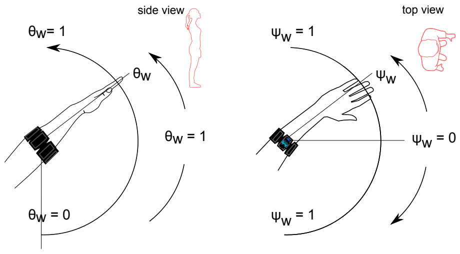

Figure 6: Rescaling of device's angles for use in subsequent mapping to the wind synthesis parameters. Left: rescaled pitch angle. Right: rescaled yaw angle.

3.3.1 Rain

Rain Amount R_a is zero with hand downwards and increases linearly with the pitch angle, going from 0 to 1 in the interval $[-60^o, 90^o]$: $R_a = max(0, 0.58 + 0.53\ \theta)$ (θ in radians).

Rain Color R_c is controlled by roll angle according to $c = min((5\phi+\pi)/(2\pi), 1)$, where factor 5 takes into account the fact that the Myo device rolls much less than the hand of the user.

Rain Throw is proportional to the modulus of the angular speed of the right Myo.

3.3.2 Wind

Wind Speed mapping adopts the following rationale: it is 0 for arm pointing forward, and increases to 1 at 90^o in both rotation directions. Also, a multiplicative factor is present: increasing from 0 with arm down up to 1 with arm at horizontal. Mathematically, the current mapping is:

$W_s = min(1, \theta_W \Psi_W + s)$, where θ_W and Ψ_W are the rescalings of pitch and yaw angles, respectively, and s is proportional to absolute arm angular speed (see Figure 6).

As each sounding source responds to a range of wind speeds, working the yaw angle Ψ_W is equivalent to "scanning" the sound sources, exciting them one by one.

Wind Oscillation amount W_o is proportional to the modulus of the left arm roll, going from 0 to 1 in the intervals $[0, \pm 25^o]$: $W_o = min(|\phi|/25, 1)$ (ϕ in degrees).

3.3.3 Thunder

Thunder is triggered when a *clenched fist* gesture is detected for the left hand. The current implementation uses the software Wekinator, which receives via OSC an array of features with 8 elements (one for each EMG sensor of the left arm Myo), each value being the average of the absolute value of the last 20 EMG samples (following the feature engineering proposed by [4]). The recognition algorithm is k-means classification. The current class of the pose is streamed via OSC to the synthesis module.

Although faster and more reliable than the algorithm of the official Myo SDK, we still had false positives. And if the user stayed in some specific arm poses, a false positive continually got retriggered.

Preliminary tests with Dynamic Time Warping in Wek-

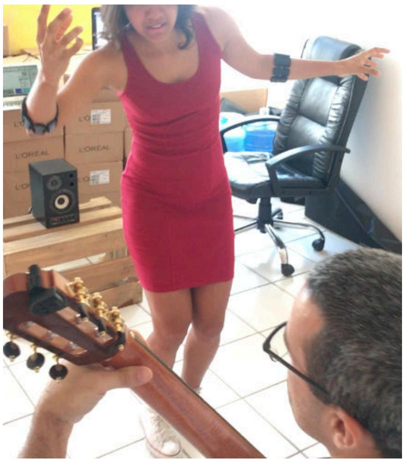

Figure 7: A musician/dancer plays the Elemental in a rehearsal with a classical guitarist.

inator seemed to achieve more reliable results. In these, we registered different clenched fist gestures as time series (again, each element being a feature array of 8 elements). We configured Wekinator to downsample each time series to 10 elements.

4. USER STUDY

To assess the quality of the system and understand its affordances, we developed a user study consisting in 2 experimentations. The pilot study consisted in experimental sessions with an audience comprised mostly of non-experts. We observed their reaction to the system and asked questions regarding ease of use, ease of learning, pleasure in using the system, intuitiveness of the mappings, efficiency exploring and refining sounds, suitability of the system for expressive applications and quality of the sounds. We then proposed our system to a second group composed mainly of experts (musicians, dancers, DMI designers) to validate it. Both experiments are first detailed followed by a discussion.

