Designing a new virtual reality interface for interacting with audio spatialization backends

Zétény Nagy
Illinois State University
Normal, IL, USA
znagy@ilstu.edu

Kristin Carlson
Illinois State University
Normal, IL, USA
kacarl1@ilstu.edu

Greg Corness
Illinois State University
Normal, IL, USA
gjcorne@ilstu.edu

ABSTRACT

xrOSC presents an alternative spatialization tool for composers working in 3D sound formats. We developed this tool to have an easier learning curve, and a low barrier of entry. *xrOSC* is an isotonic mixed reality controller interface with six degrees of freedom and direct absolute input designed for standalone extended reality devices. By utilizing hand tracking and gestural control, we can enable more natural and intuitive positioning of sound sources in 3D audio spatialization of the external composed space, as opposed to using spatialization tools on a traditional 2D display with mouse/keyboard input methods. This software is designed specifically as a control method, as there is no processing done on the device. *xrOSC* simply sends control messages over the network to a spatialization backend using the Open Sound Control protocol. This paper discusses the design process and methodological considerations in order to develop a tool for composers to more easily create spatial-considered musical compositions.

Author Keywords

Extended reality, standalone virtual reality, spatial audio, electroacoustic music, composition, object-based spatialization

CCS Concepts

• Human-centered computing → Human computer interaction (HCI) → Interaction paradigms → **Mixed / augmented reality**; • Applied computing → Arts and Humanities → **Sound and music computing**; • Human-centered computing → Interaction design → **Interaction design process and methods**

1. INTRODUCTION

Using extended reality systems for electroacoustic composition is becoming more accessible to artists. The rapidly lowering cost and barrier of entry of self-contained, "all-in-one" virtual reality systems can enable new opportunities for interacting with state-of-the-art spatial audio systems for electroacoustic composition purposes. This paper details *xrOSC*, an extended reality application for interacting with such systems. (We use the term *extended reality* as an umbrella term for the entire spectrum of virtual, augmented, and mixed realities; which we shorten as XR.) To understand the technological, methodological, and musical significance and potential of such a tool, this paper details the design process and the usage of the *xrOSC* software.

NIME'24, September 4–6, 2024, Utrecht, The Netherlands

2. BACKGROUND AND CONTEXT

In *Theory of Harmony*, 20th century composer Schoenberg essentially introduced the concept of timbre-focused composition with his tone-color-melody (*Klangfarbenmelodie*) theory [11] – and thus introduced a new *dimension* of compositional thinking [13]. Later, in Smalley's *Spectromorphology* framework, links within dimensions of sonic events or *sound-shapes* are explored in depth, and a new concept, *spatiomorphology* is introduced [12]; where the *spatial* qualities of sound, and the space the sounds inhabit, become an additional compositional dimension that electroacoustic composers must take into consideration, and can utilize as an added compositional layer, or even treat it as inseparable from other compositional processes and sonic attributes.

Now, the tools for creating spatial electroacoustic works are increasingly abundant and well-documented, with spatiality becoming an important facet of the electroacoustic canon [1]. However, the compositional aspect of the effective and expressive creation of the aforementioned *spatiomorphologies* is an area that is still being actively explored in new interfaces for spatial music expression [3].

3. METHODOLOGY, DESIGN, AND USAGE

3.1 Design goals

xrOSC is a new interface for spatial music expression, or NISME for short [3], with design goals that iterate and focus on some of the guidelines outlined by Bukvic et al. [3], such as intuitive/natural interaction and scalability/adaptability. The user input can be classified using Quiroz's methods [9] as isotonic, utilizing direct absolute input.

