Embodied Generation of Sonic Objects

Marco Fiorini
mfiori21@student.aau.dk
Aalborg University Copenhagen
Sound and Music Computing, 8th Semester
Embodied Interaction

***Abstract*—This project explores the possibilities of embodied music cognition and embodied generation of sonic object, through a new digital instrument controlled with gestures. Starting from the idea of sonic objects proposed by Pierre Schaeffer, a polyphonic granular sampler has been implemented in Max/MSP. Here, gestures recorded from a smartphone and handled through Open Sound Control have been mapped to the granular synthesis' parameters, using a set of computed mid-level movement descriptors.**

I. Introduction

The demand for new tools in terms of interactive possibilities offered by digital media technology has stimulated interest in gestural integration in musical applications [1]. Observing human motion in detail has brought up movement descriptors through which gestures in musical performances can be assessed [2]. Meaning, style and expressiveness can be communicated through a combination of multiple elements under human movement exploration [2]. The concept around gestures can be explained as a way of bridging movement and meaning, going further over the boundaries set between physical world and mental experiences [3].

As an improvising musician in the area of electroacoustic and electronic music, these researches on gesture manipulation of sound and embodied cognition and interaction with the sonic matter have represented a relevant research field throughout the last period. Considering specifically the work of Visi et al. [3] [4] on embodiment of music cognition and designing of gestural mapping strategies in music performances, I wanted to investigate the possibilities of controlling an instrument through gestures. Building upon van Dijk's framework of *Designing for Embodied Being-in-the-World* (D4EB) [5], the idea for this project was to design an hybrid artefact conceptually situated in between the inner person and the outer environment, and in between the representational world of information and the purely physical world, operating as a unified whole to support and transform the ways in which the lived body is active in relation to the *"lifeworld"* (the world in which our lived body operates [5]).

This resulted in choosing the smartphone, an everyday tool used by all kind of people to make sense of the world in which we are connected, as the object designed to control this instrument, and led the me to the following hypothesis/problem statement: *I will use gesture to interact with the generation of sonic objects, based on Schaeffer's definition and OSC mapping, using IMU data from a smartphone*

II. Problem Analysis

A. HCI Research Problem

The problem solving capacity of this research project could be related both to Empirical and Constructive research in Human-Computer Interaction (HCI) [6].

- *Empirical*: the project aims at creating and elaborating description of real-work phenomena, such as sonic objects, related to human use in a digital computer interaction.
- *Constructive*: interactive artefacts, such as smartphones, will be taken into account and investigated for the human interaction with the computational side.

1) Concepts of interaction: The basic concept of interaction in this project is based on Experience, considered as an ongoing stream of expectations, feelings and memories, rather than Embodiment, defined as acting in a social world [7]. This because my project focuses on the generative process of sonic objects from gesture mapping, but in future works this could definitely be expanded and enhanced in a participatory sense-making approach and thus cover an important aspect of social interaction in the lifeworld as defined by van Dijk in [5]. Other fundamental concepts of this projects are Dialogue, Transmission and Control, as they refer to a good theoretical investigation of the model and a good implementation of the mapping of the parameters for an optimal user experience [7].

B. Sonic Objects

The idea of sonic objects is generally ascribed to the seminal work of Pierre Schaeffer in the 1950s and 1960s and as emerging from practical work in electroacoustic composition of the so-called *musique concrète*. Sonic objects can be defined as "a fragment of musical sound, typically in the approximately 0.5-5 s duration range, a fragment perceived holistically as a coherent and somehow meaningful unit, the most basic unit in musical experience" [8]. The duration limits of a sonic object are determined at one end by the minimal duration necessary to perceive salient features and at the other end by a maximal duration for perceiving the object as a singular and coherent entity, i. e., as not readily divisible into smaller parts. Schaeffer discovered that manipulating existing sounds and listening to them innumerable times, his perception of these sound fragments changed, that they tended to shift ones attention towards more internal and subtle features of the sound itself, He called this shifting of attention *reduced listening (écoute réduite)*, signifying a shift toward perceptually salient sonic features, a shift of focus that eventually lead to a very extensive theory of sonic objects in Schaeffer's monumental *Traité des objets musicaux* [9] and related publications.

C. Somaesthetic

Somaesthetic, originally conceived by Shusterman in [10] as being under the umbrella of philosophy, or perhaps even a branch of aesthetics, has evolved into an interdisciplinary field of inquiry aimed at promoting and integrating the theoretical, empirical and practical disciplines related to bodily perception, performance and presentation. Thus, in this project, important aspects of the Soma Design manifesto [11] will be taken into account, namely:

- We design to move the passions in others and ourselves
- We are movement, through and through
- We design with ourselves
- We cultivate our aesthetic appreciation

D. Movement Descriptors

Following several researches in the field of computable motion descriptors, like Larboulette and Gibet [2] and Federico Visi [4], I was able to extract meaningful representations of motion through descriptors. Low-level motion descriptors represent dynamic or kinematic quantities directly derived from motion representations. Mid-level descriptors, such as Quantity of Motion, represent an intermediate stage between low and high ones, dedicated to specific tasks, presenting a good approximation of the abstract motion feature. High-level descriptors are based on semantic components. These are structural notations like Laban Movement Analysis (LMA), and they can describe the structural, dynamic and geometric properties of human motion. As cited in Hackney [12], LMA is defined by four basic effort factors: *flow, weight, time, space*, where each one holds two opposing dimensions described by effort qualities. Sustained effort qualities, for example, are expected to have low level of jerkiness, while sudden quick characteristics would have a higher rate of change in acceleration [4]. Therefore, jerkiness and fluidity are valuable for analysing expressive movement qualities. Nevertheless, since Laban effort elements are "qualitative inner attitudes of a person moving towards the effort factor" [4], the use of computable descriptors should not be seen as an attempt to measure effort qualities quantitatively, but more as an helpful approach to design computational models discerning movements of expressive nature. The following is then a brief overview of the descriptors later used in the implementation of this research.

