LIVE MUSIC IN VIRTUAL IMMERSIVE SPACES

G. A. Thomas[1], F. M. Rivera[1], L. Kelso[1], B. Weir [1], P. Rich [1],
O. Moolan-Feroze[2]
[1]BBC R&D, UK, [2] Condense Reality, UK

ABSTRACT

Game-like environments that offer live multiplayer capability are becoming a major form of entertainment. A large community, estimated to be 3.2M in the UK, also spend time in these environments for social & experiential reasons, rather than gaming. Music artists are using these spaces to present virtual concerts, drawing in big crowds and revenue. Broadcasters, as well as major music labels, are starting to look at how to harness games-like media to deliver live music events.

This paper presents approaches the BBC has been trialling for delivering live events into virtual immersive spaces. Trials are being run to explore the use of low-latency volumetric capture technology of the artists, to allow virtual attendees, through their avatars, to interact with the performer and each other. Other trials are looking at the capture of performers in larger spaces such as stages at a festival, relying on 2/2.5D video approaches. Results from these trials are reported, including technical aspects and audience feedback.

INTRODUCTION

Game-like environments that offer live multiplayer capability are becoming a major form of entertainment. A large community, estimated to be 3.2M in the UK, also spend time in these environments for social & experiential reasons, rather than gaming. Music artists are using these spaces to present virtual concerts, drawing in big crowds and revenue. Ariana Grande’s 2021 ‘Rift Tour’ in Fortnite, garnered 78M views across 5 events. In 2020, Travis Scott’s 5 events in Fortnite generated an estimated $12M in merchandise sales. At the same time, audiences for TV and radio are reducing. Broadcasters, as well as major music labels, are starting to look at how to harness games-like media to deliver live music events. Such approaches offer the potential for audiences to engage in a more interactive way with the performers and each other.

There are many challenges around delivering a live concert into these virtual immersive spaces. The Fortnight concerts relied on pre-generated animated avatars of the performer, making live interaction with the audience unfeasible. Such an approach would also significantly add to the production cost and make a ‘simulcast’ of a broadcast event difficult or impossible.

This paper presents approaches the BBC has been trialling for delivering live events into virtual immersive spaces.

BUSINESS, OPERATIONAL AND AUDIENCE REQUIREMENTS

Pre-animated performers rendered as 3D avatars in virtual music experiences enable fans to experience the music in shared spaces and interact with others. However, initial surveys we conducted before embarking on any user trials indicate that potential audience members and live music artists would like the audience to be able to interact with the performer in real-time through modes such as sing-along, cheering, dancing, and expression of emotions [Rivera et al. (1)]. This was later borne out through audience surveys of the trial events described later in this paper. Video-based capture of performers (as distinct from motion capture) was also favoured from a business perspective, as this makes it easier to simulcast a performance being staged for a live audience or TV show, where the performer would not want constraints on what they can wear.

We took a holistic approach for this preliminary research to elicit a requirements wish list for live immersive music events. We considered an audience perspective, the perspective of music artists, and that of extended reality (XR) practitioners with experience in the field. We treated the initial findings as points to consider, but still open to further investigation, on which subsequent sections of this paper will elaborate.

Our survey [Rivera et al. (1)] had a total of 45 responses from potential audience members, and 15 responses from workshop participants. These indicated that the ability to interact with each other and with the music artist, the agency to choose a point-of-view, adapt the audio mix, re-watch the performance, explore the environment, and take away digital artefacts, were unanimously popular features. Proximity-based audio chats were the most desired mode of interaction with friends, with text chats the next most popular. However, text chats were the most desired mode of interaction for conversing with new people, rather than audio. Personalisation of avatars was also a high priority, with the added option for gender-neutral avatars. The strongest incentives to join a live immersive music experience (apart from the music) was to socialise with friends.

The importance of supporting text-based chat was also reiterated in interviews with 4 established XR practitioners. Amongst other findings it was identified that a low latency real-time experience, supported by a text chat platform (either built into the platform or using an external service, such as Discord) was considered key to helping foster a sense of engagement for an audience attending a virtual event. This was also borne out in feedback from pilot live events as described later in the paper.

