ETHERACTION: PLAYING A MUSICAL PIECE USING GRAPHICAL INTERFACES

Jean-Michel Couturier
LMA-CNRS
31, chemin Joseph Aiguier 13402 Marseille cedex 20
couturier@lma.cnrs-mrs.fr

ABSTRACT

This paper introduces the use of graphical interfaces to interpret an electroacoustic piece, Etheraction. Electroacoustic pieces, commonly created for tape, can now be interpreted in live performance with dedicated interactive systems; the interaction between the performers and these systems can use graphical interfaces, largely implemented in nowadays computers. When using graphical interfaces for real time sound control, the tasks consist in controlling sound parameters through the manipulation of graphical objects, using pointing techniques or direct control with additional devices. The paper presents how I have designed two interactive systems dedicated to interpret in live Etheraction, a multichannel piece I have initially composed for tape. The piece is based on the motion of physical models of strings that control sound parameters. The two devices control both synthesis parameters and spatialisation parameters, are based on interactions with graphical interfaces, and use specific controllers.

1. INTRODUCTION

With progress of technique, more and more electroacoustic pieces, usually created for tape, are interpreted in live performance. The interpretation devices will depend on the nature of the piece and on the choice made by the performers. In case the piece contains a lot of synthetic sounds, those sounds can be played in real time using existing synthesizers, or using a specific device, especially designed for the piece. The modularity and the flexibility of digital and electronic tools enable to build a device dedicated to one musical piece. But designing such a device is not always easy; one must take into account which parameters one wants to control in each part of the piece: which ones can be fixed or driven by an automatic process, if some high level parameters can be defined. In a second step, one has to choose which gesture controllers to associate with the parameters.

The problematic is not the same than in designing a digital musical instrument: if in both cases, the work consists of linking controllers to synthesis parameters (mapping, [13][3]) there are lots of differences. A musical instrument is generally built to be used in several musical pieces, and those pieces are conceived while the instrument already exists; dedicated devices are built either after the piece or simultaneously to the piece creation. They are only used to play the piece. Another difference is that in a musical piece, sound processes can differ along the piece; the performer can choose if he wants to use different devices for each part or to use a unique device for the entire piece. In both cases, a lot of parameters have to be manipulated, and not necessarily all at the same time.

This paper introduces a new way of designing device that can be used for a piece interpretation, manipulating specific graphical interfaces. Graphical interfaces enable to display a lot of graphical objects that we can manipulate with the same controller; each graphical object is linked to synthesis parameters. The shapes of the graphical interface and its objects have no physical constraints; this gives more freedom to the designer.

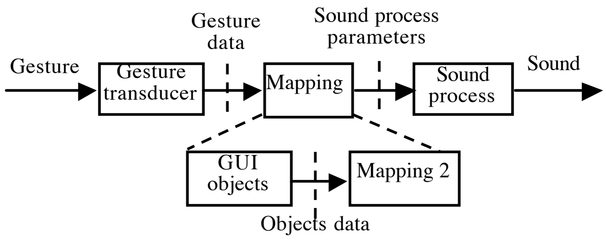

Figure 1. The usual Mapping chain links gesture data to sound parameters; with the graphical interface, there is an additional step in the mapping: graphical objects are linked to sound parameters and the gesture device can control any graphical objects.

This paper introduces how I have designed devices with graphical interfaces to interpret a specific piece, called Etheraction. Section 2 introduces the use of graphical interface in music performance; section 3 describes the Etheraction musical piece, and section 4 the design of the interfaces and their use in live performance.

2. USING GRAPHICAL INTERFACES IN THE CONTROL

The graphical interfaces are not essential in a computer-based musical device, unlike in many computer applications, but they can provide a high level of interactivity in the performance. This section introduces the use of graphical interfaces in the context of real time performance: how to act in the interface, with which controller, what are the advantages.

