

MOPS
Internet-Draft
Intended status: Informational
Expires: 7 September 2022

R. Krishna
InterDigital Europe Limited
A. Rahman
InterDigital Communications, LLC
6 March 2022

Media Operations Use Case for an Augmented Reality Application on Edge
Computing Infrastructure
draft-ietf-mops-ar-use-case-04

Abstract

This document explores the issues involved in the use of Edge Computing resources to operationalize media use cases that involve Extended Reality (XR) applications. In particular, we discuss those applications that run on devices having different form factors and need Edge computing resources to mitigate the effect of problems such as a need to support interactive communication requiring low latency, limited battery power, and heat dissipation from those devices. The intended audience for this document are network operators who are interested in providing edge computing resources to operationalize the requirements of such applications.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 7 September 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

Internet-Draft

MOPS AR Use Case

March 2022

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.

Table of Contents

1.	Introduction	2
2.	Conventions used in this document	3
3.	Use Case	3
3.1.	Processing of Scenes	4
3.2.	Generation of Images	5
4.	Requirements	5
5.	AR Network Traffic and Interaction with TCP	8
6.	Informative References	8
	Authors' Addresses	12

[1.](#) Introduction

Extended Reality (XR) is a term that includes Augmented Reality (AR), Virtual Reality (VR) and Mixed Reality (MR) [[XR](#)]. AR combines the real and virtual, is interactive and is aligned to the physical world of the user [[AUGMENTED 2](#)]. On the other hand, VR places the user inside a virtual environment generated by a computer [[AUGMENTED](#)]. MR merges the real and virtual world along a continuum that connects completely real environment at one end to a completely virtual environment at the other end. In this continuum, all combinations of the real and virtual are captured [[AUGMENTED](#)].

XR applications will bring several requirements for the network and the mobile devices running these applications. Some XR applications such as AR require a real-time processing of video streams to recognize specific objects. This is then used to overlay information on the video being displayed to the user. In addition XR applications such as AR and VR will also require generation of new video frames to be played to the user. Both the real-time processing of video streams and the generation of overlay information are computationally intensive tasks that generate heat [[DEV HEAT 1](#)], [[DEV HEAT 2](#)] and drain battery power [[BATT DRAIN](#)] on the mobile

device running the XR application. Consequently, in order to run applications with XR characteristics on mobile devices, computationally intensive tasks need to be offloaded to resources provided by Edge Computing.

Edge Computing is an emerging paradigm where computing resources and storage are made available in close network proximity at the edge of the Internet to mobile devices and sensors [[EDGE_1](#)], [[EDGE_2](#)]. These edge computing devices use cloud technologies that enable them to support offloaded XR applications. In particular, the edge devices deploy cloud computing implementation techniques such as disaggregation (breaking vertically integrated systems into independent components with open interfaces using SDN), virtualization (being able to run multiple independent copies of those components such as SDN Controller apps, Virtual Network Functions on a common hardware platform) and commoditization (being able to elastically scale those virtual components across commodity hardware as the workload dictates) [[EDGE_3](#)]. Such techniques enable XR applications requiring low-latency and high bandwidth to be delivered by mini-clouds running on proximate edge devices

In this document, we discuss the issues involved when edge computing resources are offered by network operators to operationalize the requirements of XR applications running on devices with various form factors. Examples of such form factors include Head Mounted Displays (HMD) such as Optical-see through HMDs and video-see-through HMDs and Hand-held displays. Smart phones with video cameras and GPS are another example of such devices. These devices have limited battery capacity and dissipate heat when running. Besides as the user of these devices moves around as they run the XR application, the wireless latency and bandwidth available to the devices fluctuates and the communication link itself might fail. As a result algorithms such as those based on adaptive-bit-rate techniques that base their policy on heuristics or models of deployment perform sub-optimally in such dynamic environments.[[ABR_1](#)]. We motivate these issues with a use-case that we present in the following sections.

[2.](#) Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this

document are to be interpreted as described in [[RFC2119](#)].