3.2 Hardware, underlying software, and frameworks

xrOSC is built for the Meta Quest 3. We chose this device for its relatively affordable pricing, ease of use, widespread adoption, mixed reality fidelity and development capabilities. *xrOSC* is developed in Unity, using the XR Interaction Toolkit. As *xrOSC* is a spatial controller designed to be used with a backend, no actual audio processing is done on the device itself. The end user is expected to supply their own, OSC-capable spatialization backend. For Open Sound Control communication over the local wireless network, *xrOSC* uses the uOSC[1] library. *xrOSC* is supplied to the end user via an .apk file that they can install on their Quest device. By default, the OSC messages that *xrOSC* transmits over the network are formatted as IRCAM Spat5 messages [4], however the end user is also encouraged to use OSC routing tools to alter the syntax of the messages if needed.

[1] https://github.com/hecomi/uOSC

3.3 Displaying sound sources in XR

The most widespread user interface design in object-based 3D spatialization is the use of 2D cartesian or polar coordinate systems. Some applications also use distorted 2D projections of 3D space to represent the sound field. While accurate and precise, these approaches take some time to get accustomed to – to immediately understand a combination of various 2D coordinates as a discrete position in 3D space. There are some user interfaces that use a 3D viewport, with navigation usually reminiscent of first-person video game controls or, in some cases, orbiting cameras. However, as these methods usually also use perspective distortion to display 3D information on a traditional two-dimensional display, the exact spatial internal and external relations between objects, speakers, and the listener could be hard to understand for inexperienced users encountering such a view for the first time.

While there are advantages and drawbacks to both approaches, the main advantage of XR technologies is that as opposed to methods based on traditional flat displays, users in the XR environment will likely immediately understand all spatial relations, thanks to the ability to display depth-accurate 3D information, and full head rotation and position tracking. The point sound sources are represented as floating numbered spheres in 3D worldspace to mirror the traditional method of representing them as circles on a 2D display.

3.4 Interacting with sound sources and user interfaces in XR

The sandbox-style approach to the aforementioned XR worldspace paradigm of hand-tracked spatialization is heavily inspired by Magnuson's solsticeVR [8]. Hand tracking and/or hand-based gestural control has already been proven effective in other NISMEs as well that share design sentiments with *xrOSC*, such as Monet [3] and Locus [10]. As we can see in projects such as Objects VR [5], XR manipulation of sound also provides a sense of agency over sounding bodies to the user; while examples like Cubing Sound [14] provide us with heuristic feedback on embodied interactions in AR spaces. The control interface in *xrOSC* thus defaults to hand tracking. Virtual representations of sound sources can be added and removed freely, and the per-source settings can be accessed by "selecting" a source, by either choosing it from a drop-down menu, or grabbing/hovering over them. These sound sources can be interacted with by beams protruding from the user's hands, "grabbing" the sources by pointing at them and pinching; this also enables reaching out and physically grabbing the sphere itself. The spheres inherit the velocity of the hands, so throwing sources around is supported and encouraged, as they can be retrieved at any time with the press of a button. The force and direction of gravity can also be changed on a global level, and gravity can also be toggled on a per-source basis, enabling some sources to float around in zero-gravity, and others to fall to the floor collider, which lines up with the real-life physical floor. Figures 1 and 2 show the user interacting with sources at a distance with passthrough enabled and with it disabled, respectively.

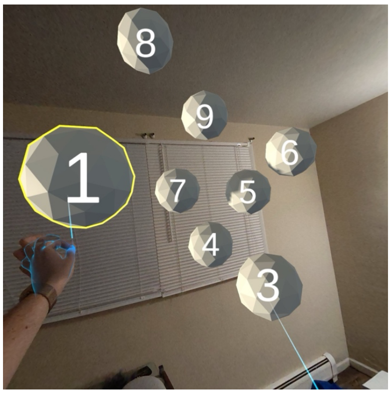

Figure 1.

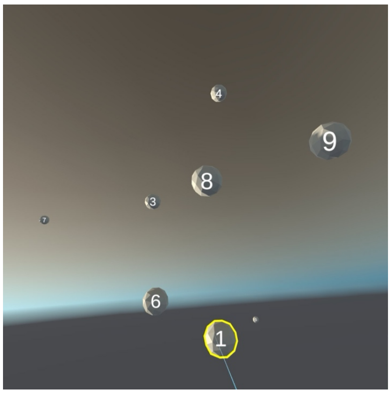

Figure 2.