2.1. Controlling graphical objects

Commonly implemented in nowadays computers, the graphical interfaces are often used in music softwares. In case of real time sound control, the tasks consist in controlling sound parameters through the manipulation of graphical objects, according to the direct manipulation principles [10]. All sound parameters are controllable via graphical objects that generally represent real objects like piano keyboards, faders, buttons, etc. The graphical interfaces tend to reproduce on screen an interaction area that is close to a real one, like front panels of electronic instruments. The aim of such interfaces is to make the user feel he has real objects in front of him.

Generally, the use of graphical interface needs a pointing device, like the mouse, that controls the position of a graphical *pointer* displayed onscreen. In order to be manipulated, graphical objects need to be activated by the user, with the pointing device. To activate a graphical object, the user has to put the pointer over the object (pointing task) and to press a button (clicking task). Once activated, the data of the pointing device are linked to the graphical object in a way predefined in the software. The graphical object is inactivated when the user releases the button (unclick task).

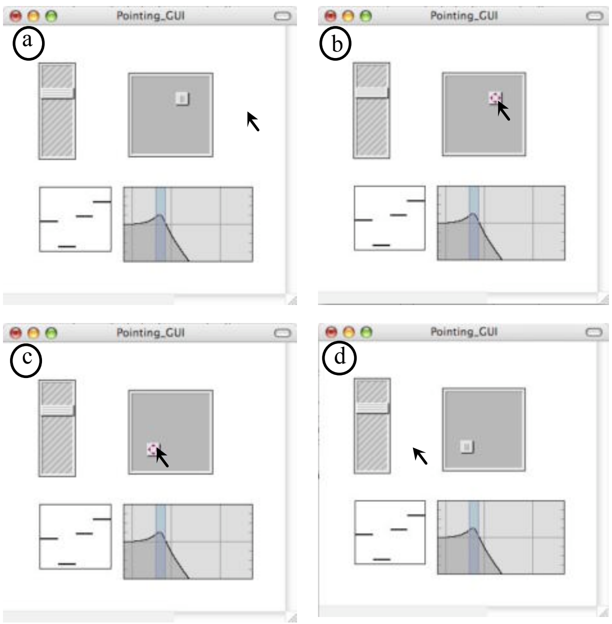

Figure 2. Different tasks in the control of graphical objects using a pointing device: activation *(a*, pointing task and *b*, clicking task), manipulation (*c*) and inactivation (*d*).

This interaction technique enables to use only one controller, the pointing device, to manipulate all the graphical objects. This technique is implemented in today computers and used to control WIMP (Windows, Icons, Menus and Pointing) interfaces, with a single mouse as pointing device. Nevertheless, complex musical tasks, with numerous parameters to control simultaneously, cannot be performed in real time with a single pointing device. Performers must use additional controllers or advanced interaction techniques, as we will see in the following paragraphs.

To have a better control on sound, music softwares usually use specific controllers, like MIDI ones, to control the graphical objects of the interface, and their associated sound parameters. Those controllers are directly mapped to the corresponding graphical objects, giving by this way a more direct access to the graphical object: there are no pointing and clicking tasks (activation task) in the interface, and several graphical objects can be manipulated simultaneously.

Figure 3. An example of direct control of graphical interface through a specific controller. The graphical sliders displayed on screen are permanently linked with the physical sliders of the controller.

This second interaction technique, which could be called *direct mapping* technique, seems better adapted to real time sound control, but is more expensive in hardware and less flexible than the pointing technique. The Etheraction devices, as it shows in section 4, use the two interaction techniques complementarily.

Beyond the single pointing technique and the direct mapping technique, advanced interfaces have been developed in the field of HCI (Human Computer Interaction). Those interfaces are more efficient than the traditional WIMP interfaces; some of them use bimanual gestures [6], mix real and graphical objects (tangible interfaces: Audiopad [9], ReacTable [7]) or use 3D graphics [8]. At NIME 2003, Daniel Arfib and I introduced the pointing finger device [4] (figure 4, 5th picture), a multifingers' touchscreen-like device. This type of system provides the most direct and intuitive possible interaction: one can, with his fingers, manipulate graphical objects as if they were real objects. The pointing fingers use six DOF (degrees of freedom) sensors attached to four fingers and switch buttons on fingertips; this device gives the position of four fingers regarding a screen. A special program manages the graphical objects and disables conflicts between the different pointers. We design two musical instruments with this device, one scanned synthesis instrument and one photosonic instrument. This interface uses the pointing technique with four pointers, which allows the control of numerous parameters simultaneously.