1) Velocity: Computes for one joint the rate of change of its position [2]:

$$v^k(t_i) = \frac{x^k(t_i+1) - x^k(t_i-1)}{2\delta t} \tag{1}$$

The speed of one joint is represented as the magnitude of its velocity [2]:

$$v^k(t_i) = \sqrt{v_x^k(t_i)^2 + v_y^k(t_i)^2 + v_z^k(t_i)^2} \tag{2}$$

2) Acceleration: In physics the definition of acceleration would be the rate of which velocity changes with time. As described in [2] by Larboulette and Gibet, it computes the instantaneous acceleration for one joint k and can be estimated by the following equation:

$$a^k(t_i) = \frac{x^k(t_i+1) - 2x^k(t_i) + x^k(t_i-1)}{\delta t^2} \tag{3}$$

3) Fluidity and Jerkiness: In the process of measuring kinematic quantities used to describe motion, "jerk" represents the variation of acceleration over time and is the third-order derivative of movement position [4]. In [13], Flash & Hogan define "jerk index" as the magnitude of the jerk averaged over the entire movement. Thus, jerkiness can be seen as the inverse of fluidity since it relates to the smoothness of the movement [4]. For one joint k it can be computed [2] as:

$$j^k(t_i) = \frac{x^k(t_i+2) - 2x^k(t_i+1) + 2x^k(t_i-1) - x^k(t_i-2)}{2\delta t^3} \tag{4}$$

From the definition for fluidity index, it can be derived that higher values of jerk correspond to lower fluidity [4]:

$$f^k(t_i) = \frac{1}{j^k(t_i) + 1\delta t} \tag{5}$$

4) Quantity Of Motion: Defined as the sum of speeds of a set of points multiplied by their mass by Fenza et al. [14], denoted as overall motion energy by Glowinski et al. [15] and computed by Visi in his modosc implementation as group of points taking into consideration the weight of each point [16], QoM is expressed by the equation [2]:

$$QoM^k(t_i) = \frac{\sum_{k \epsilon K} w_k . v^k(t_i)}{\sum_k w_k} \tag{6}$$

5) Periodic Quantity of Motion: Inspired by Quantity of Motion, in the need of a suitable descriptor that is able to describe multiple periodic gestures, PQoM was proposed as a way to measure a temporal quality in the movement [4]. It is expressed as:

$$PQoM^k(T) = \frac{1}{T}\sum_{i=1}^{T} QoM^k(t_i) \tag{7}$$

6) Time Effort: Defined by the sense of urgency, this high-level effort descriptor has two opposing dimensions - Sudden (quick) and Sustained (streched, steady) [2].

$$Time^k(T) = \frac{1}{T}\sum_{i=1}^{T} a^k(t_i) \tag{8}$$

7) Flow Effort: Explained as describing the continuity of the movement, it is denoted by the two opposite dimensions: Free (fluid) and Bound (restrained) [2]. The computation comes from the combined jerk over time. For the kth part of the body and a movement of length T, it is expressed as:

$$Flow^k(T) = \frac{1}{T}\sum_{i=1}^{T} j^k(t_i) \tag{9}$$

III. Implementation

The architecture of the whole system is presented in Figure 1. The user holds and move the smartphone, as the gestures are tracked by the phone sensors (gyroscope, accelerometer and magnetometer). The data are packed and sent to a PC using the OSC protocol [17] and mapped, using movement descriptors, to the parameters of the granular synthesizer in a Max/MSP patch [18]. Furthermore, a toggle in the Max/MSP patch enables pitch holding of the generated grain of sound.

A. Granular Synthesis

The basis of granular synthesis can be traced to the particle theory side of sound. Seemingly, the notion of the sonic grain which makes up the particles in granular synthesis, was first introduced in the Dutch scientist Isaac Beeckman's journals. The sonic grain is later found in the work of Nobel Prize physicist (holography) Dennis Gabor [19], which focuses on the theory of acoustical quanta. The coining of term grain, however, is attributed to Iannis Xenakis who has conducted much research in this area and composed musical works based on grains of sounds [20]. The basic concept of granular synthesis is quite simple: use elementary sonic particles (grains) of short duration (tipically in the range of 10-60 msec) and juxtapose them horizontally (time) and vertically (amplitude/density) to produce a sound object or sound cloud.

Thus, granular synthesis can be implemented generating small sound events and placing them over time as if in a score. It was chosen as the main synthesis model for this project because its parameters seem to logically fit with the characteristics of sonic objects as described in [9] and reported in [8]. The basic control parameters of a granular synthesizer are typically the following:

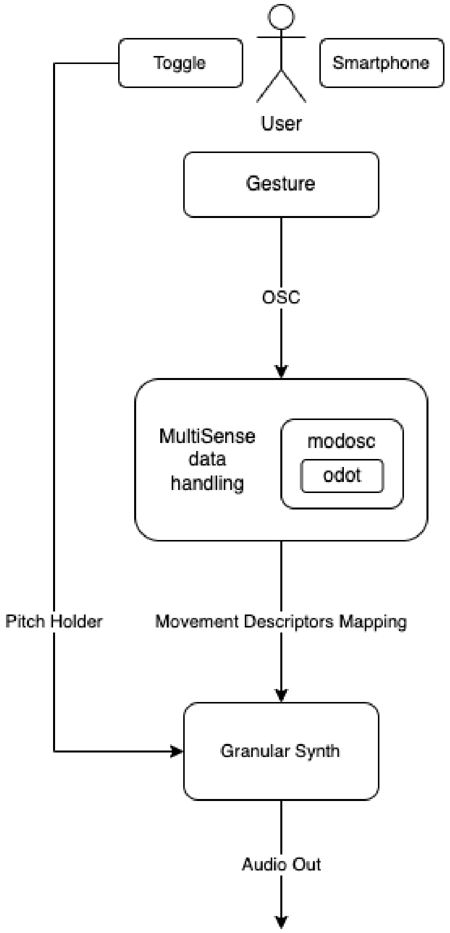


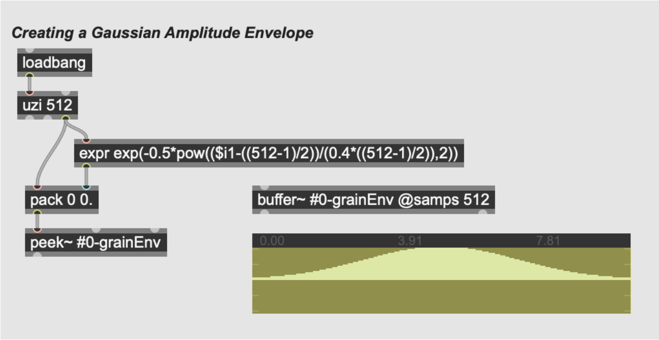
Fig. 1. Flow diagram representing the architecture of the whole system.

