IBC2022

5G EDGE-XR PROJECT: LIVE BROADCAST XR EXPERIENCES USING 5G AND CLOUD-GPU COMPUTE

A. P. Gower[1], R. G. Oldfield[2], N. Fellingham[3], A. Godfrey[4]

[1]BT, UK; [2]Salsa Sound, UK; [3]Condense, UK;
[4]The Grid Factory, UK

ABSTRACT

There is an increasing appetite for extended reality (XR) enhanced broadcast experiences that offer augmented graphics, volumetric video, and immersive spatial content. However, the computational requirements for such implementations are high, resulting in experiences that can only be enjoyed by those with the most powerful consumer hardware which limits audience reach.

In this paper we present the work of the 5G Edge-XR project, a DCMS funded collaboration of UK organisations that looked to leverage cloud-GPU compute and high-speed 5G connectivity to increase access to next generation Extended Reality experiences for the broadest range of application use cases.

INTRODUCTION

In this paper we present work undertaken in the UK Department of Culture Media and Sport (DCMS) funded, 5G Edge-XR project [1]. A collaboration between BT, Salsa Sound, Condense, The Grid Factory, DanceEast and the University of Bristol. The project explored how high-quality augmented and virtual reality immersive experiences could be broadcast to audiences with consumer AR/VR headsets, smartphones and tablets, using cloud-GPU to render XR presentations delivered over 5G networks. The goal was to democratise access to XR experiences by reducing the need for heavy processing on end-user devices.

The project has shown [2] that cloud-based GPU clusters can deliver complex real-time rendered free-viewpoint XR experiences over high-speed low-latency 5G networks, with a fidelity that cannot be easily matched by traditional client-side rendering approaches. The use-cases that the technology facilitates range from medical data imaging, retail, enhanced and immersive sport broadcast, in stadium experiences and dance education and performance. Of primary focus in this paper are the sports broadcast use-cases which include AR volumetric boxing, AR MotoGP, AR in-stadium rugby and VR 360º football.

The paper is structured as follows. We begin with a description of the overall architecture of the 5G Edge-XR system with particular focus on the cloud-GPU technologies that under-pin the system and delivery network. Following this we provide an overview of the important scene capture technologies including the volumetric video, 360º video, spatial audio and tracking data. We then describe in more detail the focused use-cases and finally present results and evaluation of the project.

ARCHITECTURE

The end-to-end production chain used to deliver 5G Edge-XR experiences starts with content assets being created/captured and encoded prior to contribution to the cloud-GPU platform. The assets are then imported into an OpenVR application scene which renders a viewpoint reflecting the position and orientation of the client device. In this configuration the requirements of the client device are relatively simple; they provide position and orientation information to the GPU processing system and decode and present the received audio-visual stream.

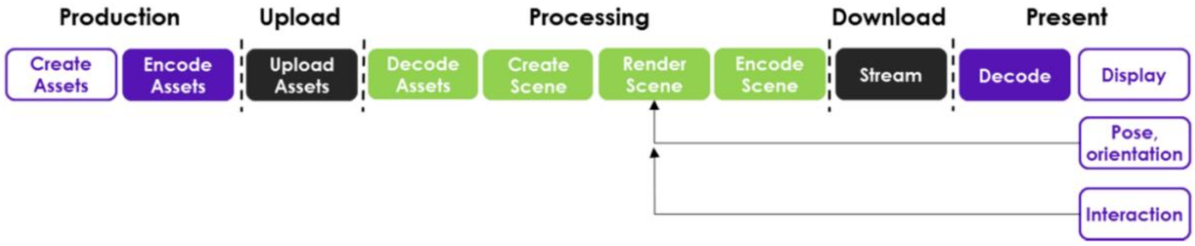

Figure 1 – Overview of the end-to-end data flow for delivering cloud-rendered XR experiences.

Network

The logical and physical location of the GPU affects the end-to-end latency. Public Internet-based cloud resources might typically add 20-30ms to the round-trip time. By siting the GPU closer to the point of presentation, say at the logical edge of the network prior to exiting the network via a peering node, we reduce the end-to-end delay resulting in a more responsive user experience. The key challenge is to minimise the latency between the position and orientation information being sent from the client device to it receiving an updated rendered frame of video. Figure 2 provides an overview of typical round-trip latency for cloud-GPU.

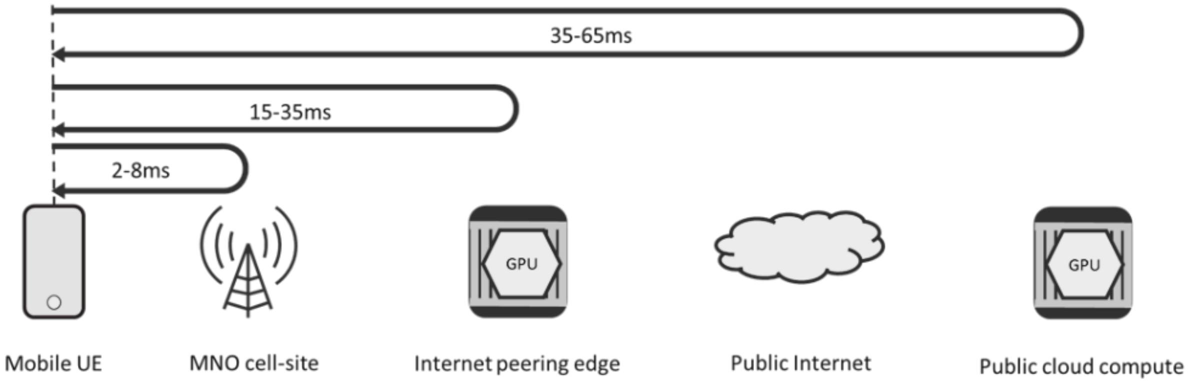

Figure 2 – Simplified network routing between the mobile device and the GPU cluster from where the experience is streamed, with associated estimated round-trip delays.