2.2. Pointing devices and additional controllers

To manipulate a 2D graphical interface with a pointing device, one should use at least one controller that enables a pointer to move on a 2D plane; at least we need XY and a button. To drive several pointers, one can use several controllers or use one controller that gives several 2D coordinates.

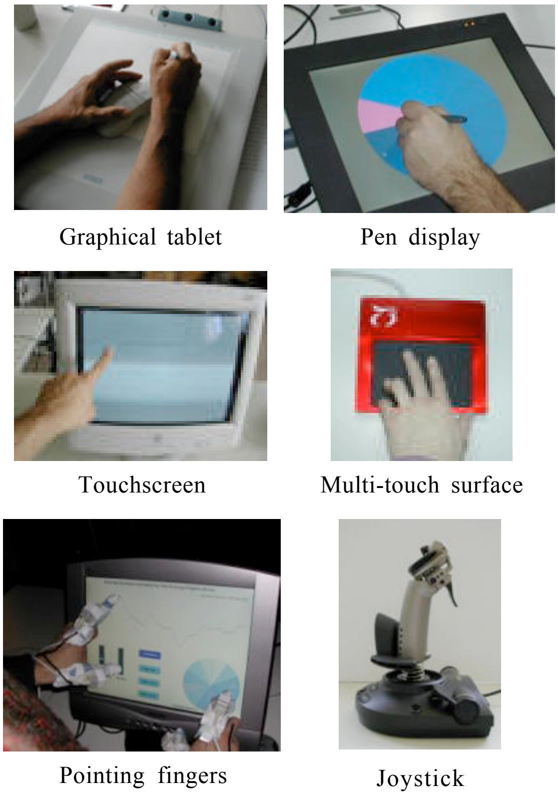

Figure 4. Different controllers that can move one or several pointers in a graphical interface. They can be bimanual, with an object to hold, with a screen, multi-fingers, invasive / non invasive, …

In some cases, one wants to use a specific controller (for example a piano keyboard or a foot controller) beside the pointing device: in this case, if necessary, the graphical interface can contain a graphical element linked to the controller. Then, the musical device can be described according to two points of view: in the first case, the controller manipulates the graphical element that is linked to the sound parameters (like in figure 3); in the second case, the controller directly manipulates the sound parameters and the graphical element is just a visual feedback. The first point of view is pertinent if the graphical element behavior is very close to the gestures that are done on the controller; in this case, when the performer uses the controller, he really has the impression that he manipulates the graphical interface (in fact the graphical element). This feeling will improve the user's immersion in the device and the interaction. Moreover, adding extra controllers will extend the amount of parameters controllable simultaneously.

2.3. Graphical interfaces and interpretation

If the graphical interfaces help to build digital musical instrument, they can be even more efficient to interpret a musical piece, especially if a lot of parameters have to be manipulated but not all at the same time, as when sound processes differ along the piece. Numerous objects can be designed, from current graphical interface buttons and sliders to specific synthesis objects (like interacting with a string through the pointing fingers device). Graphical interfaces enable to display all the parameters and the associated graphical objects in a same area, and enable to manipulate them with the same controllers; if we use other controllers, the graphical interface could integrate them and give good visual feedback. It provides an important freedom in the design of real-time sound systems.

The next sections will introduce a musical piece, Etheraction, and the control device I have built to interpret it.

3. ETHERACTION: INTERPRETATION OF AN ELECTROACOUSTIC PIECE

Etheraction is a musical piece I have composed in early 2004. The first version of the piece was a recorded one; once this version was completed, I have built devices with computer, graphical interfaces and controllers to interpret the piece in live situation. The recorded version was diffused on March 9th 2004 at the GMEM, Marseilles, and the live version was performed on April 7th 2004, at musée Ziem, Martigues, France.