Performing the Meta *Menu* system gesture toggles the visibility of the floating menu system attached to the hand. These UI elements are attached to the user's left hand and can be interacted with by poking them with an extended right index finger. The *Global Settings* menu is where the user can add or delete sources from the scene, alongside other controls, such as a passthrough toggle, altering selection behavior, floor collision toggle, and other options. In *Source Settings,* the user can change the physics parameters, such as drag, mass, or gravity, of the currently selected source, which is outlined in yellow in the XR scene. The *Recall to Hand* button located in this menu retrieves the selected source to the right hand. In *OSC Settings,* the player can specify the target IP address and port of the audio backend computer. Figure 3 displays the menu system, with the user interacting with *Source Settings,* changing the drag parameter on the selected source, which has a yellow outline.

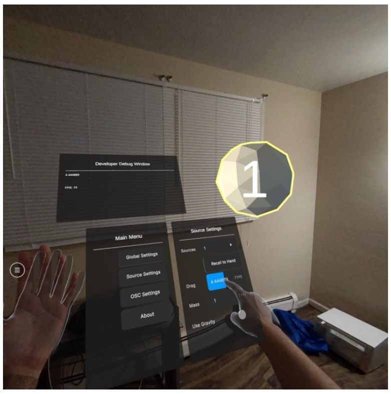

Figure 3.

3.5 Network communication with user-supplied spatialization backend

By specifying the IP address and port of a backend computer connected to the same Wi-Fi network as the Quest headset running *xrOSC,* the software will continuously send OSC messages over the network to the specified computer. This flow of information is illustrated in Figure 4. The output messages are formatted as IRCAM Spat5 messages, chosen for their clear syntax. The spheres themselves send X, Y and Z global location data. The head position of the user is also transmitted as XYZ coordinates. Furthermore, all joints of the user's fingers are also tracked, and transmit XYZ coordinate information, to be used as arbitrary gestural controls. The resulting coordinate messages can then be used for spatialization, but can also be rerouted and reformatted to be used for other perceptual parameters as well, thus enabling the creation of *spatiomorphologies* that have strong intrinsic links in both spatial, spectral and morphological qualities.

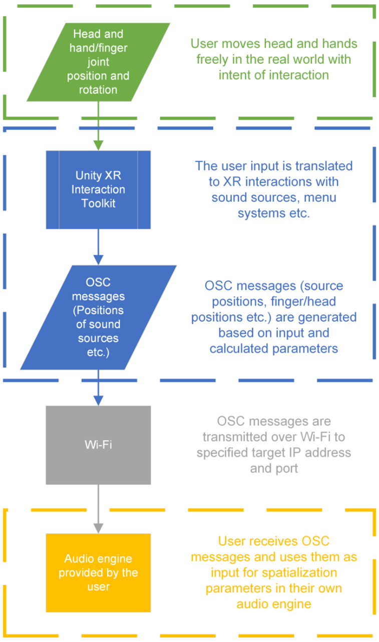
Head and hand/finger joint position and rotation
User moves head and hands freely in the real world with intent of interaction
Unity XR Interaction Toolkit
The user input is translated to XR interactions with sound sources, menu systems etc.
OSC messages (Positions of sound sources etc.)
OSC messages (source positions, finger/head positions etc.) are generated based on input and calculated parameters
Wi-Fi
OSC messages are transmitted over Wi-Fi to specified target IP address and port
Audio engine provided by the user
User receives OSC messages and uses them as input for spatialization parameters in their own audio engine

Figure 4. Block diagram of flow of information

The main use case considered while designing *xrOSC* were either using it in permanent high-density loudspeaker array [2] applications, where the user is using the software while they are inside of such an array, or using it with the user wearing headphones and listening to a head-tracked binaural sound field. Figures 5 and 6 show example workflows for these applications.

