Performance of Augmented Reality Remote Rendering via Mobile Network

K. M. Börner[1]

A. Fuhrmann[2]

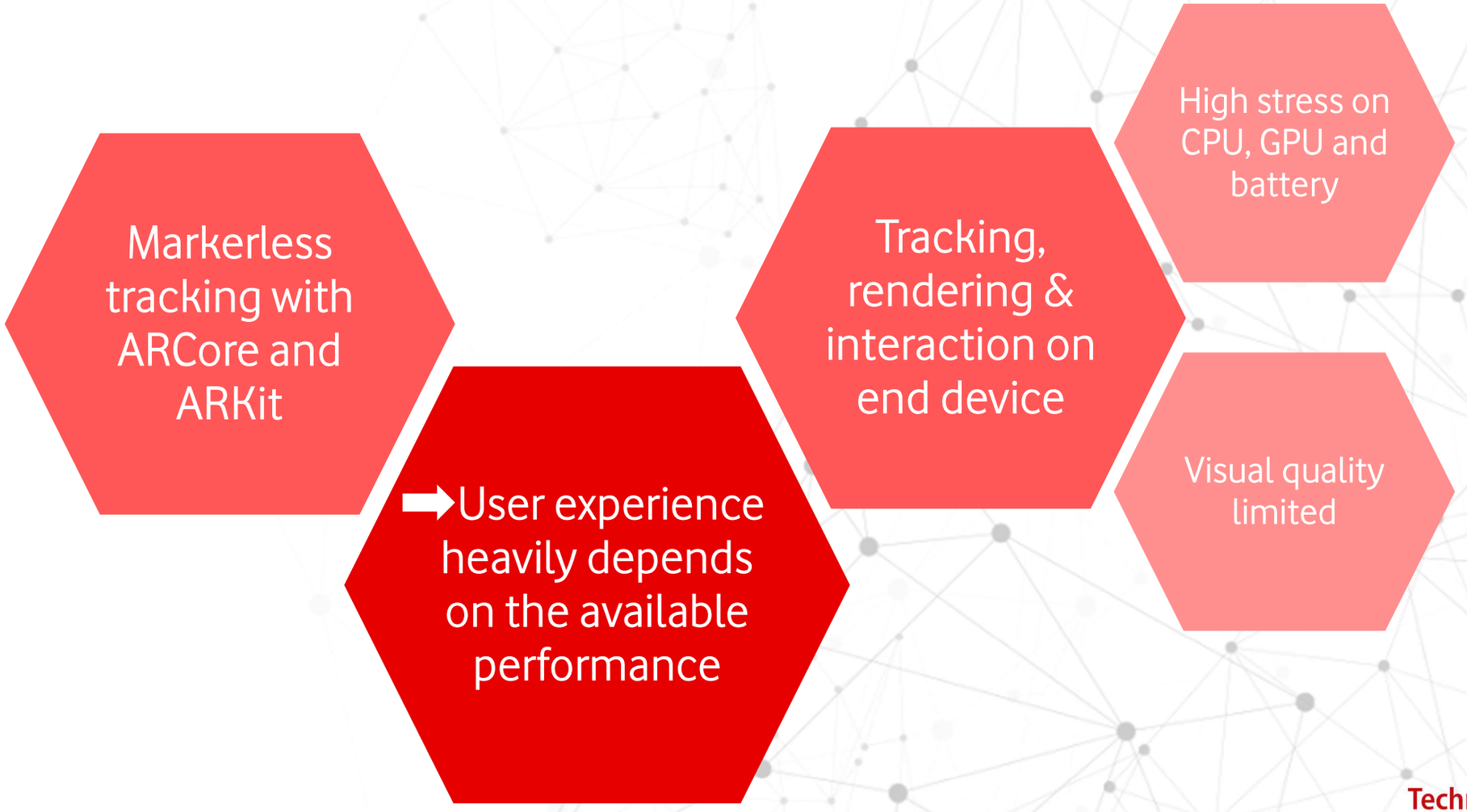
M. A. Bösinger[1]

[1]Vodafone GmbH, Germany
{kaimanuel.boerner, michaelandreas.boesinger}@vodafone.com
[2]TH Köln, Germany
arnulph.fuhrmann@th-koeln.de

