Remote Orchestral Conduction via a Virtual Reality System

Remote Orchestral Conduction via a Virtual Reality System

Leonardo Severi, Matteo Sacchetto,
Andrea Bianco, Cristina Rottondi
Department of Electronics and Telecommunications
Politecnico di Torino, Turin, Italy
{name.surname}@polito.it

Gabriele Abbate, Antonio Paolillo, Alessandro Giusti
IDSIA USI-SUPSI
Lugano, Switzerland
{name.surname}@idsia.ch

***Abstract*—This paper envisions the adoption of a Virtual Reality (VR)-based approach to provide visual feedback to remote musicians without the acquisition and transmission of a video stream, in a Networked Music Performance scenario. Focusing on a remote orchestral conduction setup, the VR headset tracks the conductor's gestures to convey them to remotely connected performers, where the conductor's hands and head pose are displayed by means of an avatar. Quantitative results suggest that the Motion-to-Photon latency introduced by the system is tolerable for NMP applications, while the proposed system achieves a substantial reduction of bit-rate requirements in comparison to traditional video streaming.**

***Index Terms*—Virtual Reality, Orchestra, Networked Music Performance, Low Latency Streaming.**

I. Introduction

A Networked Music Performance (NMP) is a real-time musical interaction among geographically distant musicians, enabled by low-latency audio/video streaming over a telecommunication network [1]. Several research efforts have focused on conveying high-fidelity audio while minimizing the "Mouth-to-Ear" (M2E) latency, i.e., the time elapsed from the moment the audio is captured by an acquisition device to when a playback device reproduces it. However, an in-presence musical interaction is not solely based on audio cues, but also relies heavily on visual cues, which musicians use to synchronize their performances and convey expressive intentions [2].

The most common form of visual feedback that a NMP system could exploit is a video stream, as typically done in videoconferencing platforms. Unfortunately, including video streaming in a NMP system increases the network bandwidth required, thus introducing scalability limitations. To reduce bandwidth requirements, the video needs to be encoded before streaming, which introduces additional latency. Since NMP systems have strict M2E latency requirements (below 30 ms [3]) to ensure a synchronous musical interplay, audio and video streams are generally transmitted and reproduced out-of-sync. Attempts to eliminate the encoding/decoding latency by streaming uncompressed video [4] resulted in a minimum network bandwidth requirement of 1 Gb/s (and even higher if the HD video configuration is used[1]).

[1]https://lola.conts.it/downloads/Lola_Manual_2.0.0_rev_001.pdf

In this study, we explore an alternative to video transmission in the context of a remote orchestra conducting scenario: we exploit a Virtual Reality (VR) headset to extract and convey the conductor's gestures to a geographically spread orchestra interconnected via a NMP system. Our proposed approach has the twofold goal of reducing i) the required transmission bit-rate, since we need to transmit only data related to the pose of hands and head of the conductor, and ii) the Motion-to-Photon (M2P) latency, i.e., the time taken for a user gesture to result in a corresponding visual change on a display [5], as pose tracking can operate at a higher frame rate than what is commonly supported by cameras.

Numerical results show that the average M2P latency achieved by our experimental setup is on average 140 ms and never exceeds 210 ms, which constitutes a promising starting point for future potential integration in a Musical Metaverse (MM) framework [6]. Indeed, the MM concept foresees the creation of a digital universe where users can interact musically via avatars, blending the physical and digital worlds by means of Musical Extended Reality (Musical XR) and Internet of Musical Things technologies [7]. In particular, we envision the adoption of our proposed system as a building block for MM-mediated orchestra rehearsals and performances, as already occurred in [8].

The remainder of the paper is organized as follows: Section II reviews related studies, Section III provides an overview on the system with technical details on its implementation, Section IV describes the methodology used to evaluate the performance of the system, Section V presents the obtained results, Section VI outlines the identified limitations and Sec. VII concludes the paper.

II. Related Work

Several studies tackled the assessment of the maximum acceptable asynchrony between audio and video in real-time streaming. In [9], the authors show that the perception of misalignment is deeply linked to the type of content considered. As a general outcome, the study highlights that when the lag between visual and audio streams is greater than 100 ms, more than a half of the subjects involved in the test campaign therein reported were able to perceive it, thus suggesting that lag should not exceed 100 ms.