We also interviewed 5 music artists (with substantial experience playing live gigs) to explore expectations from a creative performer's perspective. The artists prioritised good quality sound, with spatial audio and realistic room acoustics generally preferred, although some preferred sound to reflect their own intended sound mix. Audience reaction through real-time low latency feedback was considered essential to the performers. If playing to a live real-world crowd and simultaneously to the virtual audience, it was deemed important that their view of the virtual world did not distract from the live performance (for example having to look at a small monitor, or text messages). The concept of a large LED wall displaying the virtual world was appealing to provide more of a seamless audience. The potential to reach new audiences, and to express themselves in more immersive and creative ways was highly appealing. Some of the artists preferred being represented as realistically as possible (through volumetric capture for instance), while others were enticed by the possibilities of having more creative representations (such as custom avatars).

TECHNICAL APPROACHES

Volumetric Capture

Volumetric capture describes a group of technologies that are able to represent real world 3D objects and events in a way that can be played back and rendered from a floating “free viewpoint” camera. Typically, this type of playback is enabled via 3D game engines which allow events that have been volumetrically captured to appear alongside traditional 3D graphics assets.

The volumetric capture system used in the experiments takes a similar form to that presented by Orts-Escolano et al. (2). On the recording end, a set of 10 calibrated depth and RGB sensors capture raw image frames. These frames are passed through a “fusion” process that generates a 3D model, which is represented as a triangle mesh and UV mapped texture. The fusion process implements a real-time hole filling and an occlusion prediction process which compensates for noisy and incomplete information coming in from the cameras. During operation, the system records everything that occurs within a 3D “capture area”. Events that fall outside of this area will not be represented in the output model. The capture area is limited to 4mx4mx3m to ensure real-time playback. There are also limitations on the amount of surface area within the scene, both in terms of processing times and bit-rate limitations. Once the data is fused, the model is delivered to a cloud-based distribution system which compresses the data to support a number of different bit rates, and will generate segments similar to an HLS/DASH system. The segments are then served out via a manifest to client devices over a CDN.

As a means of injecting live content into virtual spaces, volumetric capture has a number of benefits. Compared to methods such as motion capture, it provides a much more authentic representation of the performance, allowing anything within the capture volume to be captured and represented in-game. It does not suffer from the uncanny valley effect that can result from the armature of the motion capture not being able to accurately represent the performer's movements. When compared to 2D and 2.5D video, volumetric video enables full free viewpoint representation, as well as the opportunity to shadow cast and relight the content which allows better integration into the virtual scene.

However, volumetric capture does present challenges. A calibrated camera rig needs to be set up surrounding the performer. The calibration can take time, and it is possible to disrupt the calibration during an event if the cameras are knocked. Additionally, depending on the technology employed within the cameras, certain materials and surfaces are unable to be properly captured. Some of this can be solved algorithmically, although in practice, it is simplest to put some limitations on the artist's costumes and props. The interaction between the cameras and the lighting can cause additional problems. To be able to relight the content within the game engine requires flat lighting in the capture area. This causes difficulties when recording content during in-person events where existing event lighting is in place.

2D / 2.5D Capture

A simpler approach that may suffice in situations where a fully-free viewpoint is not required, is to use one or more conventional 2D video streams (sometimes referred to as video billboards), or so-called 2.5D video (where depth data may be used to help extend the range of viewpoints) [Grau et al. (3)]. Such approaches may be particularly suitable when the range of viewpoints is naturally limited, such as for an audience viewing a stage from the front. The lack of true 3D may also be less apparent in browser-based VR use cases (which are our current focus), where the user has no stereoscopic depth cues, rather than applications using a VR headset. There is also anecdotal evidence that flat images placed within a 3D environment can trick the brain into seeing them as true 3D, through the successful use of Pepper's Ghost illusions for so-called holographic performers on stage [Grow (4)].

Adding depth data to images, or simply using an alpha mask to define the foreground area, can help make a performer captured in a single video stream look better-integrated into a 3D environment. However, often the stage background or lighting/haze effects form an important part of a live performance, so for our initial work we have focused on what can be achieved with one or more conventional video streams without any sophisticated processing, to provide a baseline for representing a live music stage performance without needing to impose constraints on the performance itself.