4.1 First Evaluation: A Pilot User Study

A preliminary user study was carried out with 23 participants (10 females and 13 males, aged 10 to 79), composed mostly of non-experts: 60.87% weren't skilled as musicians, nor dancers, actors, sound professionals, or similar.

We first explained to each participant how to use the system and the different mappings involved. We subsequently trained the Wekinator classifier with the *clenched fist* gesture for the system to learn when to trigger the thunder. The participants could then optionally train a few minutes with the system. When ready, they could perform for as long as they wanted and we recorded the produced data. When finished, we interviewed them with a detailed questionnaire.

The questionnaire was composed of 18 questions regarding ease of use, ease of learning, pleasure in using the system, intuitiveness of the mappings, efficiency exploring and refining sounds, suitability of the system for expressive applications and quality of the sounds. The participants should grade their responses on a 5-points ascendant Likert scale, where -2 would be very negative (for example, *Strongly Disagree* or *Very Bad*), 0 neutral and 2 very positive (for example, *Strongly Agree* or *Very Good*).

Note: The version in this pilot has small differences regarding the Wind: Wind Oscillation is autonomous (so, there is no Wind Oscillation Amount parameter here. see Section 3.2.5); yaw rescaling of left arm reads $0-180^o$, instead of $0-90^o$ to both directions (see Figure 6).

4.2 Second Evaluation: Evaluation of the System by Performers

Having proven in the first pilot study that the system was usable, intuitive and suitable for live experiments, we conducted an experimentation similar to the one from the pilot study. This second evaluation was conducted with mostly skilled performers. Among the 18 participants (8 females and 10 males, aged 19 to 37), 15 were skilled in one or more areas related to the experiment: 10 were skilled musicians, 8 were skilled dancers and/or actors, 1 was a skilled sound designer, 2 were digital music instrument designers.

4.3 Results of the Questionnaires

We compare the responses of the users in the pilot study and in the validation with experts (now referred to as Evaluation 1 and Evaluation 2, respectively). The questions answered in both studies are summarized below. The corresponding results, illustrated in Figures 8, 10, 9, are discussed.

How do you rate the suitability of the system for expressive applications? (Figure 10)
In Evaluation 1, it was a Suitable/Non-suitable question. Just 1 subject out of 28 rated the system as *not suitable* (not included in Figure 10).
In Evaluation 2, the question was divided in 3 sub-questions, related to: expressive applications of sonic nature (body in secondary importance – for example, music performance); bodily nature (sound in secondary importance – for example, dance, theater); or mixed nature (body and sound equally important). As shown in Figure 10, most subjects rated the system as *very suitable* in all the three scenarios, followed by *suitable.*

Was the relation between executed movements and produced sounds natural or intuitive? (Figure 8)
All mappings had the majority of ratings as either *intuitive* or *very intuitive.* We see that the results were coherent between the two validations, except for Wind Speed. It is important to recall that in Evaluation 1, Wind Speed had a different mapping, that we changed for the sake of symmetry (Section 3.2.5). Some users reported in Evaluation 2 that the range in yaw angle was too short. This may be the cause of the worst result, as the wind model is complex and has a dynamic behaviour of excitations and decaiments.

How do you qualify your experience using the system? (Figure 9, upper left)
In Evaluation 1, 17 subjects (73.9%) rated the system as *very pleasant* and 6 subjects (26.1%), as *pleasant.* In Evaluation 2, 14 subjects (77.8%) rated the system as *very pleasant* and 4 subjects (22.2%) as *pleasant.*

How do you rate the ease of learning the system? (Figure 9, top center left)
Most users rated *easy to learn* (Evaluation 1: 56.5%, Evaluation 2: 44.4%), followed by *very easy* (26.1% and 33.3%, respectively). Given the amount of elements to be controlled – 6 parameters (5 in Evaluation 1, where *Wind Oscillation* was autonomous) plus 1 gesture – this reinforces the results on intuitiveness.

