Beatscape, a mixed virtual-physical environment for musical ensembles

Aaron Albin
Georgia Tech Center for Music Technology
840 McMillan Street
Atlanta, GA 30332
aalbin3@gatech.edu

Sertan Şentürk
Georgia Tech Center for Music Technology
840 McMillan Street
Atlanta, GA 30332
sertansenturk@gatech.edu

Akito Van Troyer
MIT Media Lab
77 Mass. Ave., E14/E15
Cambridge, MA 02139-4307
nav.reyort@gmail.com

Brian Blosser
Georgia Tech Center for Music Technology
840 McMillan Street
Atlanta, GA 30332
bpb54321@gmail.com

Oliver Jan
Georgia Tech Center for Music Technology
840 McMillan Street
Atlanta, GA 30332
oliverjan@gmail.com

Gil Weinberg
Georgia Tech Center for Music Technology
840 McMillan Street
Atlanta, GA 30332
gilw@gatech.edu

ABSTRACT

A mixed media tool was created that promotes ensemble virtuosity through tight coordination and interdepence in musical performance. Two different types of performers interact with a virtual space using Wii remote and tangible interfaces using the reacTIVision toolkit [11]. One group of performers uses a tangible tabletop interface to place and move sound objects in a virtual environment. The sound objects are represented by visual avatars and have audio samples associated with them. A second set of performers make use of Wii remotes to create triggering waves that can collide with those sound objects. Sound is only produced upon collision of the waves with the sound objects. What results is a performance in which users must negotiate through a physical and virtual space and are positioned to work together to create musical pieces.

Keywords

reacTIVision, processing, ensemble, mixed media, virtualization, tangible, sample

1. INTRODUCTION

Virtuosity, in general, is defined as having advanced skills in a particular or multiple musical areas. We are especially interested in interdependent virtuosity, in which performers demonstrate and improve their skills collectively in some areas such as in a band or a symphony orchestra. We are also interested in exploring novel manners for ensemble interaction that cannot be achieved in traditional acoustic means. Our goal, therefore, is to make novel environments that encourage users to interact with one other and allow for new strategies for interaction. Specifically, we would like to:

1. Create an ensemble that has interdependencies between different performers.
2. Allow for a level of virtuosity, not as a single person but in an ensemble context, like a small chamber orchestra
3. Use physical means of creating the sounds without relying solely on a traditional computer point and click interface while allowing for some degree of gesture control.

2. BACKGROUND

There have been many studies of networked, ensemble, multiuser collaborative frameworks [9, 13]. One of the first examples of this came about through the League of Automatic Composers and later iterations of the Hub, in which users could both produce and alter each others music information [2, 7]. More recent notable examples from which have parallels to our work include the Reactable which is a tabletop interface allowing performers to use a common area, moving physical objects on a surface which are essentially generative audio synthesizers [10]. While the Reactable provides for a very virtuosic ensemble instrument, the interaction with the Reactable is limited to placing objects on a table, which does not allow for richer and more expressive gestural input such as continues hand gestures in 3D.

An example for an interdependent musical instrument is the Tooka, a wind instrument that forces the users to interact with each other by playing on two ends of a hollow tube. Players place opposite ends in their mouths and modulate the pressure in the tube, controlling sound. Coordinated button presses control the music as well, thus tending to create music that explores intimacy and cooperation [5].

The Reactogon is another instrument has the property of interconnections between sound objects causing activities like chain reactions and triggering other objects [3]. It quantizes the space and uses the idea of a harmonic table to allow for easy creation of chords and other musical patterns.

Another environment that uses a physical interface to manipulate virtual sound objects is Drile [1]. Here, musical nodes are represented as worm like objects which can be manipulated and grouped into trees residing in virtual rooms. It is a live looping musical environment by which each worm can be manipulated by scrubbing them through different tunnels causing different sound effects.

Beatscape is similar to these projects with respect to using physical objects to manipulate virtual entities. We do this using gestural and pointing devices as well as tangible interfaces. It also is an environment that encourages groups to work together to achieve musicality. The contribution we

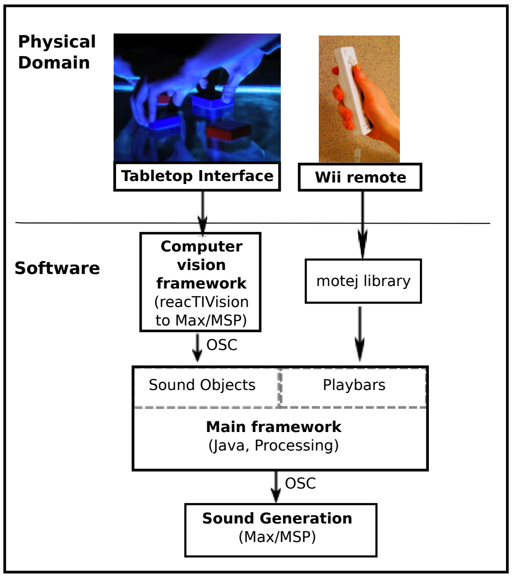

Figure 1: Beatscape Setup

present in this project is the combination of different gestural interactions by two different sets of performers, and the requirement to synchronize gestures based on calculated time delays between gestures and sound triggering.

