THE REACTABLE*

Sergi Jordà, Martin Kaltenbrunner, Günter Geiger, Ross Bencina
{sjorda,mkalten,ggeiger,rbencina}@iua.upf.es
Music Technology Group/IUA
Universitat Pompeu Fabra
Ocata 1, 08003 Barcelona

Figure 1. The reacTable*

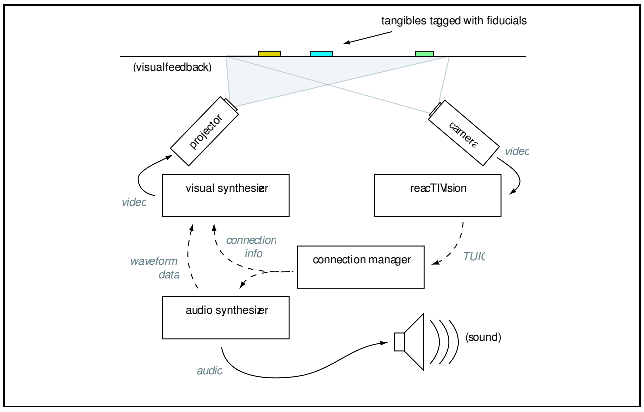

Figure 2. The reacTable* architecture

ABSTRACT

This paper describes the reacTable*, a novel multi-user electro-acoustic music instrument with a tabletop tangible user interface, which is being developed at the MTG in Barcelona. We first introduce the reacTable* project and some of the concepts behind its design and we then discuss interaction and performance scenarios.

1. INTRODUCTION

The reacTable* is a state-of-the-art music instrument, which seeks to be collaborative (local and remote), intuitive (zero manual, zero instructions), sonically challenging and interesting, learnable and masterable [17], and suitable for complete novices (in installations) and for advanced electronic musicians (in concerts). The reacTable* uses no mouse, no keyboard, no cables, no wearables. The technology it involves is, in other words, transparent to the user; it also allows a flexible number of performers that can enter or leave the instrument-installation without previous announcements.

2. CONCEPTION AND DESIGN

The reacTable* is based on a translucent round table. A video camera situated beneath, continuously analyzes the table surface, tracking the nature, position and orientation of the objects that are distributed on it. The objects are passive and of different shapes, without any sensors or actuators. Users interact by moving them, changing their position, their orientation or their faces (in the case of volumetric objects), controlling with these actions the topological structure and the parameters of a sound synthesizer. Also from beneath the table, a projector draws dynamic animations on its surface, providing a visual feedback of the state of the synthesizer. Figure 2 illustrates this architecture.

2.1. Modular synthesis and visual programming

The concept of modular synthesis goes back to the first sound synthesizers, both in the digital [11] as in the analog domains, with Robert Moogs or Donald Buchlas Voltage-controlled synthesizers [4]. Modular synthesis has largely proved its unlimited sound potential and can be considered indeed as the starting point of all the visual programming environments for sound and music, which started with Max in the late 1980s and have developed into PD [14] or AudioMulch [1], to mention a few. As shown by all of these environments, visual programming constitutes nowadays one of the more flexible and widespread paradigms for interactive music making. The reacTable* is probably the first system that seeks to incorporate all the aforementioned paradigms, in order to build a flexible, powerful and intuitive new music instrument.

2.2. Objects, connections and visual feedback

Each of these objects has its dedicated function for the generation, modification or control of sound. By moving the objects on the table surface and bringing them into proximity with each other, performers construct and play the instrument at the same time, while spinning them as rotary knobs allows to control their internal parameters. Like Max and its cousins, the reacTable* distinguishes between control and sound objects, and between control and sound connections. When a control flow is established between two objects, a line is drawn between them, showing by means of dynamic animations, the flux direction, its rate and its intensity. Audio flows, on their turn, are represented by means of instantaneous waveforms. Moreover, the reacTable* projection wraps the physical objects with virtual auras. An LFO control object, for example, will be wrapped by a blinking animation that will keep showing the frequency, the amplitude and the shape (e.g. square vs. sinusoidal) of the oscillation. First informal tests show that this visual feedback, as seen in figure 1, is crucial for the playability of the reacTable*.