[3.](#) Use Case

We now describe a use case that involves an application with AR systems' characteristics. Consider a group of tourists who are being conducted in a tour around the historical site of the Tower of London. As they move around the site and within the historical buildings, they can watch and listen to historical scenes in 3D that are generated by the AR application and then overlaid by their AR headsets onto their real-world view. The headset then continuously updates their view as they move around.

Krishna & Rahman

Expires 7 September 2022

[Page 3]

Internet-Draft

MOPS AR Use Case

March 2022

The AR application first processes the scene that the walking tourist is watching in real-time and identifies objects that will be targeted for overlay of high resolution videos. It then generates high resolution 3D images of historical scenes related to the perspective of the tourist in real-time. These generated video images are then overlaid on the view of the real-world as seen by the tourist.

We now discuss this processing of scenes and generation of high resolution images in greater detail.

[3.1.](#) Processing of Scenes

The task of processing a scene can be broken down into a pipeline of three consecutive subtasks namely tracking, followed by an acquisition of a model of the real world, and finally registration [[AUGMENTED](#)].

Tracking: This includes tracking of the three dimensional coordinates and six dimensional pose (coordinates and orientation) of objects in the real world[[AUGMENTED](#)]. The AR application that runs on the mobile device needs to track the pose of the user's head, eyes and the objects that are in view. This requires tracking natural features that are then used in the next stage of the pipeline.

Acquisition of a model of the real world: The tracked natural features are used to develop an annotated point cloud based model that is then stored in a database. To ensure that this database can be scaled up, techniques such as combining a client side simultaneous tracking and mapping and a server-side localization are used[[SLAM_1](#)],

[[SLAM_2](#)], [[SLAM_3](#)], [[SLAM_4](#)].

Registration: The coordinate systems, brightness, and color of virtual and real objects need to be aligned in a process called registration [[REG](#)]. Once the natural features are tracked as discussed above, virtual objects are geometrically aligned with those features by geometric registration. This is followed by resolving occlusion that can occur between virtual and the real objects [[OCCL_1](#)], [[OCCL_2](#)]. The AR application also applies photometric registration [[PHOTO_REG](#)] by aligning the brightness and color between the virtual and real objects. Additionally, algorithms that calculate global illumination of both the virtual and real objects [[GLB_ILLUM_1](#)], [[GLB_ILLUM_2](#)] are executed. Various algorithms to deal with artifacts generated by lens distortion [[LENS_DIST](#)], blur [[BLUR](#)], noise [[NOISE](#)] etc are also required.

[3.2.](#) Generation of Images

The AR application must generate a high-quality video that has the properties described in the previous step and overlay the video on the AR device's display- a step called situated visualization. This entails dealing with registration errors that may arise, ensuring that there is no visual interference [[VIS_INTERFERE](#)], and finally maintaining temporal coherence by adapting to the movement of user's eyes and head.

[4.](#) Requirements

The components of AR applications perform tasks such as real-time generation and processing of high-quality video content that are computationally intensive. As a result, on AR devices such as AR glasses excessive heat is generated by the chip-sets that are involved in the computation [[DEV_HEAT_1](#)], [[DEV_HEAT_2](#)]. Additionally, the battery on such devices discharges quickly when running such applications [[BATT_DRAIN](#)].

A solution to the heat dissipation and battery drainage problem is to offload the processing and video generation tasks to the remote

cloud. However, running such tasks on the cloud is not feasible as the end-to-end delays must be within the order of a few milliseconds. Additionally, such applications require high bandwidth and low jitter to provide a high QoE to the user. In order to achieve such hard timing constraints, computationally intensive tasks can be offloaded to Edge devices.

Another requirement for our use case and similar applications such as 360 degree streaming is that the display on the AR/VR device should synchronize the visual input with the way the user is moving their head. This synchronization is necessary to avoid motion sickness that results from a time-lag between when the user moves their head and when the appropriate video scene is rendered. This time lag is often called "motion-to-photon" delay. Studies have shown [[PER_SENSE](#)], [[XR](#)], [[OCCL_3](#)] that this delay can be at most 20ms and preferably between 7-15ms in order to avoid the motion sickness problem. Out of these 20ms, display techniques including the refresh rate of write displays and pixel switching take 12-13ms [[OCCL_3](#)], [[CLOUD](#)]. This leaves 7-8ms for the processing of motion sensor inputs, graphic rendering, and RTT between the AR/VR device and the Edge. The use of predictive techniques to mask latencies has been considered as a mitigating strategy to reduce motion sickness

[[PREDICT](#)]. In addition, Edge Devices that are proximate to the user might be used to offload these computationally intensive tasks. Towards this end, the 3GPP requires and supports an Ultra Reliable Low Latency of 0.1ms to 1ms for communication between an Edge server and User Equipment(UE) [[URLLC](#)].