CloudXR

The project used NVIDIA's CloudXR platform which enables Unity developed OpenVR applications to be controlled, rendered and remotely viewed by end-user clients. CloudXR leverages NVIDIA RTX powered servers running VMware GPU virtualisation software to stream XR experiences of OpenVR applications. This enables the most complex XR scenes to be rendered on cloud-based servers and streamed to any device over any high-bandwidth low-latency networks. CloudXR dynamically adjusts the encode profile of streams based on variable network conditions to maximise image quality and frame rate while minimising the impact of network latency and stuttering. However, it should be noted that CloudXR does not currently support any form of Digital Rights Management (DRM).

The following outlines the step-by-step process of starting a CloudXR broadcast stream.

1. Content is captured and streamed to the cloud-GPU cluster directly or via a CDN.
2. An OpenVR application is selected by the user.
3. The Broker Service selects a free Virtual Machine and starts SteamVR.
4. The Broker Service then starts the OpenVR application on the Virtual Machine (VM).
5. The OpenVR application requests selected content.
6. The OpenVR application requests content via an NGNIX cache.
7. Content is streamed from a local or remote CDN into the OpenVR application.
8. The application scene is delivered via SteamVR to CloudXR.
9. Tracking and current view is provided by the client to CloudXR.
10. CloudXR encodes the current view.
11. CloudXR streams the current view to the client.

Cloud-GPU Compute Cluster

The GPU cluster used in 5G Edge-XR consists of four Dell servers, each hosting three Nvidia RTX8000 graphics cards with 48GBytes of frame buffer. The cluster runs the VMware hypervisor, with each virtual machine running Windows. SteamVR and NVIDIA’s CloudXR plugin provides the remote rendering and server to client streaming. The VMware GPU virtualisation software was configured to host a maximum of two VM per GPU card. This was the optimal configuration for the use case application developed in this project, although other use cases could be supported by five or more VMs per GPU. Each VM was configured as follows for all the use cases described in this paper.

Operating System	Windows Server 2019 Datacentre
Processor	AMD EPYC 7F72 24-Core Processor
Virtual Processors	16
Speed	3.19GHz
RAM	16GB
GPU	NVIDIA GRID RTX8000P-8Q
GPU Memory	15.5GB
GPU Dedicated GPU Memory	7.5GB
GPU Shared GPU Memory	8GB

Figure 3 – Virtual Machine configuration used across the cluster for the 5G edge-XR project

TECHNOLOGY ENABLERS

Key to the success of any XR experience are the scene assets. In this section we describe the main technology enablers that are required for scene capture and presentation.

Volumetric Video

Volumetric video is seen by many as the technology that will provide tools and content for next generation immersive story telling XR experiences. In the context of a broadcast boxing event, volumetric video can be used to capture an entire boxing match in 3D, where the captured event can be watched from any angle. The broadcaster can further add statistics annotations and AR effects to enhance the viewer experience.

To created 3D volumetric content, image and depth data is captured from a number of cameras surrounding a capture space. The captured data is processed to create a full 3D representation of any object or person in the defined capture area. The reconstruction process uses algorithms to produce a set of 3D models that can then be played as a sequence. The following diagram shows the data capture and delivery requirements for both the small-area and large-area volumetric capture rig used in the boxing use case.

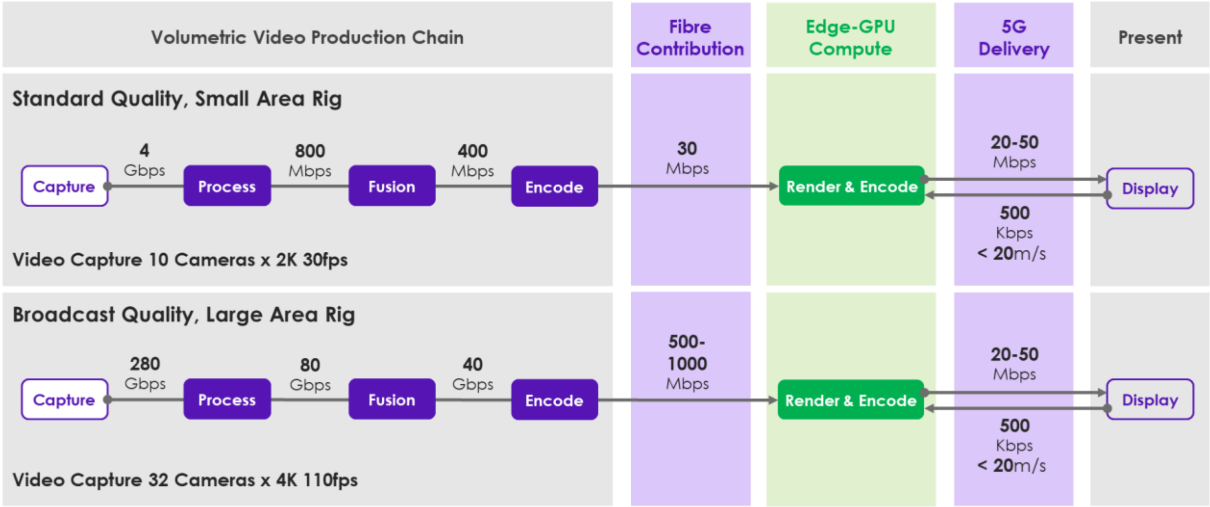

Figure 4 - Data flow for the Small and Large Area Volumetric Capture Rigs.