3. THE REACTABLE* ARCHITECTURE

Figure 2 illustrates all the reacTable* system components. In this section we briefly discuss each of them.

3.1. Vision

A vision engine has to be sufficiently fast for the needs of an expressive musical instrument, thus providing a high temporal resolution for the tracking of fast movements. The reacTable* vision component, reacTIVision, is an open source system for tracking the type, location and orientation of visual markers in a real-time video stream. The system was developed within the Music Technology Group by Ross Bencina [2] after we had developed an initial prototype using Costanza and Robinson's d-touch system [5]. It currently processes 60 frames at a resolution of 640x80 pixels in real-time on a 2GHz Athlon system, and scales without any problems to higher frame rates or resolutions, and sends the acquired data via TUIO [9] a protocol based on OpenSound control [18] to a client application.

3.2. Connection manager: dynamic patching

A central connection manager receives these OSC messages with the objects' type, position and orientation data, and calculates the actual patch network according to a simple set of rules. Unlike traditional visual programming languages, these rules manage automatic connection and disconnection between objects without requiring the user to explicitly declare them: all objects have a certain number of different in-output connectors (sound, control, sync, etc.) and each object checks its neighborhood for objects that can provide both compatible and available ports. As a result, moving an object around the table surface permanently interferes and alters existing connections, creating extremely variable synthesizer morphologies, and resulting in a highly dynamic environment.

The resulting connections, together with several additional parameters of each object (e.g. its intrinsical angle, the distance and the angle to the following connected neighbor, etc.) are then sent both to the sound and graphics synthesizer components, which dynamically construct the resulting nets and decide how to map this raw control data onto the synthesis processes.

3.3. Audio synthesizer

The reacTable* open structure favors both the existence of all types of higher level objects, such as the ones we could imagine within an environment such as Max or PD (e.g. sophisticated rhythm and melody generators, chaotic generators, pitch quantizers and harmonizers, etc.), and of all kind of lower level sound synthesis and processing algorithms and techniques. Currently, the implemented reacTable* objects can be categorized into six different functional groups: Generators (1 audio out and a varied number of control in), Audio Filters (1 audio in, 1 audio out and a variable number of control in), Controllers (1 control out), Control Filters (1 control in and 1 control out), Mixers (several audio in, 1 audio out) and Clock synchronizers. Current implementations of the synthesizer use Pure Data [14] and SuperCollider [10]. It is possible to write synthesis modules in any system that supports OSC, and allows to instantiate, connect and delete objects dynamically.

3.4. Visual synthesizer

Visual feedback is implemented in a similar way as the audio synthesizer, thus constituting a full "visual synthesizer". The connection manager sends the information about the objects' connection state and their parameters to this engine, which interprets the information and draws the objects at their correct positions with the lines that connect them. For the correct visualization of audio connections (which are represented by their waveforms in the time-domain), the graphics engine has an additional connection to the audio synthesizer, from where it receives the information about the data flows in the synthesizer. Figure 1 shows an example of the visualization.

4. PERFORMING WITH THE REACTABLE*

The reacTable* has been conceived for a wide spectrum of users, from the absolute novice in an interactive installation setup, to professional performers in concert venues. This is attempted by designing an instrument as intuitive as possible, and at the same time, capable of the maximum complexities. At the time of this writing, the reacTable* is a prototype that we are still learning how to play. However, we can already anticipate different models of playing as well as several compelling complexities that allow to simultaneously combine complementary performance approaches and techniques. We describe here four of them.

4.1. Towards the luthier-improviser continuum

Within traditional modular visual programming synthesizers, there is a clear separation between building and playing the patch (or instrument): there is an editing and an execution mode. The editing is usually a lengthy development process, which leads to a final and stable instrument patch, which then during the execution mode is controlled on screen or via any available controller device.