Regarding the next two questions, it's important to notice

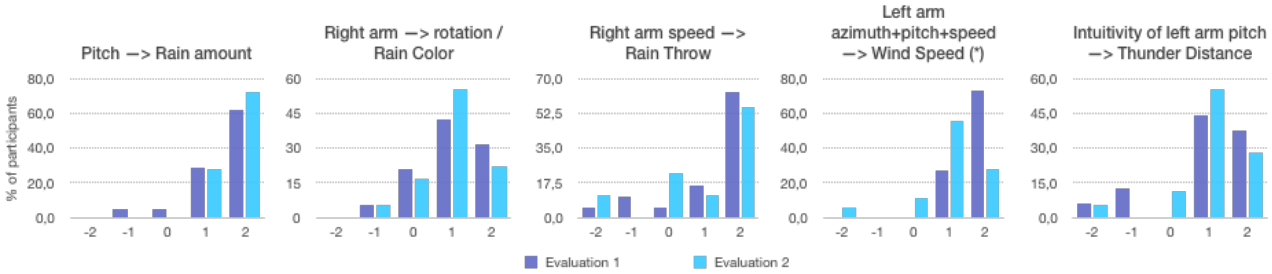

Figure 8: Assessed intuitiveness of the mappings for Evaluations 1 and 2. We see consistent results between evaluations for most mappings, except for a swap of the two values of the upper end of the Likert scale for *Wind Speed*. (*)Notice that the mappings for Wind Speed are different between Validations 1 and 2. See note at the end of Section 4.

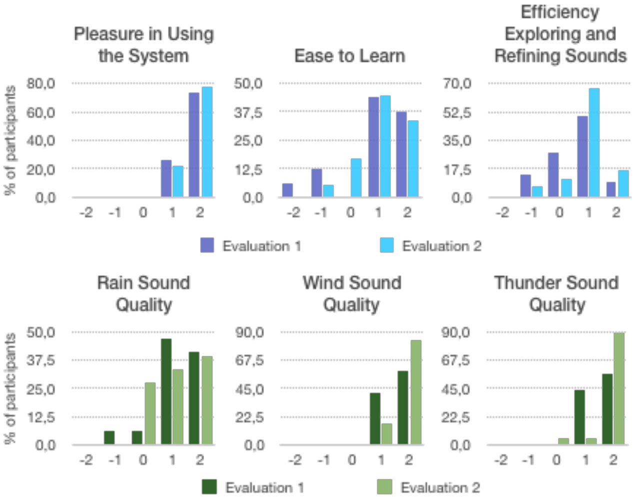

Figure 9: Assessed pleasure, ease to learn and efficiency (top row); Assessed sound quality of *Rain*, *Wind*, and *Thunder* (bottom row); for Evaluations 1 and 2.

that the group in Evaluation 2 had more than twice expert musicians than the group in Evaluation 1. These results are the ones that most vary between the two validations.

How do you rate the efficiency for exploring and refining sounds inside the system? (Figure 9, upper right)

For Evaluation 1, we have $\mu = 0.55$, $\sigma = 0.86$, while for Evaluation 2, $\mu = 0.94$, $\sigma = 0.73$, showing a rather good efficiency in manipulating sound parameters, especially in Evaluation 2.

How do you rate the quality of the sounds? (Figure 9, bottom)

Quality was rated much higher in wind and thunder than in rain (even so, rain had $\mu = 0.88$, $\sigma = 0.83$ for Evaluation 1 and $\mu = 1.11$, $\sigma = 0.83$ for Evaluation 2). Some users observed high values of rain color as less realistic. In fact, the high end of the scale of R_a represented an oversaturated sound that, in a future version, won't be available.

