Sound Rose: Creating Music and Images with a Touch Table

Alain Crevoisier, Cédric Bornand, Arnaud Guichard
University of Applied Science
Western Switzerland (HES-SO / HEIG)
Rue Galilée 15, CH-1400 Yverdon, Switzerland
{forename.surname}@heig-vd.ch

Seiichiro Matsumura, Chuichi Arakawa
University of Tokyo
7-3-1,Hongo,Bunkyo-ku, Tokyo
113-8656, Japan
sei@cfdl.t.u-tokyo.ac.jp
arakawa@mech.t.u-tokyo.ac.jp

ABSTRACT
Sound Rose is an interactive audiovisual installation created in the context of the research about new touch sensitive interfaces. This installation shows how a daily object, such as a table, can be transformed into a musical and visual instrument by using the sound produced when touching and interacting with it. This paper principally describes the hardware and software development behind the installation, as well as the design of the visual and musical interaction.

Keywords
Interactive audiovisual installation, tangible acoustic interfaces, new interfaces for musical expression.

1. INTRODUCTION
The installation was first presented at the Sound to Sense, Sense to Sound Summer School [1], held in Genoa in July 2005. It shows an application of tangible acoustic interface (TAI) as musical and visual instrument [3]. TAI interfaces are developed in the context of the European project of research TAI-CHI [2] (Tangible Acoustic Interfaces for Computer-Human Interaction), and rely on various acoustic-sensing technologies in order to detect the position of contact when touching or interacting with an object [4, 5, 6]. Similar researches have been conducted at MIT MediaLab since the late 90's for the localization of taps on various surfaces to drive graphics and sometimes audio [7]. However, the installation presented in this paper is the first one featuring the continuous tracking of finger touch using acoustic sensing technology. Continuous, multi-touch sensing has been demonstrated recently using the Frustrated Total Internal Reflection technique [9], but this approach is limited to transparent objects, such as acrylic rear-projection panes. The tracking algorithm that we are using has been developed by Politecnico di Milano, one of the partners of the TAI-CHI project, and is based on multiple sensor analysis using the dispersion property of in-solid acoustic wave propagation. In its current development, this method is able to localize and track only a single point of interaction but, on the other hand, it is compatible with a large variety of materials for flat and curved surfaces, which is a main condition if we are interested in building hybrid electro-acoustic musical instruments. In this case, the interface is not only used as a gestural controller, but also as a sound source for further real-time processing and manipulation of the sound, according to the gesture information. Therefore, the acoustic quality of the interface is of primary importance and its choice should not be subordinated to the tracking method.

In last year's first version of Sound Rose, the interaction was based on the localization of taps on a wooden table. In the new version presented in this paper, the interaction is based on the continuous touch-tracking algorithm, allowing enhanced expressiveness for the sonic and visual feedback. Other improvements include the development of a custom hardware module for signal acquisition and processing, with FireWire communication, which will be described in section 3. Accompanying software developed on the EyesWeb platform is described in section 4, and the design of the visual and musical interaction is developed in section 5. The tracking method will be detailed by its authors in a dedicated forthcoming article.

Figure 1. Audience interacting with the touch table.

2. GENERAL DESCRIPTION
The Sound Rose installation consists of a touch sensitive table with images projected from the ceiling. When users touch the table, rose-like graphics are displayed at the point of contact and

Nime'06, June 4-7, 2006, IRCAM, Paris, France.

the raw sound created by the interaction with the table is processed in real-time in order to produce more elaborated sound sequences. Continuous tracking of touch allows to generate movements in graphics and to vary parameters of sound processing. Different sound presets and graphical behaviors can be selected by tapping on virtual buttons at the four corners of the table. Tracked positions and sounds are both recorded and overdubbed in loops, allowing the users to create complex visual and sonic patterns. Another button at the bottom of the table allows to erase recorded data and sounds of the selected preset.