In [10], the authors explored how different lags between audio and video impacted a musical performance involving remote musicians and a conductor. One interesting outcome of that study is that visualization of the conductor's gestural cues was considered extremely important by the musicians, as it helped mitigate the negative impact of audio latency on the performance quality. Furthermore, high audio latency was found to be much more impactful on the musicians' capability of maintaining a stable tempo w.r.t. video latency, thus suggesting that higher video delays may be tolerated, provided that they not exceed 100-200 ms.

A preliminary attempt to bypass video streaming in NMP practices can be found in [11], where a conventional computer mouse was proposed as a replacement for the conductor's baton in a NMP scenario, with the aim of providing low bitrate and low latency visual feedback while saving network resources for audio streaming. Results show that the proposed mechanism, though not user-friendly, proved viable for conducted music, with conductor and orchestra tolerating one-way transmission delays in the range from 35 to 75 ms.

More recently, several studies investigated real-time streaming of control and/or gestural data for music-related applications in the Metaverse. In [12], [13], extended reality (XR) platforms are proposed to support bidirectional polyrhythmic interactions between players, embodied by avatars, exploiting a virtual drum circle. Instead, our proposed scenario considers VR instead of XR and does not constrain the choice of the musicians' instruments to drums. In [14], an Extended Reality Environment (XRE) for immersive NMP is presented. The framework implements a position-dependent visual and auditory representation of the users within a shared virtual space, characterized by early reflections and diffuse reverberation. Our system could be integrated with a similar audio rendering framework to enhance the feeling of co-presence experienced by conductor and orchestra members. In [15], [16], upper body tracking is exploited for avatar representation of four remote musicians, coupled with immersive audio rendering of the virtual space. Differently, our system only tracks the conductor's head and hands movements.

III. System Overview

In this Section, we present our proposed framework for transmitting visual cues for remote orchestra conduction and explore the potential approaches for connecting a VR headset worn by the conductor to a web application used by the musicians in the NMP session.

The application running on the VR headset captures head and hand movements of the wearer and displays them in an immersive scenario. This scenario is shown from the conductor's point of view, with the musicians in front of him/her. Musicians are represented as avatars, with their name reported on top of them, and are positioned in the virtual space in a way that facilitates the conductor to identify and point to each musician individually. An example of the conductor's perspective is shown in Fig. 1.

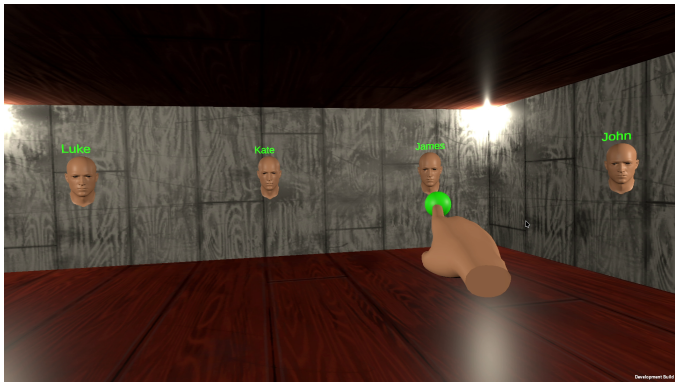

Fig. 1: Conductor's avatar pointing at *James*, seen from the conductor's perspective.

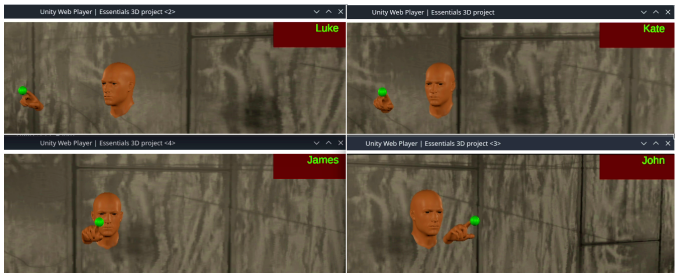

Fig. 2: The four renders displayed by the musicians' web application in the same scenario of Fig. 1

On the musician's side, a receiver device obtains the pose data from the conductor and renders it on a screen. It displays a virtual scenario depicting the avatars of the conductor and of the other musicians in the same field of view. An example of the conductor's avatar seen from the perspective of four different musicians is shown in Fig. 2. The positioning of each individual musician in the scene visualized by the conductor through the VR headset is coherent with the positioning of the conductor in the personalized visualization offered to each musician.