3.1. Etheraction recorded version

Etheraction was composed in early 2004 as an 8-channel recording. This piece is based on the motion of physical model of strings that control synthesis parameters and uses digital sounds produced with Max/MSP patches. Most of the sounds of the recorded version were generated using gestures: the synthesis parameters were driven by gesture picked up by different controllers: graphical tablet, touch surface, joystick. I have built this piece in several steps: I have first played one by one the different elements and then I have spatialized them one by one (group 2 excepted) with a custom version of Holospat from GMEM (the sound was spatialised in real time with gestures). All the gesture data were recorded (a sort of automation); then I was able to modify and adjust some parameters afterwards, without replaying all the sequence. At the end, all the parts were mixed together in Logic Audio Environment.

The piece is divided in three parts and uses different sound process, as shown in figure 5:

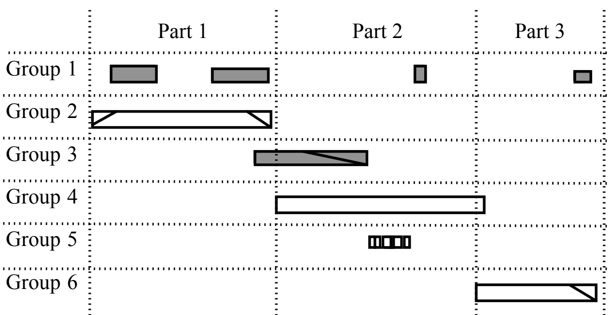

Figure 5. Etheraction overview. Each group corresponds to a different sound process. For the live version, the different groups of sound processes are dispatched on two devices (gray, white); Group 5 elements are generated auto-matically and are functions of group 4 interactions.

Group 1 uses scanned synthesis method [12] [5] with spectral aliasing. Sounds of group 2 come from 8 filters banks (with 8 filters each); the input sound is first a recorded sound and next a pink noise. Each filter banks is manipulated by a physical model of slow moving string: the string contains 8 masses and each mass position controls the gain of one filter. Each filter bank is associated to one channel (speaker). Group 3 is a mere texture of loud sound that only changes in dynamics. Sounds of group 4 come from filtering string instrument [2], with some improvements. Sounds of Group 5 are short sounds from the filtering string instrument. They are mixed to create precise responses to some group 4 events. Group 6 is very close to group 4, but has different presets, that enable the production of very different sounds.

3.2. Etheraction live version

The more important difficulty was to build devices that enable to play all the elements of the piece, from the beginning to the end: in the recorded version, all events were created one by one, spatialised one by one, and then mixed together. To interpret the piece, a lot of sound processes have to be controlled and spatialised simultaneously. For the live version, the spatialisation has been made on four speakers to reduce the computation. All the devices have been built to keep the general spirit of the original version.

To design the devices that enable to perform Etheraction in live, I have first analyzed all the sound processes that were used in the recorded version, and have tried to see what processes could be grouped together and played with the same device. The idea was to share out the group processes in a minimum number of devices, keeping the cohesion of each element. Each device has been built to be controlled by one performer: he has to be able to control alone all the parameters. All devices are based on interaction with graphical interfaces, following the method we are applying to design digital musical instruments [1], adapted to the use of graphical interface. Starting from sound process, I try to define a minimum set of control parameters, then, I try to find which parameters could be controlled trough the graphical interface and which have to be controlled by an "external" controller. The graphical objects are defined according to the nature of the parameters they are controlling; when it was necessary, I have integrated in the graphical interface some visual feedbacks of the gesture done on the "external" controllers.

As seen in figure 5, there are five groups that intervene at different parts of the piece. Those groups have been displayed on two devices; those devices are described in the next section.

4. DESCRIPTION OF THE DEVICES

As seen before, all sound processes can be controlled by only two devices; this section introduces those devices. All graphical interfaces were created with Jitter, the video complement of Max/MSP. The synthesis technique and the mapping were implemented in Max/MSP.

