

Audiopad: A Tag-based Interface for Musical Performance

James Patten

Tangible Media Group
MIT Media Lab
Cambridge, Massachusetts
jpatten@media.mit.edu

Ben Recht

Physics and Media Group
MIT Media Lab
Cambridge, Massachusetts
brecht@media.mit.edu

Hiroshi Ishii

Tangible Media Group
MIT Media Lab
Cambridge, Massachusetts
ishii@media.mit.edu

ABSTRACT

We present Audiopad, an interface for musical performance that aims to combine the modularity of knob based controllers with the expressive character of multidimensional tracking interfaces. The performer's manipulations of physical pucks on a tabletop control a real-time synthesis process. The pucks are embedded with LC tags that the system tracks in two dimensions with a series of specially shaped antennae. The system projects graphical information on and around the pucks to give the performer sophisticated control over the synthesis process.

Keywords

RF tagging, MIDI, tangible interfaces, musical controllers, object tracking

INTRODUCTION

The late nineties saw the emergence of a new musical performance paradigm. Sitting behind the glowing LCDs on their laptops, electronic musicians could play their music in front of audiences without bringing a truckload of synthesizers and patch cables. However, the transition to laptop based performance created a rift between the performer and the audience as there was almost no stage presence for an onlooker to latch on to. Furthermore, the performers lost much of the real-time expressive power of traditional analog instruments. Their on-the-fly arrangements relied on inputs from their laptop keyboards and therefore lacked nuance, finesse, and improvisational capabilities.

The most pervasive interfaces for solving this lack of available inputs have been MIDI controllers based on knobs or sliders. These commercially available devices are useful due to their modularity and their similarity to the interfaces on an analog mixing board. Unfortunately, they have obvious drawbacks. Knobs and sliders are almost too modular: musicians spend more time remembering what each knob does than focusing on the performance. Furthermore, these interfaces lack an expressive character, and it is difficult to control multiple parameters at once.

Two commercially available controllers which attempt to subvert the dominance of knobs are the Korg Kaoss Pad™[13] and the Alesis Air FX™[2]. These effects processors use novel interfaces to allow for multi-axis control of effect settings. The Kaoss Pad™ has a two axis touch pad for changing parameters, while the Air FX™ uses infrared sensing to locate the hand of the performer. Both products require performers to use factory designed effects. Furthermore, while performers can change multiple parameters in one effects

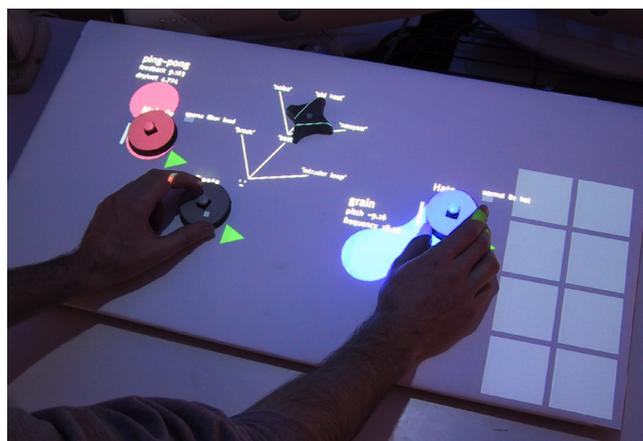


Figure 1. The Audiopad system in action.

program, they cannot simultaneously change the parameters of multiple effects. Instead, they can only modify the settings on an entire stereo mix. One research project that provides a new interface for performing electronic music is the Augmented Groove system. Users of this system can modify the way music sounds by moving vinyl records in three-dimensional space. The system tracks these motions using computer vision, and provides feedback to the user through a head-mounted display[17].

We have developed Audiopad, an interface for musical performance that aims to combine the modularity of knob based controllers with the expressive character of multidimensional tracking interfaces. Audiopad uses a series of electromagnetically tracked objects, called “pucks,” as input devices. The performer assigns each puck to a set of samples that he wishes to control. Audiopad determines the position and orientation of these objects on a tabletop surface and maps this data into musical cues such as volume and effects parameters. Graphical information is projected onto the tabletop surface from above, so that information corresponding to a particular physical object on the table appears directly on and around the object. Our exploration suggests that this seamless coupling of physical input and graphical output can yield a musical interface that has great flexibility and expressive control.

RF TAGGING

Audiopad tracks each puck using one or two RF tags. A simple type of RF tag, known as an LC tag, consists of a coil of wire and a capacitor. This circuit resonates at a specific frequency depending on its inductance and capacitance.

