Object Design Considerations for Tangible Musical Interfaces

Martin Kaltenbrunner
*Music Technology Group
Audiovisual Institute - Universitat Pompeu Fabra
Ocata 1, 08003 Barcelona, Spain
mkalten@iua.upf.es

Sile O'Modhrain
†Palpable Machines Group
Media Lab Europe
Sugar House Lane, Dublin 8, Ireland
sile@media.mit.edu

Enrico Costanza
‡Liminal Devices Group
Media Lab Europe
Sugar House Lane, Dublin 8, Ireland
enrico@mle.ie

Abstract

In this paper we describe object design considerations for the reacTable* project, a novel tangible musical instrument, developed at the Audiovisual Institute at the Universiat Pompeu Fabra. The work presented in this paper is the result of a collaboration with the Palpable Machines Group at Media Lab Europe, which focussed on haptic design aspects of the reacTable* instrument. We present a simple haptic encoding scheme for the mapping of abstract sound synthesis objects onto tangible proxy objects.

1 Introduction

The reacTable* is an electro-acoustic musical instrument in the tradition of Jordà's FMOL synthesizer (Jordà, 2002). The aim is to create a tangible electronic musical instrument that allows expressive collaborative live performances for professional musicians without the limits of many screen-based interfaces for electronic music. Many of these interfaces have very limited control possibilities and provide little feedback on the creative process for both the performer and the audience. As suggested by its name, the reacTable* is a table-based instrument, allowing direct manipulation of any object in the synthesis chain. By arranging a set of objects that are available on the table surface, the performer constructs and plays the instrument at the same time. Each of the objects has a dedicated function for the generation, modification or control of sound flow, and reacts with compatible objects near it. While the table itself is equipped with sensors for the identification and tracking of the objects' position and state, the performers do not need to wear any controller devices or sensors. In addition to the sound which is obviously produced while playing, the reacTable* also provides visual feedback by projecting a graphical representation of the sound and control flow onto the table surface. In order to create a truly multi-modal interface experience particular effort has been spent on the haptic design of the object and table properties. This paper reflects the current state of the instrument, which still differs in various aspects from the final design; especially its size will be significantly bigger than the current prototype. For a more detailed description of the original reacTable* concept see (Jordà, 2003).

2 Instrument Components

During the initial project phase we have been developing the basic reacTable* concepts within a software prototype only, simulating the tangible user interface component with a graphical interface. This approach allowed the rapid prototyping and the introduction of new synthesizer and interaction elements without worrying about sensor and hardware problems. At a second stage we added the d-touch computer vision framework (Costanza et al., 2003a), which allowed the construction of a first stage tangible prototype including a set of reacTable* objects.

The current system was implemented in a completely modular way, allowing the easy reuse or replacement of the five basic functional components. A sensor module tracks the state, position, and orientation of any object that is present on the table. These raw sensor parameters are passed to the central management component, which interprets the user gestures based on the incoming data, generating a dynamic patch network that drives the two actual synthesis components for the sonic and graphical feedback. The synthesis engine is implemented using the open-source PD language (Puckette, 1996). We are currently also integrating a graphics projection system into the prototype, and informal tests show that visual feedback will be crucial for the usability of the instrument. All these components are completely independent and are communicating via a simple proprietary network protocol, which we are considering upgrading to OpenSound Control (Wright et al., 2003) compatibility if necessary. This separation allows execution on various hardware platforms avoiding possible performance bottlenecks since each of these modules requires significant computational resources. In this paper though, we will focus on the tangible controller, which is comprised of a transparent Perspex panel and a set of hand crafted objects, which will be discussed in detail below.

3 Synthesis Object Types

The reacTable* objects can be generally categorized into seven different functional groups: Generators, Audio Filters, Controllers, Control Filters, Mixers, Clock synchronizers and Containers. There are also some exceptions that do not fit within any of these categories.

- *Generators* are sound sources that can produce various types of synthesized or sample based sound. They have an audio output and various control inputs. We are currently considering adding a sound input port to generator objects as well in order to allow FM synthesis.

- *Audio Filters* can modify incoming sound based on their internal algorithms, which can range from a simple band-pass filter to any possible sound effect. Filters have generally one or two sound inputs and a sound output as well as several inputs for control.

