ARmonica: A Collaborative Sonic Environment

Mengu Sukan Ohan Oda Xiang Shi Manuel Entrena Shrenik Sadalgi Jie Qi Steven Feiner

Department of Computer Science, Columbia University
New York, NY, 10027, USA
{mengu, ohan, feiner}@cs.columbia.edu, {xs2146, mae2135, ss3650, jq2152}@columbia.edu

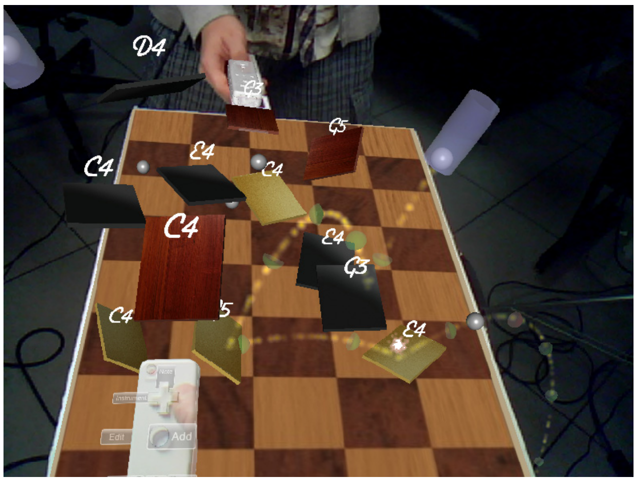

Figure 1. Two players ready to add bars to the environment. (Tracked Wii remote controllers, and parts of players' hands are visible because occluding geometric models of the remote controllers are added to the scene.)

Figure 2. Player edits the highlighted bar, changing its position and orientation. Translucent trail of ball's trajectory includes hemispherical tick marks at regular temporal intervals.

ABSTRACT

ARmonica is a 3D audiovisual augmented reality environment in which players can position and edit virtual bars that play sounds when struck by virtual balls launched under the influence of physics. Players experience ARmonica through head-tracked head-worn displays and tracked hand-held ultramobile personal computers, and interact through tracked Wii remotes and touch-screen taps. The goal is for players to collaborate in the creation and editing of an evolving sonic environment. Research challenges include supporting walk-up usability without sacrificing deeper functionality.

Author Keywords: Augmented Reality, Sound

ACM Classification Keywords: H5.1 [**Information interfaces and presentation**]: Multimedia Information Systems—*Artificial, augmented, and virtual realities*.

General terms: Design, Human Factors

INTRODUCTION

ARmonica was developed to explore collaborative audiovisual play, using heterogeneous displays and interaction devices. (Its name, while technically an organological misnomer, is derived from Benjamin Franklin's name for his mechanical glass harmonica [6].) ARmonica players work together to create a 3D augmented reality (AR) environment in which virtual balls are fired at virtual bars that, when struck, emit sounds defined by musical instrument samples, accompanied by particle effect visualizations (Figure 1). Players place bars and ball launchers that hover over a table covered by an optical marker array. Ball launchers launch virtual balls into the scene based on player-set parameters, including ball generation frequency and initial velocity. Balls and bars interact in a physically realistic manner; each time a bar is hit, it generates a sound corresponding to a player-assigned instrument and note. Players are provided with context-dependent visual overlays to assist in designing the environment.

ARmonica was inspired by several earlier systems. Budge [1] pioneered the idea of player-constructed physics-based computer games, created with a virtual pinball construction set. Lytle [3] developed software that used a score to generate physically-based animation offline of fanciful virtual percussion instruments, one of which was played by collisions with virtual balls. Later, Lieberman and Paluska's "Absolut Quartet" [2] used robotic ball launchers to play a real marimba. Vincent's Mandala [7] AR system triggered sounds when a user's silhouette intersected with 2D iconic instruments. In "Augmented Groove" [5], an AR application by Poupyrev et al., multiple participants wearing head-worn displays (HWDs) make music together by manipulating physical cards on a table to control pre-composed musical sequences. Nimoy's BallDroppings [4] is an interactive game based on a 2D physics simulation in which virtual balls fall from a point on the screen and bounce off lines drawn by the user using a mouse; each time a line is hit, a sound is generated whose pitch depends on the ball's

UIST'10, October 3–6, 2010, New York, New York, USA.
ACM 978-1-4503-0271-5/10/10.

velocity. ARmonica thus situates the design of a physically-based sound-generation structure [2–4,7] within a 3D augmented space [5].

DESIGN

Interaction

Some players wear a tracked, video see-through HWD and manipulate the scene with a hand-held Wii remote controller, tracked in 3D with a small, rigidly affixed, optical marker array. Other players hold a tracked ultramobile PC (UMPC) to fire balls at the scene. A ground plane optical marker array establishes a common 3D coordinate system.

