Audio Engineering Society

Convention Paper 10538

Presented at the 151st Convention
2021 October, Online

Interactive Application to Control and Rapid-prototype in a Collaborative Immersive Environment

Samuel Chabot[1] and Jonas Braasch[1]

[1]*Rensselaer Polytechnic Institute, Troy, NY, USA*

Correspondence should be addressed to Chabot (chabos2@rpi.edu)

ABSTRACT

Human-scale immersive environments offer rich, often interactive, experiences and their potential has been demonstrated across areas of research, teaching, and art. The variety of these spaces and their bespoke configurations leads to a requirement for content highly-tailored to individual environments and/or interfaces requiring complicated installations. These introduce hurdles which burden users with tedious and difficult learning curves, leaving less time for project development and rapid prototyping. This project demonstrates an interactive application to control and rapid-prototype within the Collaborative-Research Augmented Immersive Virtual Environment Laboratory, or CRAIVE-Lab. Application Programming Interfaces (APIs) render complex functions of the immersive environment, such as audio spatialization, accessible via the Internet. A front-end interface configured to communicate with these APIs gives users simple and intuitive control over these functions from their personal devices (e.g. laptops, smartphones). While bespoke systems will often require bespoke solutions, this interface allows users to create content on day one, from their own devices, without set up, content-tailoring, or training. Three examples utilizing some or all of these functions are discussed.

1 Introduction

1.1 Overview and Motivation

Since the conception of the CAVE in the early 1990s [1, 2], immersive systems have taken a large leap in providing immersive environments and virtual reality platforms for individuals, via headsets [3, 4], and groups, via room-centric systems [5, 6]. The latter provide unique opportunities for collaborative activities, because users can interact with each other without the hindrance of wearable devices. This typically comes at the expense of tracking-based distortion correction for the visual image, and the optimal trade-offs are usually determined by the application. Room-centered audio-visual systems often make compromises with regard to audio fidelity, because of reflective screen surfaces and restricted mounting points for loudspeakers. The main design goal for the CRAIVE-Lab, which is at the center of this report, was to develop a collaborative immersive system that provides high fidelity for both the audio and visual modalities—at the cost of reasonable compromises for the visual domain, for example the lack of ceiling projection. The design was made possible after the availability of affordable short-throw video projectors and a microperforated screen that is wrapped around the perimeter of the lab. A 128-channel loudspeaker array is mounted at ear height behind the screen, as will be further discussed in Section 1.2. Experience with the system has revealed a number of challenges, including a lack of:

1. a user-friendly control system, and
2. an interface which enables rapid prototyping of content for the system.

One needs to keep in mind that immersive systems cannot be easily operated through a separate control room, as is, for example, standard for sound recording studios, because operators must be present to be immersed as well. At the same time, system operation needs to work tetherlessly and/or through personal devices because larger gear would be counter productive to providing an immersive experience. Rapid prototyping is another challenge, because an immersive experience lives and dies with the available content. Especially in the context of education, which is one of the main applications for the CRAIVE-Lab, content must be created on a quick and affordable basis to keep up with the curriculum.

The potential of these spaces continues to move beyond the hypothesized and into the demonstrated. They are utilized as teaching environments, and recent studies have shown they can confer lasting knowledge on students [7]. Projects on the repurposing of existing pharmaceutical drugs for novel treatments have utilized these spaces for massive data visualizations [8]. And performances spanning geographic location colocate remote musicians and users within these spaces [9].

These successful applications encourage the continued development of and investment in these immersive environments. However, many of these bespoke spaces suffer from a requirement for highly-tailored content, which may not translate between environments. These spaces often prescribe specifications for content which creators must strictly adhere to. Or the tools with which a user is to engage with an environment require downloads from various code repositories and a myriad of installation steps, such as those of the Allolib related to the Allosphere [5]. While these may offer powerful methods for creating content, they burdens users, especially those with little-to-no technical experience, with difficult or tedious learning curves and deduct from often limited available production time. Additionally, a lack of robust methods for rapid prototyping of material will leave less availability for design iterations. These users and considerations should be accounted for.

