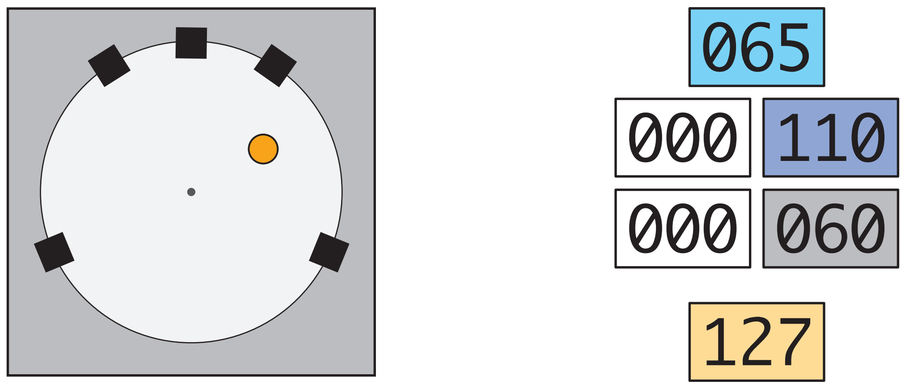
Toward an Improved Panning Interface for Spatialized Audio

Trevor Muzzy (tmuzzy@ucsd.edu)

Advisor: Peter Otto (potto@ucsd.edu)

Abstract

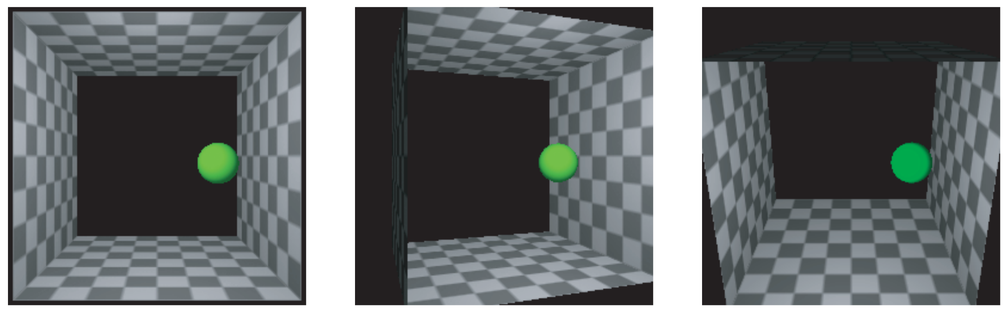
When designing audio software one must often develop data visualization schemes for complex multidimensional data. In the case of audio content creation applications for surround sound one faces the challenge of representing time-varying positional data of a sound source within an acoustic space. One possible solution involves the use of a virtual room interface with a moving icon representing a sound's location. This interface may be implemented in the Max/MSP environment for potential integration into such surround content creation applications as Vortex Designer from Immersive Media Research. This approach allows for intuitive control and monitoring of complex sound paths and allows for greater control than allowed by currently available solutions.



A typical digital audio workstation's surround panning control (left) showing the 5 speakers of a 5.1 configuration, compared to a Designer-style panning control showing a 5.1 speaker configuration (right). Both provide a general aerial view of listening space and each graphic represents a single breakpoint in an automated sound path.

Background / Motivation

IMR Vortex Designer is a surround content creation application utilizing interface and DSP technologies originally developed at UCSD. The application has matured into a very flexible tool for surround sound design. The current panning technique utilized by Designer is a power panning paradigm, in which a user specifies a relative amplitude for each speaker's signal at each breakpoint on the timeline. This allows for a sound object to move between the speakers, which must be arranged in one of the supported configurations. Sound paths are therefore dependent upon speaker configuration and a composition will not directly translate to a different setup. Designer would therefore benefit from a panning technique that allows for a more representational interface that would serve as both a data entry control and a display object for tracking a sound object's trajectory. This interface allows for the representation of height information as well. Immersive audio installations and next generation surround standards which include speakers in more than one horizontal plane call for a more dynamic panning paradigm than is currently offered in commercially-available software.