Using clever antenna geometries, these simple structures can be tracked in space using amplitude measurements of the tags' resonant frequencies [15].

There are two well known examples of RF tagging in musical interfaces. The Marimba Lumina consists of several mallets and a large surface with many sensing elements[14]. Each mallet has several RF tags that are detected by the sensing elements inside the striking surfaces. In addition to playing notes, these mallets can adjust different controls to select voices and effects. Musicians can perform operations such as sliding the mallets after striking them to adjust the pitch of a note. Other effects are available by navigating a series of menus on an LCD using the mallets.

The Musical Trinkets[8] project uses a set of RF tags embedded in physical tokens to control a collection of musical sounds. The properties of the sounds change as a function of the distance between the corresponding tag and the sensing antenna. Graphical feedback is rear-projected through a frosted glass plate inside of the sensing antenna.

Several systems employ RF tagging in computer interfaces. The Wacom Intuous uses a sophisticated system of RF coils to track up to two input devices on a two dimensional surface[19]. Sensetable [16] is a platform for developing tangible interfaces [9] based on RF tags that tracks up to nine tags on a flat surface with high resolution and low latency. Its sensing surface includes graphical feedback using a video projector mounted on the ceiling.

IMPLEMENTATION

The Audiopad hardware is a result of further development of the Sensetable system. The current implementation uses much smaller tags than the original Sensetable system as shown in figure 2. The smaller size of these tags provides more flexibility in the physical form of the objects holding the tags. In addition, the current system uses passive tags and does not suffer from the gaps in the sensing surface present in the original Sensetable.

To determine the position of the RF tag on a two dimensional surface, we use a modified version of the sensing apparatus found in the Zowie™ Ellies Enchanted Garden™ playset[4]. We measure the amplitude of the tags resonance

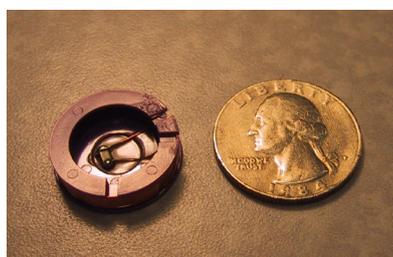


Figure 2. An RF tag used in the Audiopad system.

with several specially shaped antennas. The amplitude of the tag's resonance with each antenna varies as a function of its position on top of the antenna array. This method gives very stable 2D position data accurate to within 4 mm. Each

tag on the table resonates at a different frequency, so their positions can be determined independently.

By attaching two LC tags to a single object, we can determine its position and orientation. The relative positions of the two tags indicate the object's orientation. In objects with two LC tags, we have placed a momentary pushbutton switch in parallel with the capacitor in one of the tags. When the button is depressed, the tag does not resonate. When the tracking software does not detect this tag on the sensing surface, but does detect the other tag in the object, the system infers that the button is pressed and relays this information to the other software components in the system.

The software layer that handles this detection of button presses and orientation, known as the "tag server," also provides several other features that are useful for musical applications. The tag server allows client software applications to provide extra information about the role of each tracked object in the application. The tag server uses this informa-

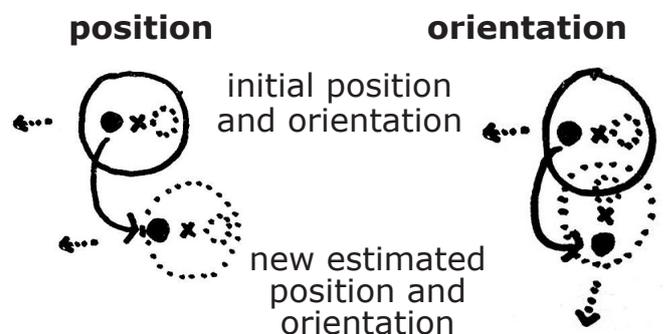


Figure 3: When one of a puck's tags is disabled due to a button press, the system can estimate the puck's state by assuming either position or orientation is unlikely to change.

tion to optimize the tag polling schedule. For example, if it is important for a button press to be detected with very low latency, the software can assign the tag connected to the button a high tracking priority. When the pushbutton on top of a puck is held down for an extended period of time, the tracking information for that tag becomes less reliable.