Methods which automate the tailoring of user content to meet the space's specifications can increase time spent on development. This project first describes the most utilized functions of a specific environment and the back-end servers required to expose them to the Internet via Application Programming Interfaces (APIs). Subsequent front-end applications can be created which interface with these APIs and streamline the presentation of content in the environment. Users must be able to manipulate and control aspects of the environment directly from the interfaces they are most comfortable and familiar with, their own personal devices, and do so with a limited on-boarding process. One such application for communicating with the back-end APIs is created and demonstrated.

The main scope of the paper is to describe the developed interactive application and how it enables ergonomic control and rapid prototyping for collaborative immersive systems, such as the CRAIVE-Lab. For this purpose the remainder of the paper is structured as follows: Subsection 1.2 introduces the CRAIVE-Lab infrastructure that was used for this research. Section 2 describes the concept and implementations to enable rapid prototyping, followed by the description of applications in Section 3. The paper concludes with a discussion and conclusion section (Section 4).

1.2 The CRAIVE-Lab

The human-scale environment in which the subsequently described Internet services and applications reside is the CRAIVE-Lab. This lab, pictured in Fig. 1, consists of an approximately 10×12 m^2 floor space which can hold up to 49 simultaneous users (per fire code). About the perimeter is a 4.3 m tall, nearly-360° projection screen, rectangular in shape with flat side walls and rounded corners (r: 1.52 m). The advantage of such a shape is a more efficient usage of available floorspace and more easily read text (as opposed to a cylindrical screen). The visuals are powered by eight short-throw projectors (Canon WUX400ST) and a single PC with two NVIDIA Quadro K5200 graphics cards. The projectors are blended together into one seamless display using software from PixelWix. The screen itself is made of a microperforated PVC material so as to remain acoustically transparent to the horizontal array of loudspeakers located behind it.

This dense array contains 128 loudspeakers (JBL 308) following the perimeter of the screen at ear-level (1.65 m). Six additional loudspeakers hang from the ceiling for elevation information. The array is accessible to networked devices via two Ferrofish Verto 64

