HAPTIC THEREMIN: DEVELOPING A HAPTIC MUSICAL CONTROLLER USING THE SENSABLE PHANTOM OMNI

William 'Pete' Moss
University of Washington
Center for Digital Arts and Experimental Media (DXARTS)

Bryan Cunitz
University of Washington
Applied Physics Laboratory

ABSTRACT

The Theremin is one of the first fully electronic musical instruments, allowing continuous control of both pitch and volume by varying hand proximity to two antennas. It allows new forms of musical expression, but is very difficult to play because it presents no tactile feedback to the performer. Haptic technology can solve this problem by providing tactile feedback. The low-cost SensAble Phantom Omni can track 6 degrees of continuous freedom and can be programmed using the feature-rich OpenHaptics C API that interfaces with OpenGL to create custom graphic and haptic applications. The authors propose using an Omni as the basis for a music controller based on the Theremin that communicates via OpenSoundControl (OSC) with the SuperCollider Audio Language to create a Haptic Musical Controller System.

1. INTRODUCTION

In order to learn more about haptic devices, and explore their use as musical controllers, the authors have taken a SensAble Phantom Omni and designed a virtual musical workspace to use for musical performance. The Omni was an easy choice, because it is the first inexpensive, almost-consumer-level device that has good force-feedback qualities and an easy to use programming API.

2. PROJECT BACKGROUND

There have been several creative attempts to provide feedback to the performer playing the theremin. Most of them use a computer at some point in the chain to process information and give additional feedback. Some important developments include:

- using a camera to track hand motion in space [3]
- using a Powerglove to track hand movement [3]
- computer simulated theremin, using mouse or touchpad (several examples, including Theremini for the Palm Computer, by Pete Moss, available at petemoss.org)

Additionally, other interfaces use haptic technology to provide feedback:

- Custom haptic interfaces to control sound (the Plank [4], vBow [2])
- Simple vibration feedback for pitch placement [3]

3. DESIGN CRITERIA

3.1. Graphic Interface

The graphic interface should be simple and tidy in appearance in order to reduce the performer's concentration upon it. The simplest design is a three-dimensional box with a cursor that represents the physical location of the Omni pointer in real-space.

3.2. Haptic Interface

The haptic rendering will of course need to be informed by the graphics, as the performer should 'feel' the walls of the performance space. But the haptic space can go beyond the graphics, by providing sensations to feel tempered pitches and the vibrations of the sound, like a real musical instrument.

3.3. Audio Synthesis

The Synthesis should be subservient to the haptic software, responding only to pitch, amplitude, and spectral commands. However, in order to reduce the complexity of the haptic loop, some transformations could be offloaded to the Synthesizer.

4. IMPLEMENTATION

4.1. Graphics

The authors have used OpenGL to render the musical space – essentially just a box inside which the performer can move freely. There is little in the way of visual clutter. Besides showing the 3D cursor, the authors have made it possible to view the pure pitches by showing lines perpendicular to the X axis. The 'shadow' of the cursor is also visible on the walls to help orient location visually.

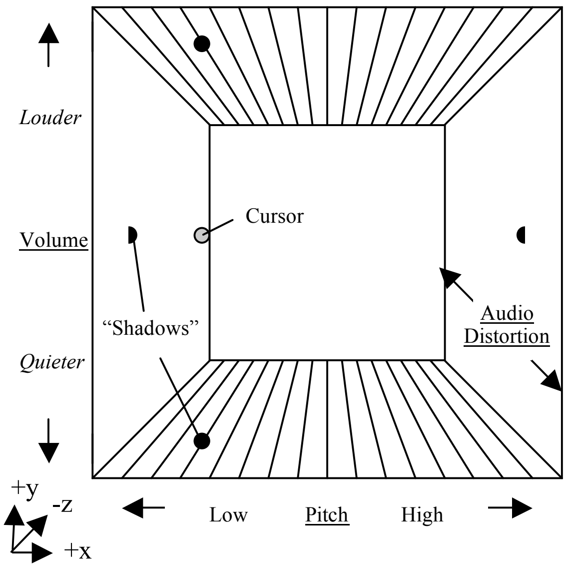

Figure 1. Representation of the Haptic & Graphic space. The X, Y, & Z axis correspond to Pitch, Volume, and Audio Distortion respectively. A cursor is visible to help orient the musician to their location in the playing space. Shadows are also included for further assistance.

4.2. Haptics

As mentioned previously, the motivation for the haptic Theremin is to provide tactile feedback to the musician so that she would be able to feel the location of notes as well as some type of vibration or stiffness analogous to that felt by someone playing a string or wind instrument.