4.1. First device

The graphical interface of the first device is controlled by an interactive pen display, a mouse and a 2D pedal (a modified joystick). The 2D pedal is dedicated to only one part of the interface, the spatialisation; the control of spatialisation consists in the displacement of a point in a 2D space. The other graphical objects are controlled by the pen and the mouse; to manage the graphical objects and to enable the use of several pointers in a same graphical interface, I use a specific Max/MSP object that I have designed for the Pointing Fingers device (section 2.1).

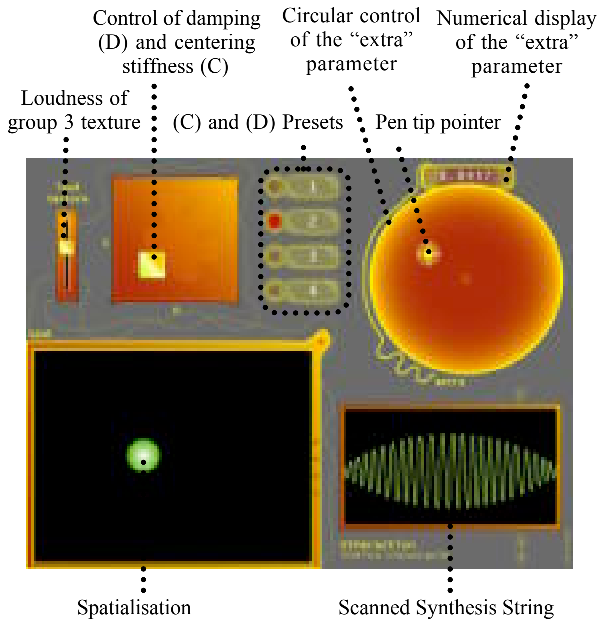

Figure 6. Graphical interface of the first device. The graphical objects can be manipulated with an interactive pen display and a mouse; the spatialisation is controlled by a 2D pedal. The pen tip pressure controls the amount of force applied on the string and corresponds to the radius of the circle in the middle of the pen tip pointer. (the mouse pointer is not on the figure).

The circular control of the "extra" parameter (a deformation of the force profile applied on the string) is incremental: turn clockwise will increase the parameter, and inversely; this enables a precise control on a very large scale. The string damping and centering stiffness are controlled by a 2D slider; to help the performer, four presets of these parameters can be loaded thanks to buttons. The loud texture of group 3 is controlled by a slider, and a visual feedback of the string shape is displayed at the bottom right.

4.2. Second device

This device is much more complex, because the sound processes are not the same along the piece, and there are numerous parameters to drive simultaneously. To lighten the graphical interface, I have created three different graphical interfaces, one for each part of the piece (figure

7); the performer can switch between the interfaces using tabs.