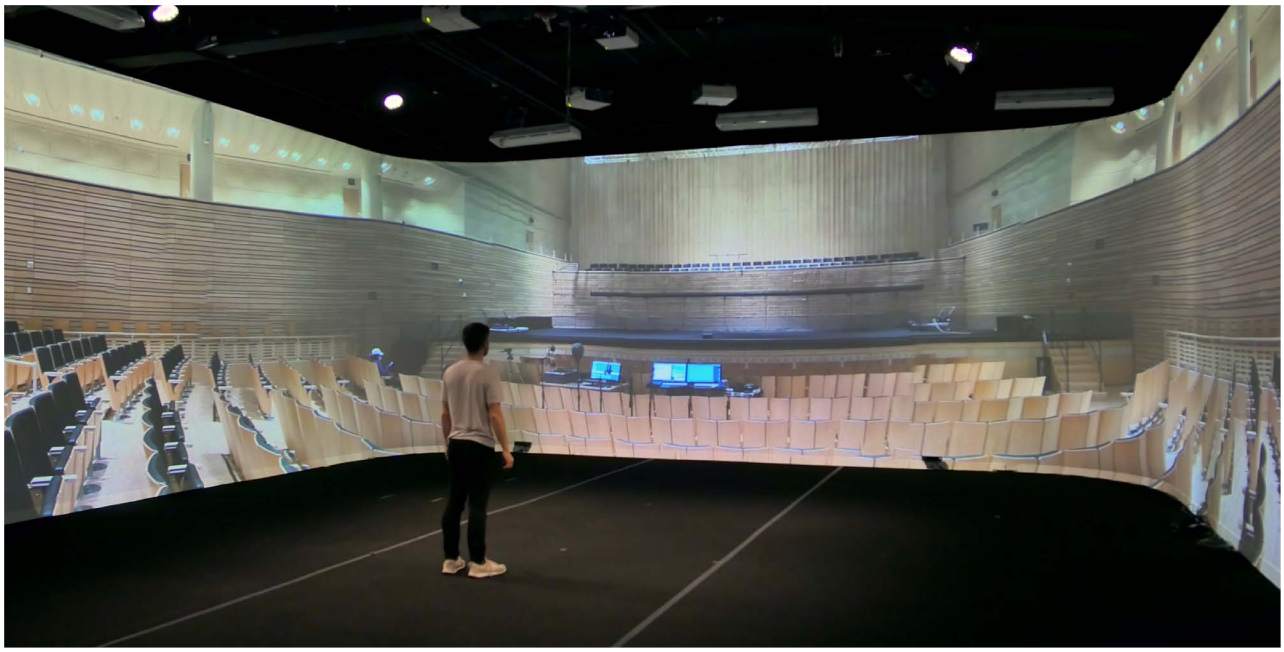

Fig. 1: A student stands at the center of the CRAIVE-Lab, a human-scale immersive environment at Rensselaer Polytechnic Institute,which hosts researchers and artists for data and reality explorations. The space facilitates ongoing collaborations amongst a variety of disciplines, including architecture, language learning, artificial intelligence, and acoustics.

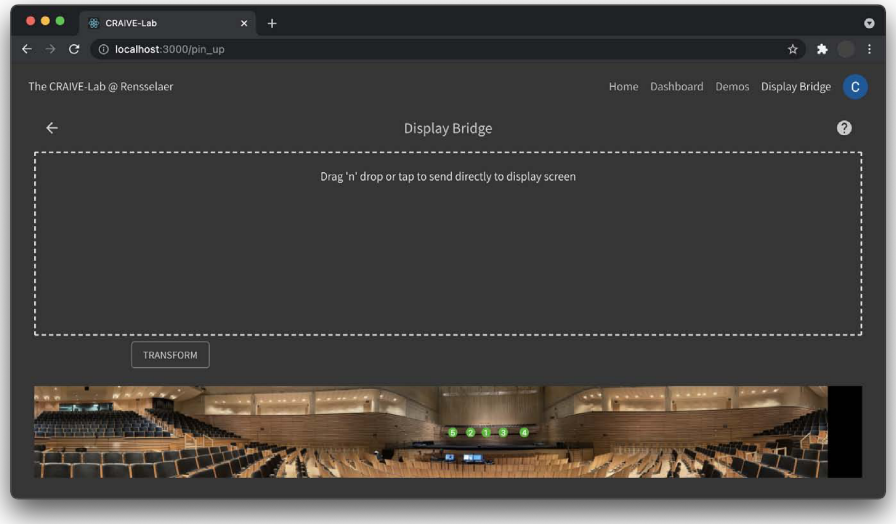

Fig. 2: A frame of the front-end user interface. This interface engages with the back-end Node servers which expose functions of the environment to the Internet. Uploaded imagery is automatically formatted, cropped, and displayed on the panoramic screen. Uploaded audio files are transmitted to the spatialization server housed in Max. Icons for each corresponding sound source are overlaid on the image. The user can drag these to the desired location for congruent audio-visual presentation.

interfaces for Dante protocol communication and capable of multiple spatial audio techniques, including higher-order ambisonic (HOA) and wave field synthesis (WFS) presentations [10, 11]. A significant advantage of the latter within collaborative environments such as the CRAIVE-Labis the production of inhomogeneous soundfields, and therefore the elimination of a single so-called "sweet spot."

Additionally, the space is outfitted with various user spatial tracking devices: a 16-channel spherical microphone array, six time-of-flight sensors, and multiple Kinect gesture sensors [12, 13]. Finally, six professional LED light fixtures (ETC D60) hung from the ceiling illuminate the workspace without casting glare and shadow on the projection screen.

The lab space is regularly utilized by classes of the School of Architecture for immersive presentations of ongoing works and renderings [14]. These students are often limited to a matter of weeks or days to complete their projects, and entire studios may be expected to present over the course of a single day. Thus, reductions in the time required to become familiar and engage with the system are valuable to the projects at hand.

2 ENABLING RAPID-PROTOTYPING

This project's web-based approach to engage and control the immersive environment focuses on rapid prototyping and content creation through a workflow incorporating users' personal devices. This method requires neither training or on-boarding, nor highly-tailored content. It consists of three main units: (1) the *Spatial Audio Worker*, (2) the *Display Bridge*, and (3) the *Front-end User Interface*—see Fig. 4. The *Spatial Audio Worker* receives transmissions of audio content for output along the loudspeaker array and position information for source placement. The role of the *Display Bridge* is to remove image distortion by automatically applying the necessary perspectival transformation and formatting before passing it to the panoramic screen for display. The user then accesses these functions and controls the system using the *Front-end User Interface*. All three subsystems are explained in the next sections.