- Grain size: length of the grain (in msec)
- Window type: envelope shape for excerpting a grain
- Grain amplitude: maximum amplitude of grain
- Grain density: number of grains per unit of time (sec). Typical grain densities range from several hundred to several thousand grains per second.
- Grain pitch: pitch characteristics.

Grains are defined as small windowed portions of an audio signal typically anywhere between a few milliseconds to 100 milliseconds in duration. The choice for window length is directly related to what sort of effect we want, including the degree of recognition of the original sound source. In order to produce a sonic grain we will need then to apply a window on a portion of the target signal. The grain is defined as a windowed portion of input samples $x[n]$ using window $w[n - L]$ where L is the delay (offset or start point where windowing takes place):

$$x_{grain}[n] = w[n-L] \cdot x[n] \tag{10}$$

To enable smooth transitions between grains, each grain has an amplitude envelope. In Gabor's original conception [19], the amplitude envelope is a bell shaped curve generated by the Gaussian method. The same approach was adopted in this research, as shown in Figure 2.



```
Creating a Gaussian Amplitude Envelope
loadbang
uzi 512
expr exp(-0.5*pow(($i1-((512-1)/2))/(0.4*((512-1)/2)),2))
pack 0 0.
peek~ #0-grainEnv
buffer~ #0-grainEnv @samps 512
```

Fig. 2. Max/MSP subpatch generating the Gaussian amplitude envelope for each grain.

The density of grains, usually defined as grains per second, in general contributes to the richness of the resulting synthesized signal. That is, the greater the density, the richer and louder the sound. As is the case with the ability to control the amplitude of each grain, it is also possible to control the pitch of each grain. The main parameters of the implemented grain generator are grain start, grain duration and grain rate. The first two are randomly selected inside a user-defined range of values, while the latter is referring to the playback speed for each grain. This can also be defined by the user, allowing negative values resulting in backward playback of the grain itself. Furthermore, a random algorithm distributes each grain in a different stereo position through the audio panorama.

B. OSC Data handling

My motivation for using OSC [17] for the sending of the data is because it has already successfully been used in two prior research projects which resulted in Max/MSP libraries for handling OSC data, namely *odot* [21] and *modosc* [16]:

- *odot* is a framework for writing dynamic programs using C-like language inside a host environment such as Max. Compared to more conventional Max objects, it provides access to advanced formatting and parsing of OSC data bundles, allowing for greater control over timing and synchronization of multiple data streams. In addition, it allows the evaluation of functions that would be difficult to implement using standard objects.

- *modosc* is a "set of Max abstractions designed for computing motion descriptors from raw motion capture data in real time. The library contains methods for extracting descriptors useful for expressive movement analysis and sonic interaction design. Moreover, modosc is designed to address the data handling and synchronization issues that often arise when working with complex marker sets. This is achieved by adopting a multi paradigm approach facilitated by odot and OSC to overcome some of the limitations of conventional Max programming, and structure incoming and outgoing data streams in a meaningful and easily accessible manner" [22].

OSC can be then understood as a more flexible alternative to MIDI, as it clears away many of the ideological and hardware constraints inherent to MIDI in favor of a open-ended, user-defined address-space model that provides arbitrary parametric control via standard networking hardware [23] [24].

To send OSC data from a smartphone to a computer running Max/MSP, MultiSense OSC was used. MultiSense OSC is an app which allows the user to send wireless sensor data via OSC protocol from their Android device[1]. It was originally developed to perform head tracking for sound engineers combined with binaural VST plugin. Part of the code is derived from Sebastian O. H. Madgwick, open-source gradient descent angle estimation algorithm [25]. The Attitude And Heading Reference System (AHRS) algorithm combines gyroscope, accelerometer, and magnetometer data into a single measurement of orientation relative to the Earth. The resulting absolute orientation (quaternion) is then formatted as OSC data and sent to Max/MSP through the User Datagram Protocol (UDP). The requirement for this procedure to succeed is that both the smartphone and the PC must be connected to the same WiFi network, and the smartphone must know the IPv4 address of the PC. After matching the destination port of the smartphone with the source port of the PC, Max/MSP can receive the data packets through the object `udpreceive`, with the number of the selected port as an argument. The addresses contained in OSC packets are then matched to extract raw values using the `o.route` function, an odot function that dispatches OSC messages according to an address hierarchy, stripping off the portion of the address that matched. The resulting data are packed, using the `o.pack` object and, using modosc syntax domain, converted with an odot `o.expr.codebox` to a modosc point of data (the phone). A point in modosc consists in data bound to an OSC address. Points can be then collected in groups, as points and groups are the two main data types on which modosc abstractions operate [16].

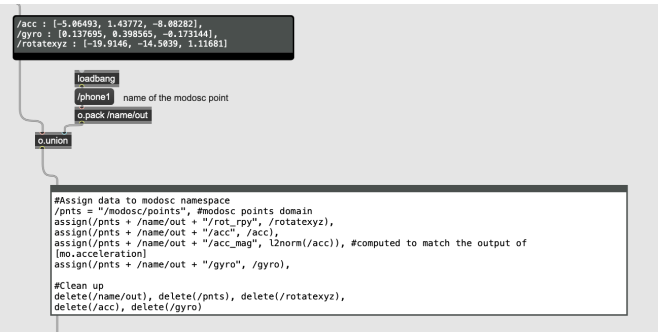
C. Implementation of Movement Descriptors

As previously stated, modosc was initially designed to work with motion capture (MoCap) data [16] [22]. Visi et al. [26] deepened the research on this field, using different kind of data, taken also from Inertial Measurement Units (IMU). This

[1] https://play.google.com/store/apps/developer?id=MultiSense+OSC

work culminated in the release of GIMLeT [27], a set of Max patches based on odot and modosc, incorporating neural network, gesture following through PoseNet [28] [29] and handling of OSC data with TouchOSC [30]. Unfortunatelly, TouchOSC is only able to send accelerometer data from the smartphone, thus limiting the number of movement descriptors which could be calculated (see Section II-D). This is why I decided to use MultiSense OSC and its algorithm to gather also data from gyroscope and magnetometer, resulting in orientation values. As shown in Figure 3, I was able to adapt the orientation-based data extracted from the smartphone to the modosc syntax using a `o.expr.codebox` from the odot library.