Control inputs permit the constant modification of the object parameters that can be controlled either by changing the spatial object properties (e.g. position, orientation, distance to the next object, angle to the next object, distance to the center, angle to the center, etc.), in some cases even its morphological properties (e.g. bending, shape), or by connecting control data flows to their control inputs. These data flows are generated by a third object type, the Controllers.

- *Controllers* generally produce their control data by algorithmic generation which can include from simple low frequency oscillators to complex chaotic or fractal generators. Like in any other object, their respective parameters (e.g. frequency and range in a low frequency oscillator) depend also on the spatial properties of the object, and can be permanently modified. Controllers do not yet have inputs but we plan to implement this feature soon. With some exceptions, controller output is generally adimensional, which means that the effect of a controller depends on the control input it connects to. *Control Filters* process control data. They have a control input and a control output, and unlike regular controllers, their output can sometimes be dimensional; the output values of a harmonizer or a chord generator, for example, are always mapped to pitch.

- The *Mixer* object can take various sound streams as an input and produces a single output stream. Inverted Mixers (*Splitters*) can split a single sound into multiple output streams.

- *Clock synchronisers* introduce a higher hierarchy; they can influence several objects in their proximity at once and in several ways, like sending them synchronised triggers or correcting their low frequencies in order to match a given pulsation. Clock synchronisers have one fundamental parameter, tempo (they also have tempo subdivision), which can be modified by repeatedly hitting the object several times.

- High-level *Container Objects* can virtually contain any pre-built set of sub-patches, allowing the construction of more complex sound structures.

The objects do not need to be connected explicitly: a set of basic connection rules automatically connects compatible objects in respect to their activation, distance or availability. This of course does not exclude the possibility of an explicit connection gesture. See (Kaltenbrunner et al., 2004) for a more detailed description of the Dynamic Patching concept.

4 Object Handling

The objects available on the table can be manipulated by the players in various ways, when placed on the table, an object is identified and activated, moving it on the table surface, its position is tracked as well as its rotation angle. Based on this position and orientation data, inter-object relations such as relative distance and angles are calculated.

Most reacTable* objects are *plain* and *passive*, meaning that they do not come with any cables, switches, buttons whatsoever. The user also does not have to wear special sensors or controller equipment for the object handling: plain hands are the only necessary controller. This, of course, does not rule out the possibility of smart objects that incorporate additional internal electronics in order to retrieve some additional sensor data coming from squeezing, bending or bouncing them, like in the case of the Squeezables (Weinberg and Gan, 2002). In any case, this has 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. A simple rubber hose is an example suggesting some of these additional control possibilities, whose state could be either determined by the computer vision or by using some bending sensors like in the Sonic Banana (Singer, 2003), can serve as a bending controller producing multi-dimensional control data.

More than manipulating the table objects, the hands can be considered to be reacTable* objects themselves, acting as a kind of meta-controller. Tracking of the hands' position and state allows the recognition of various natural hand gestures, such as pointing, painting, waving, etc. Wavetable objects, for example, allow the painting of a waveform next to them, while a simple karate style gesture on a sound flow will result in muting this connection.

5 Tangible Object Types

As already stated above, the reacTable* objects are *plain* and *passive* objects, meaning that they generally do not come with any embedded electronics. This implies that we do not have access to any *active* or computer controlled haptic feedback (vibration, force feedback, etc.), and therefore we can only provide *passive* haptic feedback as defined by the physical object properties only.

The reacTable* objects act as physical and tangible representation of the various virtual synthesis components. They are proxy objects, or *phycons* (Ishii and Ullmer, 1997), which allow the direct manipulation of any of these synthesizer components as required by the performer. Since most synthesizer objects are of rather abstract nature, we decided to reflect this in a more abstract object design as well. Complex everyday objects are used, but have some special functions as discussed below.

We have considered the various haptic dimensions such as shape, size, and material (including texture, weight, density, temperature) to create a suitable haptic encoding scheme for the various abstract object types and their variations, in order to allow rapid and accurate object identification by simply grasping them with the hand. We have been especially concentrating on haptic properties of the object's top surface, but this was mainly due to the lack of available material variations.