The reacTable* has to be built and played at the same time. Each piece has to be constructed from scratch starting from an empty table (or from a single snapshot which has been re-constructed on the table before the actual performance). This is a fundamental characteristic of this instrument, which therefore always has to evolve and change its setup. Building the instrument is equivalent to playing it and vice-versa. Remembering and repeating the construction of a building process can be compared to the reproduction of a musical score. The reacTable* establishes thus a real continuum not only between composition and performance, but between lutherie, composition and performance.

Moreover, the reacTable* connection paradigm in which by moving an object around the table surface, the performer is able to permanently interfere and alter existing connections, creates extremely variable synthesizer morphologies just at the reach of one hand.

4.2. The caresser-masseur-violinist model

Objects on the table permanently sense at least three parameters, depending on their relative position within the net topology and of their angle orientation. This allows for very subtle and intimate control. Moving and twisting delicately two objects allows to precisely control six parameters, without scarifying voluntarily brusque and sudden morphology changes and discontinuities.

4.3. Hand and fingers interaction: the bongosero–karateka and the painter models

The hands play an important role: not only can they manipulate reacTable* objects, they are treated as superobjects themselves. Cutting or muting a sound stream can be done with a finger or with a karate-style hand gesture.

This feature promotes physically intense playing. Grabbing, lifting and dropping objects with both hands, cutting flows with karate-like gestures and reactivating them by touching the objects again, will only be limited by the computer vision engine speed. The current implementation predicts that the input frame rate will not go below 30 Hz, while 60 Hz is attainable with an adequate camera.

Soon, the reacTable* will also allow free finger-drawing directly on all of the table's surface. This functionality should include drawing envelopes, wavetables or spectra, depending on which objects are situated nearby.

4.4. The reacTable* as a collaborative multi-user instrument

The reacTable* supports a flexible number of users with no predefined roles, and allows simultaneously additive (users working on independent audio threads) as well as multiplicative (users sharing control of audio threads) behaviors. Because of the way physical objects are visually and virtually augmented, the reacTable* also constitutes a perfect example of a both local and remote multi-user instrument. In a local collaboration scenario two, three or more players can share the same physical objects and their space, but this collaboration can be expanded when two or more reacTables* are connected through the net. Sharing the same virtual space, performers can still only move the physical objects on their local table, but since these are also projected onto the remote tables, their movements may modify the shared audio threads, thus provoking interactions between displaced objects. In a third collaboration scenario, remote users could join a reacTable* session with a software simulation, where the virtual table would have the same impact as remote tables, although without the tangible interaction.

5. THE REACTABLE*: FUTURE WORK AND CONCLUSIONS

For future versions of the reacTable* we are planning more complex objects such as flexible plastic tubes for continuous multi-parametric control, little wooden dummy 1-octave keyboards, combs (for comb-filters), or other everyday objects. Currently, reacTable* objects are plain and passive, meaning that they do not come with any cables, switches buttons or whatsoever, but this should not rule out the possibility of "smart" objects that may incorporate additional internal electronics in order to retrieve some additional sensor data, coming from hitting, squeezing, bending or bouncing them, like in the case of the Squeezables [16]. In any case, this would have to be -achieved in a completely transparent way, using wireless technology for example, so that the performer can treat all objects in an equal way.

To conclude, we can affirm that, unlike many new designed instruments, the reacTable* does not originate from exploring the possibilities of a specific technology, nor from the perspective of mimicking a known instrumental model. It comes from our experience designing instruments, making music with them, and listening and watching the way others have played them [8]. The reacTable* team is currently constituted by Sergi Jordà, Martin Kaltenbrunner, Günter Geiger, Ross Bencina, Hugo Solis, Marcos Alonso and Alvaro Barbosa.