As stated by Visi et. al in [4], "The data obtained from IMUs are morphologically very different from positional data returned by optical MoCap, since calculating absolute position from IMU data in real time is technically very difficult if not outright unfeasible, as the operation would require double integration of acceleration data". Nevertheless, since movement descriptors are used instead of raw data, the extracted features can be adapted to work on a cognitive level towards a different meaning of the conveyed movements of the subject. As also presented in [4], movement descriptors most commonly used with positional data can be adapted to work with IMU data.



```
/acc : [-5.06493, 1.43772, -8.08282],
/gyro : [0.137695, 0.398565, -0.173144],
/rotatexyz : [-19.9146, -14.5039, 1.11681]
```

```
#Assign data to modosc namespace
/pnts = "/modosc/points", #modosc points domain
assign(/pnts + /name/out + "/rot_rpy", /rotatexyz),
assign(/pnts + /name/out + "/acc", /acc),
assign(/pnts + /name/out + "/acc_mag", l2norm(/acc)), #computed to match the output of
[mo.acceleration]
assign(/pnts + /name/out + "/gyro", /gyro),

#Clean up
delete(/name/out), delete(/pnts), delete(/rotatexyz),
delete(/acc), delete(/gyro)
```

Fig. 3. The data coming from the smartphone's accelerometer, gyroscope and orientation (rotatexyz) are adapted through an odot o.expr.codebox to the modosc syntax, generating a modosc point named /phone1.

In this implementation the gestures data are then based on orientation, rather than on position. Furthermore, since we deal with one smartphone, every movement descriptor is calculated as an instantaneous value in the specific moment in time when it's computed. While this is enough for the implementation of most of modosc movement descriptors (such as velocity, acceleration, jerkiness and fluidity), quantity of motion requires the definition of a group of points. A group of points consisting of all the individual phone movements through time has been then defined, in order to calculate the quantity of motion as the weighted sum of the speeds of every single point (see Section II-D4). As shown in Figure 4, the selected descriptors are then gathered to be later mapped to the granular synthesis parameters.

In order to compute the periodic quantity of motion and the time and flow effort descriptors, as presented in Section II-D, the computed values over time of the single point movements have been calculated through list operators in Max/MSP, as displayed in Figure 5. Furthermore, a compensation of gravity acceleration has been performed on the z axis.

D. Mapping

Among the reasons for the creation of new instruments are the real-time control of new sound-worlds, and the control of existing timbres through alternative interfaces to enable individuals in the spontaneous creation of music [31]. The term *mapping* is used widely to indicate the mathematical process of relating the elements of one data set onto another. In computer music, mapping is often used in relation to algorithmic composition, where a parameter with a particular set of values is scaled or transformed so that it can be used to control another parameter [32]. As defined by Hunt and Wanderley in [33], in this research I consider mapping as the act of taking real-time performance data from an input device and using it to control the parameters of a synthesis engine.

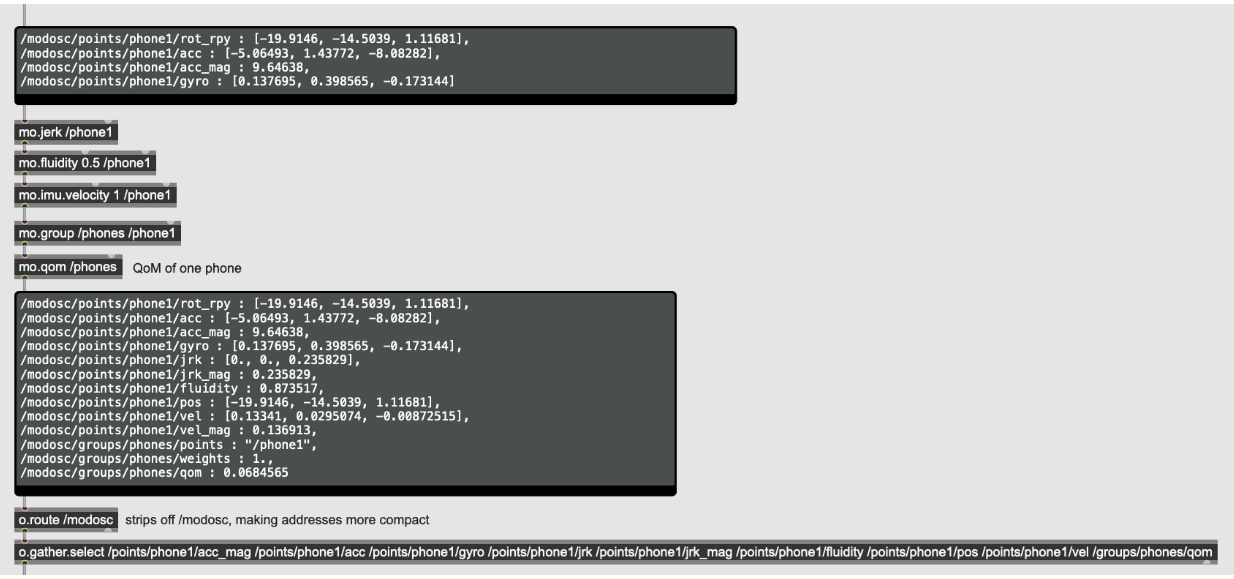
1) Different ways of mapping: The main question to be solved was related to the actual choice of which mapping strategy to implement. The ultimate goal in designing new Digital Musical Instruments (DMI) is to be able to obtain similar levels of control subtlety as those available in acoustic instruments, but at the same time extrapolating the capabilities of existing instruments [34]. Considering mapping as part of an instrument, two main directions could be deduced from the analysis of the existing literature:

- The use of generative mechanisms, such as neural networks.
- The use of explicitly defined mapping strategies.

Although I initially considered a possible mapping with a neural network, in the end my decision was to use explicit mapping strategies, presenting the advantage of keeping the designer in control of the implementation of each of the instrument's component parts, therefore providing an understanding of the effectiveness of mapping choices in each context [33].