Note that the Edge device providing the computation and storage is itself limited in such resources compared to the Cloud. So, for example, a sudden surge in demand from a large group of tourists can overwhelm that device. This will result in a degraded user experience as their AR device experiences delays in receiving the video frames. In order to deal with this problem, the client AR applications will need to use Adaptive Bit Rate (ABR) algorithms that choose bit-rates policies tailored in a fine-grained manner to the resource demands and playback the videos with appropriate QoE metrics as the user moves around with the group of tourists.

However, heavy-tailed nature of several operational parameters make prediction-based adaptation by ABR algorithms sub-optimal[[ABR_2](#)]. This is because with such distributions, law of large numbers works too slowly, the mean of sample does not equal the mean of distribution, and as a result standard deviation and variance are unsuitable as metrics for such operational parameters [[HEAVY TAIL 1](#)], [[HEAVY TAIL 2](#)]. Other subtle issues with these distributions include the "expectation paradox" [[HEAVY TAIL 1](#)] where the longer we have

waited for an event the longer we have to wait and the issue of mismatch between the size and count of events [[HEAVY TAIL 1](#)]. This makes designing an algorithm for adaptation error-prone and challenging. Such operational parameters include but are not limited to buffer occupancy, throughput, client-server latency, and variable transmission times. In addition, edge devices and communication links may fail and logical communication relationships between various software components change frequently as the user moves around with their AR device [[UBICOMP](#)].

Thus, once the offloaded computationally intensive processing is completed on the Edge Computing, the video is streamed to the user with the help of an ABR algorithm which needs to meet the following requirements [[ABR 1](#)]:

- * Dynamically changing ABR parameters: The ABR algorithm must be able to dynamically change parameters given the heavy-tailed nature of network throughput. This, for example, may be accomplished by AI/ML processing on the Edge Computing on a per client or global basis.
- * Handling conflicting QoE requirements: QoE goals often require high bit-rates, and low frequency of buffer refills. However in practice, this can lead to a conflict between those goals. For example, increasing the bit-rate might result in the need to fill up the buffer more frequently as the buffer capacity might be limited on the AR device. The ABR algorithm must be able to handle this situation.
- * Handling side effects of deciding a specific bit rate: For example, selecting a bit rate of a particular value might result in the ABR algorithm not changing to a different rate so as to ensure a non-fluctuating bit-rate and the resultant smoothness of video quality. The ABR algorithm must be able to handle this situation.

In addition to the requirements for ABR algorithms, there are other operational issues that need to be considered for AR use cases such as the one described above. In a study [[AR_TRAFFIC](#)] conducted to characterize multi-user AR over cellular networks, the following issues were identified:

- * The uploading of data from an AR device to a remote server for processing dominates the end-to-end latency.
- * A lack of visual features in the grid environment can cause increased latencies as the AR device uploads additional visual data for processing to the remote server.
- * AR applications tend to have large bursts that are separated by significant time gaps. As a result, the TCP congestion window enters slow start before the large bursts of data arrive increasing the perceived user latency. The study [[AR_TRAFFIC](#)] shows that segmentation latency at 4G LTE (Long Term Evolution)'s RAN (Radio Access Network)'s RLC (Radio Link Control) layer impacts TCP's performance during slow-start.