2.1 Spatial Audio Worker

The loudspeakers in the array can be addressed in multiple ways. One such method for producing dynamic spatialized audio, named the Spatial Audio Worker, utilizes the Cycling74 Max 8 software and IRCAM Spat5 object library, which includes objects for multiple spatialization techniques [15]. A configuration is created by specifying the loudspeaker positions in a virtual space and the desired spatialization technique (e.g. HOA). Sound sources can then be positioned relative to this virtual array. The contribution required of each virtual loudspeaker to simulate the soundfield of the sources is calculated by the spatialization object. This is output for each loudspeaker to its analog in real space to produce the simulated spatialized soundfield. A visualization of the array for the CRAIVE-Lab and three sounding sources can be seen in the *Spat5.viewer* object shown in Fig. 4.

Access via the Internet: The spatialization object accepts playback of audio within Max as sound source input. In order to expose this functionality to the Internet, the Node4Max ability to embed a Node.js server within a patch is utilized. This server is configured with an API which accepts as input audio files and position information. This server can receive both standard RESTful and real-time WebSocket transmissions from applications and interfaces that adhere to its simple protocol: audio files are transmitted via a POST method or socket connection as audio buffers and position information is updated with a JavaScript Object Notation (JSON) object.

Upon receipt at the server, an audio buffer is temporarily written as a local file. Max is pointed to this new file using a message with the local file path output by the Node4Max object. The local file is loaded into a Max *buffer~* object for playback into the spatializer object.

The Worker retains a list of currently available sound source inputs to the Spat5 spatializer and dynamically assigns incoming audio to the next available. The server responds to that application or interface which it received the audio buffer from with this sound source number. Any subsequent updates from the application for this particular sound source must include the source number.

Upon the completion of a file's playback, a message notifies the Worker, which frees the occupied source and adds it back into the list of available sources, removes the temporary file, and notifies the client application that the source has been destroyed.

Position information must include the source number which is to be updated and the new position, in degrees,