A. Transmission Setup

The main challenge of this system is the streaming of real-time pose data from the VR headset to the remote musicians' devices. First, even though any wired or wireless transmission medium could be leveraged to convey pose data to the musicians' devices (equipped with a web browser to visualize the conductor's gestures), it is highly recommended to rely on a wired Internet connection, since latency and jitter figures achievable by Wi-Fi and 4G technologies are hardly compatible with NMP requirements. Second, web applications cannot directly handle TCP connections or UDP packets. However, they can leverage WebRTC, that supports the exchange of messages in a peer-to-peer (P2P) fashion, as well ass the use of unreliable (UDP-like) data channels [17], which is the preferred choice when dealing with real-time data. Thus, our application uses WebRTC for the conductor's pose data streaming.

The application running on the VR headset is not strictly bound to any specific transport protocol. However, the VR headset has limited options when it comes to physical connections. It supports i) Wi-Fi, through an integrated Wi-Fi adapter; ii) a Wired Ethernet adapter over USB-C; iii) Android Debug Bridge (ADB)[2] on top of USB. During testing, we discovered that Ethernet support varies among VR headsets. At the time of writing, the Meta Quest 2 does not support it, while the Meta Quest 3 does. However, the graphical interface of the Meta Quest 3 does not provide consistent information about the connectivity status when using Ethernet, indicating that its integration is not yet stable. For such reasons, we opted for not leveraging an Ethernet connection, and relied instead on the ADB option since, as shown in Sec. V, it yields comparable latency results while being more stable software-wise. The ADB option has the main drawback of requiring an additional device to enable the connectivity from the VR headset to the Internet. However, since in a realistic scenario the conductor needs to receive the audio streams produced by the remote musicians, we reckon on the presence of a device, which can also act as a streaming device for pose data, that may additionally run a NMP system for audio transmission (e.g., JackTrip [18]). Note that our system is agnostic to the number of audio channels streamed by the NMP system, i.e., both stereo and immersive (3D) audio streaming can be used.

In summary, in our setup, the VR headset is connected to a streaming device via USB. The application on the VR headset communicates with other peers and the server through a TCP socket forwarded by ADB reverse forwarding. The streaming device, i.e., a generic computer running the ADB tool, handles the connections. Messages in the TCP socket are synchronously produced and consumed by the application immediately after each graphic frame generation.

B. Framework Description

Our VR-based system prototype for remote orchestra conduction consists of four main components, as shown in Fig. 3:

- **VR headset**: worn by the orchestra conductor to track his/her hands and head movements and to visualize the remote performers via avatars. In our experiments, the VR headset used was the Meta Quest 3.
- **Streaming device**: a PC connected to the VR headset via USB and to the Internet through a wired connection. It takes care of receiving pose data from the VR headset through ADB and of streaming it to the remote musicians through WebRTC.
- **Signaling server**: a server that facilitates the setup of WebRTC P2P connections.
- **Receiver(s) device**: a PC connected to the Internet with a wired connection. It takes care of receiving and displaying the conductor's movements to the musicians by means of a web application.

[2]https://developer.android.com/tools/adb

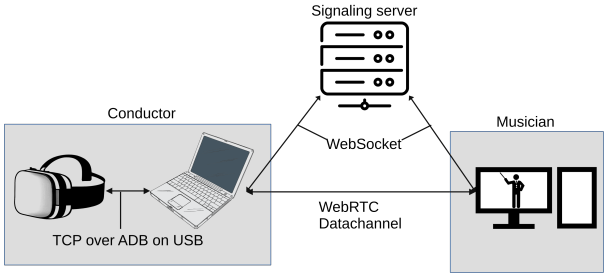

Fig. 3: System overview

Notably, the adoption of WebRTC is in line with recently proposed implementations of multi-user applications for the MM, e.g, in [19]–[21].