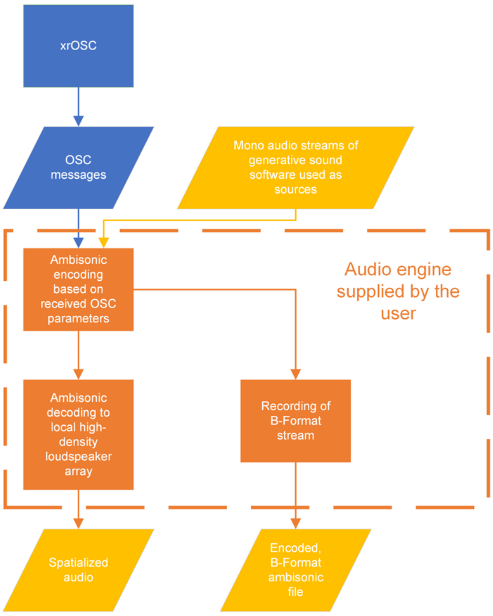
xrOSC
OSC messages
Mono audio streams of generative sound software used as sources
Ambisonic encoding based on received OSC parameters
Audio engine supplied by the user
Ambisonic decoding to local high-density loudspeaker array
Recording of B-Format stream
Spatialized audio
Encoded, B-Format ambisonic file

Figure 5. Example workflow of xrOSC. Live sound spatialization on a high-density loudspeaker array

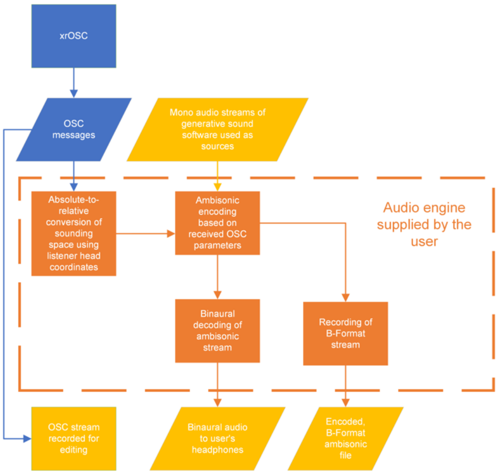
xrOSC
OSC messages
Mono audio streams of generative sound software used as sources
Absolute-to-relative conversion of sounding space using listener head coordinates
Ambisonic encoding based on received OSC parameters
Audio engine supplied by the user
Binaural decoding of ambisonic stream
Recording of B-Format stream
OSC stream recorded for editing
Binaural audio to user's headphones
Encoded, B-Format ambisonic file

Figure 6. Example workflow of xrOSC. Live sound spatialization on headphones

4. FUTURE WORK

4.1 Planned features

xrOSC in its current form is a first iteration on a concept. Other than the already listed accessibility features, possibly the largest and most difficult to implement feature we would like to add to this project is simultaneous multi-user interactions. With two or more users wearing XR headsets connected to the same backend computer and interacting with sources in the same scene, users could interact with each other, enabling more performative applications and making the experience less solitary. Another feature in the pipeline is collision with spatial meshing of real-life physical surroundings.

4.2 Accessibility

It is a crucial tenet of software development to create accessible experiences, and we want to adhere to this as closely as possible. Now that the basics of the underlying frameworks of *xrOSC* are in place, it is time to focus on implementing accessibility features. Extended reality accessibility solutions have recently been collected, outlined and synthesized by Dudley et al. [6] to form a set of up-to-date guidelines that *xrOSC* aims to implement as development goes on.