4.4 Other remarks

We report here some notable appreciations made by users.

In the First Evaluation, one user recreated the weather of his home village, another felt at a specific beach of Bretagne (France) and felt even the rain hitting her helmet, other felt at the sea in the storm and controlled the waves hitting the boat. A professional DJ and dancer wanted to have *Elemental* in his setup.

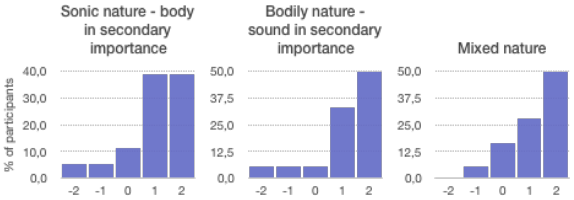

Figure 10: Assessed suitability of the system for expressive applications, for Evaluation 2.

In the Second Evaluation, users related varied approaches: some performers simulated real sounds to match their memories, others told stories from the sounds, others concentrated solely on aesthetical exploration of sounds, others created choreographies that rendered sounds at the same time interesting and harmonious with the movement.

5. CONCLUSION AND PERSPECTIVES

The ability with *Elemental* to perform and compose music at the sound level, and its suitability for multimodal performance, using both body and sound, was almost unanimously recognized, both by direct observation (Figure 10) and by the fact that many performers showed a strong interest in including *Elemental* in different artistic works. At the moment, an artistic collective[6] is rehearsing with the system and also a duo of a musician/dancer with a classical guitarist (Figure 7)[7].

In addition, users reported working both in terms of pure sound creation or in terms of simulation or recall of meteorological phenomena. These results suggest that environmental sound models may be suitable to build expressive NIMEs.

A compromise between realism (or believability) and sound control was also observed in the development of *Elemental*. Noise melodies or noise sweeps were not allowed by controlling Rain Color, because realism imposed high frequencies to be present (from the smaller fragments generated from water drops after impact). Wind Oscillation, which is stochastic, had also to be re-modelled and controlled by the performer. A complete absence of oscillation rendered the wind sound artificial, while completely autonomous oscillations represented a lack of control.

The attested ease to learn and intuitiveness of the mappings showed that with a good choice of gestures and mappings, a considerable amount of synthesis parameters (seven) could be handled at the same time.

Many possible improvements were raised from the expe-

---

[6] `https://www.fukeicollectif.com/`

[7] A video of the musician/dancer trying the system can be seen at `https://www.youtube.com/watch?v=V_Sv5HiV5zU`

riences with the users, such as the refinement of control possibilities (for example, wind synthesis has only been considered in wind speed).

We intend to continue investigating the system, increasing the dimensionality of the gestural input data and examining strategies for better representing a high-dimensional gestural interaction in this context and designing and automating the mapping to the synthesis engine.

Acknowledgments

This research has been funded by Région Bretagne and Department of Morbihan, France. We also thank MusTIC group of Federal University of Pernambuco (Brazil), specially MSc. João Tragtenberg, Dr. Giordano Cabral and Dr. Filipe Calegario for the discussions and the kind support on the tests conducted in Brazil.

6. REFERENCES

[1] M. Aramaki, R. Kronland-Martinet, and S. Ystad. Perceptual Control of Environmental Sound Synthesis. In S. Ystad, M. Aramaki, R. Kronland-Martinet, K. Jensen, and S. Mohanty, editors, *Speech, Sound and Music Processing: Embracing Research in India*, Lecture Notes in Computer Science, pages 172–186, Berlin, Heidelberg, 2012. Springer.

[2] D. Arfib, J.-M. Couturier, L. Kessous, and V. Verfaille. Strategies of mapping between gesture data and synthesis model parameters using perceptual spaces. *Organised Sound*, 7:127–144, Aug. 2003.