2) Explicit mapping strategies: The available literature generally considers mapping of performer actions to sound synthesis parameters as a *few-to-many relationship* [35]. Considering two general sets of parameters, three intuitive strategies relating the parameters of one set to the other can be devised as [33]:

- *one-to-one*, where one synthesis parameter is driven by one performance parameter,
- *one-to-many*, where one performance parameter may influence several synthesis parameters at the same time, and



```
/modosc/points/phone1/rot_rpy : [-19.9146, -14.5039, 1.11681],
/modosc/points/phone1/acc : [-5.06493, 1.43772, -8.08282],
/modosc/points/phone1/acc_mag : 9.64638,
/modosc/points/phone1/gyro : [0.137695, 0.398565, -0.173144]

mo.jerk /phone1
mo.fluidity 0.5 /phone1
mo.imu.velocity 1 /phone1
mo.group /phones /phone1
mo.qom /phones    QoM of one phone

/modosc/points/phone1/rot_rpy : [-19.9146, -14.5039, 1.11681],
/modosc/points/phone1/acc : [-5.06493, 1.43772, -8.08282],
/modosc/points/phone1/acc_mag : 9.64638,
/modosc/points/phone1/gyro : [0.137695, 0.398565, -0.173144],
/modosc/points/phone1/jrk : [0., 0., 0.235829],
/modosc/points/phone1/jrk_mag : 0.235829,
/modosc/points/phone1/fluidity : 0.873517,
/modosc/points/phone1/pos : [-19.9146, -14.5039, 1.11681],
/modosc/points/phone1/vel : [0.13341, 0.0295074, -0.00872515],
/modosc/points/phone1/vel_mag : 0.136913,
/modosc/groups/phones/points : "/phone1",
/modosc/groups/phones/weights : 1.,
/modosc/groups/phones/qom : 0.0684565

o.route /modosc    strips off /modosc, making addresses more compact
o.gather.select /points/phone1/acc_mag /points/phone1/acc /points/phone1/gyro /points/phone1/jrk /points/phone1/jrk_mag /points/phone1/fluidity /points/phone1/pos /points/phone1/vel /groups/phones/qom
```

Fig. 4. Movement Descriptors computed on the adapted IMU data using the modosc functions. In the last line of code, the selected descriptors for acceleration magnitude, acceleration, gyroscope, jerkiness, jerkiness magnitude, fluidity, position (orientation), velocity and quantity of motion are selected and gathered to be later mapped to the granular synthesis parameters.

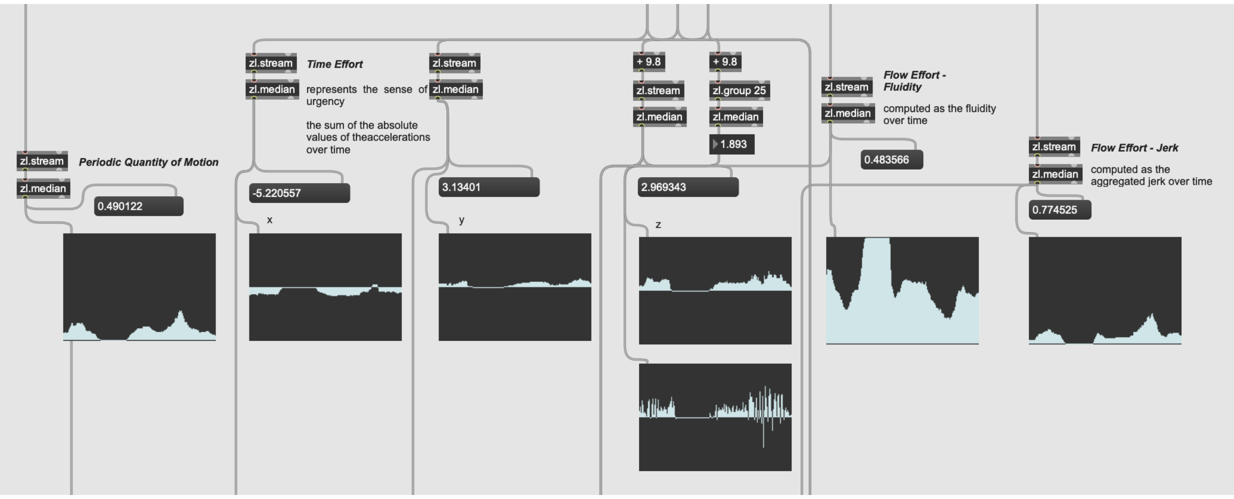


Fig. 5. Curves showing the computed movement descriptors through time. From left to right: periodic quantity of motion, time effort on the x axis, time effort on the y axis, time effort on the z axis, flow effort of fluidity, flow effort of jerkiness. On the second raw, under the time effort on the z axis, impulsive movement on the z axis have been calculated, to control the grain generation in the granular sampler. On the same axis, a compensation of gravity acceleration has been performed.

- *many-to-one*, where one synthesis parameter is driven by two or more performance parameters.

Concerning explicit mappings between two sets of parameters, many ways of abstraction of the performance parameters have been proposed, from perceptual parameters [36] to focusing on continuous parameter changes represented by gestures produced by the user [34] [37].

3) Three-layer mapping: In designing an explicit mapping for this system, I decided to adopt a three-layer mapping model. In this model, the first layer is interface-specific, since it converts the incoming sensors information into a set of chosen (intermediate or abstract) parameters that could be perceptually relevant or derived from other forms of interaction, like gesture. These are then mapped – in a second independent mapping layer – onto the specific controls needed for a particular synthesis engine. The advantages of this model is that the first mapping layer is a function of the given input device and the chosen abstract parameters, while the synthesis engine could be changed by changing the second mapping layer, e.g. from granular synthesis to physical modelling or FM synthesis, since the second mapping layer is not dependent on the control parameters directly. The proposed mapping layers

[38] are thus:

- Extraction of meaningful performance parameters.
- Connection of performer's (meaningful) parameters to some intermediate representation set of parameters (for instance, perceptual or abstract).
- Decoding of intermediate parameters into system-specific controls.