Because the button press disables one of the puck's tags, the tracking system only knows the position of the other one. Applications can specify in this case whether the user is more likely to adjust the puck's position or its orientation in the current application context. For example, if a puck's rotation controls the volume of a track, the application might ask the tag server to assume that the puck's position is fixed while the button on the tag is pressed (Figure 3). Another task handled by the tag server is tag calibration. The resonant frequency of LC tags can vary over time as a function of temperature. If the amplitude of a tag's resonance decreases below a certain level, the system recalibrates to accommodate shifts in the tag's resonant frequency.

The tag server communicates its tracking information to the video and audio components of the software. These compo-

nents translate the tag positions into graphical feedback on the table using a video projector. They also convert the information into MIDI commands corresponding to specific gestures the user makes with the tags. We chose to adopt the MIDI standard as this allowed us the flexibility of interfacing the Audiopad with any MIDI capable software or synthesizer.

We are currently using Ableton's Live™[1] software as Audiopad's musical back end. Live is a new performance tool for electronic music that arranges sets of sample loops into tracks and allows sample playback on an arbitrary number of tracks. The samples are played back in sync with each other, and are triggered with quantization to the bar. A new sample can be triggered in a track by a MIDI note or controller. Each track can be patched into its own chain of effects, and has controls for volume and pan. All effect parameters and track levels are also controllable by any continuous MIDI controller. In our setup, the tag server sends MIDI data to Live to trigger new samples and to make changes in volume levels and effects parameters.

INTERFACE DESIGN

Audiopad's system architecture provides a great deal of interface design flexibility. One focus of the design is using

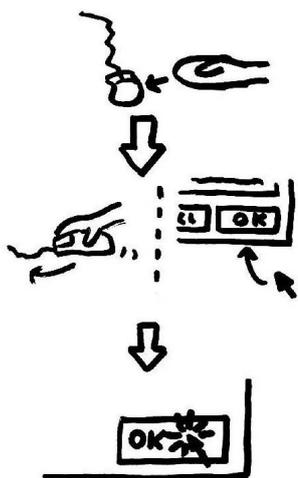


Figure 4a: The three steps of interaction with a graphical user interface.

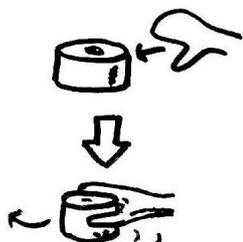


Figure 4b: The two steps of interaction with a graspable interface.

multiple physical objects to mediate a performer's interaction with the synthesis process. Giving physical form to the digital parameters of a synthesizer provides several types of benefits to a performer. First, the performer receives passive haptic feedback when manipulating the objects. This feedback can be especially important in musical applications, where users sometimes must quickly and accurately control a variety of parameters at the same time.

Second, the objects serve as persistent representations of the digital state of the system. One important technique this enables is physically arranging parts of the song on the table. For example, a performer might want to group the rhythm tracks in one area of the table. This process of "offloading computation" by modifying one's environment is a process widely used by experts in many domains to simplify complex tasks [12].

Third, using multiple physical objects in an interface allows a

more immediate level of control than is afforded by a single object. To control more than one parameter with a single physical input device may require as many as three steps (Figure 4a). First, one must grab the physical object. Then one must associate the physical object with the parameter one wishes to control. Finally, one can adjust the parameter with the object [3]. With multiple physical objects, a two state model is more appropriate (Figure 4b). First, one grabs the physical object that corresponds to the desired parameter. Then, one adjusts the parameter by moving the object [5].

Our interface design also focuses on the seamless coupling between input and output spaces. In addition to the audio output produced by the synthesizer, the system provides graphical feedback to the performer about the synthesis process. This information includes the currently selected sample on each track, the volume of each track, whether a track is currently playing, the effect associated with a track, the current parameters of that effect, and whether or not changes in the puck's position will change the effect settings.

Many existing interfaces for digital synthesis combine an expressive interface for manipulating synthesis parameters with an awkward interface for selecting the parameter to be modified, such as a few buttons and a small LCD display. The graphical feedback in our system uses the same expressive interface both to manipulate parameters and to select the parameters to be manipulated. This approach provides performers with flexibility, by making it easy to change parameter mappings in a performance setting.

Interaction Techniques

A performer begins using Audiopad by mapping pucks on the sensing surface to groups of samples in a piece to be performed, as shown in Figure 5. The user assigns a puck to a sample group by placing the puck on top of the desired track in the grid on the right side of the sensing surface. The user then moves the puck back to the middle of the table, where he can select the samples to be played, modify effects, and change sample volumes.

