iEAR: IMMERSIVE ENVIRONMENTAL AUDIO FOR PHOTOREALISTIC PANORAMAS

Paul Riker, Daniel Acevedo

King Abdullah University of Science and Technology,
Visualization Laboratory
Thuwal, 23955-6900, Saudi Arabia
paul.riker@kaust.edu.sa, daniel.acevedo@kaust.edu.sa

ABSTRACT

This paper presents iEAR, a flexible spatial audio rendering tool for use with photorealistic monoscopic and stereoscopic panoramas across various display systems. iEAR allows users to easily present multichannel audio scenes over variable speaker arrangements, while maintaining tight integration with the corresponding visual elements of the display media. Built in the Max/MSP Audio Programming Environment, iEAR utilizes well-established panning methods to accommodate a wide range of speaker configurations. Audio scene orientation is tied to the visual scene using an OSC connection with the visualization software, allowing users to render and spatialize multichannel environmental audio recordings in tandem with the changing perspective in the visual scene.

1. INTRODUCTION

In recent years, capture and display of stereoscopic panoramic imagery has become increasingly efficient. With the development of technologies such as the CaveCam, stereoscopic capture of spherical panoramas takes only a matter of minutes in the field [1]. These panoramas can then be rendered by large-scale stereo VR systems, such as the CAVE, NexCave (see Figure 1), and CAVE2, or any stereo-enabled display technology [2], [3], [4]. The user becomes immersed in this photorealistic rendering of the space and can utilize the experience for research, presentation, and experimentation. Similarly, capture of monoscopic panoramas has become simple, thanks in part to the ubiquity of mobile-device cameras and panoramic image software. Google Maps' street view layer delivers a photorealistic VR environment on an unprecedented scale using monoscopic panoramas.

While display workflow and associated technologies have developed considerably in this field, yielding impressive results in the visual domain, the lack of a purpose-built tool for rendering immersive audio within these frameworks has limited users' experience in terms of presence. The benefits of an audio-enabled display environment on a user's experience are well documented [5], [6], [7]. Many tools exist for the development of real-time spatial audio rendering (OpenAL, Supercollider, and Max/MSP, for example), but they are generally low-level and can only be deployed efficiently by audio development specialists. There are also a number of platforms for the simultaneous presentation of immersive VR and audio, but their implementation for the purpose of enhancing photorealistic VR with audio would be impractical for most facilities, given the complexity of systems or the specificity of their intended use [8], [9], [10]. A purpose-built tool that can quickly tie in to any existing framework is required.

Figure 1: The NexCave at the Visualization Laboratory of King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia.

2. iEAR

iEAR presents a simple, intuitive solution for the presentation of immersive audio in parallel with existing visualization software. After an initial calibration, the software can effectively render any spatial arrangement of up to 8 channels of audio to any quantity and configuration of outputs. For users without access to multichannel audio files, iEAR includes a time-delay faux upsampling method for limited surround presentation of monophonic audio files. For facilities lacking multichannel playback capabilities, there is a binaural panning mode for maximum immersion using headphones. iEAR is easily linked to visualization software via OSC/UDP messages that communicate a user's orientation within the visual space, along with the active scene (filename) and user interaction. System configuration and user settings are stored in a single JSON file for offline editing and quick recall.