[6.](#) Informative References

- [ABR_1] Mao, H., Netravali, R., and M. Alizadeh, "Neural Adaptive Video Streaming with Pensieve", In Proceedings of the Conference of the ACM Special Interest Group on Data Communication, pp. 197–210, 2017.
- [ABR_2] Yan, F., Ayers, H., Zhu, C., Fouladi, S., Hong, J., Zhang, K., Levis, P., and K. Winstein, "Learning in situ: a randomized experiment in video streaming", In 17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20), pp. 495–511, 2020.
- [AR_TRAFFIC] Apicharttrisor, K., Balasubramanian, B., Chen, J., Sivaraj, R., Tsai, Y., Jana, R., Krishnamurthy, S., Tran, T., and Y. Zhou, "Characterization of Multi-User Augmented Reality over Cellular Networks", In 17th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), pp. 1–9. IEEE, 2020.
- [AUGMENTED] Schmalstieg, D. S. and T.H. Hollerer, "Augmented Reality", Addison Wesley, 2016.

[AUGMENTED_2]

Azuma, R. T., "A Survey of Augmented Reality.", Presence:Teleoperators and Virtual Environments 6.4, pp. 355-385., 1997.

[BATT_DRAIN]

Seneviratne, S., Hu, Y., Nguyen, T., Lan, G., Khalifa, S., Thilakarathna, K., Hassan, M., and A. Seneviratne, "A survey of wearable devices and challenges.", In IEEE Communication Surveys and Tutorials, 19(4), p.2573-2620., 2017.

[BLUR]

Kan, P. and H. Kaufmann, "Physically-Based Depth of Field in Augmented Reality.", In Eurographics (Short Papers), pp. 89-92., 2012.

[CLOUD]

Corneo, L., Eder, M., Mohan, N., Zavodovski, A., Bayhan, S., Wong, W., Gunningberg, P., Kangasharju, J., and J. Ott, "Surrounded by the Clouds: A Comprehensive Cloud Reachability Study.", In Proceedings of the Web Conference 2021, pp. 295-304, 2021.

[DEV_HEAT_1]

LiKamWa, R., Wang, Z., Carroll, A., Lin, F., and L. Zhong, "Draining our Glass: An Energy and Heat characterization of Google Glass", In Proceedings of 5th Asia-Pacific Workshop on Systems pp. 1-7, 2013.

[DEV_HEAT_2]

Matsushashi, K., Kanamoto, T., and A. Kurokawa, "Thermal model and countermeasures for future smart glasses.", In Sensors, 20(5), p.1446., 2020.

[EDGE_1]

Satyanarayanan, M., "The Emergence of Edge Computing", In Computer 50(1) pp. 30-39, 2017.

[EDGE_2]

Satyanarayanan, M., Klas, G., Silva, M., and S. Mangiante, "The Seminal Role of Edge-Native Applications", In IEEE International Conference on Edge Computing (EDGE) pp. 33-40, 2019.

[EDGE_3]

Peterson, L. and O. Sunay, "5G mobile networks: A systems approach.", In Synthesis Lectures on Network Systems., 2020.

Internet-Draft

MOPS AR Use Case

March 2022

[GLB_ILLUM_1]

Kan, P. and H. Kaufmann, "Differential irradiance caching for fast high-quality light transport between virtual and real worlds.", In IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 133-141, 2013.

[GLB_ILLUM_2]

Franke, T., "Delta voxel cone tracing.", In IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 39-44, 2014.

[HEAVY_TAIL_1]

Crovella, M. and B. Krishnamurthy, "Internet measurement: infrastructure, traffic and applications", John Wiley and Sons Inc., 2006.

[HEAVY_TAIL_2]

Taleb, N., "The Statistical Consequences of Fat Tails", STEM Academic Press, 2020.

[I-D.ietf-mops-streaming-opcons]

Holland, J., Begen, A., and S. Dawkins, "Operational Considerations for Streaming Media", Work in Progress, Internet-Draft, [draft-ietf-mops-streaming-opcons-09](https://datatracker.ietf.org/doc/html/draft-ietf-mops-streaming-opcons-09), 1 March 2022, <<https://datatracker.ietf.org/doc/html/draft-ietf-mops-streaming-opcons-09>>.

[LENS_DIST]

Fuhrmann, A. and D. Schmalstieg, "Practical calibration procedures for augmented reality.", In Virtual Environments 2000, pp. 3-12. Springer, Vienna, 2000.