3. SETUP

In this section we will describe the physical and virtual domains of Beatscape including the hardware and software architectures used to create the environment (Figure 1).

3.1 Physical Domain

Some "conventional" instruments, such as stringed or brass, require the player to skillfully use both hands in order to generate the sound. Each hand has several different techniques to fulfill its role and the synchronized actions of the hands contribute to the pitch content, timbral quality and timing of the sounds. As an example, on a guitar, the left hand typically defines the notes to be played while the right hand causes the sound to come forth. In the guitar case, we usually have one performer controlling both hands; therefore, we need to take account of the interaction between multiple users. In Beatscape we take this paradigm and separate it to two different types of players: those who can place or set up the sound and those who can trigger them.

3.1.1 Tabletop interface

To manipulate the sound objects, we decided to use a tangible table-top interface. We used a set of children's toy blocks as the physical manifestations of the sound objects. Each block has a fiducial marker, specially designed for the reacTIVision framework [11] (explained in Section 3.2.1). The marker is taped to the bottom of the block, placed on a transparent glass table. A camera is placed beneath the table to detect the markers. To quickly identify the sounds during the performance, we drew a picture of the virtual avatar on top of the blocks. The sides of table are covered with dark cloth, so no external light might cause problems. Additionally, two diffuse-light spotlights are placed beneath the table so that they illuminate the markers sufficiently.

3.1.2 Wii remote

To create triggering objects that would be used to collide with the sound objects we decided to use the Nintendo Wii Remote with the motej library [6] to give us some degree of gestural movement. While we considered using a more advanced gestural toolkit or a machine learning mechanism for expressive gestures, we realized that the simple act of pointing using the "sensor bar" and the IR camera attached to the Wii remote was an effective method to identify the specific sound object we wanted to trigger. Triggering was accomplished merely by a simple threshold detection of the acceleration component. More advanced gestures were unnecessary since the Wii remote has buttons and a directional pad allowing us to select different options with ease.

Figure 2: Sound object avatars, one-shot playbars and reacTIVision toolkit projected side by side

3.2 Software Architecture

The software component of Beatscape can be divided into the computer vision framework, the performance framework (Figure 2) built primarily in Java using the Processing API for drawing sound objects and playbars on the screen, and the sound generator realized in Max/MSP.

3.2.1 reacTIVision

We used the reacTIVision toolkit as the computer vision framework, as it enables fast and robust way of tracking fiducial markers. It allowed us to associate the physical blocks tagged with a fiducial marker for each of our sound objects without a need for more advanced image processing. The data is sent through a Max/MSP patch that acts as a mediator between reacTIVision and the main Java application via Open Sound Control (OSC).

3.2.2 Sound Objects

When deciding how the sound objects would be visually displayed for the performers and the audience, we wanted the visual images to be easily associated with the sound and the objects to be animated upon being triggered. We decided to design the visual avatars in Inkscape using SVG format (Figure 3), because this allows sparse representations of simple images and enables us to animate different parts of an image separately. For example, one could animate the eyes nose and mouth of a face image independently. A number of different animations were explored, including rotations, size expansions, and blinking. The avatars are added, moved and removed by the commands received from reacTIVision.

3.2.3 Playbars

We settled upon three basic types of playbar objects that create sound upon collision with sound objects. Each player is represented either by a circle or a square as their base triggering object. They can point on the screen using the IR "sensor bar". In order to deal with the problem of having jittery motions caused by hand movements, we allow the wii mote users to freeze and unfreeze their cursors by pressing

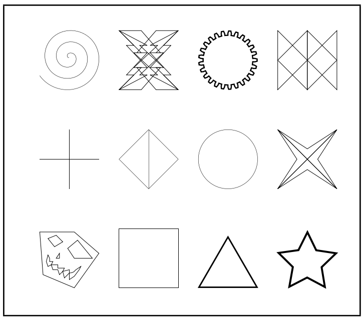

Figure 3: Sound object virtual avatars

the button "A". Then, to create a playbar object, the user must make a jerking motion.

The first type of triggering object is a "one-shot, wavefont-like" playbar where their shape expands from their cursor to some maximum size (Figure 2). The rate of expansion is controlled by the acceleration component. We set a threshold to choose two different rates of expansion: one slow and another fast. The second type of triggering object is similar to the first, except that once it reaches its maximum size, it restarts and repeats its expansion again.

