Combining Web Audio Streaming, Motion Capture, and Binaural Audio in a Telepresence System

Lawrence Fyfe, Olivier Gladin, Cédric Fleury, Michel Beaudouin-Lafon

Combining Web Audio Streaming, Motion Capture, and Binaural Audio in a Telepresence System

Lawrence Fyfe
Univ. Paris-Sud, CNRS,
Inria, Université Paris-Saclay
F-91400 Orsay, France
lawrence.fyfe@inria.fr

Olivier Gladin
Univ. Paris-Sud, CNRS,
Inria, Université Paris-Saclay
F-91400 Orsay, France
olivier.gladin@inria.fr

Cédric Fleury
Univ. Paris-Sud, CNRS,
Inria, Université Paris-Saclay
F-91400 Orsay, France
cedric.fleury@lri.fr

Michel Beaudouin-Lafon
Univ. Paris-Sud, CNRS,
Inria, Université Paris-Saclay
F-91400 Orsay, France
mbl@lri.fr

ABSTRACT

This paper describes the use of spatialized binaural 3D audio in the Digiscape telepresence system. Digiscape is a custom-built system that enables groups of users of visualization platforms, including CAVEs and wall-sized displays, to collaborate at a distance while visualizing and manipulating large and complex data sets. Digiscape supports web-based audio and video streaming as well as data sharing among its constituent platforms. Using motion capture systems, the locations of collaborators in each physical spaces are sent to the remote platforms, which spatialize the collaborators' audio streams in their local audio space. The audio spaces can be configured in a variety of ways, from a single shared audio space to mapped, adjacent, contained, split, or distorted audio spaces, facilitating the exploration of the possibilities of spatialized voice communication in telepresence systems.

Keywords

Telepresence, remote collaboration, binaural audio, motion capture, WebRTC, RTP, Opus

1. INTRODUCTION

Remote collaboration has become commonplace as networks increasingly enable people to meet and work together beyond the physical limitations of geography. However, many real-time collaborative systems, such as desktop videoconferencing or shared document editing, do not recreate many of the aspects of co-located collaboration. Hence the need to better support *telepresence*, which is defined by Buxton [2] as:

> ...the use of technology to establish a sense of shared presence or shared space among geographically separated members of a group.

As part of a larger project about high-end interactive and collaborative visualizations, called Digiscope[1], we are interested in creating a better sense of shared presence when working together at a distance. The goal of Digiscope is to facilitate collaboration among researchers working at various physical locations. The project currently connects ten platforms around Université Paris-Saclay, France. Each platform features very large, very-high-resolution displays, some of them with 3D capabilities, that can host multiple users. Supporting remote collaboration among these platforms opens up the possibility of new projects and the sharing of ideas and methods, requiring an advanced telepresence system so that multiple groups of remote collaborators can work efficiently on large, complex visualizations [5].

Each platform has microphones and high-resolution cameras that live-stream audio and video across the Digiscope network. Multiple video streams can be displayed anywhere on the platform's large displays, or on dedicated side displays. Most platforms also have a motion capture system that can track who is in the space at a given time and where they are in the space. This information is used, for example, in our CamRay system [1] to determine which video feed from an array of video cameras to send to the remote sites, and to move the display of the video feed on the remote platforms according to the location of the tracked collaborator.

To complement these video capabilities, we investigate the use of binaural 3D audio for telepresence. Binaural audio provides cues to listeners as to the locations of sounds in space. When combined with a motion tracking system, the location of each collaborator can be used to set the location of the sound in the 3D audio space. In particular, if the collaborators wear wireless microphones while moving in their motion tracking space, their speech can be spatialized in the audio spaces of remote listeners.

2. RELATED WORK

Earlier work has investigated the use of spatial audio in telepresence settings. For example, Cohen et al. [3] used binaural audio to place an artificially spatialized ringing phone sound into an office space. Hollier et al. [7] looked at the

Web Audio Conference WAC-2018, September 19–21, 2018, Berlin, Germany.

[1] http://digiscope.fr/en

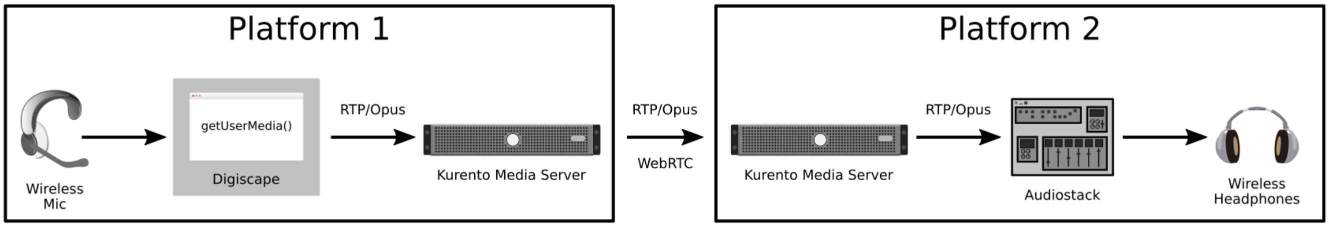

Figure 1: The flow of audio from microphones to RTP streams to headphones between two Digiscope platforms. Note that this flow can also be bidirectional.