The small-area volumetric rig can capture an area of approximately 3.5m x 3.5m x 2m, using 8-10 Microsoft Azure Kinect cameras, recording data at 2K resolution and 30fps, creating raw data stream of approximately 4Gbps. The large-area volumetric rig can capture an area of approximately 8m x 8m x3m, using 32 Emergent Vision mounted in stereo pair format. These cameras can capture raw data at 4K resolution and 110fps at approximately 280Gbps. Despite the contribution of data to the cloud-GPU being compressed to under 1Gbps, without the use of cloud-rendering the presentation of high-resolution and high-vertices count (>60K vertices) volumetric video would not have been possible.

The volumetric ingest - the process by which raw image data is converted to volumetric video - is based on a real-time non-rigid registration [3]. As frames are captured, depth and RGB information is registered with the current state of the system, allowing the incremental improvement of the geometry and texture information in the output volumetric video. This reduces the influence of sensor noise on output surfaces, as well as reducing the impact of sensor occlusion. The raw output from the ingest process is large. An initial geometric simplification and texture encoding can be used prior to delivery to the Edge-GPU compute node to reduce on-site bandwidth requirements.

The latency from capture to contribution is currently around 3.5 seconds. This is about 5ms between capture and processing, 40ms between processing and fusion, 50ms between fusion and encoding, and the remaining time for encoding. We anticipate that in the future, encoding latency can be significantly reduced to around 10ms.

To enable a volumetric video broadcast to be editorially driven, production tools were developed to change the presentation of the augmented reality scene synchronised to the main TV broadcast.

Volumetric Audio

As well as providing a volumetric video capture system, the 5G Edge-XR project also developed an immersive audio capture and rendering engine, pioneered by Salsa Sound Ltd. Fundamental to rendering immersive and personalised audio is the creation of an object-based audio scene where all individual audio sources and ambience descriptions, with accompanying metadata, are kept separate right through the signal chain. The actual rendering is then performed, in this case, on the network edge for a personalised and immersive presentation with audio matching individual viewer's preferences and viewpoints.

For the 5G Edge-XR project, a cloud-based, AI-driven mixing engine has been developed [4] to automatically define and mix the object-based audio scene. One of the benefits of running audio analysis, processing, and mixing in the cloud is the increased processing power enabled by GPU acceleration. This increases what can be done with audio analysis enabling more complex processing tasks like real-time audio object extraction, localisation, and semantic analysis on the incoming streams.

Extracting Sound Sources/Objects

Capturing an object-based audio scene presents some challenges and requires some alterations in the way that content is traditionally created/mixed to ensure that the individual sounds at an event e.g., the sound of a punch in boxing, the racket strike in tennis or the sound of a ball being kicked etc. are detected and extracted as separate sources. The way this is done at the capture end depends somewhat upon the context and audio extraction techniques hence vary accordingly. Fundamentally however we employ machine learning techniques that analyse various representations of the input audio to learn complex patterns that later allow them to detect specific audio events [5]. Once detected, events are segmented into objects with metadata and/or added into a mix to create a field-of-play stem.

Source Localisation

In addition to extracting the audio content of the objects, it's important that individual sources are accurately localised in the scene and corresponding metadata authored. This means that as the viewer navigates their visual perspective in the scene, it is possible to correctly move sound sources around so that the audio presentation matches the visual presentation.

To facilitate this, we developed an optimised triangulation routine using the time difference of arrival (TDOA) between detected sources at different microphones. Determining the TDOAs can be a challenge using traditional algorithmic methods such as cross-correlation [6] due to the high background noise of live events. Instead, we use our source extraction methods to provide the accurate time stamp of the detected source in each microphone to determine the TDOAs across multiple microphones.

An efficient method of computing the location of the audio objects based on these TDOAs uses a brute-force optimisation method which is both computationally more efficient and also less prone to errors than traditional methods. To do this we split the target zone (pitch, court, ring etc.) up into a series of candidate source positions. Once this grid has been defined, we create an array of TDOAs at each microphone which would exhibit if the source were at each of the candidate locations (given the microphone coordinates) – this forms our search grid. Once a source has been detected in at least three microphones and we have a set of TDOAs, the algorithm compares the measured TDOAs with each combination in the search grid looking for the element with the minimum Euclidean distance/error which can be inferred as the most likely source position as shown in the search grid heatmap plot in Figure 5. This method is more efficient and robust to measurement errors and high background noise.