The third type of playbar is a flashlight object in which the cursor is a maximum sized circle or square, filled in. When the flashlight object hovers over a sound object, it causes it to play at its maximum repetition rate. Additionally, by pressing the button "B", the flashlight can be removed from the cursor so that it remains in the spot, allowing the user to create a new flashlight again by making a jerking motion (Figure 4).

A repeating or the flashlight playbar can be selected with the directional pad and deleted by pressing the "-" button. Additionally, for a quick silence mechanism, the "+" button removes all playbars.

3.2.4 Sound Generation

When a sound is triggered, an OSC message containing the sample name and the playback rate is sent to a Max/MSP patch that plays the samples. We use the so-called "gating" mechanism often seen in many hip hop sampler instruments. When a sound is triggered and then re-triggered quickly, the first instance is cut off and the sample is played from the beginning. We ensure that any subsequent trigger of any sample will occur with a inter-onset interval of 100ms at its fastest. This was a decision influenced by our personal aesthetics in hip hop and popular dance and electronic music. Moreover, as an aesthetic decision, the samples are either hard-panned to left and right to achieve spatialization.

4. TECHNIQUES OF THE PERFORMERS

Through our interactions in preparing a piece, a number of different techniques were developed by the Wii remote players as well as the sound object placers.

4.1 Wii remote Players

For the Wii remote players using the one shot bars, they can point to a specific location, lock, and create multiple waves from that point. The players must take into account the time it takes for the waves to expand and collide with a sound object. If they want an immediate triggering, they must point at the object directly and then launch a wave. However, they can also have a "delayed" trigger by creating waves near object, giving a degree of spatio-temporal control over collisions. Another technique is to create a "trailing" wavefront by keeping the cursor unfixed, moving

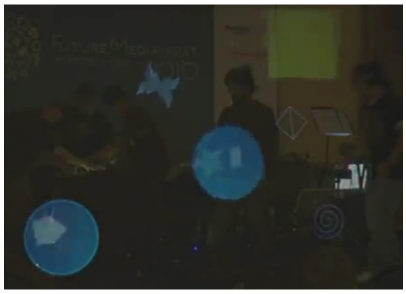

Figure 4: The virtual and physical domains: Virtual interface shows sound and flashlight objects. The sound object players (to the left) and Wii mote players (to the right) are presented in the background

it across the screen and tapping the Wii remote to create new expanding playbars.

With repeating playbars, one technique is for Wiimote players to make "minefields". This allows the sound object players to take control of the expressivity by deciding where to place the sound objects within the expanding fields. Also, by creating multiple repeating bars in the same location at different time intervals, the sound objects will be triggered in complex rhythmical patterns.

With the flashlight object, by hovering over a sound object it will cause the sound to play repeatedly at its maximum rate. By flicking the Wii remote over a sound object, it can strobe the sound object. Another feature of the flashlight is to disassociate it with the cursor and leave it behind for a sound object player to use.

4.2 Sound Object players

Sound object players have a number of techniques. The angle of a sound object is mapped to four discrete playback speeds; thus, these players are responsible for the harmonic and melodic progression of the piece. However, this is dependent upon whether or not there is a playbar to trigger the sound. This interdependency forces different instances for pitch changes: when the playbars are of type single shot, the sound object players have to prepare for rotating the object in advance, whereas in the minefield and the flashlight example, they can change the pitch quicker without need to anticipate as in the previous case.

Sound object players can influence Wiimote players' trajectories by forcing them to decide which route to take if using the "trailing" technique. Putting the objects in proximity to each other allows Wiimote players to simultaneously or consequently trigger multiple sound objects based on relative spacing between objects and playbars. Conversely, in the "minefield" approach, sound object players can decide when to trigger the objects by placing the blocks on and off of the table; the sound players can use this technique to create complex rhythmic patterns.

The sound object players can also give objects character in a limited fashion. For example, the objects can be wiggled both to get the attention of the audience as well as to get the attention of the Wii remote players in order to remind them when to trigger them at certain points in the piece. The objects which have facial or more anthropomorphic features such as the scream/jack-o'-lantern combo seem to have more personality.

5. COMPOSITION AND PERFORMANCES

For our performance, we composed a structured improvisation piece with three distinct sections that were meant

Sample Type	Name of the Sound Objects
Percussive	Snare, Bass Kick, Scratch, Click
Guitar Chords	"High", "Tension", "Bass", "Finish"
Misc. Effects	Scream, "Whoosh", "Synthesizer", "Clap"

Table 1: Types of sound samples used in the performance and sound objects associated with each type

to showcase different techniques. The first section dealt with introducing the environment to the audience and then arranging the sound objects into a grid, putting the emphasis on the Wii remote players. The second section is a "minefield" approach whereby control is handed over to the sound object players. The third section, where both sound objects and playbars are free to move, showcases the flashlight approach. The piece we performed illustrates all the techniques discussed previously.