VAR
Technology
Arts Sciences
TH Köln

1 Introduction

Motivation



Technology Arts Sciences TH Köln

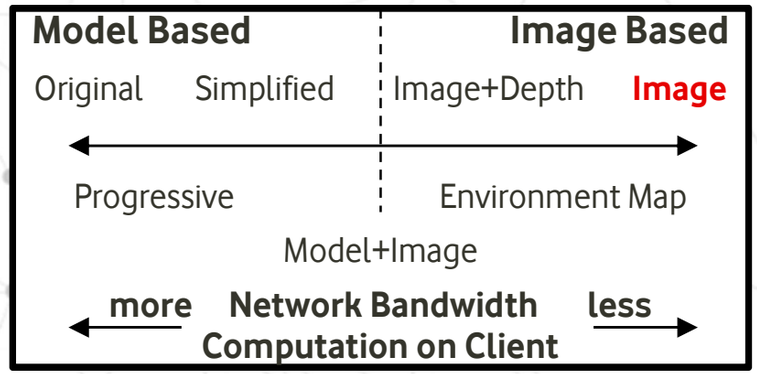
Idea

Move rendering and interaction to server

Usage of Remote-Rendering

(Mobile) Edge Computing System

Implement Remote-Rendering procedure for Unity AR Apps in order to increase visual quality



cf. [SHI15]

Technology Arts Sciences TH Köln

2

Related Work

Technology
Arts Sciences
TH Köln

Remote-Rendering

Early work on remote rendering was already performed over twenty years ago [Lev95, HS98].
Two key challenges are described by Shi and Hsu [SH15] in their survey:

- **The interaction latency**
  - time between the user interaction and the display of the result
  - Image-based approaches are more affected
  - Usage of depth map and warping can reduce latency, but artifacts can occur [SJNC09, CHC17].
- **Limitations of networks such as the bandwidth**

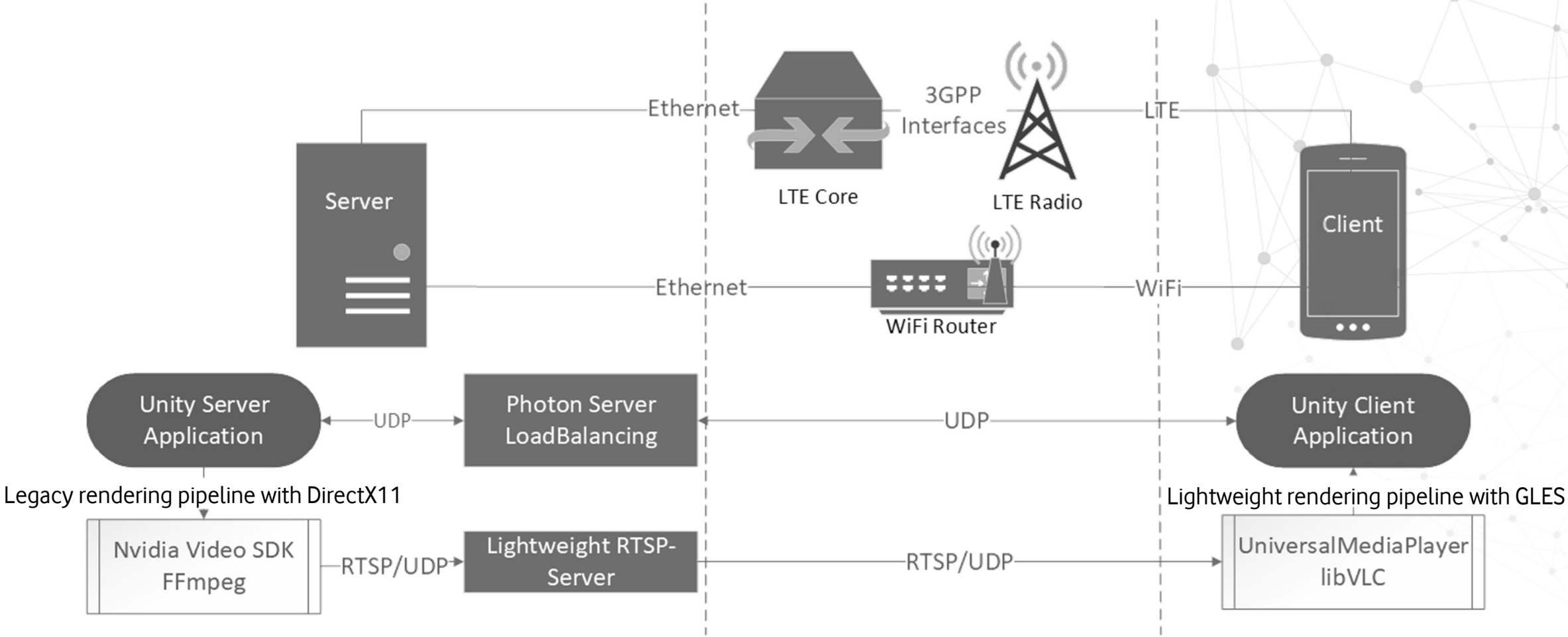
Mobile Edge Computing

- In MEC, the server is part of the providers network
  - Ideally at the mobile network tower
- Bohez et al. [BTV+13] implemented a middleware platform for collaborative applications
  - data together with its processing are shared between multiple user.
- Latency of markerless tracking and object recognition can be reduced [ZHH18]
  - offloading computation intensive tasks to the edge cloud