The application running on the VR headset is built with the Unity game engine[3] and the aid of the Oculus SDK[4]. The web application running on each receiver's device is also built with Unity and compiled as a WebGL target. It reuses the same scenario and layout of the application running on the VR headset, making the interaction and the positioning coherent among the two entities. The streaming device application and signaling server are Node.js applications.

IV. Evaluation Methodology

The M2P latency can be decomposed in:

a) Acquisition/Tracking latency: time required by software and hardware components of the VR headset to track head movements and hand gestures.

b) Processing latency: time taken by the application to incorporate the collected pose data in the message to be sent.

c) Network transmission latency: time needed to convey messages from the VR headset to the musicians' application.

d) Rendering latency (receiver device): time required to graphically render the received data.

e) Video output latency (receiver device): time required by the driver and hardware to display the rendered frames.

We consider three different latency-related measurements:

- *Network packet jitter*: message generation happens at fixed intervals (once per frame generation, i.e., $d = 1000/f$, where d indicates the delay in milliseconds and f indicates the frame rate, measured in frames per seconds (fps), of the VR headset). At the receiver side, we measure the inter-arrival time between two consecutive messages.
- *Messages Round-Trip-Time (RTT)*: we measure the time difference between the dispatch of an "echo request" to the streaming device (or musician application), to the corresponding reception of an "echo reply" message. It is measured in terms of d, which represents our time granularity for every measure performed in the VR headset.
- *M2P latency of hand movement*: using a camera with a frame rate of 119 fps, we record the movement of

[3]https://unity.com/

[4]https://developer.oculus.com/downloads/unity/

a single hand executing an upward conducting gesture, close to the screen where the musicians' application is being displayed. We then manually annotate the video frames corresponding to the beginning and the end of a movement, for both the real hand and the rendered one. The time difference between the frame annotated as the start of a given movement of the real hand and the corresponding frame for the rendered hand represents the M2P latency. It is measured in number of frames and then converted in milliseconds.

V. Results

A. Testbed Description

The test environment consists of a single machine concurrently operating as streaming device, signaling server and receiver device. The machine is a laptop with an Intel i5-7300U, 12 GB of RAM and a Linux distribution (kernel 6.8.5 PREEMPT_DYNAMIC). The musicians' web-application runs in Chromium (version 123). The laptop is connected to an external monitor (HP e24m G4) through DisplayPort over USB-C. The monitor is used to display the video output visualized in the browser, based on which all M2P measurements were collected.

B. Transmission Jitter

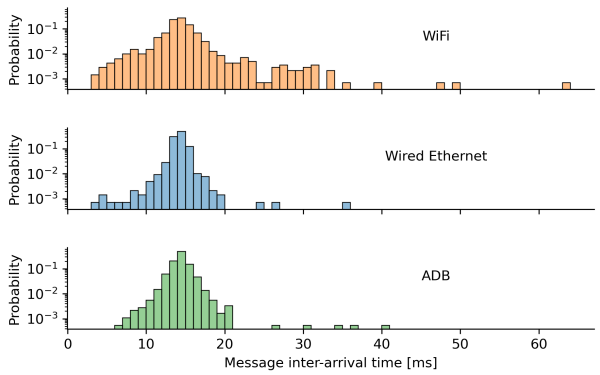

Fig. 4: Distributions of message inter-arrival times for the three transmission stacks. The ideal value should be $\sim$ 14 ms $=$ $1000/72$ fps.

Fig. 4 compares the jitter values achieved by the three different connections methods available on the VR headset. The wired-Ethernet connection (through the Ethernet to USB-C adapter) results to be the most stable one in terms of jitter, followed by ADB, whereas the Wi-Fi connection yields the worst jitter figures. While these results suggest that the best solution would be the use of the Ethernet connection through a USB-to-Ethernet adapter, we preferred the ADB option for the reasons described in Sec. III-A.