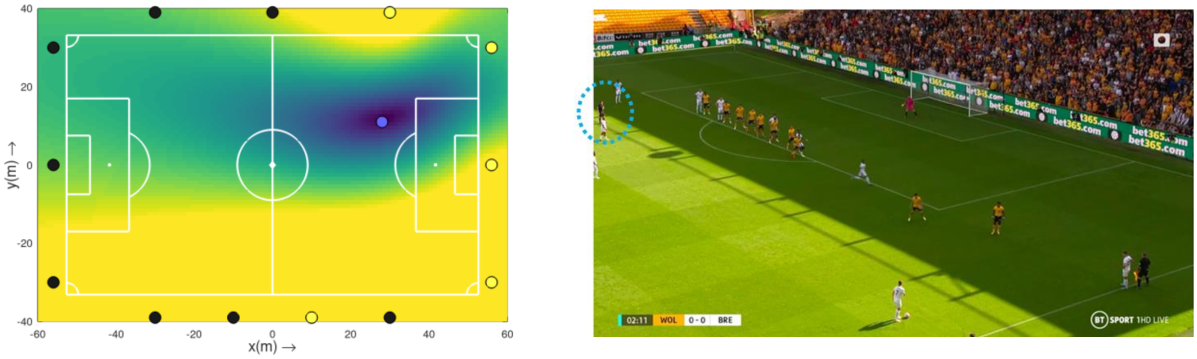

Figure 5 – Heatmap of source position probability (left), with video screen grab of actual source position (right) for an example referee whistle-blow. The black dots show microphones, yellow dots show those that detected the event. The blue dot shows the estimated source location.

The overall audio object extraction system diagram can hence be represented by Figure 6

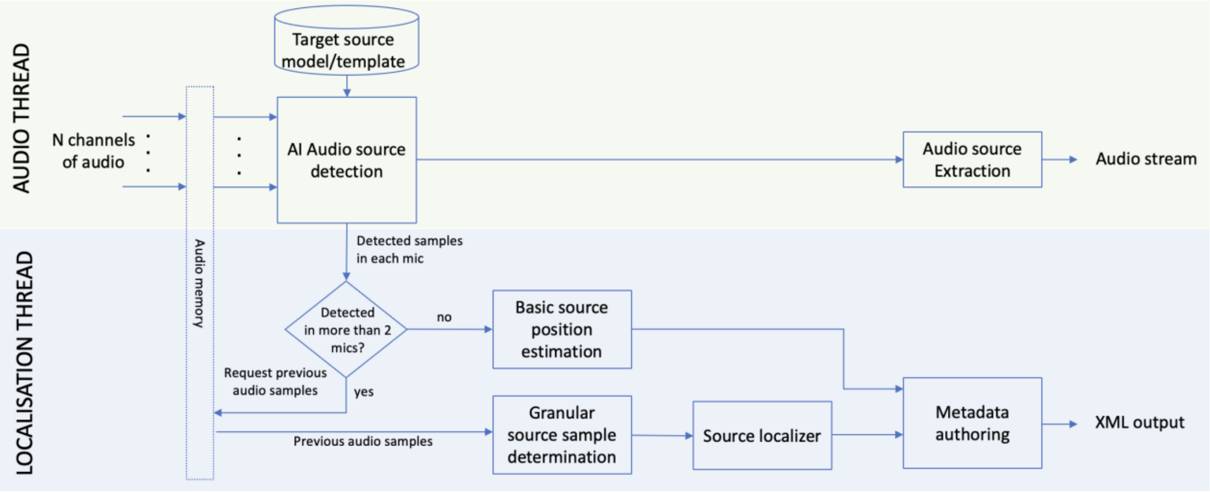
AUDIO THREAD
LOCALISATION THREAD
Target source model/template
N channels of audio
Audio memory
AI Audio source detection
Audio source Extraction
Audio stream
Detected samples in each mic
Detected in more than 2 mics?
no
yes
Request previous audio samples
Basic source position estimation
Previous audio samples
Granular source sample determination
Source localizer
Metadata authoring
XML output

Figure 6 - Source localisation workflow.

Creating audio metadata

Once the parameters have been extracted from the scene, a corresponding metadata feed is authored and streamed into the rendering engine so sources can be positioned and manipulated in the scene correctly. The continuous metadata stream includes localisation data, type of object detected and time stamp of the detection. Although in this project, this data is primarily used for spatial rendering of the audio scene, it is anticipated that this, (and other advanced audio data mining results like speech-to-text or crowd excitement data) holds value for other parts of the production workflow both for the 5G Edge-XR project and for current workflows e.g., triggering graphic/text overlays and/or highlight suggestions etc.

360 Degree Video

The 5G Edge-XR VR 360 Football experience uses 360-degree video to provide viewers with an ‘like being their’ immersive experience. We used existing video assets taken from BT Sport’s current live 8K VR 360 broadcast which is used within the BT Sport App. The video content was recorded prior to contribution in an 8K (7,680 horizontal and 4,320 vertical pixels) equirectangular video format. Some minor adjustments were made to the raw video assets to remove the jumbotron for use in the 5G Edge-XR prototype. Spatial audio was also recorded at each camera location using ambisonic microphones which enables an accurate 3D presentation of what was heard at that location.

Live Data

The use and presentation of live data within XR applications can offer sports fans insights that increase engagement and enjoyment of sports viewing. Within the 5G Edge-XR project data streams were heavily used in MotoGP where Dorna provided tracking and timing dataset and in Rugby where Sportable provided real-time ball tracking and match statistics.