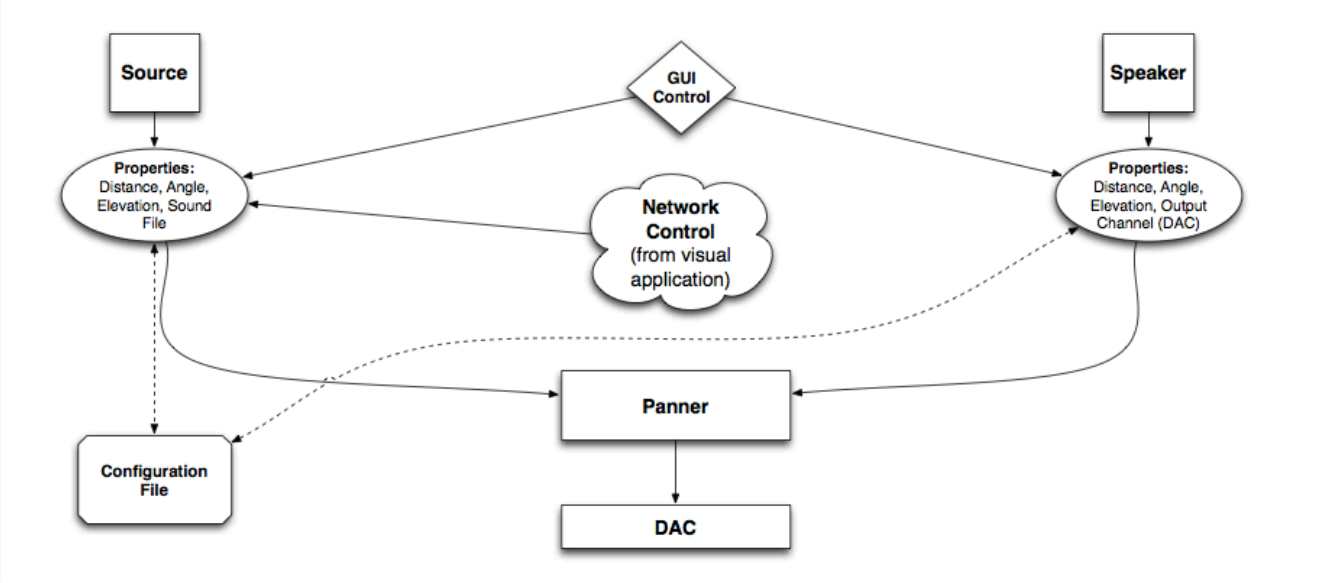

Figure 2: iEar's design defines speakers and sources as objects, whose properties inform the central panning algorithm.

2.1. System Description

Figure 2 provides an overview of iEar's structure. iEar treats inputs ("sources") and outputs ("speakers") like objects. These objects have properties, namely distance, angle, and elevation, and the user defines those properties via graphical user interface (GUI) or by editing the JSON configuration file directly. In the case of speakers, the user defines the quantity and locations of all speakers as well as the corresponding output channels of the digital-to-analog convertor (DAC) on first run, and these values (presumably) remain fixed. The user then selects the channel count for the sources to be loaded into iEar. Default orientation values then populate the GUI and the user adjusts these values to correspond with a particular recording method.

iEar's simple network interface allows the user to quickly define the UDP port for incoming network control.

The configuration file may be stored at any time, facilitating the creation of templates to suit a variety of situations. For example, storing with only speaker definitions and network port is recommended for the creation of a system template.

Once all speakers and sources have been accurately defined, the user may begin playback. Playback can be initiated manually from within the GUI, or via network command. Scene orientation is passed from the visual application to iEar via the defined network port, and the internal panning algorithm adjusts the apparent audio source locations to correspond with the changing visual scene.

iEAR's hierarchical configuration file structure is comprised of two main components: speakers and sources. "Network Port" and "Panning Method," are global settings that also appear in the configuration file. The user inputs these values via the GUI at which point they populate the current JSON configuration file, which can be saved through the GUI at any time. Every configuration file variable is dynamically assignable through the GUI and changes take effect in the software immediately. A user may also edit the configuration file directly, using an external text editor, but the changes will only be reflected in the software after a recall of the edited JSON file.

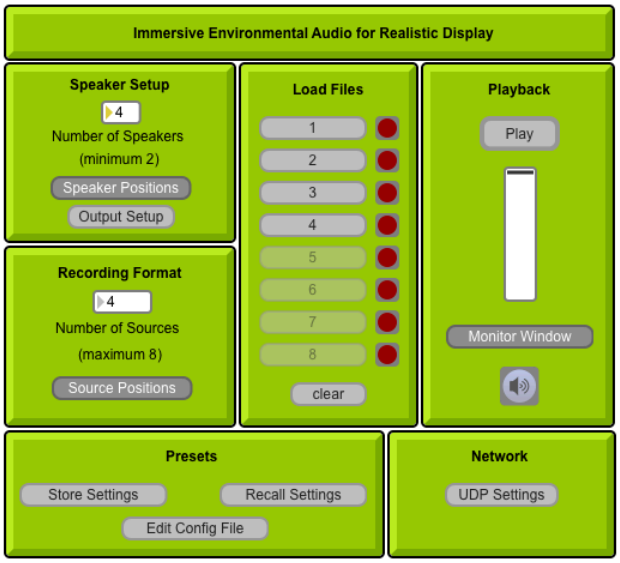

Figure 3: iEar's main window

2.2. User Interface

iEar provides a simple user interface consisting of one main window and two auxiliary windows. All other user interaction is achieved through dialog boxes. Figure 3 shows the main iEAR window. There are six panes of this window, reflecting discreet functionality:

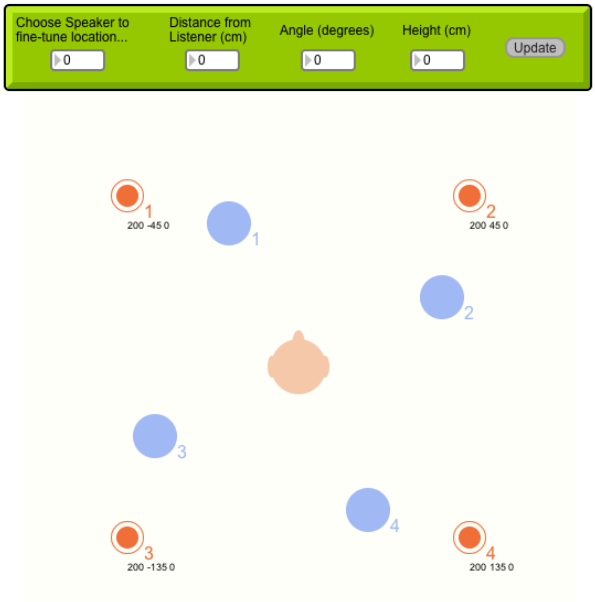

Figure 4: Speaker Positions window, showing speaker locations, source positions, and user orientation.

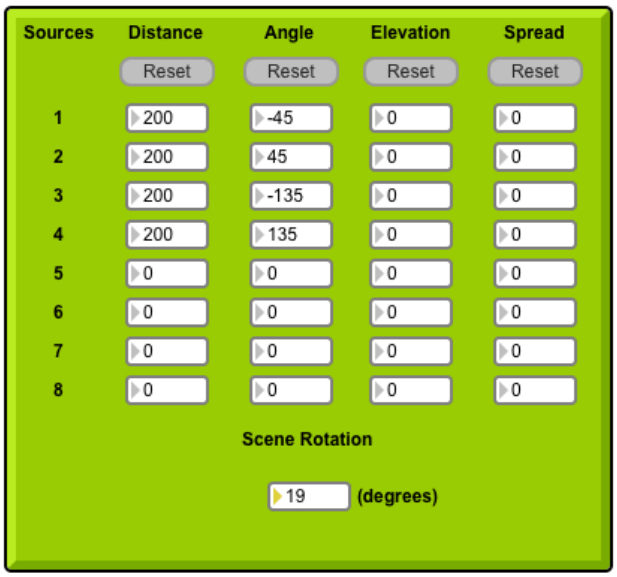

Figure 5: Source Positions window where fine adjustments can be made to the sources group. The group may also be rotated from this window.

- Speaker Setup: *Define Quantity of speakers, establish DAC mapping, and access the "Speaker Positions" window.*
- Recording Format: *Define quantity of sources and access "Source Positions" window.*
- Presets: *Store, edit, and recall configuration files.*
- Load Files: *Load audio files into iEar's buffers. Active after defining recording format.*
- Playback: *Provides manual playback functionality, master output attenuation, access to the "Monitor" window, and the global on/off button for the application.*
- Network: *Define incoming UDP port.*

Figure 4 shows the "Speaker Positions" window. This window shows the current audio "scene," consisting of speaker positions (small, solid circles surrounded by thin rings), speaker channel numbers (logical), source positions (larger, translucent circles), source numbers, and a graphic representing the user's head. The view dynamically adjusts to a user's interaction with the "Speaker Positions" control area as well as with the "Source Positions" window. It also adjusts in real-time to information received from the visualization software.

At the top of the window is the control area wherein a user can identify a speaker number and manually enter its distance from the listener in centimeters, its angle in degrees, and its height in centimeters. The positional properties for each speaker are communicated to the panning system and displayed under the speaker number for reference. For optimal panning performance, a user must complete this definition process for all speakers in the system. A recommended workflow is to perform this operation first (before adding any source files) and save the configuration as a template for the system.