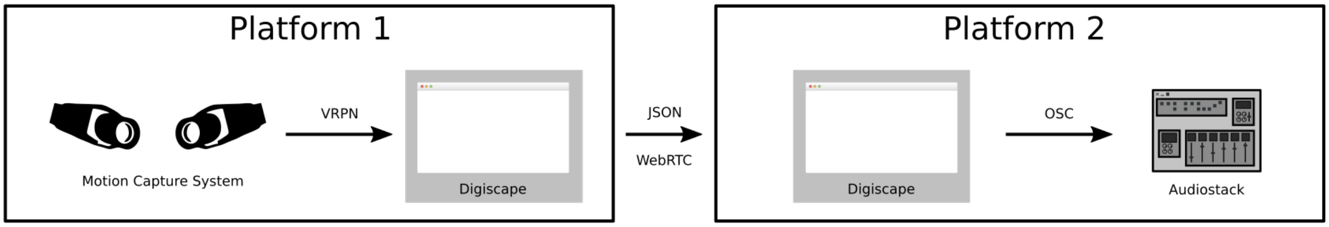

Figure 2: The flow of location data from a motion capture system to Audiostack between two Digiscope platforms. The system in Figure 2 runs simultaneously with the system in Figure 1.

use of spatial audio for telepresence in a number of virtual reality (VR) applications in which they track the locations of sound in the VR space. Keyrouz and Diepold [8] used a robot dummy head with two microphones to record and send binaural audio to a remote user. Using head tracking, the remote listener could then turn their head to change their perception of the direction of the sound source.

None of the research just described combines binaural audio with motion tracking systems. By combining motion tracking with binaural audio, video and data streaming, we have created a rich telepresence platform in which both local listeners and remote sources are tracked in local and remote audio spaces. In the next section, we describe the design and implementation of the Digiscape binaural web audio-based telepresence system.

3. TELEPRESENCE COMPONENTS

Each Digiscope platform has a different software and hardware configuration. As such, we needed a software solution that worked across the different platforms. Since web technologies make cross-platform development easier, Digiscope telepresence is built on a custom-built web-based application called Digiscape. Digiscape was created using NW.js[2] (based on Chromium and the V8 engine), Node.js[3], AngularJS[4], and a variety of other JavaScript libraries.

In order to transmit audio, video, and data around the Digiscope network, we use WebRTC[5] and WebSockets [4]. A key advantage of this approach is that no additional ports need to be opened on the different servers that comprise the network. Given the number of platforms, their different software, and firewalls between platforms, WebRTC and WebSockets make interoperability much easier.

3.1 Web Audio Streaming

Each platform in the Digiscope project is equipped with several sets of wireless microphones (AKG WMS 470s) and headphones (LD Systems MEI 100s). This setup allows collaborators to move freely inside the platform while communicating. The microphones and headphones are connected to a RME Fireface UFX II interface at each platform. Thanks to the use of web-based technologies, the Digiscape front-end runs on Linux, MacOS and Windows. It uses WebRTC media streams (via calls to `getUserMedia()`) to capture audio from the microphones.

Audio streaming uses the Opus[6] codec over RTP [10] streaming. Audio is not streamed directly from the machines connected to the microphones/headphones. Rather, it is streamed from local machines to a Kurento media server[7] installed at each platform. Digiscape handles the exchange of SDP [6] data with the Kurento media server. With Kurento, streams can be converted to different codecs though, for now, Opus is the only codec used. The Kurento media servers are useful in Digiscope because they use RTP and can distribute streams to various clients or other servers. The setup is shown in Figure 1.

3.2 Motion Capture

Most Digiscope platforms feature camera-based motion capture systems, made by either VICON or ART. Each collaborator wears or holds an object with retro-reflective markers that uniquely identify the collaborator. The motion capture system can track multiple objects/collaborators simultaneously in their physical space and send their positions and orientations using VRPN [9].

Digiscape uses a custom-built data sharing and event messaging system based on JSON. This infrastructure makes it easy for applications (such as motion capture) to communicate with one another. To work within Digiscape, the mo-

[2] https://nwjs.io/
[3] https://nodejs.org/en/
[4] https://angularjs.org/
[5] https://webrtc.org/
[6] http://opus-codec.org/
[7] https://www.kurento.org/

tion capture tracking data is converted from VRPN to JSON messages. Once the position data is converted to JSON, the location of any person in a platform's local space can be sent to any other platform. Figure 2 shows the flow of location data from the motion capture system.