For this experiment, we built a 3D "nightclub-style" environment in Unreal Engine that included a replica of the real-world stage on which we had recorded a number of musical performances. We then tested the user responses to the following scenarios:

1) A single view of a performer taken from a camera in front of the stage, placed on a 3D plane covering the virtual set stage boundaries.

2) A view of the performer from one of three different camera positions, placed on a 3D plane covering the virtual set stage boundaries. The camera feed displayed on that plane was chosen on a frame-by-frame basis according to whichever camera was closest to the player viewpoint, the idea being to switch to the "best" perspective on the musician as the user moves about.

3) As #2, but with the addition of a strobe-light effect from virtual lights in the 3D scene that triggered whenever the displayed camera view changed. Early experimentation showed that this could distract the user from noticing the camera view transition.

The video from each camera was packed into one of four quadrants in a single 3840x2160 UHD video encoded as mp4 at 20 Mbps. That meant that the resolution of a single camera feed was 1920x1080. This resolution was used for each of the three scenarios above. Since we were using at most three cameras, the fourth quadrant can hold a "broadcast-style" cut that can be used as a texture for a virtual "big screen" in the virtual nightclub (note though that we did not add the big screen for this experiment). Although we would typically stream the video live into the rendering application, for this experiment the videos were added as files into the build to ensure consistent display quality for the user.

The three cameras were positioned to record the performance as shown in the photo below. The height off the ground for each camera was chosen to match the typical viewing height of the user's camera in the 3D scene, and the distance from the stage was chosen to match the typical viewing distance of the user from the stage in the 3D scene.

Each camera view was framed such that its image encompassed the entire stage. A physically accurate model of the stage was created and placed into the 3D world, and a plane (onto which the texture of the video from the cameras was drawn) was positioned to cover the stage corners. In order to ensure that the stage corners in the video were correctly placed at the corners of the video plane, a transform matrix for each camera was calculated that mapped the UV values for the stage corners in each camera image to the corners of the video plane.

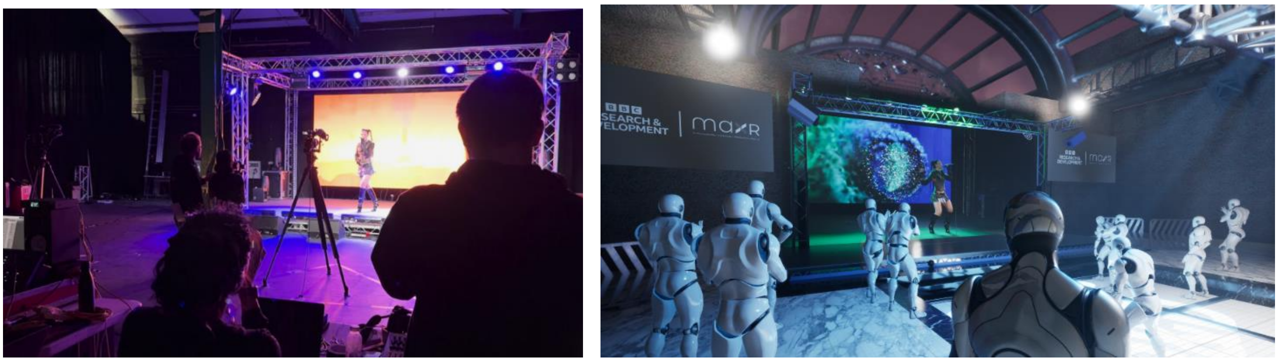

Figure 1 – Capture of a performance, with central and left-hand cameras visible (left); camera feed from left-hand camera embedded in virtual night club (right).

Delivery and Rendering

For the user to experience the performance in a game-like environment, they need to be able to run what is effectively a 3D multiplayer networked game that receives a live stream of the captured data. There are several approaches that can be taken for this.