3

System Overview

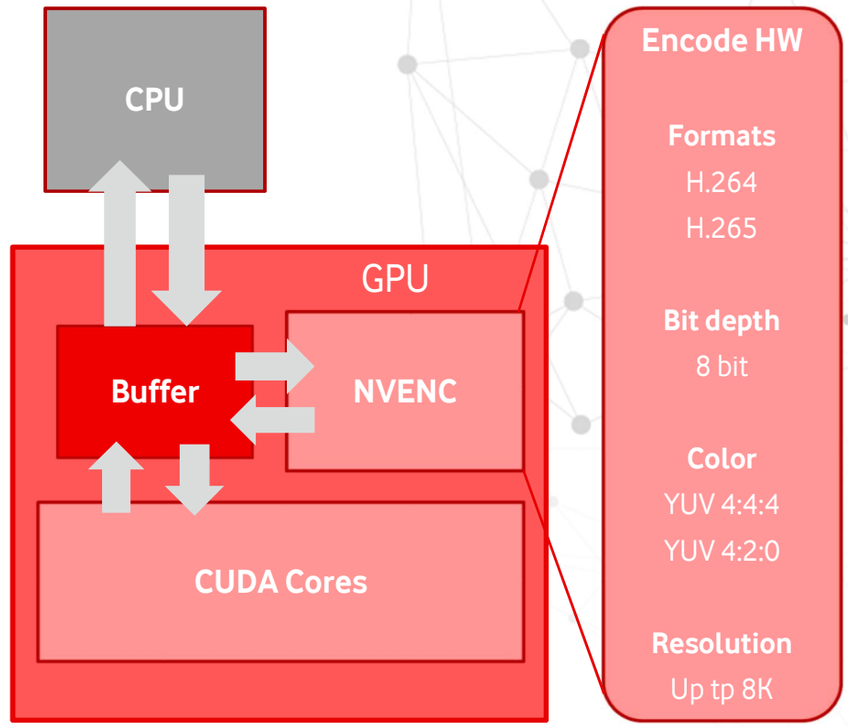
System Pipeline



Technology Arts Sciences TH Köln

Video Encoding & Chroma Keying

- Server renders in texture
- Copy texture to NVENC
  - Use native API, for example DirectX11
- Encoding on GPU
  - No copy of texture between CPU and GPU
- No transparency in AVC & HEVC
  - Client blends transmitted video in front of camera frame
  - „Real world“ pixel need to be visible
  - Transparent pixels are marked with specific colour, for example black (RGB= 0,0,0)
  - Color/brightness limit (compression) as protective area to keying color
  - Client set decoded pixel transparent



cf. [NVI20]