6. ACKNOWLEDGMENTS

We would like to thank the former interns within the reacTable* team Ignasi Casasnovas, José Lozano and Gerda Strobl for their valuable contributions to the project. We are also thankful for Enrico Costanza's contribution of his d-touch vision engine and Sile O'Modhrain's suggestions to our initial tangible prototype. This work was partially supported by the European Commission Cost287-ConGAS action on Gesture controlled Audio Systems.

7. REFERENCES

[1] Bencina, R. (1998). Oasis Rose, the composition: Real-time DSP with AudioMulch. In *Proceedings of the Australasian Computer Music Conference*.

[2] Bencina, R., Kaltenbrunner, M. Jordá, S. (2005). Improved Topological Fiducial Tracking in the reacTIVision System. PROCAMS 2005, *IEEE International Workshop on Projector-Camera Systems*. Submitted.

[3] Blaine, T., Forlines, C. (2002). Jam-O-World: Evolution of the Jam-O-drum Multi-player Musical Controller into the Jam-O-Whirl Gaming Interface. In *Proceedings of the 2002 Conference on New Interfaces for Musical Expression (NIME-02)*, Dublin, 17-22.

[4] Chadabe, J. (1975). The Voltage-controlled synthesizer. In John Appleton (ed.), *The development and practice of electronic music*. New Jersey: Prentice-Hall.

[5] Costanza E. Robinson J.A. (2003). A region adjacency tree approach to the detection and design of fiducials, in *Vision, Video and Graphics (VVG) 2003.*

[6] Fitzmaurice, G., Ishii, H. Buxton, W. (1995). Bricks: Laying the Foundations of Graspable User Interfaces. In Proceedings of CHI'95 Conference on Human Factors in Computing systems, 442-449.

[7] Ishii, H Ullmer, B. (1997). Tangible Bits: Towards Seamless Interfaces between People, Bits and Atoms, In *Proceedings of CHI 97 Conference on Human Factors in Computing systems*, Atlanta, Georgia USA, 22-27.

[8] Jordà, S. (2005).*Digital Lutherie: Crafting musical computers for new musics performance and improvisation*. PhD. dissertation, Universitat Pompeu Fabra, Barcelona.

[9] Kaltenbrunner, M., Bovermann, T., Bencina, R. Costanza, E. (2005). TUIO: A protocol for table-top tangible user interfaces. *6th International Gesture Workshop*, Vannes 2005.

[10] McCartney, J. (1996). SuperCollider a new real-time synthesis language. In *Proceedings of the 1996 International Computer Music Conference*, San Francisco, CA: International Computer Music Association, 257-258.

[11] Mathews, M. V. (1963). *The digital computer as a musical instrument*. Science, 142, 553-557.

[12] Paradiso, J. A. Hsiao, K.-Y. (2000). Musical Trinkets: New Pieces to Play. *SIGGRAPH 2000 Conference Abstracts and Applications*. NY: ACM Press.

[13] Patten, J., Recht, B. Ishii, H. (2002). Audiopad: A Tag-based Interface for Musical Performance. In *Proceedings of the 2002 International Conference on New Interfaces for Musical Expression (NIME-02)*, Dublin, 11-16.

[14] Puckette M. (1997). Pure Data. In *Proceedings of the 1997 International Computer Music Conference*. San Francisco, CA: International Computer Music Association, 224-227.

[15] Singer, E. (2003). Sonic Banana: A Novel Bend-Sensor-Based MIDI Controller. In *Proceedings of the 2003 International Conference on New Instruments for Musical Expression (NIME 03)*, Montreal, Canada.

[16] Weinberg, G., Gan, S. (2001). The Squeezables: Toward an Expressive and Interdependent Multi-player Musical Instrument. *Computer Music Journal*, 25(2), 37-45.

[17] Wessel, D. Wright, M. (2002). Problems and Prospects for Intimate Musical Control of Computers *Computer Music Journal*, 26(3), 11-22.

[18] Wright, M. (2003) OpenSound Control: State of the Art 2003,*Proceedings of the 2003 Conference onNew Interfaces for Musical Expression*, Montreal, Canada, 2003.