Interacting with 3D Reactive Widgets for Musical Performance

Florent Berthaut, Myriam Desainte-Catherine, Martin Hachet

▶ To cite this version:

Florent Berthaut, Myriam Desainte-Catherine, Martin Hachet. Interacting with 3D Reactive Widgets for Musical Performance. Journal of New Music Research, Taylor & Francis (Routledge), 2011, 40 (3), pp.253-263. hal-00633750

HAL Id: hal-00633750
https://hal.archives-ouvertes.fr/hal-00633750

Submitted on 19 Oct 2011

Interacting with 3D Reactive Widgets for Musical Performance

Florent Berthaut[1] , **Myriam Desainte-Catherine**[1] , **Martin Hachet**[2]

[1]Université de Bordeaux, Scrime, LaBRI

[2]INRIA, LaBRI

florent@hitmuri.net, myriam@labri.fr, hachet@labri.fr

Abstract. *While virtual reality and 3D interaction open new prospects for musical performance, existing immersive virtual instruments are often limited to single process instruments or musical navigation tools. We believe that immersive virtual environments may be used to design expressive and efficient multi-process instruments. In this paper we present the 3D reactive widgets. These graphical elements enable efficient and simultaneous control and visualisation of musical processes. Then we describe Piivert, a novel input device that we have developed to manipulate these widgets, and several techniques for 3D musical interaction.*

1. Introduction

Graphical musical interfaces have many advantages over hardware controllers. They provide a direct access to a high number of parameters of an unlimited number of sound processes. They can also be used to display rich information about each process and thus facilitate manipulation of multi-process instruments as stated by Jordà [Jordà, 2003]. The purpose of our work is to explore the possibilities provided by 3D interaction and virtual environments. 3D navigation can be a good metaphor for navigation in musical pieces or exploration of musical structures, thus allowing musicians to visualise and manipulate large sets of sound processes. New interaction techniques developed in these fields of research open possibilities for musical interaction and can also be combined with traditional techniques. Immersion also provides new sensations to musicians and to the audience. However, several questions arise regarding the visualisation and the manipulation of sound processes in these environments. Their appearance must be chosen in order to facilitate identification and understanding. 3D interaction techniques must be adapted to the specific needs of musical interaction, such as expressiveness, efficiency, and minimum latency.

2. Virtual reality and musical performance

Relatively little research has been done in the field of 3D interaction and virtual reality for music. Some works focus on navigation in musical environments, like the virtual groove in the Phase project [Rodet et al., 2005], or the audiovisual grains in Plumage [Jacquemin et al., 2007]. The applications developed by Wozniewski et al. [Wozniewski et al., 2006] also rely on user movements to either control the spatialization of pre-recorded sound sources, or apply effects on the sound of an acoustic instrument. Some immersive instruments are single processes instruments, i.e. instruments that allow musicians to interact with only one synthesis process, such as the Virtual Xylophone, the Virtual Membrane, or the Virtual Air Guitar developed by Mäki-Patola et al. [Mäki-Patola et al., 2005] and the sculpting instruments developed by Mulder [Mulder, 1998]. Finally, among the existing multi-process 3D instruments, part of them, like the WAVE software from Valbom et al. [Valbom and Marcos, 2005] or the application developed by Martin Naef et al. [Naef and Collicot, 2006], provide limited visual feedback and interaction since they tend to emulate hardware controllers. Other instruments rely on gaming software or devices, like the 3D instrument Fijuu [Oliver and Pickles, 2007], or the collaborative musical First Person Shooter q3osc [Hamilton, 2008]. They offer new interaction techniques and interesting visualisations, but they do not take advantage of immersive environments potential. None of these applications combine immersion, simultaneous control of multiple processes, expressive interaction techniques and complex visual feedback.

3. The 3D Reactive Widgets

3.1. Main approach

Our research focuses on using 3D immersive environments for musical interaction. These environments indeed add opportunities in terms of temporal or hierarchical navigation by means of 3D movements, as it can

be seen on figure 1. They also enable the design of new interaction techniques and paradigms, for example by benefiting from enhanced manipulation and visualisation spaces. Furthermore, immersion can improve the experience of musicians who will better perceive 3D environments with stereoscopic display and head-tracking. It will also improve the experience of the audience, if they are equipped with stereoscopic glasses, both for the spectacular aspect and for the understanding of the musicians playing.

As said in the previous section, most virtual reality instruments are single process instruments, i.e. instruments that allow the control of only one synthesis or effect process, or musical navigation tools. However, we believe that the main advantage of graphical musical interfaces is to give the opportunity of handling multi-process instruments with control on and visual feedback from the selected sound processes. This is why we chose to rely on the concept of reactive widgets described by Golan Levin [Levin, 2000] and used for example by Sergi Jordà [Jordà, 2005] in the FMOL instrument. A reactive widget is a graphical component which allows both manipulation and visualisation of a musical process. Its graphical parameters are connected to the parameters of the associated musical process. These connections are bidirectional, so that graphical changes are reflected in the sound process and that musical events are displayed in return by the widget. The efficiency of this concept lies in the shortening of the "indirection degree" described by Michel Beaudoin-Lafon [Beaudoin-Lafon, 1999] because there is direct manipulation of the "objects of interest", in our case the visualised sound processes. These observations led us to adapt the concept of the reactive widgets to 3D immersive environments, as it can be seen in figure 3.