USE CASES

The 5G Edge-XR project explored eight use cases in total. These use cases include applications for Sports Entrainment (Boxing, MotoGP, Stadium and Football), Architecture Engineering and Construction (AEC), Education (Dance), Health (Medical Imaging) and Retail. The vision illustration created for the sports broadcasting use cases are presented below, with descriptions that summarise the applications developed, tested, trialled.

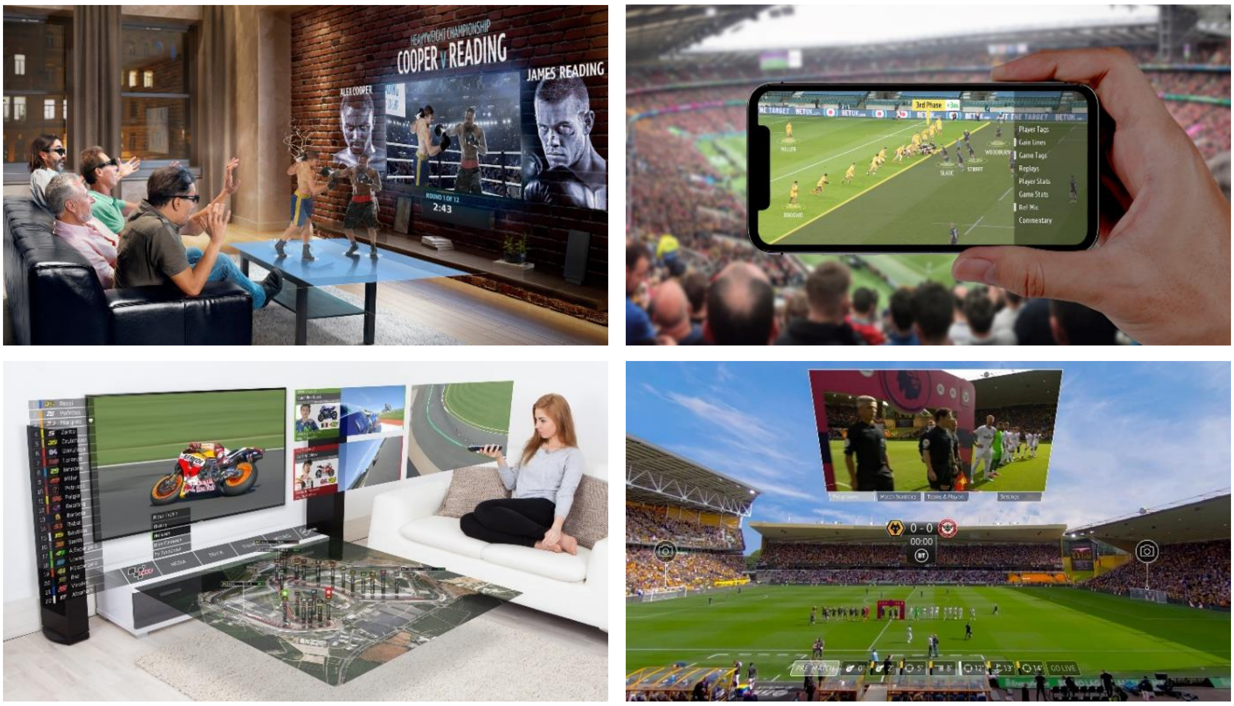

Figure 7 - 5G Edge-XR Sports Use Case Vision Illustrations, AR Boxing (top left), AR Rugby (top right), AR MotoGP (bottom left) and VR Football (bottom right).

AR MotoGP

The MotoGP use case is an AR immersive motorsports experience targeted at fans of MotoGP. It offers an immersive MotoGP Race Centre wrap-around video gallery with high-resolution 3D map showing the location of rides on the race circuit during the race and a Parc Ferme area that shows life-size virtual MotoGP bikes which fans can walk around and inspect.

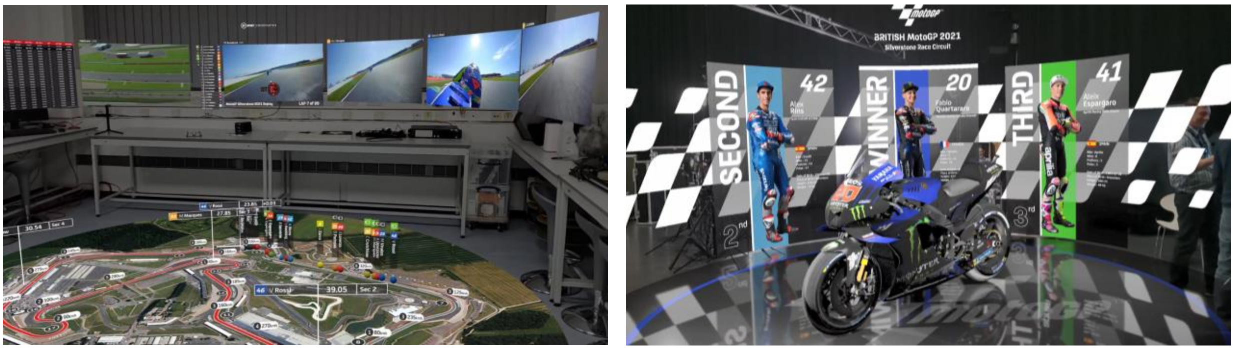

Figure 8 - MotoGP UX showing Race Centre gallery (left) and the Parc Ferme Area (right).