Pixel streaming, browser-based rendering and using a downloadable executable have a number of benefits and drawbacks. Pixel-streaming represents the most easily accessible technology for users. Typically, only a browser is required, and the bandwidth requirements are similar to those required for typical video streaming. This is because the heavy rendering tasks are performed remotely, and only the rendered output is delivered to the client. The main drawback of pixel streaming is the cost and limitations on availability of cloud rendering resources. These limitations can severely constrain the overall reach of the event. From a user's perspective, pixel streaming has a few drawbacks. First, due to additional latency incurred by sending control inputs to the cloud renderer, game play can feel laggy and unresponsive. Furthermore, due to compression artefacts, the visual presentation is not always great. These artefacts can include blocking, as well as colour space reductions.

Browser-based rendering has a number of advantages. The user only requires a modern web browser to access the event. As the content is rendered locally there is none of the input lag and compression artefacts caused by pixel streaming. Another benefit of browser rendering is that it can better integrate with web-based platforms, as no switch between applications is required when moving from the platform to the performance. The main drawback from browser-based rendering is the limitations that browsers put on access to the user's machine for security purposes. This can limit the feature-set that can be delivered to users. In the case of volumetric video, the rendering process can also require very modern browser features such as web codecs, which require users to be running the most recent version of their browser.

Rendering via a downloaded application will provide the best user experience, as the application can make full use of the user's system, with no limitations on the rendering feature set. However, as the application will need to be downloaded and installed on the user's system, this method is likely the most limited in its accessibility. Additional limitations can be imposed depending on the application distribution method. Distributing via digital platforms such as Steam or the App Store can simplify the download and install process, as well as provide users with additional assurances around the security of the application. However, these digital platforms can put limitations on monetisation options, as well as imposing significant requirements on submission.

RESULTS TO DATE

Volumetric capture

We have so far conducted four trials using volumetric capture. The first three of these relied on a pixel streaming solution and the fourth on browser-based rendering. All were accessible via the browser and did not require any additional downloading of supporting applications. Each trial was conducted as part of a live music session broadcast as part of a radio show and was therefore produced in collaboration with a studio production team. The artists involved each performed three songs and were captured in the volumetric rig which had been constructed within the recording studio space.

For the duration of these live performances, the audience were given the option of joining the immersive experience which included the ability to choose an avatar and then navigate a virtual venue using keyboard and mouse controls. The fourth trial also included the ability to join via a browser on a phone and included touch control. A volumetric capture of the live performer was visible on a virtual stage alongside fellow avatars that had joined the experience. Technical limitations meant that only 30 user avatars could be rendered in a common space, so the audience was divided into identical ‘rooms’, each holding up to 30 people plus the performer. Each performer had access to a screen as they performed which gave a fixed perspective of the virtual venue meaning they could see avatars of audience members and their movements in one of the rooms with a 10-second delay. These trials were publicised on the radio show with joining details offered on air. A ‘common stream’ was also produced from a virtual viewpoint controlled by the production team, which was made available as conventional streaming video for those unable to join virtually.

Around 150 unique users joined each trial as avatars and feedback was captured for the third trial via three online focus groups of differing ages (15-16, 17-19 and 20-24). Overall, responders found the experience intriguing and unique but their interest to attend further events was predominantly driven by the choice of artist performing. In particular, it was felt that the experience lacked social features that would better emulate the communal feeling of a concert experience. On the basis of this feedback, and our initial audience research, a Discord server was used as part of the fourth trial which allowed for interaction between participants and the radio DJ. This – in particular – was a feature that the audience seemed to value.

Our initial findings from these trials showed that the livestreaming of volumetric capture into a 3D game engine environment allows for elements of interaction with a virtual audience which is not possible via other mediums. The level of interaction from the performer was very dependent on their comfort with the technology and being briefed sufficiently, though dance moves and emotes played an important role in making this interaction feel two-way and dynamic between attendees and performer. While we are yet to test in-experience voice or text communication, Discord proved a powerful way for users to interact with each other, build anticipation and share their experience of the live experience together. Re-posting some of these messages into the in-app messaging solution also made the experience feel more alive and dynamic for those not actively participating in the Discord chat. Scaled interaction was made challenging as a result of the 30-person room limitation applicable within our trials meaning some participants found themselves in less trafficked rooms, and only one room was visible to the performer.