Technology
Arts Sciences
TH Köln

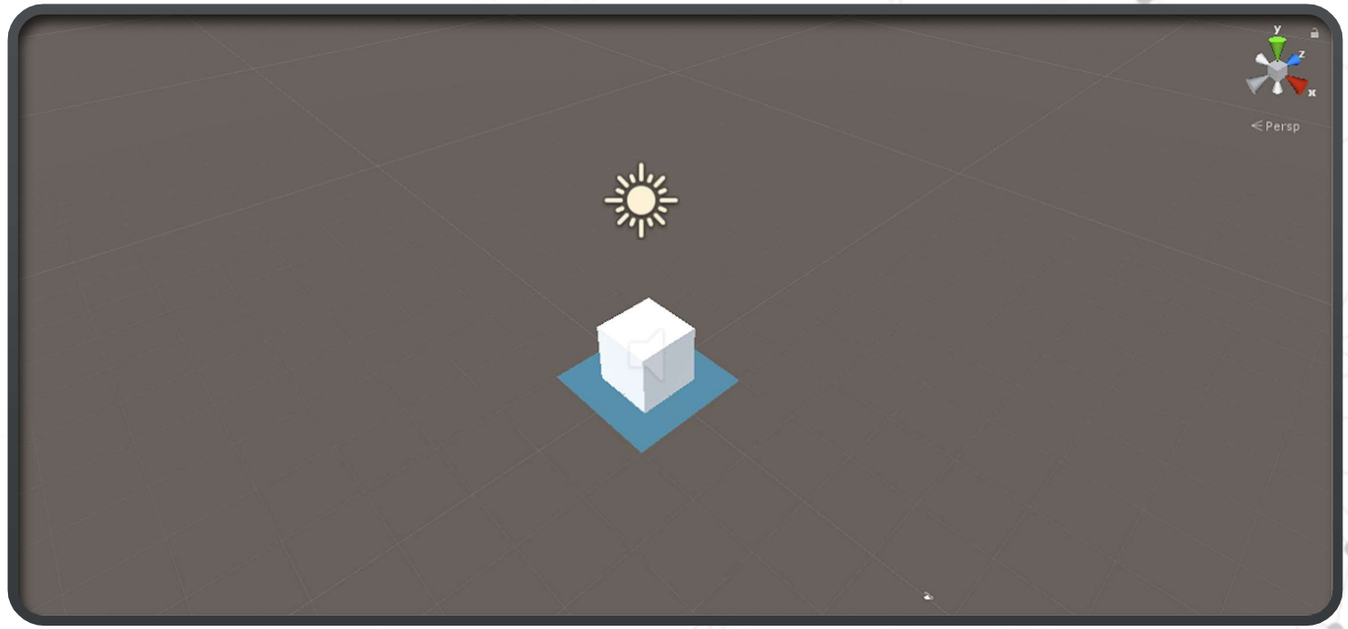
4 Results

Technology
Arts Sciences
TH Köln

Prototypes

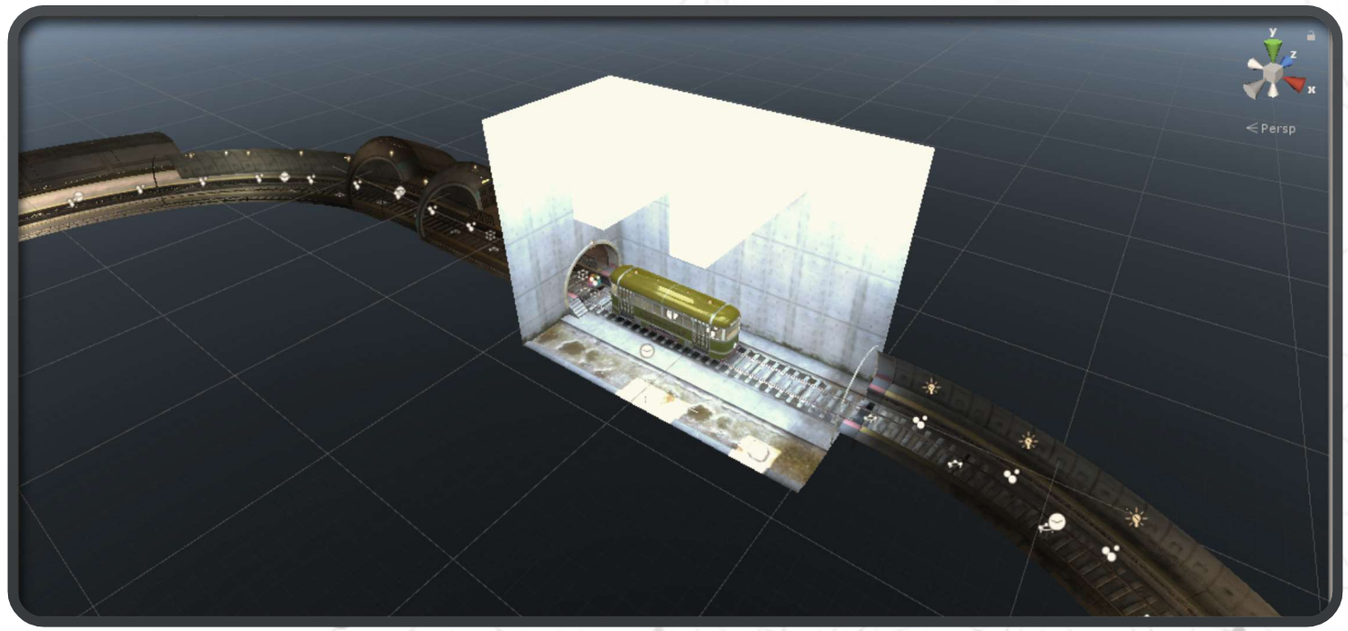
1. Prototype:

- latency measurements
- a rudimentary scene



2. Prototype:

- frame time measurements
- a complex scene

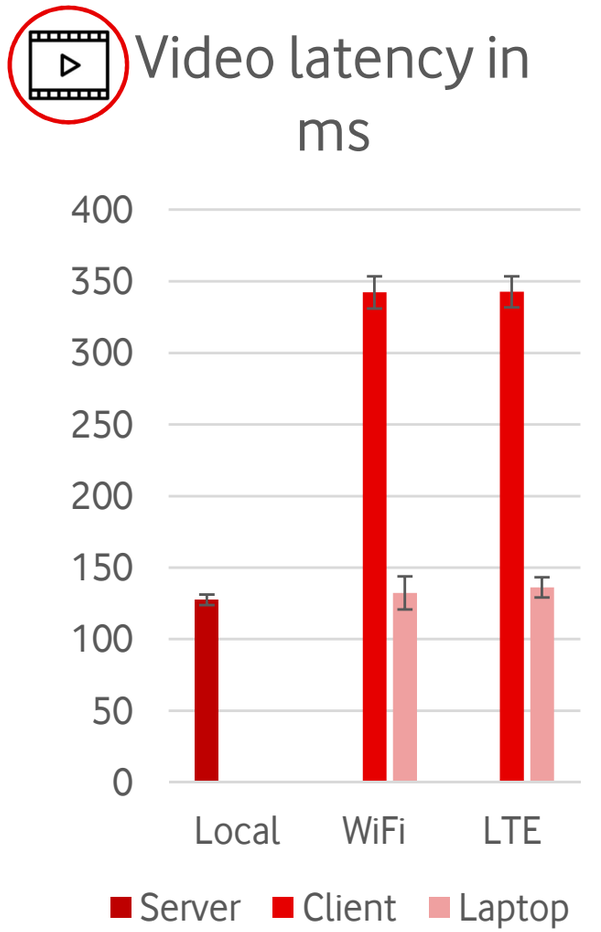


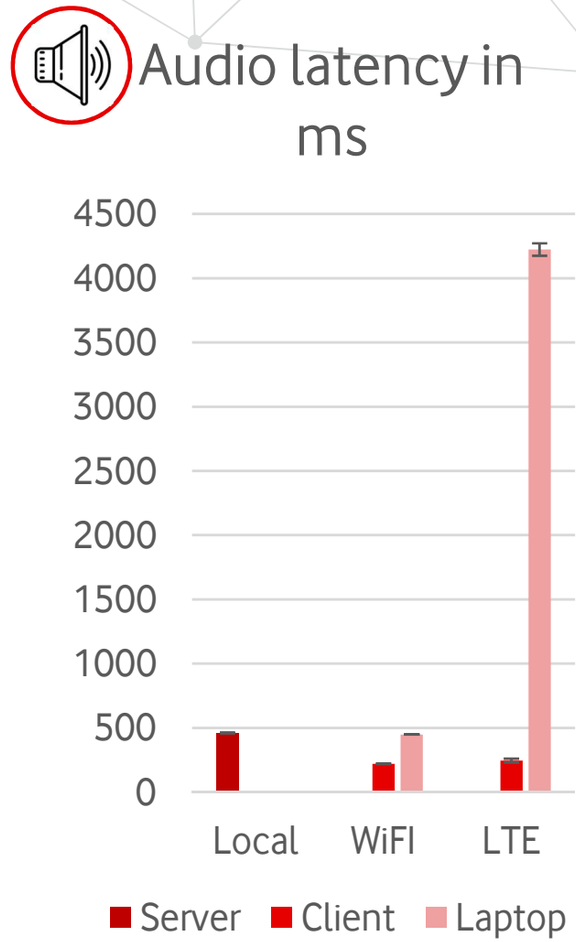
Technology
Arts Sciences
TH Köln

Video and Audio Latency

Results:

- Latency between Client and Laptop is different
- Video latency is better with VLC
- Audio latency is better with UniversalMediaPlayer