[NOISE]

Fischer, J., Bartz, D., and W. Straßer, "Enhanced visual realism by incorporating camera image effects.", In IEEE/ACM International Symposium on Mixed and Augmented Reality, pp. 205-208., 2006.

[OCCL_1]

Breen, D.E., Whitaker, R.T., and M. Tuceryan, "Interactive Occlusion and automatic object placement for augmented

reality", In Computer Graphics Forum, vol. 15, no. 3 , pp. 229-238,Edinburgh, UK: Blackwell Science Ltd, 1996.

- [OCCL_2] Zheng, F., Schmalstieg, D., and G. Welch, "Pixel-wise closed-loop registration in video-based augmented reality", In IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. 135-143, 2014.

- [OCCL_3] Lang, B., "Oculus Shares 5 Key Ingredients for Presence in Virtual Reality.", <https://www.roadtovr.com/oculus-shares-5-key-ingredients-for-presence-in-virtual-reality/>, 2014.

- [PER_SENSE] Mania, K., Adelstein, B.D., Ellis, S.R., and M.I. Hill, "Perceptual sensitivity to head tracking latency in virtual environments with varying degrees of scene complexity.", In Proceedings of the 1st Symposium on Applied perception in graphics and visualization pp. 39-47., 2004.

- [PHOTO_REG] Liu, Y. and X. Granier, "Online tracking of outdoor lighting variations for augmented reality with moving cameras", In IEEE Transactions on visualization and computer graphics, 18(4), pp.573-580, 2012.

- [PREDICT] Buker, T. J., Vincenzi, D.A., and J.E. Deaton, "The effect of apparent latency on simulator sickness while using a see-through helmet-mounted display: Reducing apparent latency with predictive compensation..", In Human factors 54.2, pp. 235-249., 2012.

- [REG] Holloway, R. L., "Registration error analysis for augmented reality.", In Presence:Teleoperators and Virtual Environments 6.4, pp. 413-432., 1997.

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997,

<<https://www.rfc-editor.org/info/rfc2119>>.

- [SLAM_1] Ventura, J., Arth, C., Reitmayr, G., and D. Schmalstieg, "A minimal solution to the generalized pose-and-scale problem", In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 422-429, 2014.
- [SLAM_2] Sweeny, C., Fragoso, V., Hollerer, T., and M. Turk, "A scalable solution to the generalized pose and scale problem", In European Conference on Computer Vision, pp. 16-31, 2014.

- [SLAM_3] Gauglitz, S., Sweeny, C., Ventura, J., Turk, M., and T. Hollerer, "Model estimation and selection towards unconstrained real-time tracking and mapping", In IEEE transactions on visualization and computer graphics, 20(6), pp. 825-838, 2013.
- [SLAM_4] Pirchheim, C., Schmalstieg, D., and G. Reitmayr, "Handling pure camera rotation in keyframe-based SLAM", In 2013 IEEE international symposium on mixed and augmented reality (ISMAR), pp. 229-238, 2013.
- [UBICOMP] Bardram, J. and A. Friday, "Ubiquitous Computing Systems", In Ubiquitous Computing Fundamentals pp. 37-94. CRC Press, 2009.
- [URLLC] 3GPP, "3GPP TR 23.725: Study on enhancement of Ultra-Reliable Low-Latency Communication (URLLC) support in the 5G Core network (5GC).", <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3453>, 2019.
- [VIS_INTERFERE] Kalkofen, D., Mendez, E., and D. Schmalstieg, "Interactive focus and context visualization for augmented reality.", In 6th IEEE and ACM International Symposium on Mixed and

Augmented Reality, pp. 191-201., 2007.

[XR] 3GPP, "3GPP TR 26.928: Extended Reality (XR) in 5G.",
<https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3534>, 2020.

Authors' Addresses

Renan Krishna
InterDigital Europe Limited
64, Great Eastern Street
London
EC2A 3QR
United Kingdom
Email: renan.krishna@interdigital.com

Akbar Rahman
InterDigital Communications, LLC
1000 Sherbrooke Street West
Montreal H3A 3G4
Canada
Email: Akbar.Rahman@InterDigital.com