4.2.1. Haptic Frets

The haptic fret algorithm uses the force feedback capabilities of the Omni to push the pen tip into the closest note node. The nodes are picked to be specific notes on a musical scale so that moving from one node up to the next would be like playing scales on an instrument. A simple way of achieving this is define a force in the x-axis with a magnitude determined by a sinusoidal equation. Figure 2 shows a force-position diagram of the haptic fret. When the Omni tip is on a node, there is no force on the pen tip as it does not require any positioning assistance. Once the tip moves away from the node a gentle restoring force tries to push the tip back to the node, but the force is weak enough so that the musician can easily wiggle the tip about the node for a vibrato effect. As the tip moves further away the restoring force gets larger to keep the tip close to the note. Once the tip moves far enough away, the restoring force starts to decrease and eventually flips direction to push it towards the next note. The pitches are linearly spaced across the x-axis, though this could be modified to the musician's preference by a frequency modulation of the sinusoidal equation.

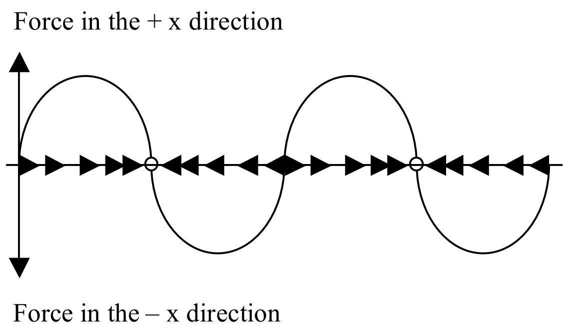

Figure 2. Force-Position diagram for the x-axis to demonstrate how Haptic frets work. The musician receives force feedback to help guide the Omni tip to ET pitches. The arrows indicate the direction of the forces.

4.2.2. Spectral Distortion Resistance

Taking advantage of a third dimension of musical manipulation that does not exist for a conventional Theremin, spectral distortion was designated to the z-axis. A spring force was applied to this axis so that as the musician pushes the tip towards the back wall (increasing the distortion), a linear resorting force pushes the tip back towards the musician. This force is consistent through the X-Y positioning of the tip.

4.2.3. Vibrations

Just as real instruments vibrate in the hands of their performers, the Haptic Theremin transmits frequency sensations to the performer by modulating the Z forces with the pitch frequency. As Chu [1] suggests, this effect is essential to the design of the instrument. However, this is proving somewhat problematic, as the Omnis three motors, while good at force-feedback, are not so well tuned for small, high frequency vibrations. We have lessened the extent of the annoyance by using a subharmonic of the fundamental frequency (currently f0/4). The effect works, but is unconvincing. For the effect to truly work, we believe that the Omni would need to be fitted with an auxiliary motor that vibrates the entire device, not just one axis as in the current incarnation.

4.3. Synth Code

SuperCollider was chosen for its ease of use, realtime performance and its ability to be completely controlled via OSC. The synthesis code merely translates the pitch number into Hz and the decibels into an amplitude scalar. The distortion parameter controls a phase distortion unit, which has the effect of increasing the amplitude of higher partials.

It is possible to track more information with the Omni, such as pitch, yaw, and roll of the stylus to control extra musical parameters, and this will be added in future versions. We are able to use the front button in the application code to hold the frequency, and the rear button to mute the sound.

5. PERFORMANCE

The performance of the device is quite responsive. It is proving quite easy to play simple melodies. With continued refinement, we hope to make a better instrument that is capable of being played competently by any intermediate musician.

To this end, we have designed the software to be as adjustable as possible. Using key commands from the computer keyboard, any user can change the force settings and musical range to be whatever they desire.

6. PLANNED ENHANCEMENTS

As mentioned above, there are some basic design criteria, such as vibration feedback, that have yet to be implemented adequately.

Aside from that, there are other parameters that the Omni device can track, the stylus pitch, yaw, and roll. The authors would like to take advantage of the extra controls to provide secondary musical parameters. Ideas include panning, pluck effects, reverb effects, as well as ways to alter the state of the workspace while playing. In fact, with a different design, one could use the Omni as a high level musical parameter controller.

The authors believe this interface could be good for learning as well as teaching. Being able to record the pointer path over time, as well as a mechanism to show what pitches to play in a song would be useful additions.

Finally, it would be interesting to transform the playing space into something more suited to human movement than a cubical box. Perhaps it would be possible to use a more irregular shaped space based in some way on the space the arm sweeps out over time.

7. REFERENCES

[1] Chu, L. "Haptic Feedback in Computer Music Performance", *Proceedings of the 1996 International Computer Music Conference*, Stanford, CA: ICMA. 1996.

[2] Nichols, C. "The vBow: A Haptic Musical Controller Human-Computer Interface", *Proceedings of the 2000 International Computer Music Conference*, Stanford, CA: ICMA. 2000.

[3] Rovan, J., Hayward, V. "Typology of Tactile Sounds and their Synthesis in Gesture-Driven Computer Music Performance." *Trends in Gestural Control of Music CDROM*, IRCAM, 2000.

[4] Verplank, B., Gurevich, M., Mathews, M. "The Plank: Designing a simple haptic controller", *Proceedings of the Conference on New Instruments for Musical Expression*, Dublin, Ireland. 2002.