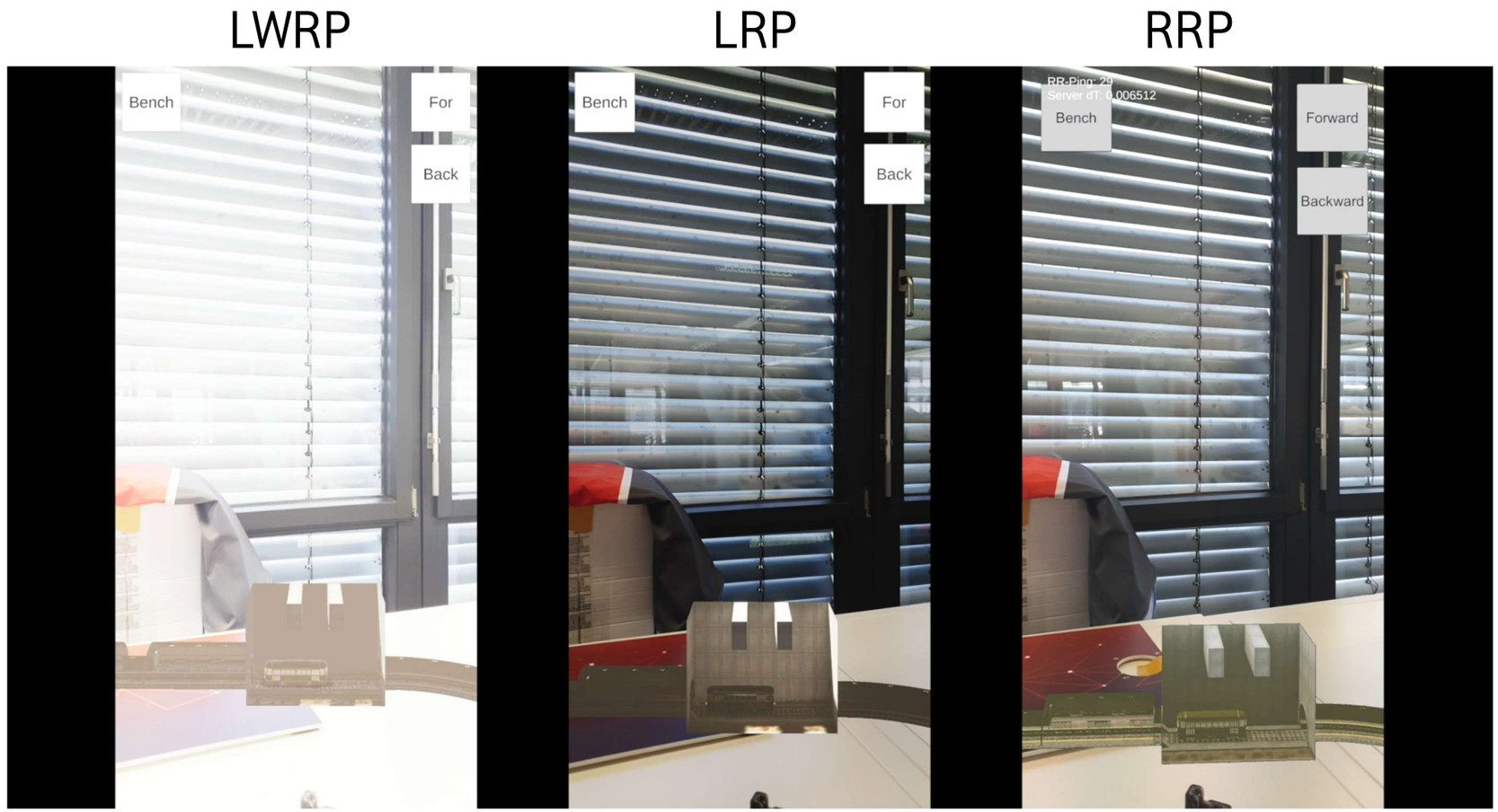




Technology
Arts Sciences
TH Köln

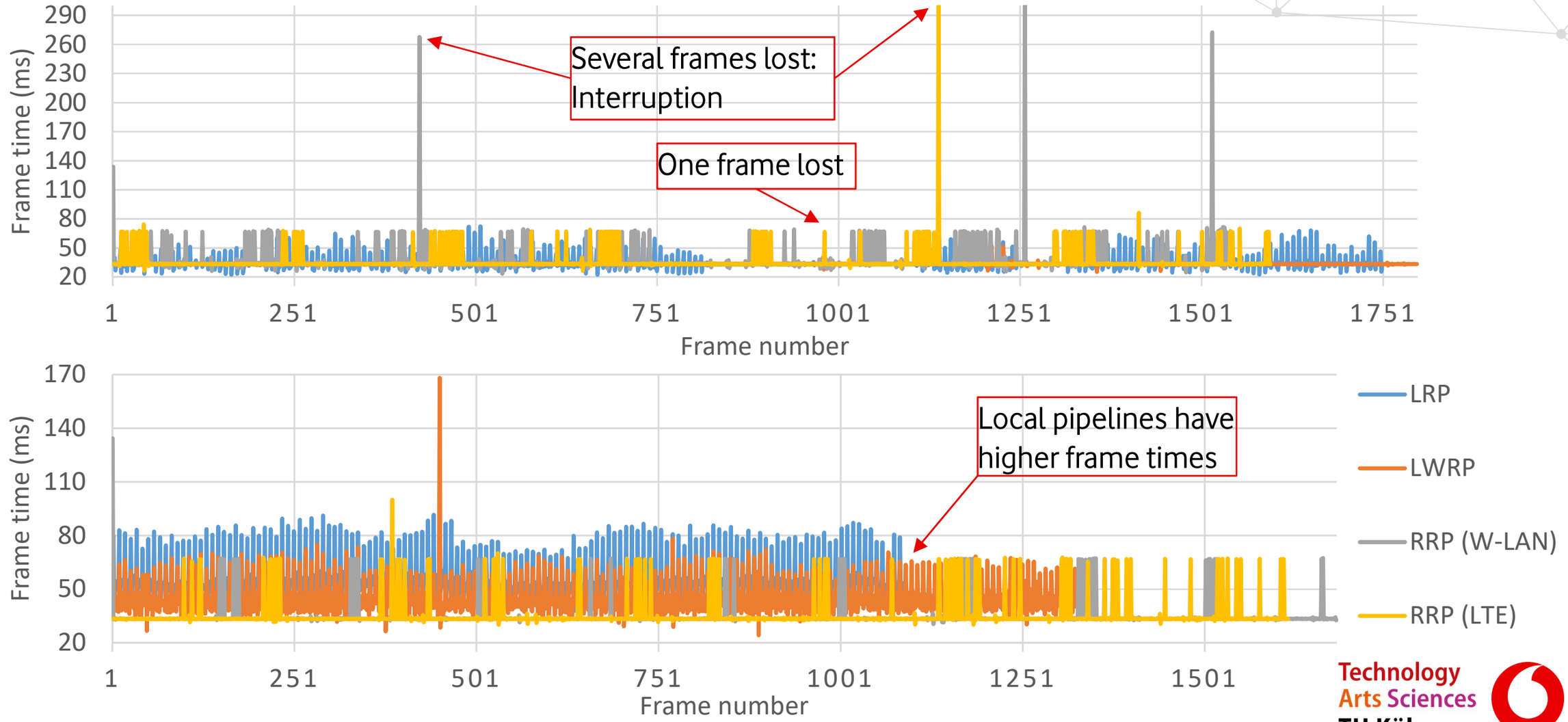
Demovideo AR

- Local pipelines
  - **LWRP**: Lightweight rendering pipeline
  - **LRP**: Legacy rendering pipeline
- RRP: Remote-Rendering
  - Server = Legacy rendering
  - Client = Lightweight rendering
- Spotlight tunnel scene
- AR app (1:100)
- 1440x2960
- Real movement synchronized