We contributed sound samples based off of our current listening interests. After collecting various sounds, we organized them into percussive sounds, chord structures, and miscellaneous effects like screams and "whoosh" (Table 1). We used very highly compressed guitar chords from the band Justice, from their debut album *Cross* [4]. All of the percussive sounds and two of the effects are from the Freesound Project [12]. The "clap" and "synthesizer" effects are from FL Studio Legacy Pack [8].

The first performance of Beatscape was in Listening Machines 2010, the annual concert series hosted by Georgia Tech Center for Music Technology, showcasing the work of masters students in the Music Technology program. The performance took place at the Eyedrum in April 2010, in Atlanta, Georgia. The piece was performed again in October, 2010 with Aaron Albin, Sertan Şentürk, Avinash Sastry, Andrew Collela and Sang Won Lee in the FutureMedia Fest 2010, hosted by Georgia Institute of Technology[1]. During the performances we used two screens: one for Reactivision output and the other for our visualization. We decided to show Reactivision for the audience who could not see the table from the back of the hall, so that they would understand how the avatars are controlled.

6. EVALUATION

For the Wii remote players, it was very easy to trigger sounds on the screen and set up different playbars. For the sound object players, the table top interface also proved to be a very intuitive. However the real challenge and skill came about through cooperation both within the Wii remote players and the sound object players so that the performance wouldn't become cluttered. Although we did not conduct user studies to assess the learning curve and the development of skill, by practicing we learned to cooperate with each other and give each other some space, thereby achieving our objective to work along as an ensemble.

The performances were well received. Through informal discussions with audience members, they were seen as self-explanatory and both visually and aurally appealing.

For future iterations, we would like to allow importing any type of image as well a means to associate an arbitrary sound to an image in real time, which could make for an interesting networked variation that adds more spontaneity to the improvisation. We would also like to change the animation with respect to the pitch of the sound object and give the avatars more of a personality for better visualization, feedback to the players and more association to the visual to the audio for the audience. The creation of sound objects can further be associated with custom animations associated with the sounds, giving the sense that they have an intention. Additionally we hop to conduct user studies to further explore collaboration and usability of Beatscape.

[1]Both performances are available online at http://vimeo.com/11676226 and http://vimeo.com/16113180

7. CONCLUSIONS

We have presented Beatscape as a virtual/physical environment that encourages ensemble virtuosity. While sound generation is a simple process of a sound object colliding with a playbar, by separating the tasks among different players, we force the users to work together to create coherent pieces. Thus Beatscape is easy to understand and start out playing but requires a group effort in order to achieve something musically meaningful.

8. ACKNOWLEDGMENTS

We would like to thank Avinash Sastry, Andrew Collella and Sang Won Lee, for assisting with the performance in FutureMedia Fest 2010, Atlanta, GA.

9. REFERENCES

[1] F. Berthaut, M. Desainte-Catherine, and M. Hachet. DRILE: an immersive environment for hierarchical live-looping. In *Proc. of NIME 2010*, pages 192–197, 2010.

[2] J. Bischoff, R. Gold, and J. Horton. Music for an interactive network of microcomputers. *Computer Music Journal*, 2(3):24–29, 1978.

[3] M. Burton. reacTogon. http://www.youtube.com/watch?v=AklKy2NDpqs, 2008.

[4] Cross. Justice. Album, June 2007.

[5] S. S. Fels and F. Vogt. Tooka: Explorations of two person instruments. In *Proc. of NIME '02*, pages 116–121, May 2002.

[6] V. Fritzsch. motej, a slim Java library for Wiimote communication. Sourceforge web-site: http://motej.sourceforge.net/index.html, 2009.

[7] S. Gresham-Lancaster. The aesthetics and history of the Hub: The effects of changing technology on network computer music. *Leonardo Music Journal*, 8:39–44, 1998.

[8] Image-Line. FL Studio Legacy Pack. Software.

[9] S. Jordà. Multi-user instruments: models, examples and promises. In *Proc. of NIME '05*, pages 23–26. National University of Singapore, 2005.

[10] S. Jordà, M. Kaltenbrunner, G. Geiger, and R. Bencina. The reacTable*. In *Proc. of ICMC '05*, pages 579–582, 2005.

[11] M. Kaltenbrunner and R. Bencina. reacTIVision: a computer-vision framework for table-based tangible interaction. In *Proc. of the 1st international conference on Tangible and embedded interaction*, pages 69–74. ACM, 2007.

[12] D. Murphy, Raphael "HardPCM" Couturier, M. Carrier, J. Steiner, cdrk, and hgavin. Sound samples. The Freesound Project: www.freesound.org.

[13] G. Weinberg. *Interconnected Musical Networks - Bringing Expression and Thoughtfulness to Collaborative Music Making*. PhD thesis, MIT, 2002.