Figure 1: Example of an immersive instrument with complex musical structures and navigation
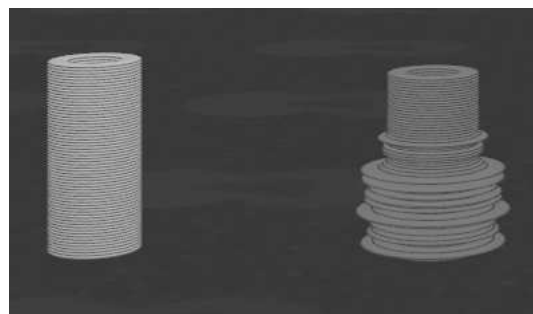

Figure 2: Two example 3D Reactive Widgets: the one on the right shows the spectrum of its associated sound process using its shape.

Figure 3

3.2. Audiovisual Mappings

Several questions emerge from the concept of the 3D reactive widgets. First of all, one must choose which sound and musical parameters should be controlled and visualised. Symmetrically, one must decide which graphical parameters should be used to manipulate and display the sound processes. One must also ensure a correct representation of the sound parameters using mappings between those audio and graphical parameters. Finally, since several mappings can be combined on a single widget, it is essential to study how many, and which, mappings can be used simultaneously without disrupting the perception of each parameter.

These questions have already been discussed in some user studies, such as [Giannakis, 2006]. But these studies investigate user preferences in a music composition perspective, i.e. with static sound parameters, and using two dimensional graphics. In [Berthaut et al., 2010a], we describe a user study that focuses rather on mapping performances and dynamic parameters, using perceptive sound parameters (volume, pitch, spectral centroid and spectral irregularity) and mostly preattentive visual features, on 3D objects. This study is composed of three experiments such as mapping preferences, mapping performances and

mapping combinations performances. The performances are tested in an audiovisual object identification task. Among the results of this study, we find preferences similar to previous studies, mostly corresponding to physical metaphors such as size and volume, distance and volume or size and pitch but also metaphors of hardware controllers. Interestingly, it appears that performances depend more on mapping scales than on the choice of the mappings themselves. Finally, mapping combinations only affect performances when several graphical parameters disrupt each other. This leads us to the conclusion that mapping presets should be defined. These presets should use visually independent graphical parameters and appropriate mapping scales in order to ensure correct visualisation of the sound processes.

4. Interacting with the 3D reactive widgets

3D reactive widgets ease the representation of sound processes with many parameters in virtual environments. Research in virtual reality and 3D interaction has led to the development of many techniques for efficient manipulation of 3D entities. However, these techniques are generic and do not take into account the specificities of musical interaction. We propose several interaction techniques, described in [Berthaut et al., 2010c] and [Berthaut et al., 2009], fitted to the manipulation of sound processes in virtual environments. They rely on instrumental gestures categories defined by Cadoz [Cadoz, 1999] such as selection gestures, modulation gestures and excitation gestures. We use these techniques, as depicted on figure 4, to interact with *3D reactive widgets*.
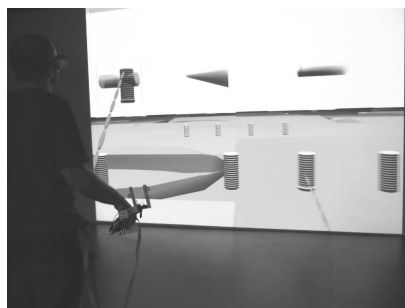

Figure 4: An immersive musical environment with four *3D reactive widgets*, *three tunnels* and *Piivert* (first prototype). Four other widgets can be seen further in the environment.

4.1. Piivert

4.1.1. Overview

While graphical interfaces may facilitate interaction with multiple sound processes, musical interaction has some specific requirements that need to be addressed. For example, excitation gestures, which produce the sound in acoustic instruments, usually require expressiveness, temporal accuracy and haptic feedback (at least passive). However, most 3D input devices are composed of joysticks/buttons and a position/orientation tracking system. Tracking systems allow the manipulation of 3D entities, e.g. virtual tools or objects. These manipulations can be done with sufficiently high resolution and low latency but the tracking systems usually lack precise haptic feedback, which is hard to provide when users need to move freely in front of a large screen or when there are several users collaborating. On the other hand, buttons and joysticks provide passive haptic feedback but lack expressiveness because of their low resolution.

We believe that splitting interaction with *3D reactive widgets* according to Cadoz's categories of musical gestures may improve 3D interaction with sound processes. Excitation gestures may be performed using Force Sensitive Resistors (FSR), i.e. pressure sensors, and directly transmitted to the sound processes. This overcomes the latency problem induced by tracking data analysis, data transmissions and graphical rendering or collision detection threads. Moreover, these sensors provide passive haptic feedback, which is fundamental for excitation gestures.

Selection gestures, by which musicians choose instrument components, like a key on a piano, can then be done graphically. 3D interaction techniques may give direct and efficient access to sound processes. Furthermore, latency problems due to tracking systems may be overcome for these gestures using specific techniques.

Modulation gestures may be performed using sensors, but graphical modulation provides some advantages, as explained in the following section.