5.1 Haptic Encoding

Shape defines the various generic object types, such as generators (square), processors (circle), controllers (star) and mixers (triangle). Simple shapes are easily accessible both visually and haptically, and provide a suitable encoding for the abstract object types. Color would meet similar requirements but is only accessible in the visual domain. Simple geometric shapes can be identified quite easily with a grasp or hand enclosure. More complex shapes would require time consuming contour following with the hand (Lederman et al., 1996) and cannot always be identified completely. Therefore we only defined a small set of easily distinguishable geometric shapes.

Size was not chosen as an encoding dimension, because, in traditional instruments, size often correlates to pitch (tuba – trumpet). Nevertheless we evaluated three different sizes: 4,6, and 9 cm diameter, which can be held and manipulated with three, four, or five fingers, at least by an average adult player. Both 2D (flat) and 3D (cubic) objects were constructed, although this feature is not used for encoding. We are using a wooden cube as a sample player; for example, where each of the six sides represents a different sound sample.

Surface texture was chosen for encoding of the object subsets. We are using two methods to create haptic surface cues. The first is laser engraving onto plastic surfaces to encode abstract haptic feedback, while attaching various materials such as felt or sanding paper onto the objects top surface can represent certain timbral properties of the sounding object. A simple clean sine wave, for example, can be represented with a clean surface; whereas a saw-tooth generator would come with a rough surface. Noise generators have a completely irregular texture and different types of sanding paper can represent a granular synthesizer. Further formal testing will evaluate the correct mapping between the perceived surface and the sonic experience provided by the corresponding synthesis object.

Material We are using both natural and synthetic material with different weight, density, thermal, and texture properties. For each functional object, we are trying to choose a material which haptically represents a close match to the sonic properties of the virtual sounding object. For synthetic sounds, for example, we choose synthetic materials, such as plastic. A sound sampler therefore, is best represented using organic materials such as wood. This early symbolic mapping needs to be evualated in later testing.

Some examples: A sine-wave oscillator is a synthetic sound source with a smooth sonic appearance. According to our haptic encoding scheme, this can be represented by a plastic square with a smooth surface. A simple band-pass filter therefore results in a round plastic disk with a deep engraving through its centre. One of the sound effect filters was constructed by attaching felt on top of a round plastic disk. Furthermore, a wooden cube would be a sample source, while a cube made of a synthetic grainy material represents a granular synthesizer. This scheme was used to encode current reacTable* objects in the most meaningful way to the authors. In the future informal subject test will refine these mappings. Figure 1 shows the first set of reacTable* objects as used in the current tangible prototype.

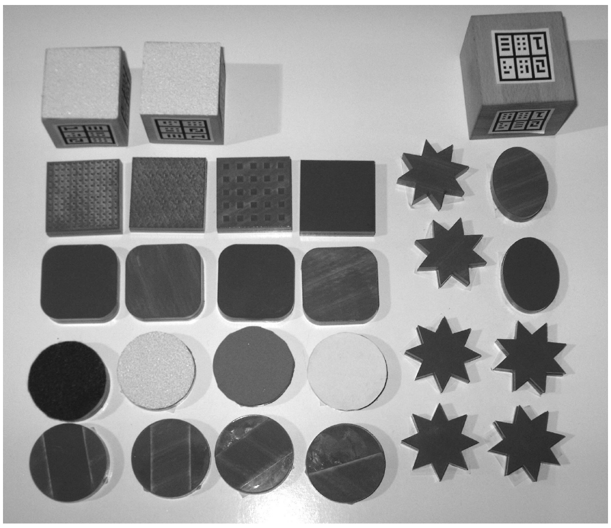

Figure 1: Some reacTable* objects

5.2 Everyday Objects

Ready made *everyday objects* are considered rather for the symbolic meaning and mechanical properties rather than matching them into the haptic encoding scheme. Within the reacTable*, these objects basically have three different functions:

- *Containers* are known and tagged objects that are part of the provided object set, and can be used as sub-patch containers. Due to their highly symbolic meaning, sub-patch containers should be easily identified and remembered by the player. They can include any possible everyday object such as coffee mugs, chocolate bars or rubber ducks.