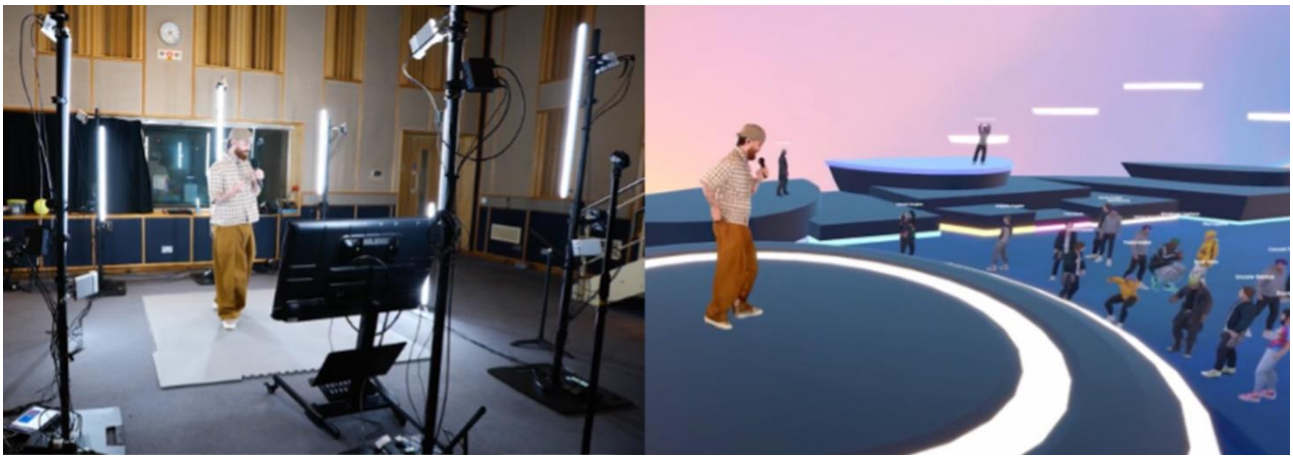

Figure 2 – Capture of a performance, with monitor to show view of audience (left); volumetric render of captured performer being watched by audience avatars (right).

The volumetric capture and delivery system is able to measure bitrate and latency metrics while recording. During the browser-based rendering event we compressed the volumetric content at two bitrates. The “low bitrate” was produced to be consumed by browsers and the “high bitrate” to be consumed by the team producing the common stream, where there were no bandwidth limitations. Of the 3 types of streamed content (audio, mesh geometry, and texture), for the “low bitrate” stream the audio was ~1mbit/s, the mesh geometry ~8mbit/s and the texture data ~10mbit/s. This resulted in a total bandwidth requirement of ~20mbit/s. For the “high bitrate” stream, the equivalent values are ~1mbit/s, ~24mbit/s and ~20mbit/s for a total of 45mbit/s.

For the events which were handled via pixel streaming, we ran with a single bitrate. This was set quite high as there were no bandwidth limitations on the cloud infrastructure handling the pixel streaming. The values were ~1mbit/s for audio, ~30mbit/s for mesh geometry and ~35mbit/s for the texture data.

The latency of the pipeline was measured in a number of places. The total in-rig latency - which measures the on-site processing time – was around 100ms. Once delivered to the cloud, the compression added around ~1000ms and the segment processing and delivery to the CDN incurred a further ~3500ms of latency. In total this adds up to ~4600ms of latency. This gives plenty of head room to allow the clients to playback at a consistent 10-second latency. Although this sounds high, it can largely be mitigated by briefing the performer to not expect instant reactions from the attendees. Interactions between the attendees themselves had a much lower latency.

2D / 2.5D Capture

We conducted a multi-day test shoot with four music acts at Production Park studios, to trial our technology in a setting resembling a live concert/festival. We live streamed into our virtual nightclub in Unreal using our three locked-off cameras approach (on the left, middle and right-hand sides of the stage).