4.1.2. Input Device

The Piivert device draws inspiration from The Hands developed by Waisvisz [Waisvisz, 1985]. As depicted on figure 5, it is composed of infrared targets placed on its extremity and of pressure sensors located below the thumbs, index fingers, middle fingers and ring fingers of each hand. Data from the sensors is encoded on 12 bits (compared to 8 bits of joysticks or 1 bit of push buttons) and latency is of approximately 5ms. More details are given in section 5.
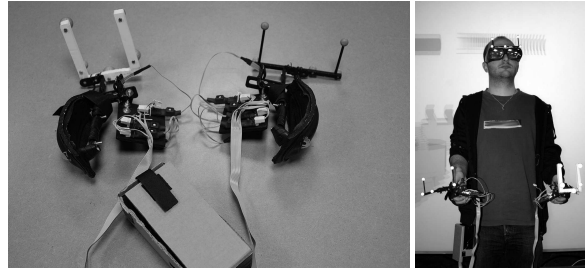

Figure 5: Current Piivert device

4.1.3. Interaction Techniques

Two classes of interaction techniques have been developed for Piivert. Techniques of the first category rely on the FSR sensors and consist of excitation gestures and other musical controls. These techniques are based on percussion gestures and can be divided in low-level and high-level gestures. We have defined a notation for these gestures, which can be seen on figures 6 and 7. There are three low-level gestures. The *Hit* gesture is a short pressure on one of the FSR sensors. The *Pressure* gesture is a slow and long pressure on the sensor. The *Release* gesture is performed by taking the finger off the sensor. These low-level gestures are particularly fitted to excitation gestures because of their immediacy. Furthermore, they provide a velocity parameter, and a duration parameter for *Pressure* gestures.
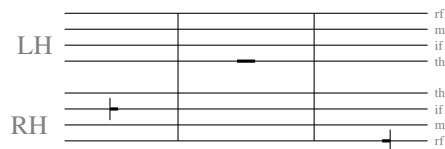

Figure 6: Low level gestures notation (RH: right hand, LH: left hand, th/if/mf/rf: fingers). From left to right: Hit, Pressure, Release

They are combined to constitute high-level gestures. For example, a *Flam* is a succession of two *Hit* gestures, while a *Roll* is a succession of three *Hit*. These gestures provide several parameters, such as fingers used, number of hands used, direction of gesture, duration and velocity of component gestures. Therefore, they may be used to trigger musical controls or complex excitation gestures, while still integrating perfectly in the stream of low-level excitation gestures.
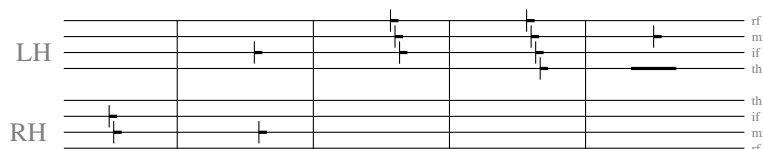

Figure 7: High-level gestures notation. From left to right: One hand Flam, Two hands Flam, Three strikes Roll, Four strikes Roll, Combination of "Hit" and "Pressure"

Techniques of the second category, which can be seen on figure 8, rely on 6DOF tracking and allow for selection gestures. In order to select and to grab the *3D reactive widgets*, musicians manipulate a virtual ray, which is commonly used in virtual reality applications [Bowman and Hodges, 1997]. It was evaluated as an efficient technique especially for near objects, by Poupyrev et al. [Poupyrev et al., 1998]. It gives a feeling of continuity from the real world and provides sufficiently accurate and fast pointing in the

virtual environment. However, it needs to be modified to fit the specificities of musical interaction. First of all, we define an *interaction plane* at a fixed depth (facing users point of view, on the X-Y axis) where *3D reactive widgets* are located in order to facilitate access and manipulation. The virtual ray technique is more efficient with near objects and we can reduce manipulation errors, especially for collision of several widgets, by limiting the movements on Z-axis. Depth can then be used for small translations of the 3D reactive widgets, or to display structures, such as hierarchical structures used in [Berthaut et al., 2010b], without interfering with most manipulations. We also expand the bounding boxes of widgets, so that they are bigger than the actual 3D object. Thus widgets get selected before musicians point them exactly. This improves the playing of fast sequences of excitation gestures. Furthermore, the movements of the ray in the bounding boxes are low-pass filtered to avoid unwanted jumps while exciting the widget. The *Split Ray* technique allows musicians to select and manipulate several widgets simultaneously with one hand. Musicians select several widgets and the virtual ray is split in several rays, one going to and grabbing each selected widget. They may then excite independently each widget with separate fingers and simultaneously move all the widgets in the environment. Here graphical interaction offers more flexibility than one could get from hardware controllers. The last technique developed is the *vibrating ray*. Its main purpose is to preserve the immersion by representing the excitation gestures done with the pressure sensors as vibrations propagating along the virtual rays.
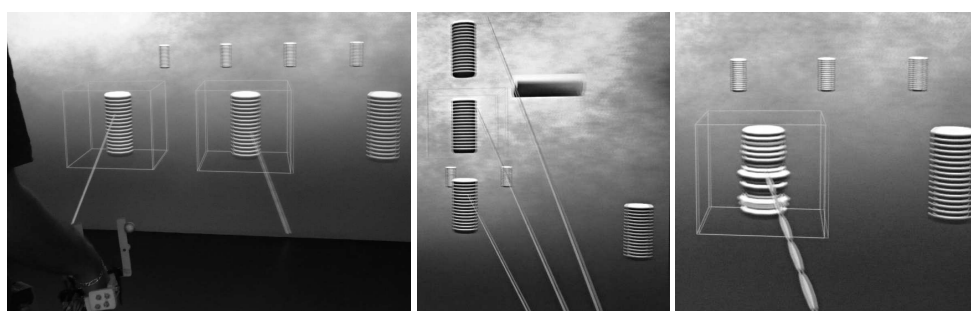