C. Messages RTT Measurements

Regarding RTT, we performed two measurements:

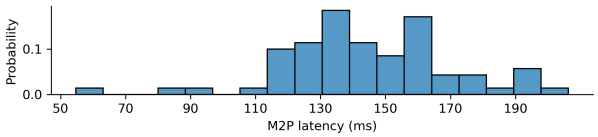

Fig. 5: M2P latency of the whole chain.

a) VR headset to streaming device (short path): the VR headset application sends an "echo request" at time t_1, and the application in the streaming device replies with an "echo reply". The VR headset receives the answer at time t_2. A single measure of the RTT (R) is equal to $t_2 - t_1$.

b) VR headset to musician application (long path): the application in the VR headset runs in the same way as above. The application in the streaming device forwards the "echo request" to a single connected musician's application, which replies to it. The streaming device forwards back the "echo reply" to the VR headset through ADB.

The short path RTT measurements yielded 99.7% of the measures exhibiting a value of $R = 1d$ and the remaining 0.3% exhibiting a value of $R = 2d$. Instead, the long path RTT experiment yielded 83.4% of the measures exhibiting a value of $R = 1d$, 13% with value $R = 2d$, 2.5% with value $R = 3d$, and the remaining 1.1% with values greater than $3d$. The RTT measured in terms of d can be converted in milliseconds by knowing the frame rate of the VR headset (f). The RTT expressed in ms (R_{ms}) is bound within the range $(R-1) <= \frac{R_{ms} \cdot f}{1000} <= R$. The total chain RTT (i.e., from the VR headset application to the musician's application) deserves further investigation but, assuming a symmetric RTT, we can infer that the one-way transmission latency is less than 21 ms in 99% of the cases (note that $f = 72$ fps on our VR headset). Pose data is rendered as it arrives, since no buffering is performed at the receiver's device. This has the effect of minimizing the latency at the cost of occasional graphical frame stuttering.

D. M2P Latency of Hand Movement

Fig. 5 depicts the measured M2P latency of hand movements. The camera used for recording had a frame rate of 119 fps. The median latency value is 17 frames, i.e., $\sim$ 143 ms. Based on the previously discussed RTT measurements, we know that latency due to transmission is lower than 21 ms. According to Abdlkarim et al. [22], the hand tracking latency for the Meta Quest 2 VR headset should be in the order of 40 ms. Other sources [23] state that the Meta Quest 3 should have a slightly lower hand tracking latency than the of Meta Quest 2, i.e., around $\sim$ 30 ms. This would imply that at most $\sim$ 90 ms are introduced by the processing time on board of the VR headset plus the rendering and output at musician's application side.

E. Bit-rate Requirements

Bit-rate requirements can be computed deterministically, assuming that both hands are tracked and that messages are sent once per frame generation. For each message we track the

6D pose of 24 bones per hand, for both hands, plus one pose representing the head. Overall, 49 6D poses are tracked. Each 6D pose consists of the Cartesian position and rotation data, both represented as triplets of 4 bytes floating-point values, totaling $2 \cdot 3 \cdot 4 = 24$ bytes per bone. Therefore, each message is 1176 bytes long, with an overhead of 24 bytes for our protocol implementation, thus yielding 1200 bytes. It follows that the bit-rate required for each communication channel director-musician amounts to 86.4 KB/s, i.e., 691.2 Kb/s. It is worth noting that the communication is stateless, i.e., every message conveys the current state of the system. Thus, a single loss only affects the receiver's state until the next message is received, avoiding any error accumulation issues.

It should be noted that further reductions of the bit-rate requirements could be achieved by diminishing the bit-depth of position and rotation data representations, at the price of a small degradation of the accuracy of the rendered pose. However, such degradation would not worsen over time, due to the absence of error accumulation.

VI. Limitations of the Proposed Approach

Although the results obtained are promising, the proposed prototype has some limitations in its setup. Firstly, the current prototype is not designed to accommodate more than one musician per device. Indeed, a single device is represented as one avatar from the conductor's perspective, making it impossible to distinguish between multiple musicians. Secondly, the prototype relies on embedded motion tracking features, which are based on a camera paired with deep-learning algorithms, both of which introduce substantial latency. Sensor-based alternatives could be investigated. Lastly, to the best of our knowledge, commercially available VR headsets offer limited customization options from both hardware and software standpoints. For instance, state-of-the-art NMP applications often rely on custom scheduling, as seen in JackTrip [18], and on the use of wired Ethernet connections, which was not feasible with the selected VR headset.