The MotoGP Race Centre experience provides an immersive video gallery that offers a view of up to 17 concurrent live race video feeds on virtual screens, a replay screen where race highlight clips can be viewed, and a timing screen which shows race timing statistics. An interactive leaderboard showing the placement of riders in the race is also provided. A virtual signer is also offered which is positioned in front of the video gallery and automatically moves to remain insight wherever the viewer is looking. Below the video gallery is a high-resolution room scale configurable 3D map of the racetrack showing the full circuit and position of all riders, track information and fastest sector times. The Parc Ferme experience enables fans to walk around three full-size virtual 3D MotoGP bikes (approx. 65K polygons per bike model) which use raytracing (mirror floor and shadows), particle systems (exhaust smoke) and spatial audio (engine idle audio) to provide a compelling life-like experience.

To demonstrate the experiential benefits of a CloudXR rendered application compared to a 'standard' client-side rendered application, two applications were built and tested. For the CloudXR app the number of available concurrent video streams was increased from 5 to 17 streams, the map texture was increased by a factor of twelve, from 4096px x 4096px to 16,384px x 12,288px. Similarly, a new Parc Ferme area was created for the CloudXR app to showcase high polygon 3D model rendering with raytracing and particle systems, which would not be possible within a client-side app.

VR 360 Degree Football

The VR 360-degree Football use case demonstrates a sports broadcaster driven live immersive stadium experience that enables sports fans to be virtually present at the match using immersive video and spatialised audio presented on Oculus Quest2 VR headsets, tablets, and smartphones. Utilising the existing 8K 360-degree production workflow of BT Sport, the 5G Edge-XR prototype demonstrated the benefits of edge-cloud rendering using CloudXR to deliver a low delay immersive video football experience.

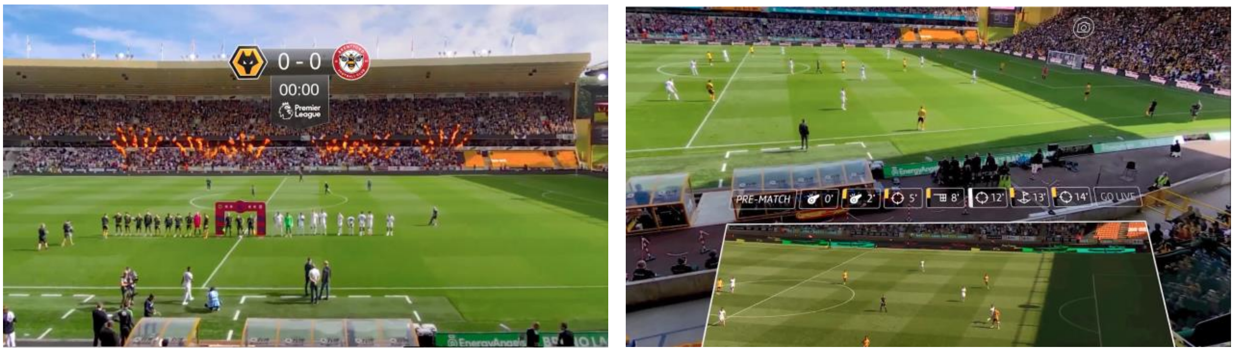

Figure 9 - VR 360 Degree Football User Interface.

The experience offers access to high-resolution 8K 360-degree immersive audio-visual streams from three camera locations within the stadium. A user-controlled zoom facility is offered enabling the 8K resolution to be fully exploited. The experience also supports personalised audio presentation where the ambient bed and foreground commentary audio levels can be changed by the user. Ambisonic audio has been used to faithfully capture and reproduce what is heard at each camera location, providing a spherical sound field which responds to changes in head rotation. This provides a fully audio-visual immersive experience that would be difficult to achieve without cloud rendering.

The prototype further offers users access to the editorial TV broadcast through an embedded interactive media panel, enabling viewers to watch replays with commentary within the immersive experience. Match statistics data and team information is also made available via the media panel. Cloud-render application exploits viewpoint tracking to enable the media panel to follow the orientation of user, so it always remains conveniently close at hand. An interactive timeline is provided which enables viewers to quickly jump to key events within the match and to re-join live. As the user jumps around the match timeline, the immersive 360-degree video and TV programme video stream presented on the media panel remain synchronised as does the embedded score clock which provides information on the competition, team score and match time.

AR Volumetric Boxing

The AR Boxing use case exploits volumetric video and spatial audio to offer fans an editorially driven live immersive augmented reality boxing experience that is streamed to a viewer's smartphone, tablet, or AR headset. The user experience created for a live broadcast boxing event was developed to demonstrate how volumetric video could enhance and complement the tradition TV editorial content created by broadcasters such as BT Sport.

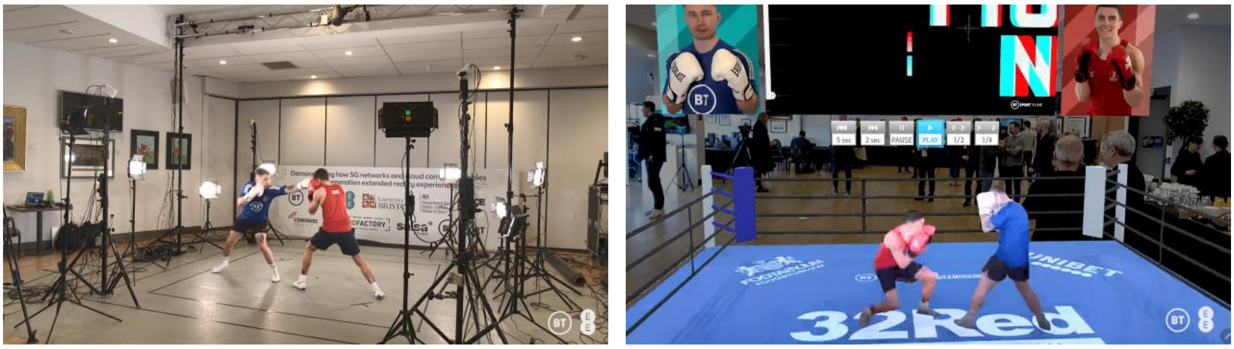

Figure 10 - AR Volumetric Boxing capture rig (left) and live in-App presentation (right).