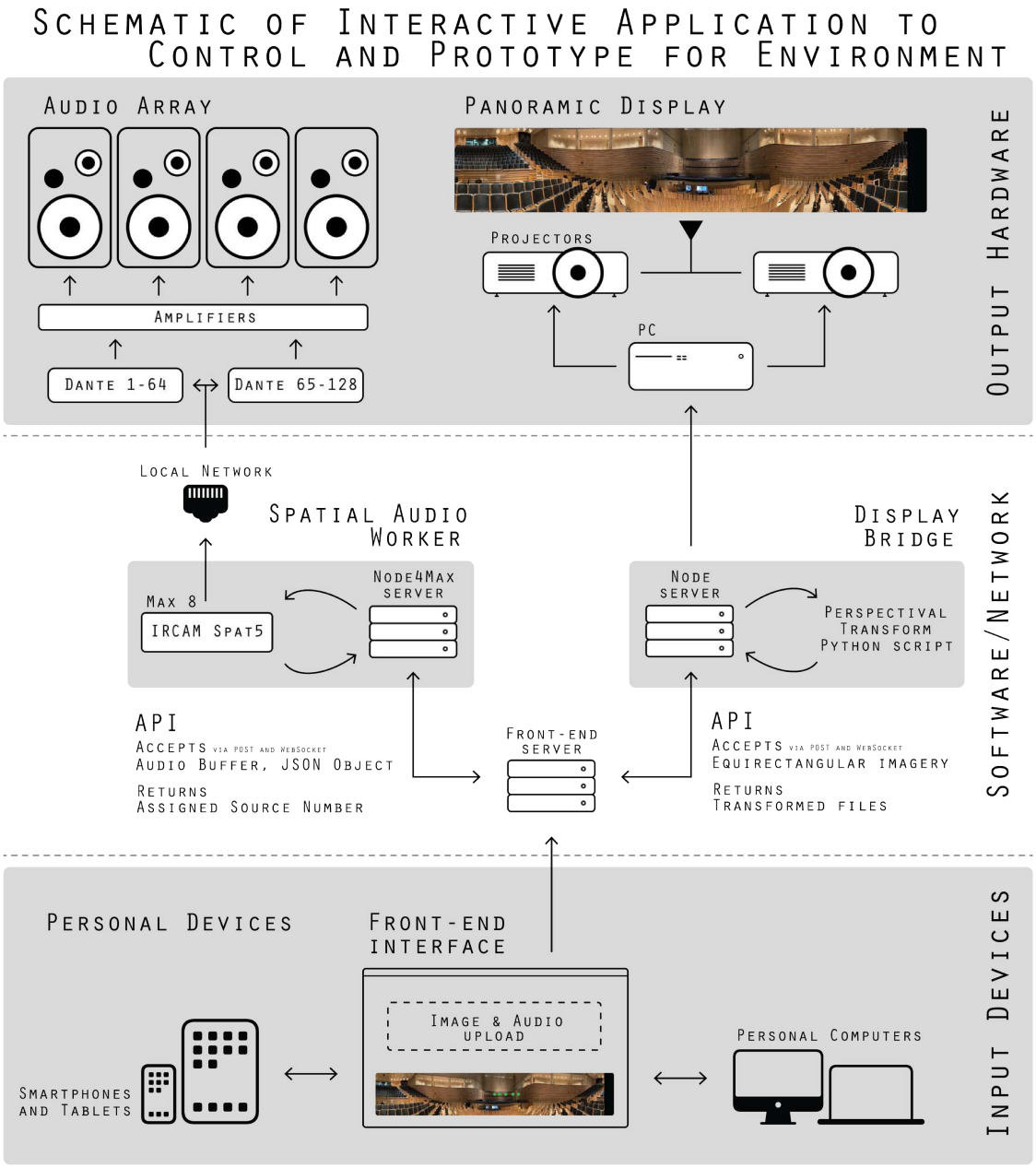

Fig. 3: Schematic diagram of the interactive application for rapid prototyping and environment control depicting hardware outputs, device inputs, and the software and network layer between. The loudspeaker array and panoramic display are accessible via Node servers and their respective API. A front-end interface configured to communicate with these servers gives users simple and intuitive control over complex functions of the environment (e.g. audio spatialization) from their personal devices.

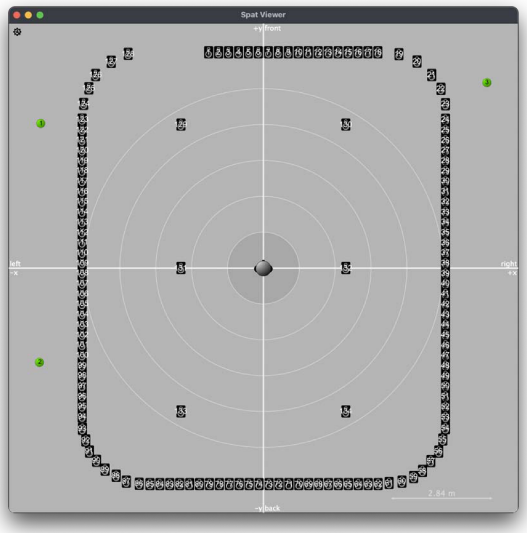

Fig. 4: This *spat5.viewer* Max object shows the loudspeaker array for the CRAIVE-Lab (shown in black) and three sounding sources (shown in green). The configuration takes in the loudspeaker and sound source positions in virtual space, and can be tailored to specific arrays.

it is to be moved to. The server then outputs this information via formatted message to the spatializer for update. Additional controls exposed by the Worker API include level gains and playback controls. These are also addressed along with the target source number in a JSON object.

2.2 Display Bridge

As a single continuous PC desktop, displaying panoramic imagery on the CRAIVE-Lab screen only requires an image or video which adheres to the display dimensions (15360-by-1200 px) and ultra-wide aspect ratio (12.8:1). Because of the blended overlap between projectors, the system actually renders the horizontal resolution of 15360 pixels over the space of only 11636 pixels. As a result, imagery must be appropriately stretched in order to counter the introduced squeezing effect. Additionally, due to the nonuniform nature of the rectangular screen (that is, not a spherical projection surface) a perspectival transform is necessary to counter introduced distortion.

An important consequence of removing this distortion, beyond visual accuracy, is the maintained congruence of onscreen visuals and spatialized audio objects. At the most affected points, deviations between the original distorted and transformed projections reach over 200 pixels [16]. This translates to over half a meter of visual deviation on the screen, more than enough to disrupt the congruence of audio-visual presentations.