The Wii remote controllers support add and edit modes. In *add* mode, a player can select the object to add (bar or ball launcher) and place it at a desired position and orientation, optionally selecting parameters (e.g., a bar's pitch) prior to adding it. In *edit* mode, a player can select an existing bar or ball launcher and change its position or orientation (Figure 2), modify its parameters, or delete it.

In contrast, while the tracked UMPC allows the player to view the augmented scene through its integrated camera, its only influence on the scene is to fire balls out of its back when the player taps the touch screen. Thus, it provides a simple "one-click" way for a hesitant player to interact with ARmonica without needing to wear an HWD or learn how the Wii remote controller works.

Visualization

To visually indicate the instrument assigned to a bar, we associate each instrument with a different texture (e.g., rosewood for marimba, metal for vibraphone, and black lacquer for piano). Ball launchers (Figure 1) are translucent cylinders placed by the players. New balls are generated according to player-assigned frequency and velocity. Once a ball is generated, it moves from one end of the cylinder to the other with constant velocity, to visualize the launch frequency. When the ball reaches the other end of the cylinder, it is launched into the world.

We have tried to make it easy for casual players to construct sonically and visually interesting environments. The Wii remotes are overlaid with tracked virtual labels that document each button's current function. To guide the placement of bars and ball launchers, a player can specify that the next ball emitted from a launcher should produce a translucent trail (Figures 1–2). Tick marks in the form of transparent hemispheres are displayed along a trail at locations that segment the trail into equal temporal intervals. These can visually assist a player in establishing a regular rhythm, if desired, by positioning bars tangent to the hemispheres. (Because the balls accelerate and decelerate when acted on by forces such as gravity, equal temporal intervals generally do not correspond to equal spatial intervals.) Players can also enable or disable the display of note names on bars in scientific pitch notation.

IMPLEMENTATION

Software

We developed ARmonica using GoblinXNA, a 3D software framework based on Microsoft XNA Game Studio 3.1. GoblinXNA supports external networking, vision-based tracking, and physical simulation libraries. Networking uses the Lidgren Networking Library, which supports message sharing between server and client machines. For 3D position and orientation tracking, we rely on the ALVAR tracking library, which supports optical marker-based tracking. Physics-based simulation in ARmonica is accomplished with the Havok Physics engine, which allows virtual objects with physical properties to be manipulated using the tracked Wii remote controllers, interfaced through Bluetooth using WiimoteLib under .NET. We feed the position and orientation information from ALVAR to Havok to produce physical forces corresponding to motion observed by the camera.

Client-Server Architecture

The physical simulation is performed on a server machine; all connected client machines receive updated positions and orientations of physically simulated objects in the scene, so that they share a consistent, synchronized scene state, which each client renders from its own position and orientation.

Each client machine, excluding the UMPCs, is paired with a Wii remote controller to interact with the environment. The location and orientation of the controller is tracked locally by its associated client machine, and transmitted to the server to update the position and orientation of the bar or ball launcher attached to it for physical simulation in add or edit mode, or to perform selection in edit mode. Even though each controller is associated with a client, its button presses are recognized by the server, and the consequences are transmitted to the clients. The UMPCs simply transmit screen taps to the server, and a ball is added for each tap on the server side with a linear velocity and direction based on the position and orientation of the UMPCs.

CONCLUSIONS AND FUTURE WORK

We have presented ARmonica, a collaborative AR application for creating and editing a sonic environment. Ongoing work includes developing additional interaction and visualization facilities to aid players in positioning and orienting ball launchers and bars.

ACKNOWLEDGMENTS

This work was funded in part by gifts from Microsoft Research, VTT, and Vuzix.

REFERENCES

1. Budge, B. *Pinball construction set*. BudgeCo, Piedmont, CA, 1982.
2. Lieberman, J. and Paluska, D. Absolut Quartet, http://www.aec.at/archiv_project_en.php?id=16387.
3. Lytle, W. More bells and whistles. In *SIGGRAPH '90 Film Show*, 1991.
4. Nimoy, J. *BallDroppings*, http://balldroppings.com.
5. Poupyrev, I., Berry, R., Kurumisawa, J., Nakao, K., Billinghurst, M., Airola, C., Kato, H., Yonezawa, T., and Baldwin, L. Augmented groove: Collaborative jamming in augmented reality. In *SIGGRAPH 2000 Conference Abstracts and Applications*, 2000, p. 77.
6. Rossing, T.D. Acoustics of the glass harmonica. *The Journal of the Acoustical Society of America*, vol. 95, pp. 1106-1111, 1994.
7. Vincent, V.J. *Mandala*. http://www.vjvincent.com/videoclip.htm.