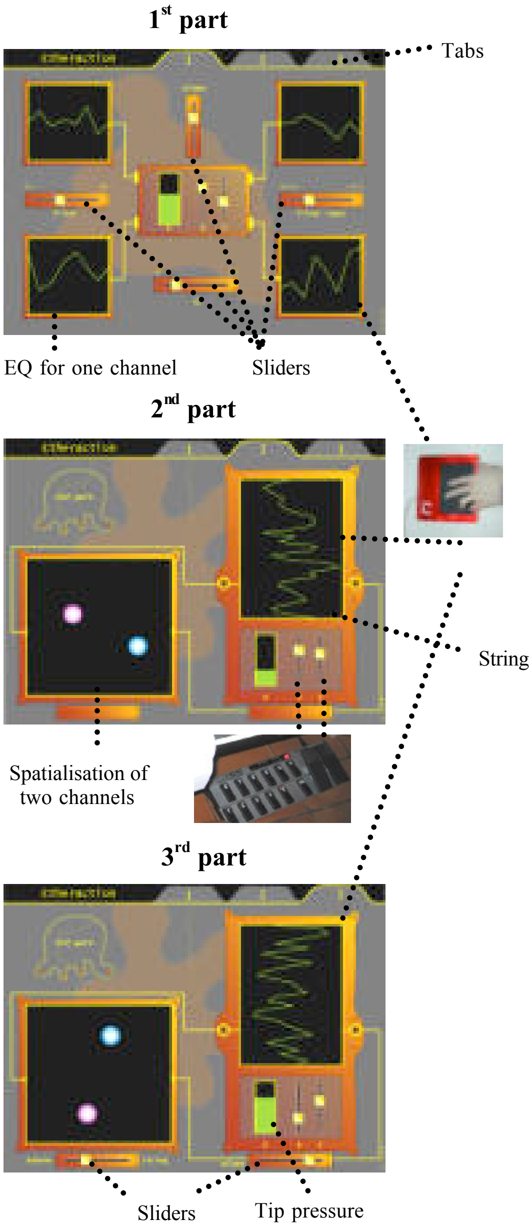

Figure 7. Second control unit. This device uses three different graphical interfaces, corresponding to the 3 parts of the piece. The performer can switch between the different interfaces using tabs. In all the parts, the strings are exited by forces controlled by a multi-finger touchpad.

This device uses three different controllers: a graphical tablet (which gives the angular position of the pen: tilt), which is used to control the graphical interface and the spatialisation; a Tactex [11] multi-finger touch surface that controls the forces applied on the different strings; and a foot controller with switches and two expression pedals. The pedals control two string parameters and a visual feedback of the pedal positions is displayed on the interface; the switches are used to choose one of the three interfaces and lock or unlock the pen on the control of the spatialisation.

The spatialisation control uses the pen tip coordinates and the tilt: the displacement of the two extremities of the pen (perpendicularly to the tablet) controls the position of the two channels that are spatialised.

4.3. Use in live performance

I have used those graphical interfaces on stage with Denis Brun, Electroacoustic student, at the "Concert Emergence" at Martigues, in April 2004.

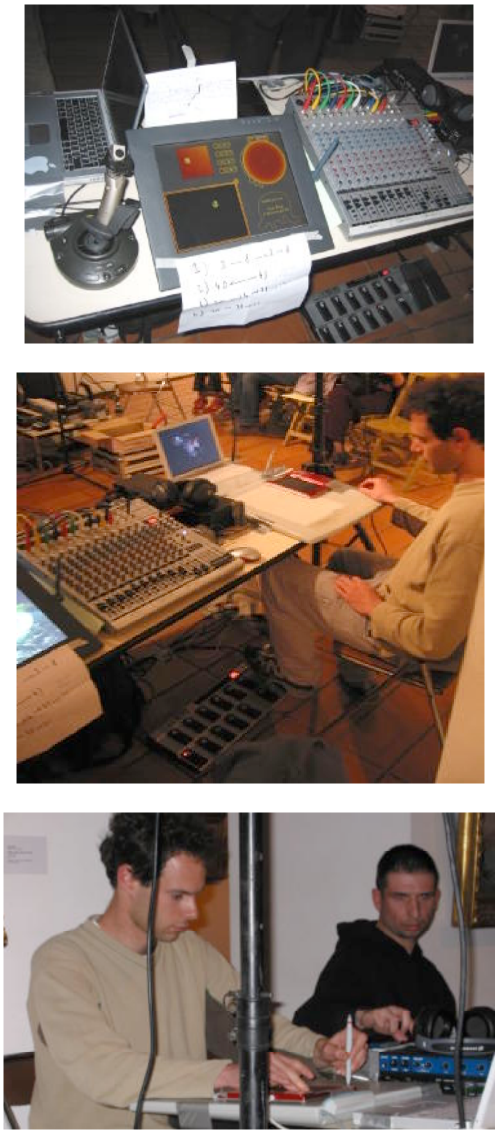

Figure 8. Martigues' concert, the two Etheraction devices and the performers in action.

For this first live performance, the devices were not as developed as described in previous sections. In the first device, the mouse was not used, a joystick was used instead of the 2D pedal and the string was not displayed. In the second device, only one graphical interface (instead of three) had been used.