We evaluated a number of streaming options for delivering live video into an Unreal application and settled on the Millicast plugin [Dolby (5)]. This provided the desired functionality, a managed streaming service, and compatibility with existing applications built by our partners in the MAX-R project in which this work was carried out. Millicast provides support for HD resolution video with low latency for live video. This meant that as we were using a quad video incorporating multiple camera views then each camera view was reduced to 960x540 pixels. However, for the user trials for testing camera layouts, we used 4K video files rather than a live stream, with each camera view ending up as 1920x1080 pixels. A fork of the OBS-Studio video streaming software was chosen to provide the Web-RTC video stream required by Millicast.

Streaming the multi-cam setup to our game environment enabled users joining as avatars to see the most relevant point of view for the position they moved to. Although this approach provides a more limited range of viewpoints in the 3D spaces than volumetric capture, it enables capturing performances with challenging staging conditions that could interfere with volumetric capture. For example, our test shoot included theatrical flashing stage lighting, and dynamic graphics displayed on an LED wall backdrop, with some haze. Dynamic virtual stage lighting was triggered by lighting in the videos of the performers to facilitate a better visual connection between the real performance and virtual space. Two of the music acts also made use of the entire stage area which was significantly larger than the volumetric capture approach discussed above can easily handle.

We conducted an online study using this set up, to explore potential preferences for the following viewing conditions for the performances:

- C1: Single centre camera view only. This provides seamless viewing within the virtual venue, but skewed perspectives to either side of the stage
- C2: Multi-cam view with 3 cameras. This provides more accurate viewing perspectives from the sides, and thus potentially more sense of depth, but (currently) at the cost of a visual glitch when the camera plane updates in response to the avatar's position in the room.
- C3: Multi-cam view as in C2, but with flashing stage light distractions the point of camera change as the player avatar's viewpoint changes at from one camera to another.

Consistent viewing angles and duration for each condition were provided by the point-of-view from a player avatar following a specified path around the virtual venue. We used video clips of the experience recorded in the gaming engine for each condition for 3 of our artists to open the study to those who are not familiar with 3D navigation in games. Participants were instructed to view each video and respond to whether the appearance of the musical performance was satisfactory. We also asked whether the lighting in the scene enhanced the viewing experience since we had used lights to distract from camera changes. We had also received indications from some pilot test participants that lighting might impact perceived levels of realism, and engagement. In a third question, we asked about interest in joining an interactive experience based on the examples shown. Figure 3 shows some example views from the study.