Figure 5 shows the "Source Positions" window. Here a user establishes the desired distance, angle, elevation, and spread for each source[1]. Once established, the entire audio scene can be rotated in tandem with the corresponding visual scene.

3. PANNING METHODS

A primary goal of iEAR is to bring convincing, immersive audio to any VR system, regardless of speaker arrangement. The central panning method employed by iEAR is Vector Base Amplitude Panning (VBAP), which affords the desired scalability across varied speaker arrangements while offering computational efficiency [11].

The flexibility and precision facilitated by iEAR allows for a multitude of capture methods to be realized against disparate physical speaker configurations via the internal panning methods. For example, given a 5-channel recording with a center microphone at 0 degrees, a stereo pair at +/- 35 degrees, and a surround pair at +/- 100 degrees presented on a quadrophonic speaker system, an audio specialist would need to perform significant preprocessing or manual mixing to achieve a convincing recreation. Rotation of that scene would be tedious. With iEAR, each element of the multichannel source is treated both as an independent audio object and as a member of a larger audio scene. Sources can therefore be localized efficiently within a reasonable distance of the listener, and then adjusted, as a collection, along with the user's perspective on the VR space. iEAR's panners ensure that all elements of the scene are optimally rendered for the current distribution of sources, regardless of speaker count.

[1] The "spread" parameter is testing phase, see section 5 for more information.

Figure 6: KAUST Research Scientist and cyber archeologist Dr. Neil Smith explores a stereoscopic panorama of the tombs at the ancient site of Dedan (Al-Khuraybah) in al Ula, Saudi Arabia. 5-channel audio was captured on site in al Ula and is rendered with iEar.

4. APPLICATION OF MULTIMODAL PANORAMAS AT KAUST

This section describes the use of iEAR within the context of the advanced visualization systems of the King Abdullah University of Science and Technology (KAUST) Visualization Laboratory. Stereoscopic panoramas are displayed and explored quite frequently at KAUST, whether for the purpose of demonstrating the quality of the VR system for visitors, or for the exploration of archeological and historical sites. The primary system on which these panoramas are displayed is the NexCave: a hemispherical array of passive-stereo LCD panels with infrared head and hand tracking, and an atypical 5.1 speaker arrangement (see Figures 1 and 6). The displays are driven by 11 render nodes and 1 headnode, all running Scientific Linux 6, tied together via a 10Gb network infrastructure.

The software used to display the stereoscopic panoramas on KAUST's VR systems is CalVR [12]. Within this framework we have developed a custom plugin that, via OSC protocol, communicates the user's orientation within the virtual scene to our audio server, along with information about the particular scene being visualized.

Because iEar receives filename information from the visualization software, users need only load an audio-enabled panorama via CalVR and they become enveloped by a vivid, multimodal scene.

5. CONCLUSIONS AND FUTURE WORK

We have presented iEar, a purpose-built tool for the rendering of multichannel audio environments in tight correspondence with photorealistic panoramas. We described iEar's system design, citing the "object-oriented" conception of both speakers and sources. We detailed iEar's configuration and user interface, drawing attention to the ease with which a user can transparently generate a configuration file while making adjustments to the GUI. We finaly took a brief look at how iEar is employed at the KAUST Visualization Laboratory.

There are several improvements to pursue and options to explore in future development. Firstly, the central VBAP method will be expanded to include other panning algorithms, ensuring optimal performance irrespective of speaker placement. As described above, VBAP was chosen precisely because of its performance across various speaker configurations. However, employing panning methods like ambisonics and DBAP will give the user more freedom to most effectively utilize their speaker distribution.

While iEar performs best when rendering a multichannel audio recording, it also supports the creation of multichannel audio scenes from disparately recorded audio files. For example, to add sound to a landscape panorama for which no corresponding multichannel audio recording exists, a sound designer could assemble a convincing multichannel scene by using sounds from a similar location, marking prominent acoustic features accordingly (water, birds, trees, etc). The application of decorrelation for apparent source width could aid in creating more convincing aural experience when this type of scene design is attempted. Therefore, we are in the process of implementing a cost-effective audio decorrelation method, drawing on previous work here at KAUST [13].