The learning period of the devices have been shortened thanks to the use of graphical interfaces. Displaying the shapes of strings increases the feeling of immersion in the device: we don't have a set of complex parameters to control but a physical object which we are interacting with. The position of the pedals and the pressure of the pen tip are difficult to evaluate; seeing them on screen provides a great help for their manipulation.

Nevertheless, with the second interface, I have encountered some difficulties to control both sound and spatialisation, in spite of the help of the graphical interface; whatever the quality of the musical device, a complete learning phase is always necessary to play the device.

5. CONCLUSION

With the live version of Etheraction, I have tried to experiment the use of graphical interface to interpret an electroacoustic piece. Etheraction uses complex sound processes with a lot of parameters to control. The graphical interfaces gives a visual representation of the sound processes as well as a control area adapted to them, making possible their control and their spatialisation in real time.

This experience shows me that a lot of strategies can be used to create an interface, according to the constraints and the musicians (composers and interprets) preferences. Creating such interface is very different than creating a digital musical instrument: here the interface has to be adapted to the music.

6. ACKNOWLEDGMENTS

I want to thanks Denis Brun for testing and performing the interface with me on stage and Magnolya Roy for the graphic design of the interfaces.

7. REFERENCES

[1] Arfib D., Couturier J.-M., Kessous L.: "Design and Use of Some Digital Musical Instruments", in *Gesture-Based Communi-cation in Human-Computer Interaction*, Lecture notes in Artificial Intelligence, LNAI 2915, pp: 509-518, ed: A. Camurri and G. Volpe, Publisher : Springer verlag, 2004

[2] Arfib D., Couturier J.-M., Kessous L. : "Gestural strategies for specific filtering processes", *proceedings of DAFx02 conference*, , pp. 1-6, Hamburg, 26-28 sept 2002.

[3] Arfib D., Couturier J.-M., Kessous L., Verfaille V.: " Strategies of Mapping between gesture data and synthesis parameters using perceptual spaces ", *Organised Sound* , Cambridge University Press Vol 7/2, August 2002.

[4] Couturier J.M, Arfib D., "Pointing Fingers: Using Multiple Direct Interactions with Visual Objects to Perform Music", *Proceedings of the 2003 Conference on New Interfaces for Musical Expression (NIME-03)*, Montreal, Canada, pp. 184-187, 2003

[5] Couturier J.M., "A scanned synthesis virtual instrument", *Proceedings of the 2002 Conference on New Instruments for Musical Expression (NIME-02)*, Dublin, Ireland, May 24-26, 2002.

[6] Guiard, Y., "Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model," *J. Motor Behavior*, 19 (4), pp. 486-517, 1987.

[7] Kaltenbrunner, M. Geiger, G. Jordà, S.. 'Dynamic Patches for Live Musical Performance' *Proceedings of International Conference on New Interfaces for Musical Expression (NIME-04)* Hamamatsu, Japan, 2004.

[8] Mulder A., Fels S. and Mase K., "Design of Virtual 3D Instruments for Musical Interaction", *Proceedings of Graphics Interface '99*, pp. 76-83, Toronto, Canada, 1999.

[9] Patten J., Retch B., Ishii H., "Audiopad: A Tag-based Interface for Musical Performance", *Proceedings of the 2002 Conference on New Instruments for Musical Expression (NIME-02),* Dublin, Ireland, May 24-26, 2002.

[10] Shneiderman, B., "Direct Manipulation: a Step Beyond Programming Languages", *IEEE Computer*, 16(8), pp 57-69, 1983.

[11] Tactex, touch surfaces, http://www.tactex.com/.

[12] Verplank B., Mathews M., Shaw R., "Scanned Synthesis", *Proceedings of the 2000 International Computer Music Conference*, pp. 368-371, Berlin, Zannos editor, ICMA, 2000.

[13] Wanderley M., "Performer-Instrument Interaction: Applications to Gestural Control of Music", *PhD Thesis*, Paris, University Pierre et Marie Curie - Paris VI, 2001.