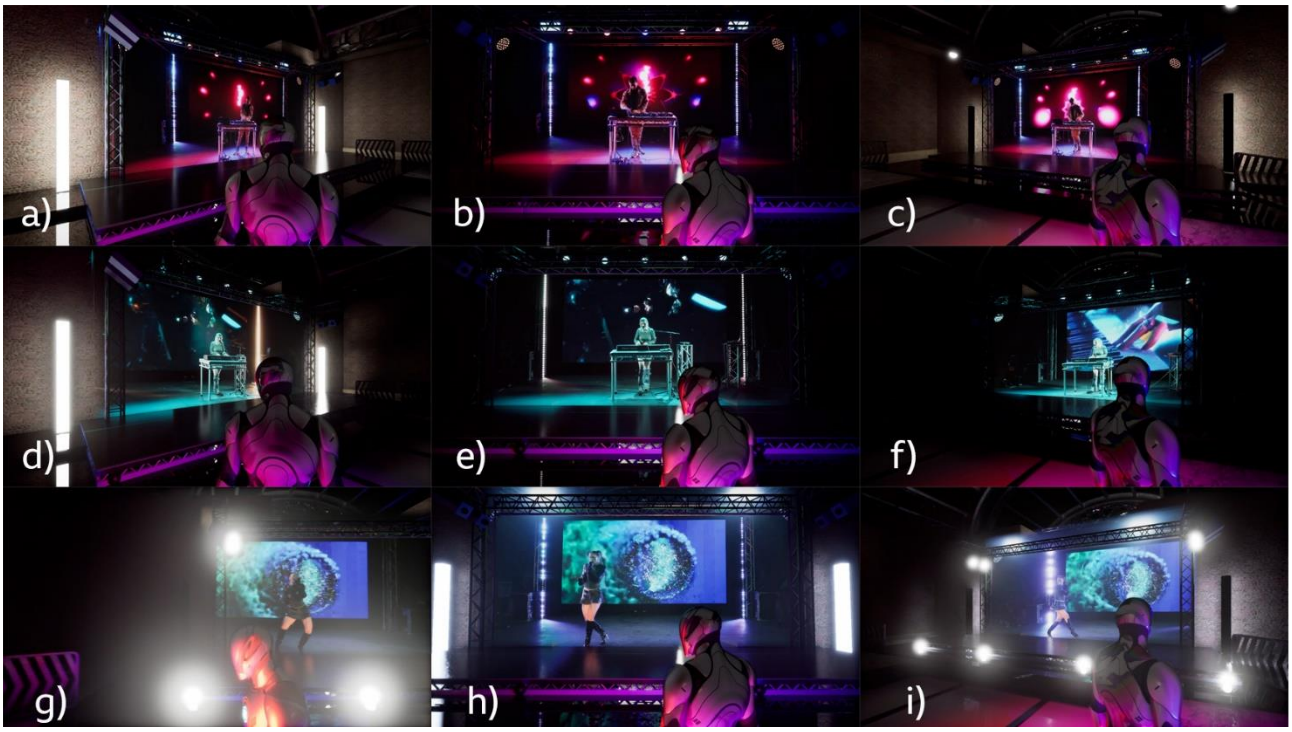

Figure 3 - Example views from the study. The top row (a), (b), (c) show single camera views (artist KDYN) from either side of stage and centre. The middle row (d), (e), (f) show multi-cam views (artist TWST). And the bottom row (g), (h), (i), show the multi-cam views (artist Badliana) when the distraction lights are triggered (g), (i) as the player avatar point of view changes when moving to the side.

As shown in Figure 4, results from our 75 respondents reflect that when considering C3 (multi-cam with distractions) 55.2% of respondents agreed/strongly agreed with our three propositions regarding satisfaction of viewing angles of the performer, the lighting enhancing the experience, and interest in joining an experience based on the example. 27.7% disagreed/strongly disagreed, with the remaining 17% neutral. C2 produced similar results with 53.6% positive (agreed/strongly agreed), 28.3% more negative (disagreed /strongly disagreed), and the remaining 18% were neutral. C1 (the single-cam) option had similar neutral responses of 18%, with 50.4% agreeing/strongly agreeing, and 31.5% disagreeing/strongly disagreeing.

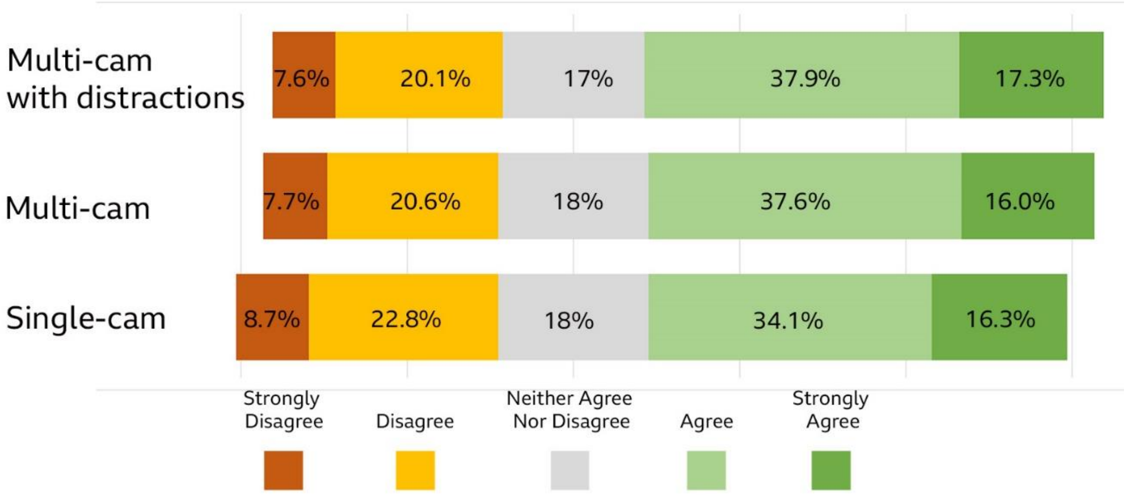

Figure 4 - Overall study results, showing percentage of respondents' agreement/disagreement levels to our propositions across our three conditions

We also asked respondents at the end of the study to prioritise up to 3 factors about the experience that they would like to have improved. The most popular options were to make the 3D performer seem more a part of the 3D world (31%), with some supporting comments suggesting for integration of lighting from real-world to virtual world (e.g. "match colours of performance with in 3D space light colour"). 13% would also like to have the performer fill up more of the stage, and 9% would like to have more music genres.