Another planned improvement is the expansion of supported network protocols. While OSC-UDP is fairly common and easily incorporated into the current visualization software, adding support for incoming TCP, UDP, OSC-TCP, and multicast protocols would provide implementers with more options for tethering their display applications with iEar.

Finally, Google Maps' Street View application links panoramas together into a vast web of single-point-of-view locations. Applying iEar's concepts to larger collections of panoramas such as these will involve further expansion of its current functionality. Firstly, it is impractical to capture a multichannel recording for each point-of-view. Therefore, intelligent morphing of audio scenes must be explored to allow subsampling of audio data for panoramic collections to be rendered seamlessly. Secondly, relative orientation metadata for audio recordings will be required to facilitate the crossfading of audio signals in a convincing manner.

6. ACKNOWLEDGMENT

The authors would like to thank the staff of the Visualization Lab at King Abdullah University of Science and Technology with special thanks to Rob Collins and Neil Smith for their editorial assistance.

7. REFERENCES

[1] R. Ainsworth, D. Sandin, A. Prudhomme, J.P. Schulze, T. DeFanti "Acquisition of Stereo Panoramas for Display in VR Environments", *IS&T/SPIE Electronic Imaging*, The Engineering Reality of Virtual Reality 2011, San Francisco, CA, January 25, 2011, doi:10.1117/12.872521.

[2] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti, "Surround-Screen Projection-Based Virtual Reality: The Design and Implementation of the CAVE," *Computer Graphics,* 27, pp. 135-142, 1993.

[3] T. A. DeFanti, D. Acevedo, R. A. Ainsworth, M. D. Brown, S. Cutchin, G. Dawe, K. Doerr, et al., "The Future of the CAVE," *Central European Journal of Engineering 1*, no. 1, pp. 16-37, 2011.

[4] A. Febretti, A. Nishimoto, T. Thigpen, J. Talandis, L. Long, J. D. Pirtle, T. Peterka, A. Verlo, M. Brown, D. Plepys, D. Sandin, L. Renambot, A. Johnson, and J. Leigh, "CAVE2: a Hybrid Reality Environment for Immersive Simulation and Information Analysis," in *Proc. SPIE 8649, The Engineering Reality of Virtual Reality*, 2013

[5] P. Larsson, D. Västfjäll, M. Kleiner, "Ecological Acoustics and the Multi-Modal Perception of Rooms: Real and Unreal Experiences of Auditory-Visual Virtual Environments," in *Proc. of the 2001 International Conference on Auditory Display (ICAD)*, Espoo, Finland, 2001, pp. 245-249.

[6] S. Serafin, "Sound Design to Enhance Presence in Photorealistic Virtual Reality," in *Proc. of the 2004 International Conference on Auditory Display (ICAD),* Sidney, Australia, July 2004.

[7] P. Chueng and P. Marsden, "Designing auditory spaces to support the sense of place: the role of expectation," in *CSCW*, 2002.

[8] D. Poirier-Quinot, D. Touraine, and B. F.G. Katz, "Blendercave: A multimodal scene graph editor for virtual reality," in *Proc. of the International Conference on Auditory Display (ICAD)*, Łódź, Poland, July 2013.

[9] G. Wakefield and W. Smith, "Cosm: A toolkit for composing immersive audio-visual worlds of agency and autonomy," in *Proc. of the International Computer Music Conference* (ICMC), 2011.

[10] L. Valbom and A. Marcos, "Wave: Sound and music in an immersive environment," *Computers & Graphics*, vol. 29, no. 6, pp. 871-881, 2005.

[11] V. Pulkki, "Virtual Sound Source Positioning Using Vector Base Amplitude Panning," *J. Audio Eng. Soc.,* vol. 45, no. 6, pp. 456-466, June 1997.

[12] J. P. Schulze, A. Prudhomme, P. Weber, and T. A. DeFanti, "Calvr: an advanced open source virtual reality software framework," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2013, pp. 864 902–864 902.

[13] Z. Seldess, S. Yamaoka and F. Kuester, "Sonnotile: Audio annotation and sonification for large tiled audio/visual display environments," in *Proc. of the International Conference on Auditory Display (ICAD)*, Budapest, Hungary, June 2011.