# INSIDE: Intuitive Sonic Interaction Design for Education and Entertainment

Alain Crevoisier and Cécile Picard-Limpens

Haute Ecole de Musique de Genève (HEM)
Rue de l'Arquebuse 12, CP 5155,
CH-1211 Genève, Switzerland
alain.crevoisier@hesge.ch, ccl.picard@gmail.com

**Abstract.** The project INSIDE - Intuitive Sonic Interaction Design for Education and Entertainment, aims at offering children and adults without previous musical experience the possibility to create sounds and make music on a very intuitive and playful manner. We develop a concept of tangible interaction using objects that can be placed on any conventional surface, like a table. The objects can be fitted with meaningful icons representing various aspects and functions related to sound and music, such as sound sources, sound modifiers, or mixers.

**Keywords:** interface, interaction, sound, education, entertainment.

## 1 Intuitive Sonic Interaction Design

We define as object any element that can be grasped and put on a surface. The project explores new ways of making music and creating sounds, by placing objects on the surface and combining them together. Each object is set to a meaning and function related to sound and music: sound sources (sounds of nature, animals, musical instruments, etc), sound modifiers (echo, reverb, filtering, etc.), players, mixers, controllers, etc. Playing sounds, mixing different sound sources, applying sound effects are performed by simply moving objects on the surface. We focus on inventing rules and interactive scenarios that are playful and intuitive, based on the way objects are combined together. For this purpose, we define three object relations: no relation, neighborhood, and object-in-object. Chains are created when two or more objects enter in relation and can be seen as sequences of objects ordered in the 2D space, as shown in Figure 2, bottom-right.

We developed an ad-hoc application, the Surface Editor[1], to facilitate the design of the interaction, by offering the possibility to set the behavior of objects very easily.

[1] http://www.surface-editor.net

A. Camurri, C. Costa, and G. Volpe (Eds.): INTETAIN 2011, LNICST 78, pp. 236–239, 2012.

The Surface Editor receives the object information via TUIO, using a simple webcam hooked to ReacTIVision[2]. This way, various interactive scenarios can be tested and evaluated very conveniently. However, this application is transparent for users. Once an interactive scenario is loaded, the application is running in the background and no intervention from users is required, except for playing with the objects.

Optionally, multi-touch sensing can be added to the setup by using the Airplane controller[3], a device developed in previous projects for transforming any ordinary surface into a multi-touch interface. This is offering extended interactive possibilities. For instance, an object representing a sound source can be played simply by touching it. However, since the Airplane controller is detecting touch by watching fingers crossing a plane of IR light placed a few millimeters above the surface, objects must be thin enough for not interfering with the plane.

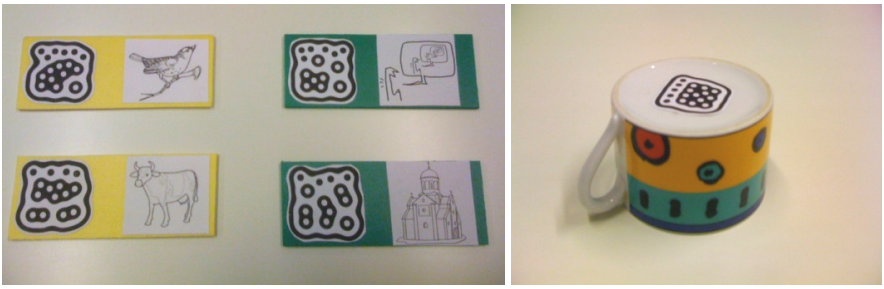

**Fig. 1.** Right : Examples of icons for objects involved in sound interactions scenarios for novice users: sound sources (animals) and sound modifiers (echo, reverberation). Left : Everyday objects, such as a cup, can be used for the interaction.

Figure 1 shows examples of icons for objects involved in sound interactions scenarios. One interesting aspect is that everyday objects can be used in the interactions.

## 2 Description of the Demo

Our aim is to introduce sound design and music creation in a playful manner. For simplicity, we will not interact with touch gestures for the demo presented at INTETAIN. This offers also more freedom to choose thicker objects that are easier to grasp for young users. The interactive scenario we have defined for this demo involves three aspects related to sound and music, with a pedagogical perspective: playing sounds, creating sounds and recording sounds.