Figure 8: Graphical Interaction techniques: (from left to right) Larger Bounding Boxes, Split Ray, Vibrating Ray

4.1.4. Evaluation

The goal of this user study was to compare Piivert with a graphical excitation technique, which we have called *Virtual Drums*, in a simple sound triggering task. The *Virtual Drums* technique is analog to most gestural controls used to play virtual drums, for example with the Nintendo Wii remote-controller [Bott et al., 2009]. In particular, we wanted to know if the approach we propose may improve temporal accuracy, and reduce error rates for such a task. Since the final users of Piivert are musicians, the chosen subjects for this study were sixteen right-handed experienced musicians, 15 men and 1 woman, aged between 23 and 41. It should be noted that, as musical instruments usually imply a learning process, this user study may only give an indication about Piivert's efficiency. A second study after a long learning period may be more conclusive. Other techniques and devices may be tested but for this preliminary evaluation, we focused on simple drums sounds triggering.

With the Virtual Drums technique, subjects trigger sounds by hitting the objects with the ray, as one would do with a drumstick. With Piivert, subjects trigger sounds by pointing the objects with the ray and performing a low-level hit gesture with one finger. In addition, larger bounding boxes are used with Piivert, which is impossible to do with the Virtual Drums technique since subjects must know exactly when they are going to hit the object. The two techniques are tested in random order, at two different tempi, i.e. musical speed, 120 beats per minute (BPM) and 170 BPM. The tempi were chosen so they are neither too slow nor too fast. In addition, a metronome is used so subjects are not disturbed by the slow tempo which may be harder to follow. For each tempo, four trials have to be completed. The trials are composed of four sequences, with an additional training sequence at the beginning of each technique, to reduce the learning effect. These sequences of one to four notes are randomly generated using a set of three sounds corresponding to the three boxes displayed. During a sequence, notes are played and their corresponding boxes change color accordingly. Subjects are asked to play over these sequences, i.e. trigger the correct objects with the correct rhythm. The sequences are played twice before subjects repeat it, to prevent memorization capabilities of subjects from interfering with the results.

For each condition, we recorded the temporal accuracy, i.e. times differences between sequences' notes and subjects actions, in milliseconds, and the error rates, i.e. missed or extra hits. The results can be
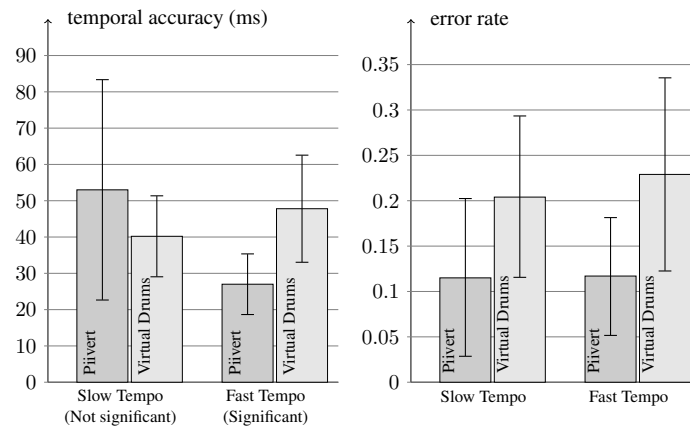
seen on figure 9.

Figure 9: Temporal Accuracy and Error rates of Piivert and Virtual Drums at fast and slow tempi.

The results of a two-way ANOVA showed no main effects of techniques or tempi for temporal accuracy. In contrast, the interaction term was significant ($F_{(1,15)} = 6.655, p < 0.05$). Paired t-test analyses were performed for each tempo. The results for slow tempo revealed no significant difference. At fast tempo a significant difference was found between techniques ($T_{15} = 3.737, p < 0.05$), indicating that Piivert is more accurate than the Virtual Drums. Concerning the error rates, a two-way ANOVA revealed a significant difference between techniques ($F_{(1,15)} = 7.484, p < 0.05$), with less errors for Piivert. Neither the difference between tempos nor the interaction effect were significant. T-test analyses were also performed for each tempo. The results showed significantly less errors for Piivert at slow tempo ($T_{15} = 2.35, p < 0.05$) and fast tempo ($T_{15} = 2.239, p < 0.05$).

At slow tempo, Piivert is slightly, but not significantly, less accurate than Virtual Drums. This may be due to the necessity of learning how to use Piivert, in contrast to the Virtual Drums, which most subjects reported as more straightforward. Indeed, all subjects commented that it took them some time to get used to the fingers movements range with Piivert, and to the "point and play" movements. Slower tempo may also induce more hesitations for the fingers movements, resulting in less accuracy. On the other hand, at fast tempo, accuracy slightly decreases for Virtual Drums due to large and fast movements required. We can presume that at faster tempi, the Virtual Drums technique would be even harder to use, and even less accurate. In contrast, accuracy increases for Piivert and is significantly higher. This may indicate that subjects got used to Piivert. Musical sequences at faster tempo are easier because there is more temporal continuity between notes, and thus hesitation between hits is reduced. Such an effect can be seen with the results of Piivert. At both tempi, Piivert gives less errors than the Virtual Drums technique. Indeed, subjects reported having difficulties to hit objects correctly with the Virtual Drums. Either they missed the objects because the tempo implied too fast and thus inaccurate movements, or they hit them several times by going through them and re-entering them, again because of too fast movements. They also reported that the Virtual Drums technique lacked haptic feedback, in contrast to Piivert. One must also note that the three audiovisual objects were at the same distance on the Z-axis. Different depths may have been even more difficult to handle with the Virtual Drums technique. Finally, this first study indicates that Piivert, after some learning, improves accuracy of selection and excitation gestures in an immersive environment.