Previously, this transformation, as well as the cropping and formatting for display, was performed by hand in Adobe Photoshop. In order to streamline the transformation and formatting process, a python script has been written which accepts as input an equirectangular image or video and exactly transforms, crops, and outputs the file formatted for the CRAIVE-Lab display. This transformation uses matrices defining the pixel coordinates of a spherical projection and the CRAIVE-Lab projection to interpolate the output from the input image. This process can be applied to other screens of irregular geometry by adjusting the output matrix.

Access via the Internet: In order to render the transformation and the display accessible via the Internet, a Node server is created and an API is configured. This server listens for incoming uploads of image and video files via a RESTful POST method. When a file is received, the Node server spawns an instance of the python transformation and injects the received file as input. Upon completion of the transformation a folder containing the output images and videos is transmitted back to the original client.

It can be included in the request header that the server also forward the output file to the panoramic screen for display. To receive this, a simple web page is configured. When it is opened, a WebSocket connection with the Node server is established. Incoming files from the server fill respective image or video HTML tags. These tags are set to fill the entire web page viewport. When opened in fullscreen on the display, this has the effect of seamlessly filling the screen with the incoming imagery.

2.3 Front-end User Interface

The respective Node servers of the audio array and display expose the audio spatialization, perspectival transform, and projection display to the Internet. Due to their APIs, an application or interface can be written to communicate with these services. A front-end user interface is designed to provide holistic access to these hooks. Through a React.js single page browser application, users are able to engage with this application

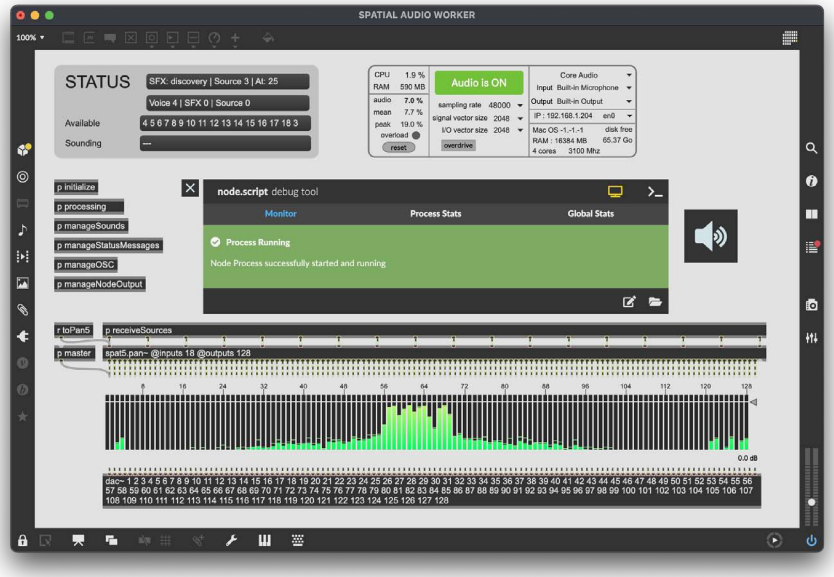

Fig. 5: The Spatial Audio Worker receives incoming audio files and their respective position information to dynamically spatialize user content. The Worker incorporates a Node.js server with an API which allows it to receive data from various applications and interfaces. One such demonstrated interface allows users to transmit audio files and manipulate their position from a browser on a personal device.

Fig. 6: An example of an equirectangular image: the concert hall of the Experimental Media and Performing Arts Center at Rensselaer. The equirectangular projection will be automatically formatted and cropped by the Display Bridge for presentation on the projection screen.

using the ubiquitous devices they are most comfortable and familiar with: their own personal devices.

A frame of the interface is shown in Fig. 2. Uploading an image through the service will post it to the display server, process the perspectival transformation and formatting, and both display it on the panoramic screen and return a copy to the user. The user can also use the interface to upload audio files, which are transmitted to the Node server of the Max patch. There, Max assigns the file an available sound source. A confirmation message is transmitted back to the interface with the assigned audio source number. The interface will generate an icon overlaid on the panoramic image indicating the current audio source position along the array with respect to the onscreen visuals. Here, the user can click and drag the icon to the desired location to align a congruent audio-visual presentation. Doing so relays the updated position for the source back to the Max patch, which adjusts accordingly.