VII. Conclusions

This paper proposed a remote orchestra conduction setup for networked music performances, consisting of a VR headset that tracks the conductor's hands and head movements and streams them to remote musicians, who visualize pose data via a web-based application. We assessed the feasibility of the proposed system by measuring the achievable M2P latency and bit-rate requirements, based on the connectivity options available on the VR headset.

Results showed that latency values lie in the same order of magnitude as the human visual reaction time (i.e., around 600 ms as reported in [24]) and are comparable to those reported in [25], where the minimum noticeable delay is estimated to range between 160 and 320 ms for remote drum players[5]). Moreover, bit-rate requirements are at least one order of magnitude lower than those of state-of-the-art video streaming.

Future research efforts will be devoted to define and perform subjective tests able to assess the usability of the setup (e.g., to evaluate the impact of latency on the quality of experience perceived by the performers). As a future development, we also plan to integrate facial tracking as a mean to convey intentions and emotions to orchestra members via the conductor's facial expression. Moreover, a system enabling remote musical conduction based on motion tracking also paves the way for forthcoming research endeavors aimed at conveying the conductor's gestures to visually-impaired musicians by means of haptic feedback mechanisms, coupled with visual representation. Furthermore, we foresee a robot conductor that leads an orchestra by actuating in real time the gestures performed by a remote conductor. In this context, we plan to study the interaction between human musicians and the robotic conductor, and how the audience perceives the proposed technology.

Acknowledgment

This work has been partially supported by the Italian Ministry for University and Research under the PRIN program (grant n. 2022CZWWKP). Leonardo Severi's PhD Programme is funded by the European Union in the framework of the Resiliency and Recovery Plan (RRP), within the NextGenerationEU initiative.

References

[1] J. Lazzaro and J. Wawrzynek, "A case for network musical performance," in *Proceedings of the 11th International Workshop on Network and Operating Systems Support for Digital Audio and Video*, ser. NOSSDAV '01. New York, NY, USA: Association for Computing Machinery, 2001, p. 157–166. [Online]. Available: https://doi.org/10.1145/378344.378367

[2] F. A. Seddon, "Modes of communication during jazz improvisation," *British Journal of Music Education*, vol. 22, no. 1, p. 47–61, 2005.

[3] C. Rottondi, C. Chafe, C. Allocchio, and A. Sarti, "An overview on networked music performance technologies," *IEEE Access*, vol. 4, pp. 8823–8843, 2016.

[4] C. Drioli and N. Buso, "Networked performances and natural interaction via lola: Low latency high quality a/v streaming system," *Lecture Notes in Computer Science*, vol. 7990, pp. 240–250, 01 2013.

[5] B. Iribe, "Virtual reality-a new frontier in computing," Oculus VR, 2013.

[6] L. Turchet, "Musical metaverse: vision, opportunities, and challenges," *Personal and Ubiquitous Computing*, vol. 27, no. 5, pp. 1811–1827, 2023.

[7] L. Turchet, C. Fischione, G. Essl, D. Keller, and M. Barthet, "Internet of musical things: Vision and challenges," *Ieee access*, vol. 6, pp. 61 994–62 017, 2018.

[8] G. Martín, "Social and psychological impact of musical collective creative processes in virtual environments; te avatar orchestra metaverse in second life," *Music Technol.*, vol. 75, pp. 75–87, 2018.

[9] R. Steinmetz, "Human perception of jitter and media synchronization," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 1, pp. 61–72, 1996.

[5]It is worth mentioning that it is common practice for conductors to anticipate their gestural cues w.r.t. the desired timing of the orchestra reaction, in order to leave time to the musicians to process and react to the visual hints. Thus, we expect conductors to be able to compensate for the M2P delay introduced by our proposed system by adapting the anticipation time of their gestures.