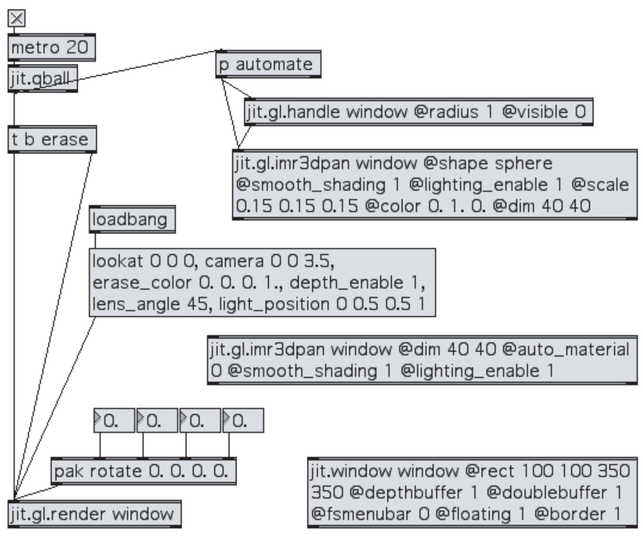
Prototype implementation of improved panner, shown in different states of rotation to illustrate different possible viewing perspectives.

Progress

A version of this interface was built utilizing OpenGL objects in Max/MSP's video processing environment, Jitter. This prototype may be easily integrated into the Designer interface in place of the current panning control. Before any such integration could take place it was necessary to perform basic benchmark testing to determine the feasibility of this version becoming part of a commercial release. Due to the real-time nature of music production applications it is necessary to minimize CPU usage wherever possible. Benchmark testing was done to determine the CPU resources required to run a single instance of the Jitter control, as well as multiple instances at once. In a production application it may be desired to have multiple instances running simultaneously in order to spatialize multiple sources in real-time.

Another Max/MSP external was built which directly implements OpenGL to build the interface. This version was written in C and was aimed at testing for computational advantages over the Jitter implementation. This version was created to be as similar as possible in order to make a valid comparison. The refresh rates of each version were regulated by a **metro** object sending out a **bang** message every 20 ms. The findings were conclusive and it was decided that the straight OpenGL implementation is more efficient by far. (The findings can be seen in the table to the right.)

This non-Jitter version will require further refinements in preparation for possible integration into an application but the current state of both versions was sufficient to perform benchmark testing.



Max/MSP patch used to build prototype interface. Custom Jitter object **jit.gl.imr3dpan** was created for this version and rendering is done by **jit.gl.render**.

Benchmark data

Test machine: Apple MacBook Pro (Core 2 Duo 2.33ghz / 2GB RAM / ATI RadeonX1600 256MB)

Test condition	CPU requirement
Max/MSP idle, no patches open	14.10 %
Jitter patch open, 20ms refresh	20.40 %
Jitter patch - 2 instances	26.65 %
Straight OpenGL version, 20ms refresh	15.20 %
Straight OpenGL - 2 instances	16.30 %

Possible Applications

There are many applications for an intuitive and flexible panning interface. Beyond Vortex Designer this control would benefit any application aimed at surround content creation. In addition, this could be a powerful tool for controlling immersive sound installations of any type. As both a control object and monitor object this interface would be ideally suited for future spatialization experiments with the Varrier display or StarCAVE at CRCA. Future projects in collaboration with CRCA's Experimental Game Lab could prove quite interesting, helping to refine this interface as a control/display for intelligent audio spatialization for game design.

Further Work

For applications requiring a single control/display object the Max/MSP + Jitter implementation is quite capable. CPU requirements make it clear that the most efficient implementation of this interface is done using pure OpenGL rather than OpenGL as offered within Jitter. An implementation in OpenGL is easily ported to a variety of systems so as to allow usability in a wide variety of contexts, whether it be in academic research or commercial applications. Combination with a spatialization technique such as ambisonics would provide a highly flexible panning tool to help streamline the surround content creation workflow.