The boxers are presented within a virtual ring as a volumetric video 'hologram' in sync with the live TV broadcast and commentary from BT Sport. The AR virtual ring sits below a virtual TV screen with boxer stats panels either side. Further virtual signage is provided showing the broadcaster brand, competition title, boxer names, together with a round indicator and round countdown timer. User controls for selecting rounds for replays is also provided.

During a live boxing round the broadcast TV feed, volumetric video and spatial audio are synchronised to provide an immersive enhanced experience. After the completion of a boxing round the presentation of volumetric video automatically changes to show a short highlight clip which has been defined by the production team. Replay clips are presented as a close-up view without the full ring being shown so the viewer can gain a better understanding of the key moments within that round. Options to rewind and change the playback speed are provided to support better analysis of the action.

The viewers experience an audio ‘bed’ with the main (background) audio feed but as they add/remove content and navigate within the scene a bespoke audio feed is overlaid which matches the visual representation. By isolating each audio source and localise it to the correct place in the scene, the sound of each punch and shout of the trainers can be panned to the correct location relative to the viewing angle of each viewer.

AR Rugby Stadium

The in-stadium rugby use case demonstrated an AR rugby experience for sports fans watching a live match within a stadium environment. The prototype used cloud-GPU rendering to deliver a live ‘through the lens’ AR view of the pitch on handheld mobile devices. Spectators from any location in the stand could hold up their mobile device to view statistics and information insights overlayed over the live action on the pitch.

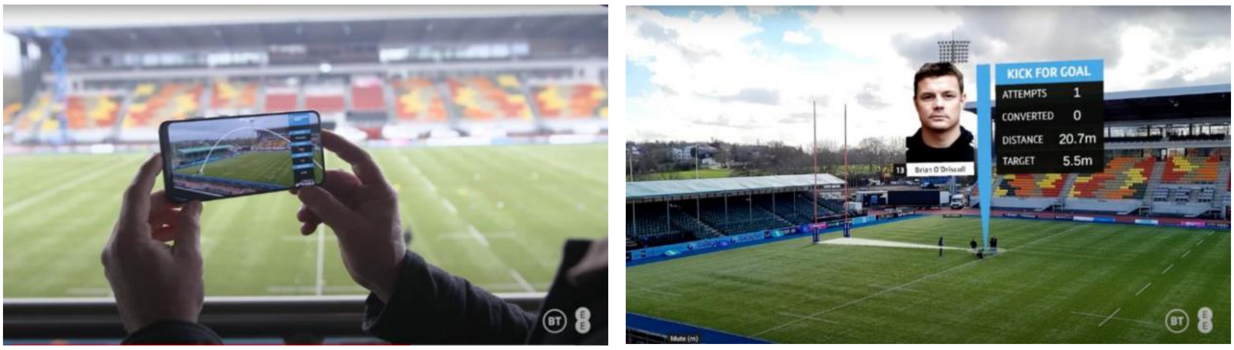

Figure 11 - AR Rugby Stadium demonstrator showing the presentation on a spectator’s phone.

We collaborated with Sportable to use their smart rugby ball tracking technology (developed in collaboration with Gilbert) that combined with their pitch side sensors provided a data feed for the ball location during live play. Sportable also provide a virtual pitch survey with co-ordinate points including all the key line points of the physical pitch. This data was used to re-create a virtual 2D pitch within a 3D scene, with smart ball co-ordinates defining the position of any virtual ball graphics that are rendered in the 3D space in relation to the location of the physical ball on the real pitch. To calibrate the spectator’s device to the stadium, a simple user interface was created which prompts the users to target all four corners of the pitch, to generate a custom virtual pitch calibration.

The Live Ball marker graphic provided the most impressive user experience as accuracy was good. Run Mode was hugely impactful for specific line breaks as all graphics appeared synchronised with the movement of the physical ball on the pitch. Kick Mode was the most 3D visually impactful of all the AR features due to the ability to render an accurate 3D arc of the kick trajectory. Kick for Posts mode was one of many examples of breaks in play (e.g., scrums and lineouts) where specific pitch locations (based on ball position) can be used to drop flag markers to contextualise match statistics.

OUTCOMES & RESULTS

Following the test and trial of the use cases described in the paper, several key insights have emerged that indicate how future cloud rendered XR experiences could be facilitated.

Architecture

Technically we found the Cloud-GPU architecture and CloudXR solution worked well and could deliver compelling XR experiences with higher visual fidelity than would be achievable on consumer mobile devices. However, CloudXR requires each user to have their own VM which could be considered inefficient when many users in a sport broadcasting context are viewing the same scene from their own viewpoint. An architecture that enables many (hundreds or thousands) virtual cameras to be used in a single scene might be more resource efficient if the number of concurrent encoding streams could be increased.