Specifically, as shown in the full mapping model presented in Figure 6, the first mapping layer extracts meaningful parameters from the control gesture variables of the smartphones and computes the movement descriptors, as described in Sections II-D and III-C. The second mapping layer then connects the abstract parameters derived by the first layer to the granular synthesis parameters, using the following strategies:

- Grain Duration range: boolean comparison between Fluidity and Jerk (many-to-one)
- Pitch: Periodic Quantity of Motion (one-to-one)
- Grain Start range: Time Effort on the X axis and Time effort on the Z axis (many-to-one)
- Play: Acceleration on the three axis (many-to-one)
- Pitch Holder: Toggle in the Max/MSP patch (one-to-one direct mapping)

This latter parameter has been implemented outside of the smartphone to give better control of the pitch. In a previous research, a Forse-Sensing Resistor (FSR) attached to the smartphone has been used but evaluation on a narrow range of participants highlighted the low affordance of it, as well as its low comfort. In this implementation, then, I decided to use a toggle button in the Max/MSP patch to provide a more intuitive control of this parameter. Furthermore, in this way the hand holding the phone is free from any wiring connections, (as on the contrary happened with the FSR) while the other hand is free to push the toggle using the computer's trackpad or mouse.

IV. Results

To present the application in a user-friendly context, the patch has been featured with a presentation mode, where only the object essential for the interaction are displayed. As shown in Figure 7, an audio file can first be selected. Then, a subpatch takes care of the synchronisation of the smartphone with the PC, and the sending and handling of OSC messages. The user is now able to move the smartphone and play the granular sampler through gestures. As previously reported, the only control that has been kept out of the gesture tracking is the one related to pitch holding. This can be performed by pressing a toggle directly on the Max/MSP patch.

A video demo of the application in the context of a free exploration using a piano sample is available[2].

Furthermore, data for the computed descriptors have been recorded during the video demo, and plotted in Figure 8. Here we could see the relation of some of the descriptors (e.g. Periodic Quantity of Motion and Pitch) as well as the different meaning of the visual representation curves of different levels of descriptors (e.g. the lower level descriptors of acceleration compared to higher level descriptors of Time Effort along the relative axis).

V. Discussion

The goal of this project was to design a new digital instrument, based on granular synthesis and played through gesture. Specifically the idea was to relate to the studies of Pierre Schaeffer in terms of sonic objects and their typology and morphology. As reported in [8], "typology can be summarized as denoting the overall shape, or envelope, of any sonic object, with regards to its dynamic (loudness), timbre and pitch-related content". This can be seen as a first sorting of sonic objects, considering their most prominent perceptual features.

Considering the three general dynamic envelope categories proposed by Schaeffer (sustained, impulsive and iterative), I believe that the proposed implementation successfully managed to relate them to relative motion categories, through the computed mid-level descriptors. On the other hand, a task that has been proven to be particularly difficult to fulfill efficiently is the one related to the classification of pitch (tonic stable, tonic varying and nontonic). In this project, pitch has been mapped efficiently and logically to the amount of movement performed by the user through time, computed as Periodic Quantity of Motion. Even though this seemed a good way of interpreting the pitch feature, further investigation is needed in terms of pitch classification and pitch holding. A better mapping of the pitch content could lead to a more complete instrument, enabling a clearer division of the pitch catergories.

Regarding morphology, considered by Schaeffer as the more internal features of the sonic objects, the main two dimensions of gait (slower fluctuations) and grain (fast fluctuation) have been covered thanks to the grain start and duration settings of the sampler. Nevertheless, more control could be added in future, either on the grain generation aspect (thus working on the amplitude envelope) or implementing some internal modulation that could add more morphological features like vibrato or tremolo. Many other features of the wide taxonomy of sonic object could be added to the project in future developments, like manipulation of the texture or introduction of sound hierarchies with composite and concatenated objects. However, I believe that this project stands out as a valid personal starting point in the research of gestural interaction

[2] https://drive.google.com/file/d/1XAone0SUF7BIyeNA1mu3v0QioulnC606/view?usp=sharing

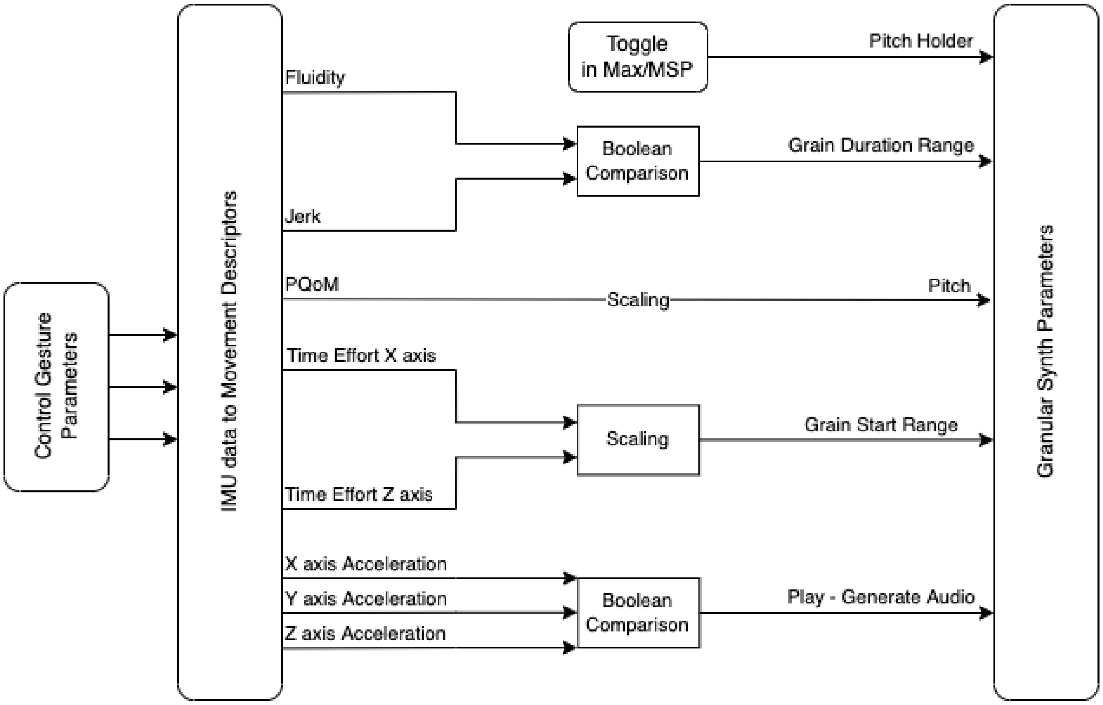


Fig. 6. Three-layer mapping model implemented in this research. The control gesture variables from the smartphone were mapped to a first layer, extracting meaningful parameters and computing mid-level movement descriptors. The abstract parameters derived by the descriptors were mapped to the synthesis parameters of the granular synthesizer. A toggle in the Max/MSP patch controls the holding of the pitch, giving a direct mapping outside the embodied environment of movement descriptors.