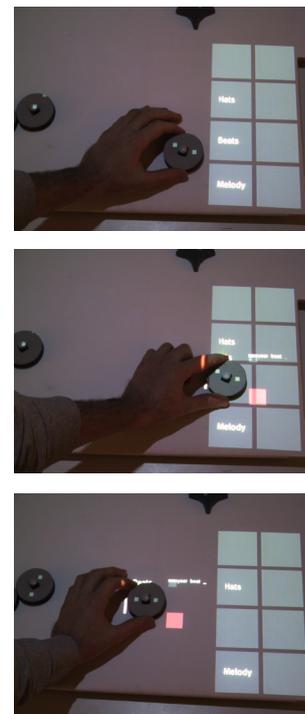


Figure 5. The process of binding a track to a puck.

The use of several physical objects combined with the display of graphical information on and around them enables a rich set of interaction techniques. One such technique is the ability to dynamically associate pucks with tracks. This allows musicians to perform with numerous tracks using relatively few pucks. The track manager on the right side of the

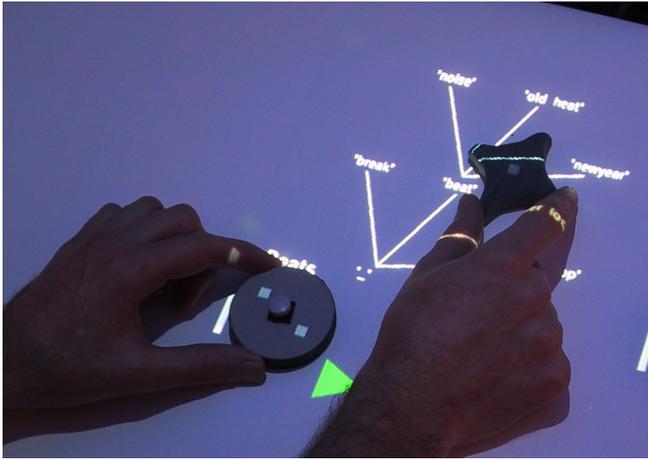


Figure 6. Selecting a sample from the tree using the selector puck.

interface holds all of the tracks that are not currently associated with pucks. The performer can associate a puck with a track by placing the puck on top of it. To remove this association between puck and track, the performer brings the puck back to an empty slot in the track manager.

Once a track has been associated with a puck, the performer can select from a tree of samples using the selector puck, as shown in Figure 6. To reduce visual clutter, the sample tree is not normally shown. The performer activates it by touching the name of the current sample with the selector puck (Figure 7). He can then browse the tree by moving one or both of the pucks; the tree moves along with its associated puck, while the selector puck selects nodes in the tree. When a node is selected, all children of that node on the tree are shown. The terminal nodes represent samples. When the user selects a sample, the system replaces the display of the tree with the name of the newly selected sample. The two-handed tree navigation technique employs the left hand to orient the samples in the tree, and the right hand to select the appropriate target with a tool. This approach mirrors the asymmetric division of labor between the hands suggested by the Kinematic Chain theory [7].



Figure 7. Two Audiopad pucks. The right puck can be associated with groups of samples. The selector puck is on the left.

Users can control the remaining parameters for each track by manipulating the corresponding puck in several ways. The performer can rotate a puck to adjust the volume of the corresponding track. The current volume of the track is displayed to the left of the puck. When the performer presses the button on top of the puck, the system displays information about the effect settings of the track, and movement of the puck controls these settings. The interaction is shown in Figure 9. Pressing the button again removes the display of effect information, as well as the ability to change it. The user can then move the track around on the surface as he wishes without accidentally changing the effect settings.

One important design decision in the development of this interface was whether to use an absolute or a relative mapping between the position of a puck and the effect parameter settings (Figure 8). After experimenting with both approaches, we decided to use a relative mapping. We chose this mapping for several reasons. First, when testing the interface we would usually verbally express changes to the synthesis process in relative terms. For example we might say “increase the filter cutoff a bit” rather than “set the filter cutoff to 8kHz.” If we usually think about making changes to the music in terms of adjustments of the current settings, then the interface should support this representation as well.

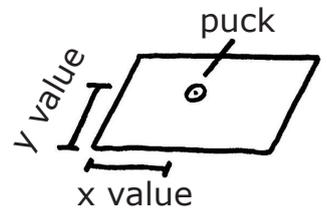


Figure 8a. Use of a puck’s absolute position to determine an effect parameter setting.