- *Super-Controllers* Ready made toys such as a (flexible) wooden snake, can be introduced as a multi-dimensional super controller. This of course requires the previous programming of the behavior of such an object, as well as the mapping of the various control parameters. The object state should be tracked completely by computer vision without any changes to the object itself. Only in special cases invisible and wireless sensor technology should be added.

- *Visitor objects* In the context of a public installation one can expect visitors will place their own objects onto the table surface expecting them as well to interact with the intrument. Since one can anticipate somehow what visitors will carry (mobile phones, keys, glasses,) these objects should be identified and integrated into the table: e.g. a mobile phone starts to play an annoying melody, or keys a rattling sound.

5.3 Further haptic design considerations

Haptic orientation cues The table edges are marked with a simple tape, which provides a haptic cue for the table dimensions, because, moving an object over this edge can be felt easily. The same principle was used to mark the table centre by applying a symbol made of transparent tape. This is both haptically and visually accessable, but does not interfere with the computer vision sensor. The localization of the table centre is important for the overall dynamic patch system.

Magnetic objects We also have been experimenting with magnets in order to provide a simple connection or compatibility cue. This idea produces a nice haptic effect, but is unfortunately not very flexible. Ordinary magnets can produce three object classes: positive, negative and neutral. This problem could be overcome with electromagnets that can be switched on or off, and even can change their strength, but this would require significant electrical power, which is not likely to meet our requirements for *plain object* design.

6 Design Constraints

The current prototype is based on computer vision. This has the advantage of simplicity and low cost, requiring only an off-the-shelf USB web-cam. The d-touch framework is based on the localization and recognition of fiducial markers, namely black & white graphical symbols that can be printed on labels and simply attached onto the objects. This system is quite robust thanks to the concurrent design of the markers and detection algorithm (Costanza and Robinson, 2003). The obvious downside of this approach is the need for tagging the objects with visible labels, which is partly overcome by attaching the markers onto the object's bottom side. The choice to place the camera below the transparent table also prevents occlusion of the objects by the player's hand and body during the performance.

The label size, and thus the object size is constrained by the system resolution. This depends not only on the optical resolution of the camera, but more significantly, on the available processing power. The lower the image resolution, the bigger the objects have to be for correct recognition. However, the image processing algorithm's computational cost has been observed to be approximately linear with the number of pixels. In fact, increasing the image resolution over 640 by 480 pixels would result in an unacceptable temporal resolution, which is around 7Hz on our test system based on a 1 GHz Intel Pentium III processor. As discussed in (Costanza and Robinson, 2003), the marker size is also related to the maximum number of different objects supported by the system. We are using a marker set of 120 different symbols, which is currently sufficient, but could be easily exceeded by a larger collection of objects, although it is unlikely that these would be used within a single session.

Currently the label size is around 3 by 3 cm on an interaction surface of around A3. This is acceptable compared to the desired object's size, based on our ergonomic considerations. Additionally, the topological approach used for the recognition (Costanza and Robinson, 2003) allows the design of labels of different shapes allowing more object and symbol variations, such as circles.

Computer vision generally has some considerable performance limitations, such as visual and temporal resolution, as well as some side effects that are degrading recognition performance, such as poor lighting or motion blur. Even scratches on the table surface or dirt on the symbol markers affect the performance significantly.

We are considering the option of employing a hybrid system for a later version of our instrument. RFID tags could be used for the identification and tracking of the reacTable* objects, while computer vision would be utilized for hand gesture recognition and for tracking objects introduced by the player without previous tagging. This should allow faster, more robust, and computationally more efficient object tracking, at a much higher system cost of course.

7 Observations on related tangible musical interfaces

Audio d-touch (Costanza et al., 2003b) is a collection of three tangible interfaces for music composition and performance: the *Augmented Musical Stave*, the *Tangible Drum Machine* and the *Physical Sequencer*. Like the current implementation of reacTable, it is based on the d-touch framework. In the layout chosen for audio d-touch a web cam observes the interactive surface from above so the fiducial labels are clearly visible to the user. This approach suffers from the occlusion problems mentioned above, but permits a simpler system setup. Audio d-touch was conceived as a desktop instrument that can be used on any table; for example, in a house or a school. By arranging the interactive objects on the interactive surface the user can play notes and understand the musical score notation, create drum beats, or record and arrange audio samples in a loop. The interactive area is covered with a printed piece of paper where visual cues give hints about the mapping between the block position and the sound generation parameters.