2.1 Layout and Construction

The touch table stands in the middle of the room (about 4x4 m), surrounded by four speakers in the corners (see Figure 2). The video projector is fixed on a stand at the back of the table, about 2 meters above the projection surface. The table is made of a 100x70 cm MDF board laminated with white plastic. A black frame is delimiting the sensitive projection area to 80x60 cm. The tabletop has a tilt of 10° for ergonomic reason. The empty central piece supporting the table allows for hiding all electronic devices. There are two sets of acoustic sensors, eight dedicated to the location of touch (three sensors only are necessary but a higher number increases robustness) and two to pick up the raw sound of the table for further processing. The former are fixed slightly outside the projection area, on the upper face of the table (corners and mid-sides), and the latter under the table, at a position determined experimentally for providing the best sonic response.

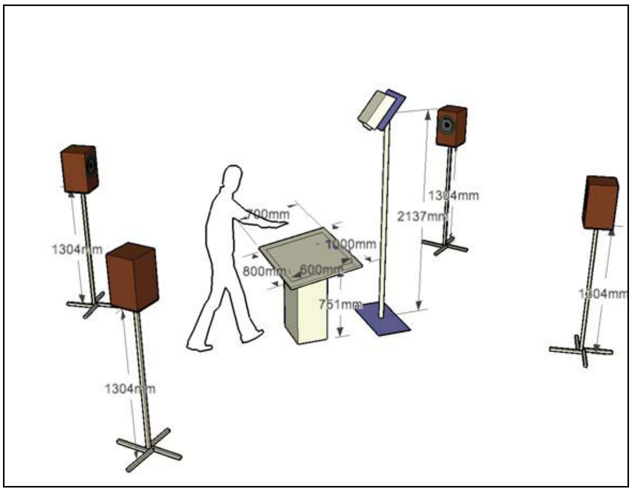

Figure 2. Layout of the installation.

2.2 System Setup

Sensors are connected to the eight preamp inputs of the hardware DSP module. In the current implementation, signals are converted to digital and sent through FireWire to EyesWeb for tracking the touch position (see Figure 3). In the future, the position of contact will be calculated directly on the DSP and sent through Ethernet using OSC (Open Sound Control) [9]. EyesWeb is also detecting if a virtual button is pressed and sends its number to a second computer (Apple, Powerbook G4, 1GHz) running Max, along with the tracked position and signal amplitude. Max is recording the position and amplitude in loop in the custom live sequencer and uses it internally to control the graphic display with Jitter.

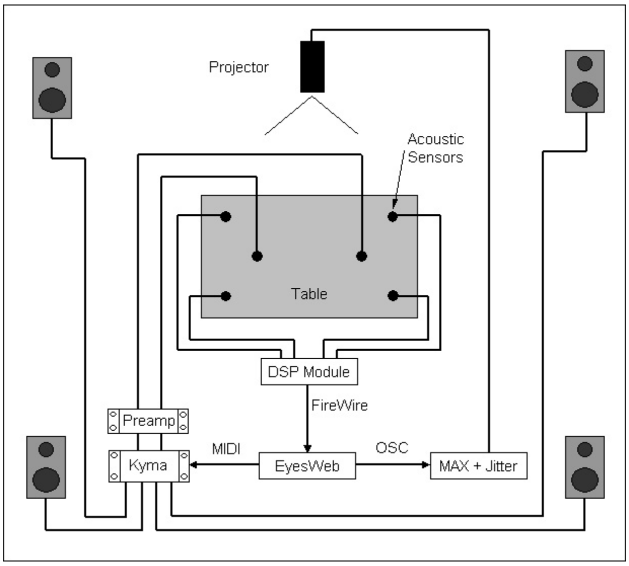

Figure 3. System Diagram.

Position and selected button numbers are also sent via MIDI from EyesWeb to KYMA, which is processing the raw sound of the table, spatializing it, and overdubbing it on a multitrack looping recorder (see figure 4).