Second, if the system were to use absolute puck position for effect settings, the performer would not be able to move the pucks around on the table to organize them, to perform two-handed tree navigation, or to reassign pucks to different tracks.

Third, the effect and volume settings of a track are two conceptually different types of parameters. If absolute puck position were used to control effect settings, users might inadvertently change effect settings while changing volume. This interface would suggest a causal link between volume and effects where there is none. Past research in multidimensional input device selection suggests that users may have a harder time setting parameters with a multidimensional input device when the device uses related physical manipulations to adjust perceptually different parameters [10]. We wanted to differentiate the input gestures for volume and effects, and

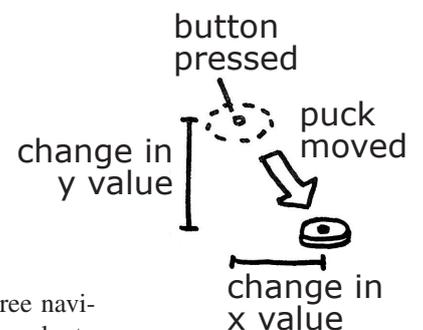


Figure 8b. Use of a puck’s relative motion to determine an effect parameter setting.

this was difficult using an absolute position scheme for effect settings.

EVALUATION

While developing this system, we iteratively refined the interface through an informal process of performance and observation. Below we discuss several of the strengths and weaknesses of the Audiopad interface.

In the initial interface prototype, users could alter the mapping between tracks and effects in the middle of the performance. The intent of this feature was to provide the performer with an added dimension of timbral control. However, in practice we found that performers did not want to change this mapping, since each track in the larger arrangement was generally best suited to one type of effect. For example, our melody tracks were compatible with a configurable delay effect, but were lost using a low pass filter. Pre-assigning effects to tracks helped reduce the interface complexity.

In early versions of the interface, users could start or stop a track using the button on top of the puck. Changes in effect settings could be enabled on a track by touching the selector puck to the bottom of the corresponding puck. With this technique, making a small change to an effect parameter was an awkward process, because users had to activate the effect change mode with a second puck, then make the change, then deactivate the effect change mode. At the same time, we noticed that the buttons on top of the pucks were rarely being used. Performers would typically start all of the tracks near the beginning of the song and not stop them until near the end of the song. We found that a more effective design was to automatically start a track playing when a sample from that track was selected. The track stops when returned to the track



Figure 9a. The user presses the button on a puck to change its effect settings.

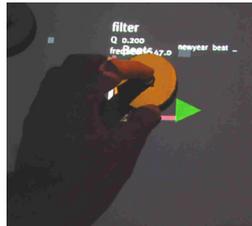


Figure 9b. Audiopad responds by highlighting the position of the puck and showing the effect settings.



Figure 9c. As the user moves the puck, the settings change, and the highlighted area stretches between the puck's initial position and the new position.

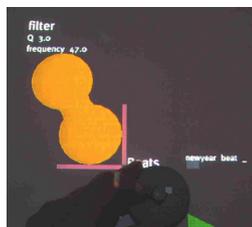


Figure 9d. Here the user exceeds the valid range of parameters for this puck. The stretched color area ceases to follow the puck past the valid region. Two red bars indicate that the valid range is exceeded.

manager on the left side of the interface. If the performer wishes to silence a track without returning it to the track manager, he can spin the puck quickly to reduce the volume to zero.

A technical limitation of the interface is its dependence on expensive video projection from above. We could eliminate the projector by integrating the display with the sensing surface. However, preventing interference between the display hardware and the sensing mechanism would pose a daunting engineering problem. On the other hand, video projectors are increasing in resolution and brightness while they decrease in cost and size, so cost will become less of an issue with time. To make the system more portable, we have developed a prototype of a tabletop projection system using a mirror and a projector resting on the table, facing upward.

On the whole, our users found the system very satisfying to use. They commented that the interface allowed them to accomplish things that are more difficult with other interfaces, such as changing samples on one track while simultaneously changing effect parameters and volume on another track. Our also users found the system visually compelling. In particular, the graphical feedback during the process of changing parameters on an effect helped clarify the relationships between these changes and the corresponding sound output.

CONCLUSIONS AND FUTURE WORK

Our experience with this interface suggests that interacting with electromagnetically tracked objects on a tabletop surface with graphical feedback can be a powerful and satisfying tool for musical expression. The integration of input and output spaces gives the performer a great deal of flexibility in terms of the music that can be produced. At the same time, this seamless integration allows the performer to focus on making music, rather than using the interface.