4.2. Tunnels

Modulation gestures can be done using the pressure sensors of Piivert. In addition, graphical interaction enables multiple modulation scales, variable resolution, modulation combinations and visual feedback. In order to modulate graphical parameters of *3D reactive widgets*, we have thus developed 3D tools called *Tunnels*.

4.2.1. Overview

These *Tunnels* are 3D objects associated to one or several graphical parameters with one or several scales. Each of them is composed of aligned hollow cylinders whose appearance reflects the variation of the graphical parameters along the *Tunnel*, as depicted on figure 10. For example, a *Tunnel* that is dedicated to the control of the height (scale on the Y axis) is a cone-shaped widget, small at one end and tall at the other end. When a *3D reactive widget* is moved through a *Tunnel*, its graphical parameters and the associated sound parameters are modified according to the parameters scales.
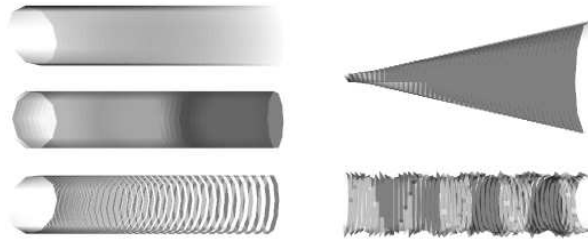
Figure 10: *Tunnels* with a single parameter (graphical/sound): (left) Transparency/Noisiness, Hue/Pitch, Scattering/Reverberation, (right) Scale/Volume, Rotation Speed/Tempo

Compared to standard 1D or even 2D graphical sliders, the *Tunnels* ease the use of non-standard scales (discrete/non-linear) by giving a visual feedback on the modulated parameters. By combining the *Tunnels* with *3D reactive widgets* one may also lower the number of graphical elements displayed. For example, if one needs to modulate 5 parameters on 10 sound processes, one will go from 50 standard 1D graphical sliders to only 15 graphical elements (10 *3D reactive widgets* + 5 *Tunnels*). Furthermore, modulations can be activated in several ways as described in section 4.2.2. The *Tunnels* also enable combination of modulations while preserving distinct representations of each sound parameter.

The *Tunnels* may be located behind the *Interaction Plane* to prevent from overloading it and to avoid accidental modulations when moving widgets on the plane. They may then be easily accessed by simply pushing *3D reactive widgets* towards them. Here the depth provided by virtual 3D environments ease the use of *Tunnels*, especially their activation.

4.2.2. Use

There are two ways to the *Tunnels*. When the musician wants to modulate one *3D reactive widget* only, he may grab it and move it through a *Tunnel*. By entering the *Tunnel* vertically, the musician can jump to a specific value. By starting horizontally, the modulation begins by the value at the end of the graphical parameter scale. Modulation is then done horizontally. If the musician moves the widget away from the *Tunnel* on the same side as it entered, the widget takes back its previous parameter value. On the other hand, if the musician releases the widget on the other side, i.e. if the widget crosses the *Tunnel*, the parameter keeps the new value. This allows musicians to make quick and temporary modulations or to restore a value that is not on the current scale. This "single object" modulation can be seen on figure 11. The *Tunnels* use the affordance principle [Gibson, 1977] since their shape suggests the action of passing the widgets through them.
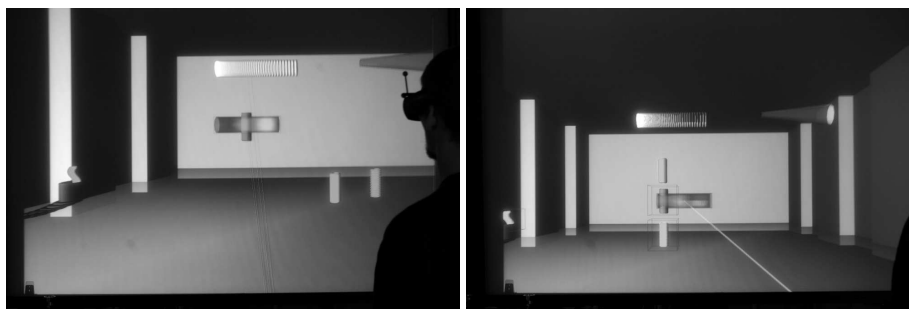

Figure 11: Single widget modulation (left) where the widget is moved inside a *Tunnel* and simultaneous modulation of several widgets by moving the *Tunnel* (right)

The musician may also want to modulate the same parameter on several widgets simultaneously. To do so, he can grab a *Tunnel* and move it over the widgets as depicted on figure 11. Depending on how the widgets are aligned, this technique enables setting the same value for their graphical parameters, or applying other effects.