We deem the small differences between C1 (single cam) and the multi-cam conditions worth further exploration in an interactive version of the study, where participants can freely explore the space and their own viewing angles. The qualitative feedback also indicates that the relationship between the lighting in the virtual space and in the real-world performance is worth further exploration.

Audience interactivity

As mentioned previously, audience experience and interactivity came out loud and clear in audience feedback from our surveys and trials. There were calls for more distinctive storytelling in the event, including countdowns to the event beginning, more dynamism/sense of progression during the event and clearer follow-on journeys at the end. Customisable avatars were a popular addition, though there were issues for some with the controls, especially amongst those less familiar with game environments.

In terms of audience participation, an ability to communicate between users was seen as key, with this working particularly well when the radio DJ hosting the broadcast was able to seamlessly link the immersive experience, radio show and Discord with shout outs and interactions.

CONCLUSION

This paper has presented a summary of work in progress to evaluate technical approaches and audience feedback for presentation of live music events in a game-like multi-user shared virtual space. Further developments and trials are currently taking place, and we expect to have new results to present by September 2024.

From the results to date, it is clear that good visual representation is important, and this includes the appearance and lighting of the virtual venue as well as the appearance of the artist themselves. There are also many other aspects that need to be considered in order to provide a good experience for users, including a clear narrative for the event in terms of how it starts and ends, and support for audience participation including communication with each other and with the artist.

Over the coming months we intend to advance the volumetric capture system in two ways. First, we will look at improving the compression rates of our volumetric video. By better leveraging the temporal consistency of the data, we believe we can maintain existing visual quality at lower bandwidth, and increase the accessibility of the events. Second, we are looking at moving some of our existing on-site processing to the cloud. This will allow us to increase the reliability of the capture system by running multiple redundant pipelines, as well as relieve some of the resource constraints we have due to running on-premises hardware.

We also plan to further study the 2D video approach, in particular for larger events, with experiments planned using content from a music festival.

REFERENCES

1. Rivera, F., Thomas, G. et al. 2023. D2.2 Report on Scenario Use-Cases for Pipelines using Virtual and XR Production (see Annex 1). MAX-R project public deliverable available at https://www.max-r.eu/documents

2. Orts-Escolano, S. Et al. 2016. Holoportation: Virtual 3D Teleportation in Real-time. UIST '16: Proceedings of the 29th Annual Symposium on User Interface Software and Technology, pp. 741–754. https://dl.acm.org/doi/abs/10.1145/2984511.2984517

3. Grau, O., Price, M., Thomas, G. 2002. Use of 3-D Techniques for Virtual Production. BBC R&D White Paper WHP033. https://www.bbc.co.uk/rd/publications/whitepaper033

4. Grow, K. 2019. Live After Death: Inside Music's Booming New Hologram Touring Industry. Rolling Stone Magazine, Sept. 2019. https://www.rollingstone.com/music/music-features/hologram-tours-roy-orbison-frank-zappa-whitney-houston-873399/

5. Dolby. 2024. Dolby.io Real-time Streaming Player Plugin for Unreal Engine. https://docs.dolby.io/streaming-apis/docs/unreal-player-plugin

ACKNOWLEDGEMENTS

The authors would like to thank the music artists who took part in these trials, including Badliana, KDYN, TWST and Sam Tompkins. They would also like to thank staff at Production Park, UK and students at the Academy of Live Technology for help with the 2D test shoot. Some of the work reported here was carried out as a part of the MAX-R project, which is co-funded by Innovate UK and the European Union's Horizon Europe Research & Innovation Programme under Grant Agreement No. 101070072.