3.3 Binaural Audio

For binaural audio, we use the Audiostack[8] server by Aspic Technologies. Audiostack uses an HRTF for sound source spatialization. In addition to sound spatialization, Audiostack can accept incoming RTP streams in either L16 (lossless) or Opus audio formats. For this project, one of Audiostack's most important features is the ability to set the position of sources (incoming RTP streams) in the binaural audio space via Open Sound Control (OSC) messages [11]. This feature, when combined with motion capture tracking, means that remote collaborators voices can be streamed via RTP while the remote tracking system simultaneously sends the position data.

In addition to using OSC messages to control the position of sound sources, Audiostack can also control both the position and rotation of listeners. With position control, listeners (wearing wireless headphones and motion capture markers) can move around their local space and the position of the sources they hear will change relative to the listener's position. With rotation control, listeners can "turn their head" to change their perception of the position of sources in the audio space. Audiostack therefore uses data from both the remote locations (position and audio feed from remote collaborators) and the local space (position and orientation of local collaborators) in order to properly create the binaural 3D audio space. Audiostack supports multiple listeners, with position and rotation data for each listener. Figure 3 shows web audio and tracking data coming into Audiostack and binaural audio being sent to a listener wearing wireless headphones.

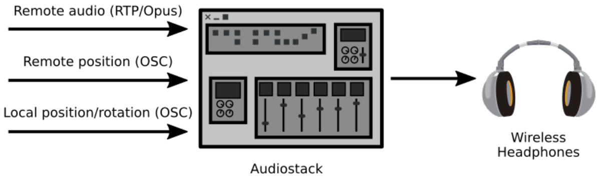

Figure 3: RTP audio stream and tracking data go into Audiostack and binaural audio comes out.

4. TELEPRESENCE AND AUDIO SPACES

Spatialized binaural 3D audio creates an *audio space* so that each listener can understand the layout of the space, therefore increasing the sense of "co-presence" in telepresence. The combination of audio streaming, motion tracking and 3D spatialization places the remote collaborators into the local audio spaces of the listeners. Audio spaces can be further defined by altering the quality and/or position of the sound based on the positions of both sources and listeners in the audio space. For example, we used an attenuation effect to cut off the source sound when it is outside of the physical area defined by the motion capture system.

The dimensions (width, height, and depth) of each tracking space, as defined by the locations of the cameras, are used as parameters for setting the size of the local audio space. When location data is sent via Digiscape, the data is normalized relatively to this space: $(0, 0, 0)$ corresponds to the center of the space, and each dimension has a $[-1, +1]$ range. We now describe various mappings that we have found useful between the remote and local audio spaces.

4.1 Mapped audio spaces

Mapped audio spaces (Figure 4) simply scale the remote space to the size of the local space. Since location data in our telepresence system is normalized, the size of each platform's space is stored and used to map the remote spaces. This default configuration is useful when the spaces share the same content on displays of different sizes.

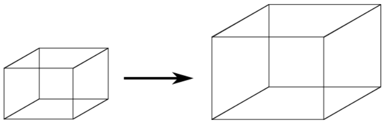

Figure 4: A smaller audio space mapped to the size of a larger one via normalization.

4.2 Adjacent audio spaces

Adjacent audio spaces (Figure 5) are spaces in which the local and the remote audio spaces are virtually placed directly adjacent to one another, creating a larger audio space. This configuration works well, e.g., for teleconference settings where the remote collaborators are "on the other side" of the wall display.

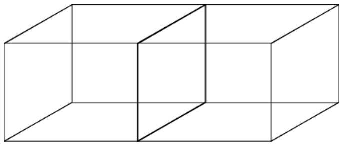

Figure 5: Horizontally adjacent audio spaces where geographically separate spaces are combined into a single audio space.

4.3 Contained audio spaces

Contained audio spaces (Figure 6) can be used when one space is of a different size from the other. This means that either a smaller remote space is contained inside of a larger local space or that a smaller local space is contained in a larger remote space. In order for this kind of audio space to work, the audio in the smaller audio space should be cut when the remote collaborator is outside of the smaller space. Otherwise, the smaller space would simply be mapped to the size of the larger space. The smaller space can be positioned anywhere inside the larger one. This setup can be used, e.g., to place a remote collaborator in a specific location of the larger local space. Communication is enabled only when the local collaborator enters the remote collaborator's space.

[8] https://www.aspictechnologies.com/audiostack/

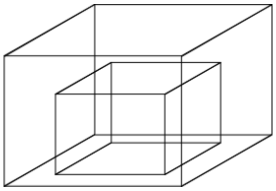

Figure 6: Contained audio spaces.