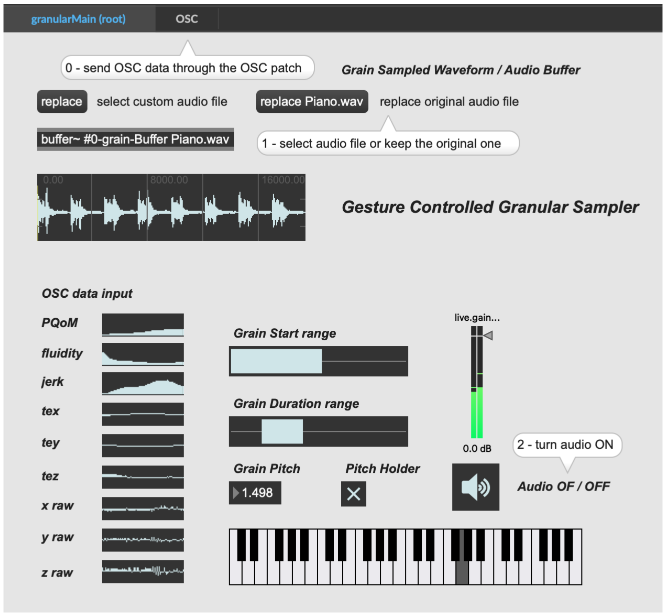


Fig. 7. The complete Max/MSP patch in presentation mode. This consists of an essential overview of the main elements of the project, to make the interaction more intuitive.

with sonic object, especially for the awareness of mapping as a very important component of such an application and for the computation of higher level descriptors. It can be concluded that when the mid-level descriptors are used, after the initial excitation given by the first gesture, in the interaction context the movement can be considered as representing a higher level of hierarchy, driving the resulting sonic outcome. Here, a feedback perceptive loop between the user and the system enhances the possibilities of the system itself, resulting in a perceptually evolving sonification of the gestures.

VI. Conclusions

The present implementation of embodied interaction techniques contributes to a larger context of studies on movement descriptors and creative usage of them. It is hoped that the research done during this study can work as a personal foundation for future investigations in these fields, as well as in the area of tools for digital assisted composition and improvisation.

References

[1] E. M. Leman and D. Cirotteau, “Sound to sense, sense to sound: A state-of-the-art,” 2005.

[2] “A review of computable expressive descriptors of human motion,” *ACM International Conference Proceeding Series*, vol. 14-15-August-2015, pp. 21–28, 8 2015.

[3] F. Visi, R. Schramm, and E. Miranda, “Gesture in performance with traditional musical instruments and electronics: Use of embodied music cognition and multimodal motion capture to design gestural mapping strategies,” *ACM International Conference Proceeding Series*, pp. 100–105, 2014.

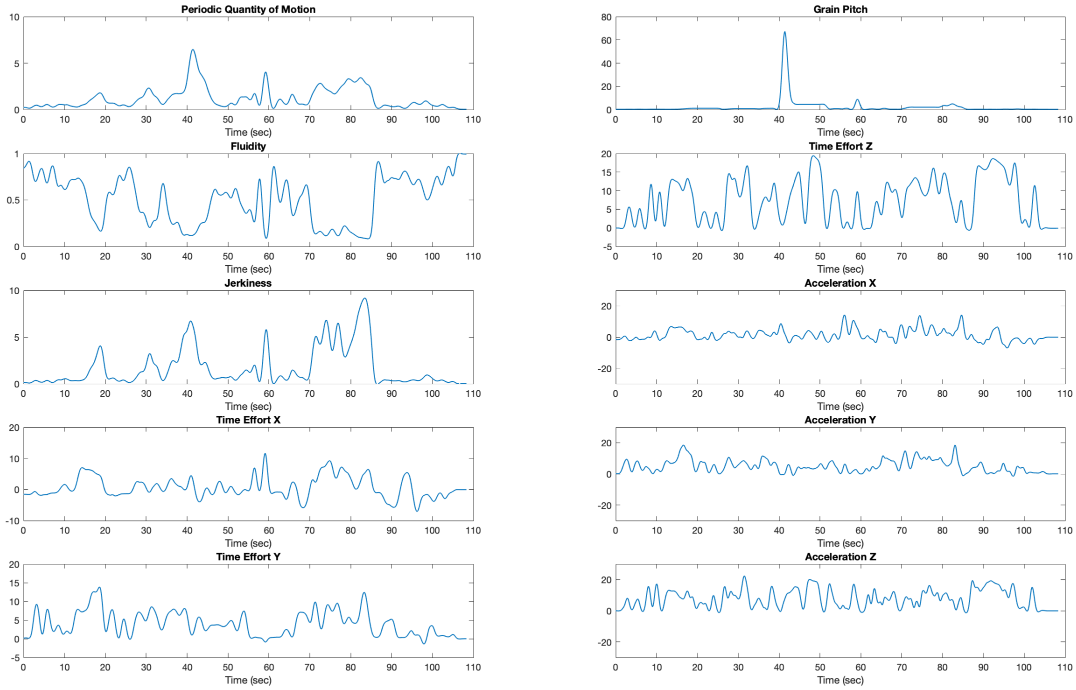


Fig. 8. Plots showing the variations of data through time, in the form of the computed descriptors for Periodic Quantity of Motion, Fluidity, Jerkiness, Time Effort and Acceleration over the three axis. The data have been recorded during a short demo of the application, available at https://drive.google.com/file/d/1XAone0SUF7BIyeNA1mu3v0QioulnC606/view?usp=sharing. A second order Butterworth IIR Lowpass filter with a cutoff frequency of 0.75 Hz was implemented to reduce the noise of the data.

[4] F. Visi, E. Coorevits, R. Schramm, and E. R. Miranda, “Musical instruments, body movement, space, and motion data: Music as an emergent multimodal choreography,” *Human Technology*, vol. 13, pp. 58–81, 2017.