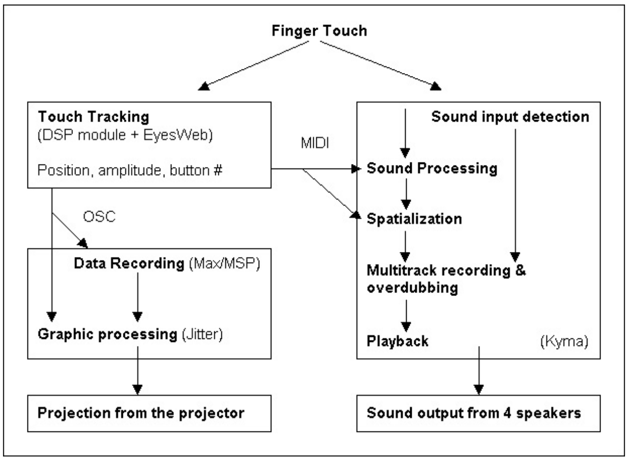

Figure 4. Process Flow-Chart.

3. HARDWARE

3.1 Architecture

The hardware used for the Sound Rose installation is part of an ongoing development concerning the realisation of a modular hardware toolkit for the processing of multimodal interfaces. It can be considered as an electronic Lego for interactive interfaces, capable of processing any kind of signals, such as audio, images, or sensors, thanks to an embedded Digital Signal Processor (DSP). The modular architecture is basically comprised of three different types of cards that can be combined together:

- Acquisition (audio, video, sensors)
- Processing (fixed point, floating point)
- Communication (Midi, Ethernet, FireWire, Wireless)

Several options exist or are under development for each kind of card, allowing to choose the right combination for each application. Cards are staked one above the other, thanks to a common bus for data transfer that is crossing them vertically.

For the Sound Rose application we have used a high sampling rate acquisition board for acoustic sensors, a fixed point DSP board and a FireWire communication board to connect to the PC.

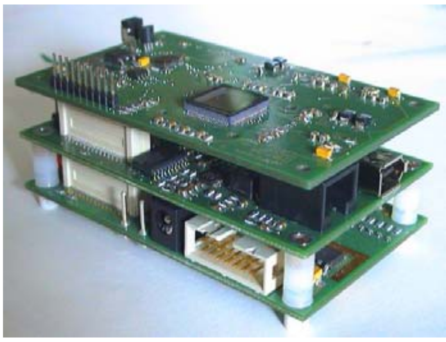

Fig. 5. The Modular Embedded Signal Processing System

3.2 Acquisition

An analog to digital conversion board was designed with 8 simultaneous acquisitions at 500 KHz with 12 bit resolution. A low noise signal conditioning circuit was developed for the acoustic signal amplification, filtering and the signal edge detection. In order to avoid signal loss and interference, a preamplifier placed in the vicinity of the sensor was designed with high input impedance and balanced, low impedance output.

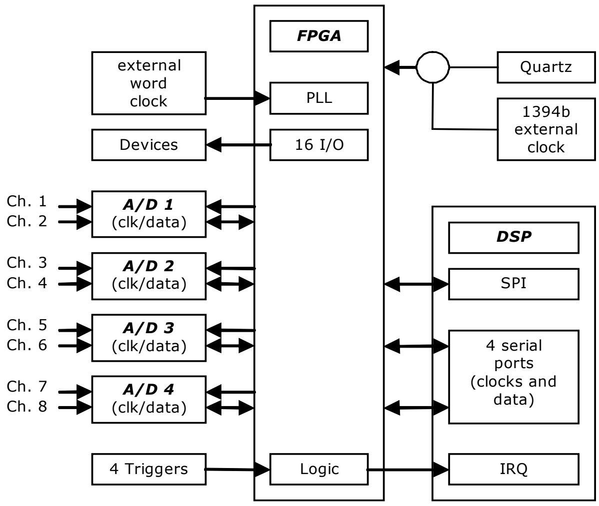

Fig. 6 A FPGA is controlling and interfacing the ADC

3.3 Processing

The processor can be either a floating point SHARC ADSP21161 or a fixed point BlackFin ADSP-BF533 of Analog Devices. The latest is running at a clock frequency up to 600 [MHz] with several interfaces such as 4 serial ports, a 16 bits asynchronous bus, a SPI and a 16 bits PPI and offers a very interesting power/price ratio. A double Blackfin version is under development, with cores running up to 750 MHz.