Additionally, environment-level controls for devices such as the projectors and LED lighting system can be found in this interface. Users therefore have access to

many of the laboratory's components through a unified interface.

3 APPLICATIONS

The following examples will outline two research applications which utilize the user interface and an artistic venture which achieves finer control by communicating with the APIs more directly. The process and the detailed signal flow are also depicted in the flow diagram of Fig. 3.

3.1 Concert Hall Auralization

An auralization of musicians performing on the stage of the Experimental Media and Performing Arts Center at Rensselaer constitutes the first environment. Figure 6 shows an equirectangular projection of the concert hall. Using the interface, this image is uploaded through the Display Bridge. There, the appropriate horizontal cross-section to be displayed is extracted and processed through the perspectival transform. This formatted image is then transmitted directly to the screen for display. The result of this is shown on the CRAIVE-Lab screen in Fig. 1.

For audio input sources, the stem recordings of each section of an opera excerpt from Mozart's *Don Giovanni* [17] are each uploaded to the Spatial Audio Worker. These automatically become movable sound objects: their relative positions along the audio array are shown overlaid on the display image and can be moved to align with their congruent visual counterparts. In this example, each recording of a musician or section is shown positioned about the stage.

3.2 Architectural Perception Research

One specific example of this workflow utilized for architectural research is a study of architectural design preferences. Rebecca Elder's research seeks to determine whether discrepancies exist in design preferences for architectural renderings when experienced using traditional poster board printouts or immersive settings [18]. Various renderings of architectural projects are embedded into panoramic images taken of New York City streetscapes. These are paired with multi-channel, in-situ soundscapes to provide a congruent aural and visual presentation.

The usage of the Display Bridge and Spatial Audio Worker allow for the immediate presentation of the produced and recorded content without the need for its tailoring to the environment. Her experimentation demonstrates that users' preferences are affected by the presentation style of the architectural renderings: users specifically cite the ability to experience a design amidst its contextual visual and aural surroundings as having altered their preferences. The streamlined interface described in this report makes possible the usage of the lab for regular presentation of architectural renderings.

3.3 Finer Control for Moving Sources

Those users desiring other specific behaviors and with familiarity engaging an API can create their own interfaces or servers to communicate with the back-end services directly. One such piece created for the space, *Biorhythm* by the artistic duo GREYMAR, utilizes biometric data gathered from one artist over a period weeks to inform a visual and aural experience [19]. For example, heartbeat data alters the periodicity of sound sources moving about the environment. Such functionality is achieved using a fresh Node server dedicated to the project. It continuously calculates, dependent on the biometric data, the desired position of each source and forwards this information to the Spatial Audio Worker for realization via an established WebSocket connection.

4 DISCUSSION AND CONCLUSIONS

This work demonstrates how a rich and complex environment can be utilized "out of the box" by large numbers of users. The Spatial Audio Worker, Display Bridge, and front-end interface abstract from users much of the complexity of engaging with a technological facility like our lab space. This allows them to prototype and present content much more rapidly than if they were required to learn these individual components and tailor their content to the environment. Furthermore, an Internet-connected audio and video display is capable of remote presentation by researchers and artists alike.

In an ongoing effort, data streams from the multiple sensor platforms will be combined in future work for contextual information about the users occupying the space. This tracking data will be made accessible to the end user via an API. This spatial data can then be utilized to manipulate the content and the environment,

and possibly in turn the user, creating a feedback loop of influence in the space.

While the APIs offer specific functionality that some users will find helpful, others may seek more powerful or bespoke presentation and processing abilities. Those users are not the intended audience of these services as they are likely more advanced and familiar with many of the environment's complexities or have more to time learn what they do not.

There is value to be had in constructing these immersive facilities. However, the variety of end users should be considered when designing services and interfaces for their operation. The variety of end users calls for a variety of user interfaces and services as well.

5 ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant IIS-1909229 and by the Cognitive and Immersive Systems Laboratory (CISL).

References

[1] Cruz-Neira, C., Sandin, D. J., DeFanti, T. A., Kenyon, R. V., and Hart, J. C., "The CAVE: Audio visual experience automatic virtual environment," *Communications of the ACM*, 35(6), pp. 64–73, 1992.