[5] J. van Dijk, “Designing for embodied being-in-the-world: A critical analysis of the concept of embodiment in the design of hybrids,” *Multimodal Technologies and Interaction 2018, Vol. 2, Page 7*, vol. 2, p. 7, 2 2018. [Online]. Available: https://www.mdpi.com/2414-4088/2/1/7/htmhttps://www.mdpi.com/2414-4088/2/1/7

[6] A. Oulasvirta and K. Hornbæk, “Hci research as problem-solving,” *Conference on Human Factors in Computing Systems - Proceedings*, pp. 4956–4967, 5 2016. [Online]. Available: http://dx.doi.org/10.1145/2858036.2858283

[7] K. Hornbaek and A. Oulasvirta, “What is interaction?” *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, 2017. [Online]. Available: http://dx.doi.org/10.1145/3025453.3025765

[8] R. I. Godøy, “Sonic object cognition,” *Springer Handbooks*, pp. 761–777, 2018. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-662-55004-5_35

[9] P. Schaeffer, *Traité des objets musicaux*, Éditions du seuil ed., 1966. [Online]. Available: https://www.seuil.com/ouvrage/traite-des-objets-musicaux-pierre-schaeffer/9782020026086

[10] R. Shusterman, “Thinking through the body : essays in somaesthetics,” p. 368, 2012.

[11] K. Höök, “A soma design manifesto,” *Designing with the Body*, 11 2018. [Online]. Available: https://direct.mit.edu/books/book/4131/chapter/170684/A-Soma-Design-Manifesto

[12] P. Hackney, “Making connections: Total body integration through bartenieff fundamentals,” 09 2003.

[13] T. Flash and N. Hogans3, “The coordination of arm movements: An experimentally confirmed mathematical model’,” *The Journal of Neuroscience*, vol. 5, pp. 1688–1703, 1985.

[14] D. Fenza, S. Canazza, and A. Rodà, “Physical movement and musical gestures: A multilevel mapping strategy.” [Online]. Available: www.eyesweb.org

[15] D. Glowinski, N. Dael, A. Camurri, G. Volpe, M. Mortillaro, and K. Scherer, “Toward a minimal representation of affective gestures,” *IEEE Transactions on Affective Computing*, vol. 2, pp. 106–118, 4 2011.

[16] F. Visi and L. Dahl, “Real-time motion capture analysis and music interaction with the modosc descriptor library.” [Online]. Available: https://github.com/motiondescriptors/modosc

[17] “OpenSoundControl.” [Online]. Available: https://ccrma.stanford.edu/groups/osc/index.html

[18] “Max/MSP, Cycling ’74.” [Online]. Available: https://cycling74.com/

[19] D. Gabor, “Acoustical quanta and the theory of hearing,” *Nature*, vol. 159, pp. 591–594, 1947. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/20239709/

[20] T. H. Park, *Introduction to digital signal processing : computer musically speaking*, W. S. P. Company, Ed. World Scientific, 2009.

[21] J. Maccallum, R. Gottfried, I. Rostovtsev, J. Bresson, and A. Freed, “Dynamic message-oriented middleware with open sound control and odot - archive ouverte hal.” [Online]. Available: https://hal.archives-ouvertes.fr/hal-01165775

[22] L. Dahl and F. Visi, “Modosc: A library of real-time movement descriptors for marker-based motion capture,” *ACM International Conference Proceeding Series*, 6 2018.

[23] M. Wright and A. Freed, “Open soundcontrol: A new protocol for communicating with sound synthesizers.” [Online]. Available: http://www.cnmat.berkeley.edu/People

[24] A. Freed and A. Schmeder, “Features and future of open sound control version 1.1 for nime.”

[25] S. O. H. Madgwick, “AHRS algorithms and calibration solutions to facilitate new applications using low-cost mems.” [Online]. Available: https://ethos.bl.uk/OrderDetails.do?uin=uk.bl.ethos.681552

[26] F. G. Visi and A. Tanaka, “Interactive machine learning of musical

gesture," *Handbook of Artificial Intelligence for Music*, pp. 771–798, 11 2020. [Online]. Available: https://arxiv.org/abs/2011.13487v1

[27] F. Visi, "GIMleT – Gestural Interaction Machine Learning Toolkitk," 2020. [Online]. Available: https://github.com/federicoVisi/GIMLeT

[28] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A convolutional network for real-time 6-dof camera relocalization."

[29] "Real-time human pose estimation in the browser with tensorflow.js | by tensorflow | tensorflow | medium." [Online]. Available: https://medium.com/tensorflow/real-time-human-pose-estimation-in-the-browser-with-tensorflow-js-7dd0bc881cd5

[30] "TouchOSC, Fully modular touch control surface for OSC and MIDI." [Online]. Available: https://hexler.net/touchosc-mk1

[31] R. Kirk, M. Abbotson, R. Abbotson, A. Hunt, and A. Cleaton, "Computer music in the service of music therapy: the midigrid and midicreator systems," *Medical Engineering Physics*, vol. 16, pp. 253–258, 5 1994.

[32] T. Winkler, "Composing interactive music : techniques and ideas using max," p. 350, 1998.

[33] A. Hunt and M. M. Wanderley, "Mapping performer parameters to synthesis engines," *Organised Sound*, vol. 7, pp. 97–108, 2002.

[34] M. M. Wanderley and P. Depalle, "Gestural control of sound synthesis," *Proceedings of the IEEE*, vol. 92, pp. 632–644, 2004.

[35] M. Lee and D. Wessel, "Connectionist models for real-time control of synthesis and compositional algorithms." *Proc. of the 1992 Int. Computer Music Conference*, pp. 277–280, 1992.

[36] G. Garnett and C. Goudeseune, "Performance factors in control of high-dimensional spaces," *Proc. of the 1999 Int. Computer Music Conf.*, pp. 268–271, 1999.

[37] A. Mulder, S. Fels, and K. Mase, "Empty-handed gesture analysis in max/fts," *Kansei, The Technology of Emotion. Proc. of the AIMI Int. Workshop*, 1997.

[38] A. Hunt, M. M. Wanderley, and M. Paradis, "The importance of parameter mapping in electronic instrument design," *International Journal of Phytoremediation*, vol. 21, pp. 429–440, 2003.