4.4 Split audio spaces

Split audio spaces (Figure 7) are such that different parts of the local space are connected to different remote platforms. This is different from adjacent audio spaces in which two spaces are combined into a single space. In a split audio space, when a local collaborator moves into another section of the local space, he or she leaves the remote space for that section and enters into another remote space. This setup lets local collaborators choose which remote collaborators they want to communicate with simply by moving around in the larger space.

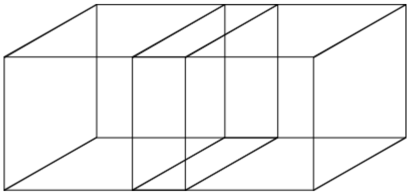

Figure 7: An audio space split into three sections.

4.5 Distorted audio spaces

All the above configurations have assumed that the position of the sound source corresponding to a remote collaborator in the local space matches the position of that collaborator in the remote space, modulo the possible translation, scaling, and clipping required by the configuration. However, this position can be further distorted, or mapped entirely differently.

For example, in our CamRay system [1], it is possible to locate the video feed of a remote collaborator in front of the local collaborator, no matter what the position of the remote collaborator is. In other words, the remote collaborator virtually follows the local one. For this virtual face-to-face to work, the location of the audio source should match the position of the display rather than that of the collaborator. Distorted audio spaces therefore open the way to a wide variety of possibilities.

5. CONCLUSION AND FUTURE WORK

This paper described a telepresence system that combines web audio streams, motion capture tracking, and spatialized 3D binaural audio. Our web-based audio streaming architecture is integrated in the Digiscape telepresence software and allows collaborators to work at a distance with an improved sense of shared presence. We introduced the concept of *audio spaces* and described a number of configurations that demonstrate the flexibility of the approach, opening up promising directions for future research.

Beyond audio spaces, there are further opportunities to use audio cues and effects to either change the quality of the streamed audio or to add new audio sources. This would place telepresence into the realm of augmented reality (AR) but with an emphasis on audio rather than images and/or video. How can collaboration be facilitated with the addition of augmented reality sounds? Can real-time effects on streaming audio be used to enhance the sense of telepresence? We intend to investigate these and other questions as we continue to develop and experiment with the Digiscape telepresence system.

6. ACKNOWLEDGMENTS

This work was partially supported by EquipEx DIGISCOPE (ANR-10-EQPX-26-01) operated by the French Agence Nationale de la Recherche as part of the program “Investissement d’Avenir” Idex Paris-Saclay (ANR-11-IDEX-0003-02). The Digiscape project is made possible by contributions from Inria, CNRS and Université Paris-Sud.

7. REFERENCES

[1] I. Avellino, C. Fleury, W. Mackay, and M. Beaudouin-Lafon. CamRay: Camera Arrays Support Remote Collaboration on Wall-Sized Displays. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI ’17, pages 6718 – 6729. ACM, May 2017.

[2] W. Buxton. Telepresence: Integrating Shared Task and Person Spaces. In *Proceedings of the Conference on Graphics Interface*, pages 123–129, 1992.

[3] M. Cohen, S. Aoki, and N. Koizumi. Augmented audio reality: telepresence/VR hybrid acoustic environments. In *Proceedings of 1993 2nd IEEE International Workshop on Robot and Human Communication*, pages 361–364, 1993.

[4] I. Fette and A. Melnikov. The WebSocket Protocol. RFC 6455, RFC Editor, December 2011.

[5] C. Fleury, I. Avellino, M. Beaudouin-Lafon, and W. E. Mackay. Telepresence systems for Large Interactive Spaces, April 2015. Workshop on Everyday Telepresence: Emerging Practices and Future Research Directions, ACM conference on Human Factors in Computing Systems (CHI 2015).

[6] M. Handley, V. Jacobson, and C. Perkins. SDP: Session Description Protocol. RFC 4566, RFC Editor, July 2006.

[7] M. P. Hollier, A. N. Rimmell, and D. Burraston. Spatial audio technology for telepresence. *BT Technology Journal*, 15(4):33–41, October 1997.

[8] F. Keyrouz and K. Diepold. Binaural Source Localization and Spatial Audio Reproduction for Telepresence Applications. *Presence: Teleoperators and Virtual Environments*, 16(5):509–522, 2007.

[9] J. Spittka, K. Vos, and J.-M. Valin. VRPN: A Device-independent, Network-transparent VR Peripheral System. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST ’01, pages 55–61, 2001.

[10] R. M. Taylor, II, T. C. Hudson, A. Seeger, H. Weber, J. Juliano, and A. T. Helser. RTP Payload Format for the Opus Speech and Audio Codec. RFC 7587, RFC Editor, June 2015.

[11] M. Wright. The Open Sound Control 1.0 Specification, 2002.