The design of the interactive objects' shapes has been mainly driven by ease of construction, leading to the use of simple rectangular blocks. These blocks are marked with machine-readable fiducial symbols as well as human-readable cues related to the object function. The musical notes used in the augmented stave have obvious meaning. The tangible drum machine blocks are as small as the system resolution allows them to be: in this case there are only two types of blocks (loud and quiet), so they are differentiated by the color of the sides.

Clearly, the cues currently used are merely visual. Several possibilities to improve the simple block design and make them distinguishable by touch are under consideration. For example, the block's physical size can be related to the note length or to the drum sample volume. In the sequencer application, different block types can be associated to different geometrical shapes. Functional areas on the interactive surface can be carved with different tactile textures.

The Audiopad (Patten et al., 2002) was primarily designed as a tangible instrument controller, the physical objects are mainly used to control a projected graphical user interface. Therefore, the objects, in this case mainly circular pucks, have the basic function of knobs like in a standard MIDI interface, mimicking their tactile and visual appearance. An additional object, the *Selector*, is shaped in a different functional way, which adds directional cues to make it easier to point to the desired selection areas. Both object types have a simple push button on their top side, which allows the triggering of certain actions associated to each object. The Audiopad is using two RF tags for each object to track position and rotation. Due to physical limitations the current system can only track up to nine different objects.

The Music Table (Berry et al., 2003) uses the AR Toolkit (Kato and Billinghurst, 1999) computer vision engine, and reduces the tangible object design to a minimum by attaching the necessary symbols for the vision system onto simple cards. These card symbols are readable both by the user and the computer vision system. Rather than crafting physical objects, the Music Table places virtual 3D objects onto the card surface; a common augmented reality technique. While the physical table contains the set of tangible proxy objects, the player is actually controlling a screen based instrument representation. The system defines an interesting set of musical objects, and also tries to overcome the object-container problem as discussed in (Kaltenbrunner et al., 2004), by defining a manipulation card for virtual objects.

The Musical Trinkets (Paradiso and Hsiao, 1999) are a collection of tiny plastic toys equipped with wireless magnetic ID tags. These objects are pre-loaded (by mapping sounds to their ID) with a certain musical behavior, "such as bird calls, shakers and percussive things" which is activated when an object is placed or moved towards a reader device. Distance to the sensor and speed of movement control the object's sound. Other objects are modifiers, such as pitch-shifters or sound effects including vibrato. The Musical Trinkets also generate visual feedback, which is projected onto the instrument's surface.

BlockJam (Newton-Dunn et al., 2003) uses, unlike the previously listed instruments, a set of sophisticated synthesis objects, which in this case aren't simply proxies for virtual processing elements, but do actually carry the necessary circuits for sound processing within. Basically, they are square boxes with simple plugs on the edges, which allow the assembling of physical sound processing patches. The boxes also come with a small LED display array to provide visual feedback on the object's state, and a touch-sensitive controller to program the object's behavior using a dial gesture.

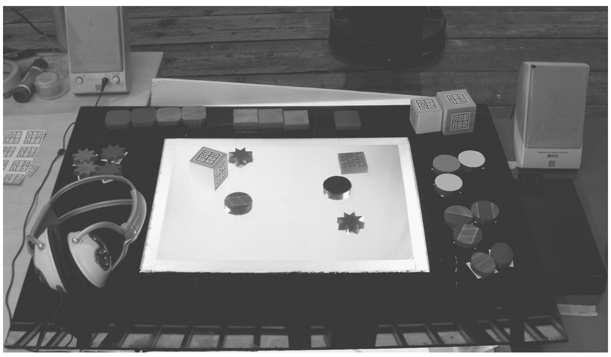

Figure 2: the reacTable* prototype

8 Future Work

In continuation of this work, we are planning to adapt the tangible reacTable* interface as a test platform for a formal evaluation of strategies for object-to-sound mappings in tangible musical instrument interfaces. We are planning to use this platform for the further development and evaluation of our haptic encoding scheme; especially focussing on the tactile surface and material properties and their mapping to sound timbre.