[2] Cruz-Neira, C., Sandin, D. J., and DeFanti, T. A., "Surround-screen projection-based virtual reality: The design and implementation of the CAVE," in *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '93, pp. 135–142, ACM, New York, NY, USA, 1993, ISBN 0-89791-601-8, doi:doi.acm.org/10.1145/166117.166134.

[3] Garon, M., Boulet, P.-O., Doiron, J.-P., Beaulieu, L., and Lalonde, J.-F., "Real-time high resolution 3D data on the HoloLens," in *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pp. 189–191, IEEE, 2016.

[4] Chessa, M., Maiello, G., Borsari, A., and Bex, P. J., "The Perceptual Quality of the Oculus Rift for Immersive Virtual Reality," *Human–Computer Interaction*, 34(1), pp. 51–82, 2019, doi:10.1080/07370024.2016.1243478.

[5] Höllerer, T., Kuchera-Morin, J., and Amatriain, X., "The Allosphere: A Large-Scale Immersive Surround-View Instrument," in *Proceedings of the 2007 Workshop on Emerging Displays Technologies*, San Diega, CA, 2007, doi:doi.org/10.1145/1278240.1278243.

[6] Papadopoulos, C., Petkov, K., Kaufman, A. E., and Mueller, K., "The Reality Deck — an Immersive Gigapixel Display," *IEEE Computer Graphics and Applications*, 35(1), pp. 33–45, 2015.

[7] Divekar, R. R., Drozdal, J., Chabot, S., Zhou, Y., Su, H., Chen, Y., Zhu, H., Hendler, J. A., and Braasch, J., "Foreign language acquisition via artificial intelligence and extended reality: design and evaluation," *Computer Assisted Language Learning*, pp. 1–29, 2021, doi:doi.org/10.1080/09588221.2021.1879162.

[8] McGuinness, D., McCusker, J., Yan, R., Solanki, K., Erickson, J., Chang, C., Dumontier, M., and Dordick, J., "A Nanopublication Framework for Biological Networks using Cytoscape.js. In Proceedings of International Conference on Biomedical Ontologies," in *Proceedings of International Conference on Biomedical Ontologies (ICBO 2014)*, Houston, TX, 2014.

[9] Chabot, S., Mathews, J., Su, H., and Braasch, J., "Co-locating remote collaborators in immersive virtual environments using telematic systems (A)," *The Journal of the Acoustical Society of America*, 148(4), pp. 2771–2771, 2020, doi:10.1121/1.5147712.

[10] Malham, D., "Higher Order Ambisonic Systems for the Spatialisation of Sound," in *Proceedings of the International Computer Music Conference*, volume 1999, pp. 484–487, Beijing, China, 1999, ISSN 2223-3881.

[11] Verheijen, E. N. G., *Sound Reproduction by Wave Field Synthesis*, Ph.D. thesis, Delft University of Technology, 1997.

[12] Sharma, G., Braasch, J., and Radke, R. J., "Interactions in a Human-Scale Immersive Environment: the CRAIVE-Lab," *Cross-Surface 2016, in conjunction with the ACM International Conference on Interactive Surfaces and Spaces*, 2016.

[13] Mathews, J. and Braasch, J., "Real-Time Source-Tracking Spherical Microphone Arrays for Immersive Environments," *Journal of the Audio Engineering Society*, 2018.

[14] Leitão, C., "IN/OUT/INTO/INFRA," in *Proceedings of the IEEE Games Entertainment & Media Conference*, IEEE, 2019.

[15] IRCAM, "Spatialisateur 5," v5.1.3.

[16] Carter, J. P., *Immersion: A Framework for Architectural Research*, Ph.D. thesis, Rensselaer Polytechnic Institute, Troy, NY, 2019.

[17] Lokki, T., Pätynen, J., and Pulkki, V., "Recording of anechoic symphony music," *The Journal of the Acoustical Society of America*, 123, p. 3936, 2008, doi:doi.org/10.1121/1.2936008.

[18] Elder, R. L., *Influence of Immersive Human Scale Architectural Representation on Design Judgment*, Master's thesis, Rensselaer Polytechnic Institute, Troy, NY, 2017.

[19] Grey, I., "IGREY," 2019, (accessed Mar. 27, 2021).