[2] http://reactivision.sourceforge.net/
[3] www.future-instruments.net/fr/airplane.php

Playing sounds can be performed in three different ways, manually, with the 'Loop Player' object or with the 'Circular Sequencer' object. To play sounds manually, users have to hold their hand briefly above any 'Sound' object to hide its visual tag. When they remove the hand and the tag become visible again for the camera, then the sound is triggered. The volume of the sound is adjusted by rotating the object. It takes 180° to go from min to max, so there is never any abrupt jump in the value of the volume. If 'Sound' objects are put in the 'Loop Player' object, then they are repeated continuously. The 'Loop Player' object is a larger, thin object that can contain several smaller objects. It illustrates the object-in-object relation (see Figure 2, top-right). Finally, the 'Circular Sequencer' object allows constructing sequences by arranging 'Sound' objects around it. The timeline is similar to a clock hand doing a circular movement.

Creating sounds is performed by taking a base sound and adding 'Modifier' objects (of 'Effect' objects) next to it. This is creating a sound chain and multiple sound modifiers can be added. A special object, the 'Transmitter' object, can be added to the chain in order to be able to represent a whole chain by a single object. This is very practical to play the chain in one of the three ways described above. The emitter is put in the chain and the receiver then behaves like any other 'Sound' object. Thus it can be put in the 'Loop Player' object or used with the 'Circular Sequencer' object. 'Modifier' objects usually have a single parameter that users can adjust by rotating the object (for instance the feedback in the 'Delay' object).

In order to record sounds, users need to put a blank 'Sound' object in the 'Record' object, which is looking similar to the 'Loop Player' but with a red color. Any sound produced will be recorded in the blank 'Sound' object. Silence is not recorded, which means that if a sound is produced, followed by some silence and another sound, then only the last sound will be recorded.

All variables and information events are sent through OpenSoundControl[4] (OSC) using the Surface Editor. The data are collected in Processing[5], using the oscP5[6] library, an OpenSoundControl (OSC) implementation for Processing, and analyzed to generate and control the corresponding sound generators and effects with the use of the Minim[7] audio library.

## 3 Requirements

For our setup, we will bring a webcam, a support to hold it, and a set of objects. We will only need a table and power. However, a loudspeaker would be appreciated. Setup time is approximately of 30 minutes.

[4] http://opensoundcontrol.org/
[5] http://processing.org/
[6] http://www.sojamo.de/libraries/oscP5/
[7] http://code.compartmental.net/tools/minim/

## 4 Setup

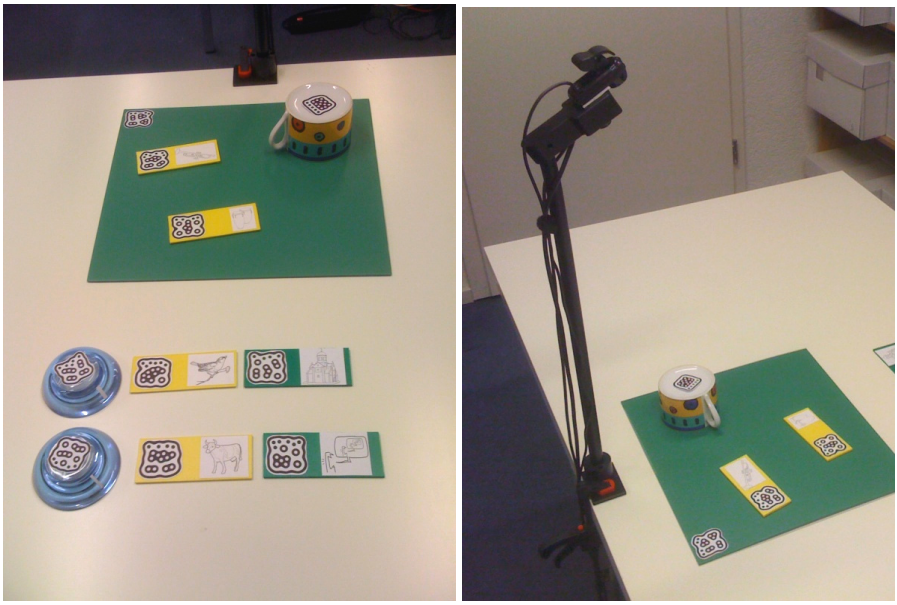

**Fig. 2.** Right : Example of object-in-object relations (top) and chains (bottom). Left : Webcam fixed on a support to detect objects.

## 5 Interest in Participating to Kids' INTETAIN

One of our objectives is to make sound design and music accessible to a wider audience, including children starting from 5 or 6 years old. Participating to Kids' INTETAIN could be the occasion for us to have useful feedback about the playability of the scenario we developed and possibly test some variations. The hope is that the research conducted within this project will be useful for music pedagogy in schools as well as entertainment at home. In future work, we could imagine letting users design their own interactive scenarios. For this, we would need to introduce them to the Surface Editor platform first, which could be possible for teachers or older users.