In order to automate the start-up and close-down of applications a Broker Service was developed to control the state of VMs, SteamVR and OpenVR applications. A NGINX Media Cache was also created on the GPU cluster to serve cached content to multiple XR apps running the same experience, thereby reducing the number of external CDN server streams.

Production & Delivery

A key difference between an XR broadcast and a traditional video broadcast is the use of a 3D XR scene presented as an interactive immersive environment. The production of this scene can be lengthy but can offer a rights holder or broadcaster with new opportunities to engage audiences and monetise participatory interaction.

Contribution of an XR scene to a Content Distribution Network (CDN) of cloud-GPU streaming servers, will typically take a similar amount of time as traditional video contribution up to a CDN. The exception to this is volumetric video, where currently the latency from capture to contribution is around 3.5 seconds. However, we believe this could be slashed to around 100ms using faster encoding techniques. Delivery of cloud-rendered XR experiences requires real-time streaming, meaning that scenes need to be rendered on cloud-GPU, encoded, transmitted to the client, decoded, and displayed in less than 30ms. So, if we compare the glass-to-glass delivery latency of a cloud-rendered XR broadcast with a traditional broadcast video pipeline, we see that the production and contribution legs up to a CDN are roughly equivalent, but the delivery leg is significantly faster for cloud-rendered XR broadcasts. Fundamentally, the use of CloudXR negates the need for a CDN to encode video at different bitrates or manage the delivery of segmented video using HLS, DASH, etc.

Through trials we found that real-time rendered XR can be delivered with CloudXR encoding streams at between 15Mbps to 50Mbps, depending on the framerate, resolution, and the number of streams required by the client (stereo streams for XR headsets). Both 5G and WiFi networks were found to support round trip latency times that were sufficiently low (<30ms) so that users did not report feelings of nausea induced by excessive latency.

Experiences

When comparing the results of user trials undertaken within the project with current solutions deployed it is apparent that the cloud-GPU architecture does enable exciting use case capabilities that are not possible to deliver using traditional client-side rendering methods. The provision of multi-stream video galleries, high-framerate high-polygon volumetric models, 8K 360 video, and GPU intensive rendering effects such as ray tracing and particle effects demonstrate the power of cloud-GPU rendering.

The OpenVR experiences created were easily deployable across a variety of client device types including phones, tablets, VR headsets and AR glasses that use iOS or Android OS. Of particular note was the extremely positive feedback on the use of consumer AR headsets for viewing XR sports broadcasts which proved to be very acceptable in terms of user immersion and comfort for long-form content viewing.

Business Model

Results from studying the business model for this kind of technology suggests that there is still a lot of work to do. For the broadcast centric XR experiences we developed, we found the CloudXR architecture (using NVIDIA RTX8000 GPUs) could support up to four simultaneous users per GPU, or twelve simultaneous users per server. Scaling this to a service to support say 10,000 simultaneous users for a live event, would be expensive.

The current cost basis for licensing of a cloud-GPU cluster (Windows-based, using VMware and CloudXR) for the XR sports use cases described in the paper is high. High costs can be OK if they generate high revenues. However, fundamentally, the usage of cloud-GPU needs to be diversified to increase utilisation and to amortise costs, thereby reducing the price per user. Furthermore, the number of simultaneous users on a GPU needs to be increased which might likely be achieved through a split-rendering architecture, optimised XR application design, flexibility in specification of VMs or using more powerful GPU cards.

CONCLUSIONS

This paper has described the work undertaken in the UK DCMS funded 5G Edge-XR project which explored how immersive live sports XR experiences could be delivered to viewers consumer devices using cloud-GPU compute, CloudXR and 5G networks. Findings from the project suggest that cloud-GPU can offer compelling real-time XR experiences that are unattainable on consumer hardware alone. Although the scalability of CloudXR has yet to be proven for use in mass consumer audience broadcasting, the outlook for cloud-rendering looks promising and will likely be key to enable next-generation immersive XR experiences.

REFERENCES

1. 5G Edge-XR project website. https://www.5gedgexr.com, Accessed July 2022
2. BT and EE redefine immersive experiences with 5G. March 2022. Available at YouTube https://youtu.be/0LHwXxVhnLE
3. Deng, B., Yao, Y., Dyke, R.M. and Zhang, J. 2022, A Survey of Non-Rigid 3D Registration. Computer Graphics Forum, 41, 559-589.
4. Oldfield, R., and Walley, M., et al., 2021. Cloud-based AI for Automatic Audio Production for Personalised Immersive XR Experiences. Proceedings of the International Broadcasting Convention. December 2021.
5. Oldfield, R., Shirley, B., & Spille, J., 2015. Object-based audio for interactive football broadcast. *Multimedia Tools and Applications*, *74*(8), 2717-2741.
6. Knapp, C. and Carter, G., 1976. The generalized correlation method for estimation of time delay. IEEE transactions on acoustics, speech, and signal processing, 24(4), pp.320-327.

ACKNOWLEDGEMENTS

The authors would like to thank all the partners involved in the 5G Edge-XR project for their contributions to this work as well as BT Sport, Dorna, Nreal, Sportable and the University of Strathclyde. We would also like to thank the International Broadcasting Convention for permission to publish this paper.