In the near future though we will continue to work on the completion of the sound synthesizer functionality as well as on the integration and refinement of the visual feedback. The final prototype will then also be subject to informal user tests and will be explored within first experimental musical performances. We are also planning to focus on the various aspects of collaborative musical performance.

9 Acknowledgments

The authors would like to thank the following people who have contributed to this work: The reacTable* team: S. Jordà, G. Geiger, and I. Casasnovas at the Pompeu Fabra University. D-touch framework: S. B. Shelley and J. Robinson at the University of York. Media Lab Europe for providing the facilities and materials for object manufacturing, and Sam Inverso for his proof reading. This work was partially supported by a European Commission Cost287-ConGAS action grant for a short term scientific mission.

References

R. Berry, M. Makino, N. Hikawa, and M. Suzuki. The augmented composer project: The music table. In .*Proceedings of the 2003 International Symposium on Mixed and Augmented Reality*, Tokyo, Japan, 2003.

E. Costanza and J. Robinson. A region adjacency tree approach to the detection and design of fiducials. In *Proceedings of Vision, Video and Graphics*, Bath, UK, 2003.

E. Costanza, S. B. Shelley, and J. Robinson. D-touch: A consumer-grade tangible interface module and musical applications. In *Proceedings of Conference on Human-Computer Interaction (HCI03)*, Bath, UK, 2003a.

E. Costanza, S. B. Shelley, and J. Robinson. Introducing audio d-touch: A tangible user interface for music composition and performance. In *Proceedings of the 2003 International Conference on Digital Audio Effects*, London, UK, September 8-11 2003b.

H. Ishii and B. Ullmer. Tangible bits: Towards seamless interfaces between people, bits and atoms. In *Proceedings of CHI 97 Conference on Human Factors in Computing systems*, Atlanta, USA, 22-27 March 1997.

S. Jordà. Fmol: Toward user-friendly, sophisticated new musical instrument. *Computer Music Journal*, 26(3): 23–39, 2002.

S. Jordà. Sonigraphical instruments: From fmol to the reactable*. In *Proceedings of the 3rd Conference on New Instruments for Musical Expression (NIME 03)*, Montreal, Canada, 2003.

M. Kaltenbrunner, G. Geiger, and S. Jordà. Dynamic patches for live musical performance. (submitted to NIME04), 2004.

H. Kato and M. Billinghurst. Marker tracking and hmd calibration for a video-based augmented reality conferencing system. In *Proceedings of the 2nd International Workshop on Augmented Reality (IWAR 99)*, San Francisco, USA, October 1999.

S. J. Lederman, C. Summers, and R.L. Klatzky. Cognitive salience of haptic object properties: Role of modality encoding bias. *Perception*, 25:983–998, 1996.

H. Newton-Dunn, H. Nakao, and J. Gibson. Block jam: A tangible interface for interactive music. In *Proceedings of the 2003 International Conference on New Interfaces for Musical Expression*, Montreal, Canada, May 22-24 2003.

J. Paradiso and K. Hsiao. A new continuous multimodal musical controller using wireless magnetic tags. In *Proceedings of the 1999 International Computer Music Conference*, pages 24–27, Beijing, China, October 22-28 1999.

J. Patten, B. Recht, and H. Ishii. Audiopad: A tag-based interface for musical performance. In *Proceedings of the 2002 International Conference on New Interfaces for Musical Expression*, Dublin, Ireland, May 24-26 2002.

M. Puckette. Pure data. In *Proceedings of the International Computer Music Conference*, San Francisco, USA, 1996. International Computer Music Association.

E. Singer. Sonic banana: A novel bend-sensor-based midi controller. In *Proceedings of the 3rd Conference on New Instruments for Musical Expression (NIME 03)*, Montreal, Canada, 2003.

G. Weinberg and S. Gan. The squeezables: Toward an expressive and interdependent multi-player musical instrument. *Computer Music Journal*, 25(2):37–45, 2002.

M. Wright, A. Freed, and Momeni A. Opensound control: State of the art 2003. In *Proceedings of the 3rd Conference on New Instruments for Musical Expression (NIME 03)*, Montreal, Canada, 2003.