Modulations depend on the position of the widgets in the *Tunnels*. By changing the length of continuous *Tunnels*, one can increase the number of values in their graphical parameters scales. Thus modulation resolution is variable. Furthermore, extending a *Tunnel* with a discrete scale and many values facilitates the access to each of these values.

Tunnels can be located freely on the *Interaction Plane*, allowing musicians to create different sets
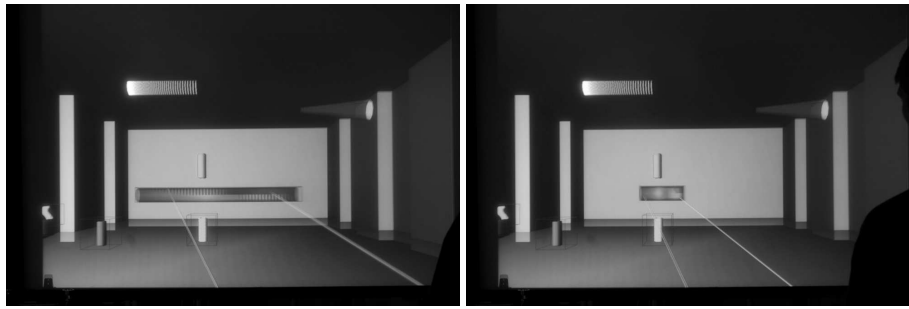
Figure 12: *Tunnels* can be stretched to increase their resolution.

of these musical tools, or to organize them in order to develop new musical gestures, such as concatenation of modulation. Furthermore, several *Interaction Planes* can be defined in the virtual environment, each with different sound processes and *Tunnels*. These locations may correspond to different parts of a musical performance, or to separated sets for musical collaboration.

4.2.3. Parameters scales and combinations

For each parameter of the *Tunnels*, one may define several scales, which can be selected in real-time, as it can be seen on figure 13. This enables both discrete and continuous modulations of the *3D reactive widgets*. Continuous scales are defined by minimum and maximum values, and discrete scales by arrays of values.

Several graphical parameters can be combined on a single *Tunnel*. For example, as shown on figure 13 (right), one can modulate the Scale and Color Hue of widgets at the same time. This enables one-to-many mappings, as defined by [Hunt and Kirk, 2000], between the gestures and the sound parameters, while preserving the separation of the graphical representation of the parameters.
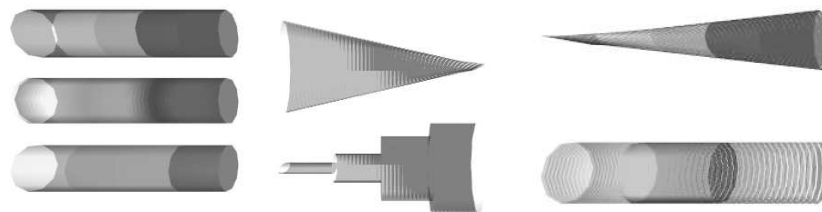

Figure 13: (Left) Different scales for the Hue and Scale parameters. (Right) Combinations of 2 and 3 parameters on two *Tunnels*. On the top, Scale (continuous) and Color Hue (discrete). On the bottom, Transparency (discrete), Hue Color (continuous) and Scattering (continuous).

4.2.4. Evaluation

In this section, we present an informal evaluation of the *Tunnels* based on user comments. Users were able to test the *Tunnels* on several occasions, including demos at our virtual reality center at the University of Bordeaux and demos during special events such as "La Nuit des Chercheurs" in Bordeaux. After they got used to manipulating the virtual rays and grabbing the *3D reactive widgets*, they understood quickly how to use the *Tunnels*. They were usually able to modify sound parameters while triggering the sounds. Users tended to modify only one widget at a time, maybe because it seemed more straightforward to move the sound process towards the effect rather than the opposite. They also figured out quickly how to change *Tunnel* scales by hitting them.

Among the comments made, four in particular may lead to improvements of the *Tunnels*:

- Fast modulations:
  Users reported difficulties to perform melodies, especially fast ones, using a *Tunnel* associating color hue to pitch. Indeed, they could not play melodies with both hands because only one hand at a time could grab and modulate a widget. A possible solution would be to duplicate widgets released inside a *Tunnel*, creating one "clone" for each value for discrete *Tunnels*, and a reasonable amount of clones for continuous ones. The clones would be placed along the *Tunnel*. This would give a kind of keyboard, allowing users to play modulations of the widgets with both hands, thus

facilitating fast modulations. Moreover, any sound process parameter, such as filter, distortion and so on, could be modulated using this keyboard feature.

- Duplication:
  When several users are playing simultaneously, they often want to modulate the same parameter, i.e. use the same *Tunnel*, but they have to wait for other to finish. One possible solution would be to allow users to duplicate the*Tunnels*, for example by grabbing one of them with one hand and performing a cut gesture, i.e. crossing with the ray, with the other hand.

- Combining:
  In the same way, they felt that it should be possible to configure the *Tunnels* while playing. While it may be difficult to entirely define the scales, combinations could be modified simply by stacking up two *Tunnels* that would then be merged into a single one. Grabbing and stretching vertically this new *Tunnel* would separate the original ones.