[3] S. Baldan, S. Delle Monache, and D. Rocchesso. The Sound Design Toolkit. *SoftwareX*, 6:255–260, Oct. 2017.

[4] B. D. Donato, J. Bullock, and A. Tanaka. Myo Mapper: a Myo armband to OSC mapper. In *NIME*, 2018.

[5] A. Farnell. *Designing sound*. MIT Press, Cambridge, Mass, 2010. OCLC: ocn494275436.

[6] R. Fiebrink, D. Trueman, and P. Cook. A Meta-Instrument for Interactive, On-the-Fly Machine Learning. Jan. 2009.

[7] J. Françoise. *Motion-Sound Mapping by Demonstration*. PhD thesis, Mar. 2015.

[8] D. Gabor. Theory of communication. *Journal of the Institution of Electrical Engineers - Part I: General*, 94(73):58–, Jan. 1947.

[9] S. Hardjowirogo. Instrumentality. On the Construction of Instrumental Identity. In *Musical Instruments in the 21st Century: Identities, Configurations, Practices*, pages 9–24. Dec. 2017.

[10] C. Heinrichs and A. Mcpherson. Mapping and interaction strategies for performing environmental sound. In *2014 IEEE VR Workshop: Sonic Interaction in Virtual Environments (SIVE)*, pages 25–30, Mar. 2014.

[11] J. D. Hincapié-Ramos, X. Guo, P. Moghadasian, and P. Irani. Consumed endurance: a metric to quantify arm fatigue of mid-air interactions. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pages 1063–1072, Toronto, Ontario, Canada, 2014. ACM Press.

[12] A. Hunt, M. M. Wanderley, and M. Paradis. The Importance of Parameter Mapping in Electronic Instrument Design. *Journal of New Music Research*, 32(4):429–440, Dec. 2003.

[13] M. Karolina, R. Jarosław, K. Tadeusz, and Z. Piotr. The perception of short-time acoustic pulses. *INTER-NOISE and NOISE-CON Congress and Conference Proceedings*, 259(7):2540–2547, 2019.

[14] S. Liu, C. Haonan, and Y. Tong. Physically-based statistical simulation of rain sound. *ACM Transactions on Graphics*, 38:1–14, July 2019.

[15] J. N. Oppenheim and M. O. Magnasco. Human Time-Frequency Acuity Beats the Fourier Uncertainty Principle. *Physical Review Letters*, 110(4):044301, Jan. 2013.

[16] S. O'Modhrain and G. Essl. 2004: PebbleBox and CrumbleBag: Tactile Interfaces for Granular Synthesis. In A. R. Jensenius and M. J. Lyons, editors, *A NIME Reader: Fifteen Years of New Interfaces for Musical Expression*, Current Research in Systematic Musicology, pages 165–180. Springer International Publishing, Cham, 2017.

[17] M. Puckette. Pure Data: another integrated computer music environment. *Proceedings of the second intercollege computer music concerts*, pages 37–41, 1996.

[18] B. Ruviaro. From schaeffer to* lorks: An expanded definition of musical instrument in the context of laptop orchestras. In *Proceedings of the 1st Symposium on Laptop Ensembles & Orchestras*, pages 23–26, 2012.

[19] C. Verron, M. Aramaki, R. Kronland-Martinet, and G. Pallone. A 3-D Immersive Synthesizer for Environmental Sounds. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1550–1561, Aug. 2010.

[20] C. Verron and G. Drettakis. Procedural audio modeling for particle-based environmental effects. *133rd Audio Engineering Society Convention 2012, AES 2012*, 2, Oct. 2012.

[21] M. Wanderley and P. Depalle. Gestural Control of Sound Synthesis. *Proceedings of the IEEE*, 92:632–644, May 2004.

[22] M. Wanderley, N. Schnell, and J. Rovan. ESCHER-modeling and performing composed instruments in real-time. In *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.98CH36218)*, volume 2, pages 1080–1084, San Diego, CA, USA, 1998. IEEE.