5. NEXT STEPS

We plan on continuing development on *xrOSC* for the near future, to keep it up to date with current spatial audio and XR technology standards. The next definite step in the project is to conduct focus group studies and evaluations, centering on compositional, technical, and accessibility standpoints, with a diverse and inclusive group of participants. Both quantitative and qualitative methods will be used to gather information and further development. After this, development will continue based on the data extracted from the research, and a public pre-release build will be published. The name of the project will also be changed, as it conflicts with another NIME project, *OSC-XR*. [7]

6. CONCLUSION

xrOSC is a first step on the way to an ideal, completely natural spatialization interface. The accurate positional tracking and 3D display capabilities of an XR headset aims to make visualization clearer, more easily understandable, and more concise. The hand tracking enables natural interactions with the sound sources. The hybrid system's reliance on a backend raises the barrier of entry, however, with the current state of standalone XR technology, this is necessary to not have to sacrifice audio processing quality, and guarantee the flexibility often required by spatial audio composition and production.

7. REFERENCES

[1] Natasha Barrett. 2007. Trends in electroacoustic music. In *The Cambridge Companion to Electronic Music* (1st ed.), Nick Collins and Julio d'Escrivan (eds.). Cambridge University Press, 232–255. https://doi.org/10.1017/CCOL9780521868617.015

[2] Natasha Barrett. 2016. A Musical Journey towards Permanent High-Density Loudspeaker Arrays. *Comput. Music J.* 40, 4 (December 2016), 35–46. https://doi.org/10.1162/COMJ_a_00381

[3] Ivica Ico Bukvic, Disha Sardana, and Woohun Joo. 2020. New Interfaces for Spatial Music Expression. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2020. . https://doi.org/10.5281/zenodo.4813342

[4] Thibaut Carpentier. 2018. A new implementation of Spat in Max. In *15th Sound and Music Computing Conference (SMC2018)*, July 2018. Limassol, Cyprus, 184–191. Retrieved from https://hal.science/hal-02094499

[5] Thomas Deacon, Tony Stockman, and Mathieu Barthet. 2016. User Experience in an Interactive Music Virtual Reality System: An Exploratory Study. In *Computer Music Modeling and Retrieval*, 2016. . Retrieved from https://api.semanticscholar.org/CorpusID:33911920

[6] John Dudley, Lulu Yin, Vanja Garaj, and Per Ola Kristensson. 2023. Inclusive Immersion: a review of efforts to improve accessibility in virtual reality, augmented reality and the metaverse. *Virtual Real.* 27, 4 (December 2023), 2989–3020. https://doi.org/10.1007/s10055-023-00850-8

[7] David Johnson, Daniela Damian, and George Tzanetakis. 2019. OSC-XR: A Toolkit for Extended Reality Immersive Music Interfaces. 2019. . Retrieved from https://api.semanticscholar.org/CorpusID:203646478

[8] John Moody. 2020. Virtual composer: Professor creates new musical worlds with solsticeVR. *Redbird Scholar*. Retrieved from https://news.illinoisstate.edu/2020/03/155090/

[9] Diego Quiroz. 2022. Gestural control for 3D audio. In *3D Audio*, Justin Paterson and Hyunkook Lee (eds.). Routledge, Focal Press, 64–81.

[10] Disha Sardana, Woohun Joo, Ivica Ico Bukvic, and Gregory D. Earle. 2019. Introducing Locus: a NIME for Immersive Exocentric Aural Environments. In *Proceedings of the International Conference on New Interfaces for Musical Expression*, 2019. .

[11] Arnold Schoenberg. *Theory of Harmony*. University of California.

[12] Denis Smalley. 1997. Spectromorphology: explaining sound-shapes. *Organised Sound* 2, 2 (1997), 107–126.

[13] Andrea Szigetvári. 2012. A multidimenzionális hangszíntér vizsgálata. DLA doctoral thesis. Liszt Ferenc Zeneművészeti Egyetem.

[14] Yichen Wang and Charles Martin. 2022. Cubing Sound: Designing a NIME for Head-mounted Augmented Reality. In *NIME 2022*, June 16, 2022. .