- Interaction planes:
  Users reported conflicts when moving widgets on the interaction plane. For example they may unintentionally collide with a *Tunnel*. Even if the modulation is not kept, this may be a problem during a performance. One solution would be to put the *Tunnels* on a distinct plane behind the widgets. Then one would have to grab and move the *Tunnels* towards the widgets, or grab and move the widgets towards the *Tunnels*, in order to perform modulations. When released, *Tunnels* and widgets would jump back to their respective planes.
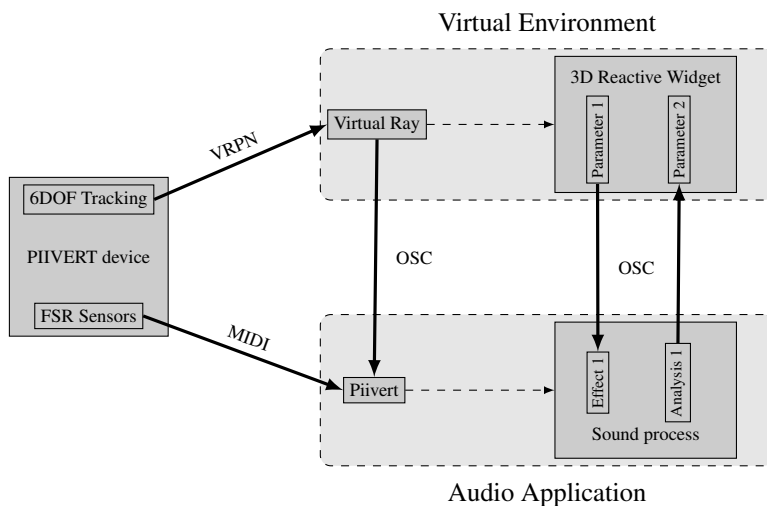
5. Implementation

Figure 14: Block diagram of the current implementation.

5.1. Input

In the current implementation, as it can be seen in figure 4, the musician is equipped with head-tracked stereoscopic glasses and with the Piivert device. Tracking is done with the A.R. Tracking 6 degrees of freedom DTrack system[1], in order to get position and orientation of the Piivert and of the musician's head. Images captured by two infrared cameras are processed by a dedicated computer which identifies targets and get their pose. Tracking data is transmitted to the application handling the virtual environment using the VRPN [Taylor et al., 2001] protocol. Piivert pressure sensors are connected to an Interface-Z acquisition board which converts the analogue data to MIDI messages, in turn transmitted to a sound card and to the sound application.

5.2. Environment

The virtual environment, composed of 3D reactive widgets, tunnels and virtual rays, is rendered by an application that we have developed, based upon the OpenSG scene graph library[2]. This application communicates with a custom audio application using the OpenSoundControl [Wright, 2005] protocol.

---

[1]http://www.ar-tracking.de/
[2]http://opensg.vrsource.org/trac

5.3. Display

The 3D application sends the scene graph to two display servers that run on separated computers. Each server renders a view of the environment for one eye. These views are then projected on a large screen by two Infitec projectors. Users wear passive stereoscopic glasses in order to separate the images for each eye.

5.4. Audio

The audio application uses the Jack sound server[3]. Sound processes are composed of sound files and audio effects. Sound files are triggered by using low-level gestures: *Hit* gestures play the entire sample and *Pressure* gestures control granular synthesis. Effects, associated to graphical parameters of the *3D reactive widgets*, are modulated using *Tunnels*. Other parameters of the sound processes, such as reading head position or spectrum, are visualised using different graphical parameters of the widgets.

In the current implementation, three audio effects are used: Pitch-shifting, Distortion and Reverberation. These audio effects are actually LV2[4] plugins loaded by the audio application. These are respectively associated to the following graphical parameters: Color Hue, Transparency and Scattering. Play head positions are highlighted by rotations of the widgets, and spectrum by their shapes. Furthermore, high-level gestures such as *Flam* allow musicians to record live sequences of sound triggering and effects modulations. The resulting setup is shown in figure 14.

6. Conclusion

Immersive virtual environments provide new possibilities for musical interaction such as new interaction techniques, additional graphical parameters and structures, navigation metaphors, and immersion. In particular, they seem valuable for multi-process instruments. In order to explore these possibilities, we defined the *3D reactive widgets*, which are complex 3D objects associated to sound processes. By means of audiovisual mappings carefully chosen, these widgets allow musicians to simultaneously manipulate and visualise the processes. They enable expressive and efficient musical interaction in virtual environments by relying on instrumental gestures while taking advantage of 3D graphical interaction techniques such as the Virtual Ray. To interact with them, we developed a new interaction device called Piivert, associated with different interaction techniques such as the *Tunnels* and percussions gestures.

The *Tunnels* may be an improvement on traditional 1D or 2D graphical sliders even for 2D graphical interfaces. The Piivert device and techniques could and will be used in other immersive instruments, and even in non-musical applications requiring expressiveness and accuracy. Moreover, the percussion gestures may be used with other interaction devices, possibly adding expressiveness. For example, a *Flam* gesture performed using the left and right buttons of a simple mouse provides a direction parameter but above all a duration parameter, which may be used as a continuous velocity parameter.

Finally the *3D reactive widgets* may be adapted to other immersive instruments, with different shapes, audiovisual mappings and even behaviour. For example, they may form 3D avatars in a collaborative immersive instrument, each avatar representing the sound process controlled by a musician.

---

[3]http://jackaudio.org/
[4]http://lv2plug.in/

References

Beaudoin-Lafon, M. (1999). *Interfaces homme-machine et création musicale*, chapter Moins d'interface pour plus d'interaction, pages 123–141. Hermès Sciences.