In the center of this classical architecture, an FPGA shown in Fig. 6 is used mainly to synchronize the signal acquisitions with one of the available clocks: internal (quartz), real-time FireWire bus (1394a external clock), and serial ports of the DSP or external word clock. The handling of the multiple interrupt signals is also done by the FPGA. Finally, 16 digital I/Os are also available on the FPGA for the configuration of the signal conditioning electronics. The 1394 FireWire board was added to the system to provide communication between the DSP module and the PC.

4. SOFTWARE

4.1 IEEE 1394a communication layer

A 1394a communication layer has been realized allowing fast transfer (up to 400 Mbits/s) of video, digital signals and parameters. The communication layer can be divided into two distinguish parts: a DSP library based on a fully compliant 1394 stack and a C/C++ library build upon a 1394 driver for the PC side. Isochronous and asynchronous communications are both available.

The combination of these two elements provides end-users or developers an easy and versatile interface with the hardware. It is possible to configure directly several parameters of the acquisition system. For instance, the user can select the sampling rate of the A/D converters or the frame rate and size of the acquired video. This communication layer consists of a generic library suitable for several platforms such as: EyesWeb which is described in the next paragraph, Matlab, Visual Studio for C/C++ programming, etc.

4.2 EyesWeb

After acquisition and pre-processing, the eight signals are transmitted to EyesWeb [11] using a specific self-build block based on the communication layer (1394 driver) described above.

The tracking algorithm has been implemented in a custom module, as well as the detection of the depressed virtual button. EyesWeb's OSC and MIDI blocks are used to transmit the touch position, button number and signal amplitude to Max and Kyma. The achieved accuracy for the tracking system is about 1 cm.

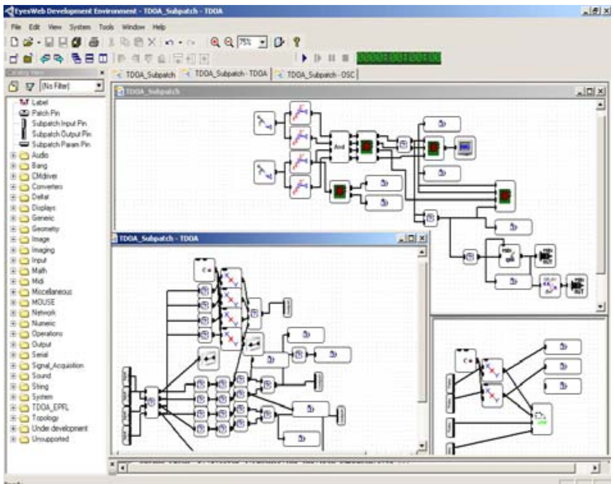

Fig. 7 EyesWeb patch and subpatches developed for the Sound Rose installation.

5. INTERACTION DESIGN

5.1 User Interface

As mentioned before, the user interface is a touch table with graphics projected on it. Interaction can take place in two ways:

a) By tapping on the virtual buttons at the four corners and at the bottom of the display area (see Figure 8).

b) By dragging or tapping one finger anywhere else in the display area.

The four virtual buttons in the corners are used for selecting a different preset. Each preset corresponds to a different sound and a different graphical behavior. The active button is highlighted with a brighter color. The virtual button at the bottom of the display area is used for erasing the sounds and graphics corresponding to the active preset.