One feature we plan to add to the system is the ability to apply multiple effects to a single track at the same time. We would also like to explore a richer set of interactions between pucks on the table. For example, if the performer were to bring two pucks close to each other, the tracks associated with those pucks might musically affect each other in some way.

We are also excited about increasing the number of tags that the sensing hardware is capable of tracking simultaneously. This will give performers the ability to physically interact with a larger number of tracks at the same time.

We intend to apply this interface to a variety of synthesis techniques and software packages. One possible technique to which this interface seems well suited is Scanned Synthesis [18]. Because Scanned Synthesis involves the manipulation of a simulated mechanical system which varies over time, the graphical feedback coincident with the physical objects in this interface could be quite helpful in the synthesis process.

In addition, we are interested in exploring the role of this system in the context of musical composition, rather than just performance. One potential use of the system could be the

construction of patches for a modular synthesizer. This interface could allow users to rapidly prototype these patches in a way that made experimentation quicker and easier.

Most importantly, the further evaluation and development of the Audiopad will require road testing in live performance settings. It is perhaps only in this type of environment that we can truly appreciate strengths and weaknesses of this interface for the electronic musician.

ACKNOWLEDGEMENTS

We would like to thank the Things That Think and Digital Life consortia of the MIT Media Lab for supporting this work. We would also like to thank Dan Maynes-Aminzade, Gian Pangaro and Jason Alonso for helping to make this work possible.

REFERENCES

1. Ableton AG, <http://www.ableton.com>
2. Alesis, <http://www.alesis.com>
3. Buxton, W. A. "Three-State Model of Graphical Input," *Proceedings of Human-Computer Interaction - INTERACT '90* (1990), 449-456.
4. Dames, A. N. Position Encoder, U.S. Patent No. 5,815,091 (September 29, 1998).
5. Fitzmaurice, G., Graspable User Interfaces. Ph.D. Thesis, University of Toronto, 1996.
6. Fletcher, R, A Low-Cost Electromagnetic Tagging Technology for Wireless Identification, Sensing, and Tracking of Objects. Master's Thesis, Massachusetts Institute of Technology, 1997.
7. Guiard, Y., "Asymmetric Division of Labor in Human Skilled Bimanual Action: The Kinematic Chain as a Model," *J. Motor Behavior*, 19 (4), 1987, pp. 486-517.
8. Hsiao, K. and Paradiso J., "A New Continuous Multimodal Musical Controller Using Wireless Magnetic Tags." *Proc. Of the 1999 International Computer Music Conference*, October 1999, pp. 24-27.
9. Ishii, H. and Ullmer, B., "Tangible Bits: Towards Seamless Interfaces between People, Bits, and Atoms," in *Proceedings of Conference on Human Factors in Computing Systems (CHI '97)*, ACM Press, pp. 234-241, 1997.
10. Jacob, R. J. K., and Sibert L. E., "The Perceptual Structure of Multidimensional Input Device Selection," *Proc. ACM CHI'92 Human Factors in Computing Systems Conference*, pp. 211-218, Addison-Wesley/ACM Press, 1992.
11. Keyfax inc., <http://www.keyfax.com>
12. Kirsh, D., "The intelligent use of space," *Journal of Artificial Intelligence*, 73(1-2), 31-68, 1995.
13. Korg USA, <http://www.korg.com>
14. O, L., "Marimba Lumina: This is not your mother's MIDI controller," *Electronic Musician*, 16(6), June 2000.
15. Paradiso, J., Hsiao, K., Strickon, J., Lifton, J., and Adler, A., "Sensor Systems for Interactive Surfaces." *IBM Systems Journal*, Volume 39, Nos. 3 & 4, October 2000, pp. 892-914.
16. Patten, J., Ishii, H., Hines, J., Pangaro, G., Sensetable: A Wireless Object Tracking Platform for Tangible User Interfaces, in *Proceedings of Conference on Human Factors in Computing Systems (CHI '01)*, ACM Press, pp.253-260, 2001.
17. Poupyrev, I., et al. Augmented Groove: Collaborative Jamming in Augmented Reality. in SIGGRAPH'2000 Conference Abstracts and Applications. 2000: ACM. <http://www.csl.sony.co.jp/~poup/research/agroove/>
18. Verplank, W., Matthews, M., and Shaw, R., "Scanned Synthesis." 2000. <http://www.billverplank.com/ScannedSynthesis.PDF>
19. Wacom Technology, <http://www.wacom.com>