Berthaut, F., Desainte-Catherine, M., and Hachet, M. (2009). Interaction with the 3d reactive widgets for musical performance. In *Proceedings of Brazilian Symposium on Computer Music (SBCM09)*, pages 13–20, Recife, Brazil.

Berthaut, F., Desainte-Catherine, M., and Hachet, M. (2010a). Combining audiovisual mappings for 3d musical interaction. In *Proceedings of the International Computer Music Conference (ICMC10)*, pages 357–364, New York, USA.

Berthaut, F., Desainte-Catherine, M., and Hachet, M. (2010b). Drile : an immersive environment for hierarchical live-looping. In *Proceedings of New Interfaces for Musical Expression (NIME10)*, pages 192–197, Sydney, Australia.

Berthaut, F., Hachet, M., and Desainte-Catherine, M. (2010c). Piivert: Percussion-based interaction for immersive virtual environments. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 15–18, Waltham, Massachusetts, USA.

Bott, J. N., Crowley, J. G., and LaViola, Jr., J. J. (2009). Exploring 3d gestural interfaces for music creation in video games. In *FDG '09: Proceedings of the 4th International Conference on Foundations of Digital Games*, pages 18–25, New York, NY, USA. ACM.

Bowman, D. A. and Hodges, L. F. (1997). An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *SI3D '97: Proceedings of the 1997 symposium on Interactive 3D graphics*, pages 35–ff., New York, NY, USA. ACM.

Cadoz, C. (1999). *Les nouveaux gestes de la musique*, chapter Musique, geste, technologie, pages 47–92. Éditions Parenthèses.

Giannakis, K. (2006). A comparative evaluation of auditory-visual mappings for sound visualisation. *Organised Sound*, 11(3):297–307.

Gibson, J. J. (1977). *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*, chapter The Theory of Affordances, pages 67–82. Lawrence Erlbaum.

Hamilton, R. (2008). Q3osc: or how i learned to stop worrying and love the game. In *Proceedings of the International Computer Music Association Conference*, Belfast, Ireland.

Hunt, A. and Kirk, R. (2000). Mapping strategies for musical performance. *Trends in Gestural Control of Music*, pages 231–258.

Jacquemin, C., Ajaj, R., Cahen, R., Olivier, Y., and Schwarz, D. (2007). Plumage: design d'une interface 3d pour le parcours d'échantillons sonores granularisés. In *Proceedings of the 19th International Conference of the Association Francophone d'Interaction Homme-Machine*, IHM '07, pages 71–74, New York, NY, USA. ACM.

Jordà, S. (2003). Interactive music systems for everyone: exploring visual feedback as a way for creating more intuitive, efficient and learnable instruments. In *Proceedings of the Stockholm Music Acoustics Conference (SMAC03)*, Stockholm, Sweden.

Jordà, S. (2005). *Crafting musical computers for new musics' performance and improvisation*. PhD thesis, Universitat Pompeu Fabra.

Levin, G. (2000). *Painterly Interfaces for Audiovisual Performance*. PhD thesis, Massachusetts Institute of Technology.

Mäki-Patola, T., Laitinen, J., Kanerva, A., and Takala, T. (2005). Experiments with virtual reality instruments. In *Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (NIME05), Vancouver, BC, Canada*.

Mulder, A. G. (1998). *Design of virtual three-dimensional instruments for sound control*. PhD thesis, Simon Fraser University, Canada.

Naef, M. and Collicot, D. (2006). A vr interface for collaborative 3d audio performance. In *Proceedings of the 2006 International Conference on New Interfaces for Musical Expression (NIME06), Paris, France*.

Oliver, J. and Pickles, S. (2007). Fijuu2: a game-based audio-visual performance and composition engine. In *NIME '07: Proceedings of the 7th international conference on New interfaces for musical expression*, pages 430–430, New York, NY, USA. ACM.

Poupyrev, I., Ichikawa, T., Weghorst, S., and Billinghurst, M. (1998). *Egocentric Object Manipulation in Virtual Environments: Empirical Evaluation of Interaction Techniques*, volume 17, pages 41–52. Blackwell Publishers Ltd.

Rodet, X., Gosselin, F., Mobuchon, P., Lambert, J.-P., Cahen, R., Gaudy, T., and Guedy, F. (2005). Study of haptic and visual interaction for sound and music control in the phase project. In *Proceedings of the 2005 International Conference on New Interfaces for Musical Expression (NIME05), Vancouver, BC, Canada*.

Taylor, II, R. M., Hudson, T. C., Seeger, A., Weber, H., Juliano, J., and Helser, A. T. (2001). Vrpn: a device-independent, network-transparent vr peripheral system. In *VRST '01: Proceedings of the ACM symposium on Virtual reality software and technology*, pages 55–61, New York, NY, USA. ACM.

Valbom, L. and Marcos, A. (2005). Wave: Sound and music in an immersive environment. *Comput. Graph.*, 29:871–881.

Waisvisz, M. (1985). The hands. In *Proceedings International Computer Music Conference*, pages pages 313–318.

Wozniewski, M., Settel, Z., and Cooperstock, J. (2006). A spatial interface for audio and music production. In *Proceedings of the International Conference on Digital Audio Effects (DAFx), 2006*, Montreal, Canada.

Wright, M. (2005). Open sound control: an enabling technology for musical networking. *Organised Sound*, 10(3):193–200.