[10] A. Olmos, M. Brulé, N. Bouillot, M. Benovoy, J. Blum, H. Sun, N. W. Lund, and J. R. Cooperstock, "Exploring the role of latency and orchestra placement on the networked performance of a distributed opera," in *12th annual international workshop on presence*, vol. 10. IWP Los Angeles, 2009, pp. 1–9.

[11] A. Carôt and G. Schuller, "Towards a telematic visual-conducting system," in *Audio Engineering Society Conference: 44th International Conference: Audio Networking*, Nov 2011. [Online]. Available: https://www.aes.org/e-lib/browse.cfm?elib=16137

[12] B. Van Kerrebroeck, K. Crombé, S. M. de Leymarie, M. Leman, and P.-J. Maes, "The virtual drum circle: polyrhythmic music interactions in mixed reality," *Journal of New Music Research*, pp. 1–21, 2024.

[13] T. Hopkins, S. C. C. Weng, R. Vanukuru, E. A. Wenzel, A. Banic, M. D. Gross, and E. Y.-L. Do, "Ar drum circle: Real-time collaborative drumming in ar," *Frontiers in Virtual Reality*, vol. 3, p. 847284, 2022.

[14] R. Hupke, S. Preihs, and J. Peissig, "Immersive room extension environment for networked music performance," in *Audio Engineering Society Convention 153*. Audio Engineering Society, 2022.

[15] A. Hunt, H. Daffern, and G. Kearney, "Avatar representation in extended reality for immersive networked music performance," in *Audio Engineering Society Conference: AES 2023 International Conference on Spatial and Immersive Audio*. Audio Engineering Society, 2023.

[16] P. Cairns, A. Hunt, D. Johnston, J. Cooper, B. Lee, H. Daffern, and G. Kearney, "Evaluation of metaverse music performance with bbc maida vale recording studios," *Journal of the Audio Engineering Society*, pp. 313–325, 2023.

[17] M. T. R. Jesup, S. Loreto, "Webrtc data channels," Internet Requests for Comments, RFC Editor, RFC 8831, January 2021. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc8831

[18] J.-P. Cáceres and C. Chafe, "Jacktrip: Under the hood of an engine for network audio," *Journal of New Music Research*, vol. 39, no. 3, pp. 183–187, 2010.

[19] A. Boem and L. Turchet, "Musical metaverse playgrounds: exploring the design of shared virtual sonic experiences on web browsers," in *2023 4th International Symposium on the Internet of Sounds*. IEEE, 2023, pp. 1–9.

[20] D. Dziwis, H. Von Coler, and C. Pörschmann, "Orchestra: a toolbox for live music performances in a web-based metaverse," *Journal of the Audio Engineering Society*, vol. 71, no. 11, pp. 802–812, 2023.

[21] D. Dziwis and H. von Coler, "The entanglement: Volumetric music performances in a virtual metaverse environment," *Journal of Network Music and Arts*, vol. 5, no. 1, p. 3, 2023.

[22] D. Abdlkarim, M. Di Luca, P. Aves, M. Maaroufi, S.-H. Yeo, R. C. Miall, P. Holland, and J. M. Galea, "A methodological framework to assess the accuracy of virtual reality hand-tracking systems: A case study with the meta quest 2," *Behavior Research Methods*, vol. 56, no. 2, pp. 1052–1063, Feb 2024. [Online]. Available: https://doi.org/10.3758/s13428-022-02051-8

[23] "Vision pro and quest 3 hand-tracking latency compared," accessed on May 01, 2024. [Online]. Available: https://www.roadtovr.com/apple-vision-pro-meta-quest-3-hand-tracking-latency-comparison/

[24] C. Orosy-Fildes and R. W. Allan, "Psychology of computer use: Xii. videogame play: Human reaction time to visual stimuli," *Perceptual and Motor Skills*, vol. 69, no. 1, pp. 243–247, 1989.

[25] T. Hopkins, S. C.-C. Weng, R. Vanukuru, E. Wenzel, A. Banic, M. D. Gross, and E. Y.-L. Do, "Studying the effects of network latency on audio-visual perception during an ar musical task," in *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2022, pp. 26–34.