Figure 8. Snapshot of the display area

5.2 Graphics

This part is handled by the second computer, running Max and Jitter. Max is used for recording the data sent from the first computer running EyesWeb. Data include the position's coordinates of taps or continuous touch, and the value of the sound's amplitude. The program features a 4 tracks data sequencer capable of recording and playing back data simultaneously. The Jitter part has 2 functions. One is to draw 5 virtual buttons projected on the table. Buttons colored red, blue, yellow and purple, at the corners, are used for selecting the recorded track. A green one indicated at the bottom is used for erasing data. The other function is to convert the set of data to drawing rose-like shapes and control movements of shapes in OpenGL. The direct data received when touching the table and the data played back by the sequencer both trigger the graphics drawn by Jitter. Therefore when the user touches the table, rose-like shapes appear at the same location. The user can add touches one by one to build up the visual sequence or erase data (cf. above). The amplitude value determines the size of each shape. Periodic appearances of rose-like shapes leave traces of user's interaction.

5.3 Music

This part is handled by Kyma [12], in conjunction with the position and control data received from EyesWeb. According to the preset selected by tapping on the virtual buttons, the input sound is directed to a different branch of the program patch. Each of the four sub-patches is featuring a different processing algorithm, with two parameters controlled by the x, y position of the touch provided by the tracking system. Examples of controlled parameters are the cut off frequency and gain of a resonant filter, or the amplitude and frequency of the carrier in modified FM synthesis [10], where the modulating oscillator is replaced by the picked up audio signal.

At the same time that processing parameters are varying, the sound is spatialized and panned with the same x, y coordinates. It is then recorded on a 4x4 multitrack loop recorder. There are four tracks for each preset or sub-patch, in order to keep the panning information of the four speakers. When the input sound exceeds a certain threshold, the processed sound is "punched in" the tracks, by mixing the new material with the one recorded previously. This process could also be described as automatic overdubbing. When the erase button is pressed, the four tracks of the selected preset are erased synchronously.

6. ACKNOWLEDGMENTS

This project has been realized with the support of the TAI-CHI project. The tracking method was developed at Politecnico di Milano by Prof. Augusto Sarti, Diego Rovetta, Gabriele Scarparo, Giovanni De Sanctis, and Prof. Stefano Tubaro.

7. REFERENCES

[1] S2S2 Summer School 2005 - www.s2s2.org

[2] TAI-CHI project - www.taichi.cf.ac.uk

[3] Crevoisier A. and Polotti P. Tangible Acoustic Interfaces and their Application for the Design of new Musical Instruments. *Proceedings of NIME-05*, Vancouver, Canada, 2005.

[4] Rolshofen W, Pham D.T. and al. New Approaches in Computer-Human Interaction with Tangible Acoustic Interfaces. *IPROMs 2005 Virtual Conference*, May, 2005.

[5] Polotti P., Sampietro M., Sarti A., Crevoisier A. Acoustic Localization of Tactile Interactions for the Development of Novel Tangible Interfaces. *Proc. of the 8th Int. Conference on Digital Audio Effects* (DAFX-05), Madrid, Spain, 2005.

[6] R. K. Ing, N. Quieffin, S. Catheline, M. Fink. *Tangible interactive interface using acoustic time reversal process.* Submitted to Applied Phys. Lett., 2005.

[7] Paradiso J. and Leo, C.-K. Tracking and Characterizing Knocks Atop Large Interactive Displays. In *Sensor Review* (special issue on vibration and impact sensing), Vol.25, No.2, 2005, pp. 134-143.

[8] Han, J. Y. Low-Cost Multi-Touch Sensing through Frustrated Total Internal Reflection. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology,* 2005.

[9] Wright M., Freed A. and Momeni A. Open Sound Control: State of the Art 2003. *Proceedings of the 2003 Conference on New Interfaces for Musical Expression* (NIME-03), Montreal, Canada, 2003, pp.153-160.

[10] Poepel C. *Synthesized String for String Players*. Proceedings of the 2004 Conference on New Interfaces for Musical Expression (NIME-04), Hamamatsu, Japan, 2003.

[11] EyesWeb Open Platform - www.eyesweb.org

[12] Symbolic Sound Corporation - www.symbolicsound.com