Technology
Arts Sciences
TH Köln

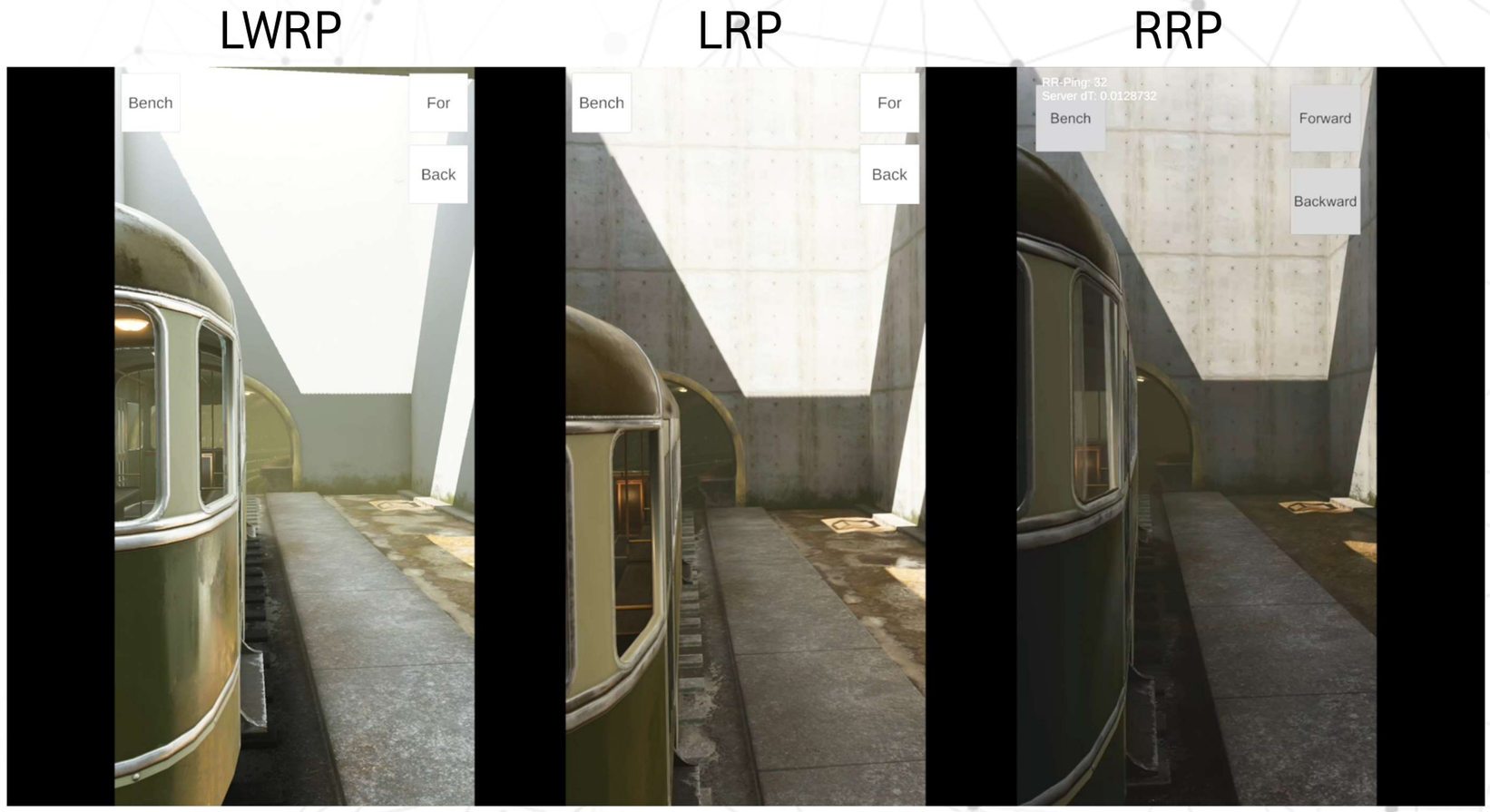
Performance Comparison AR HD1080 & HD1440



Technology
Arts Sciences
TH Köln

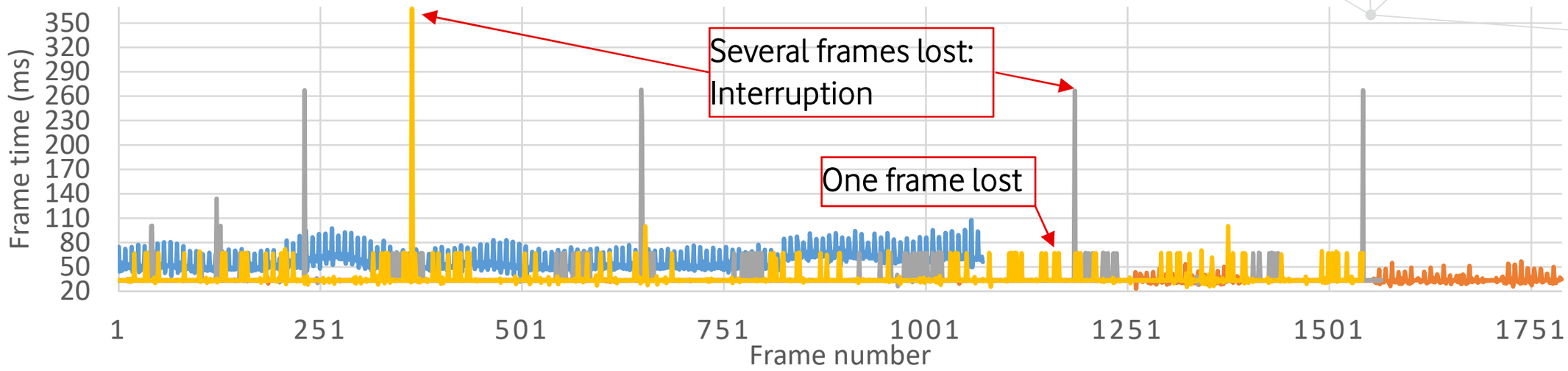
Demovideo Fullscreen

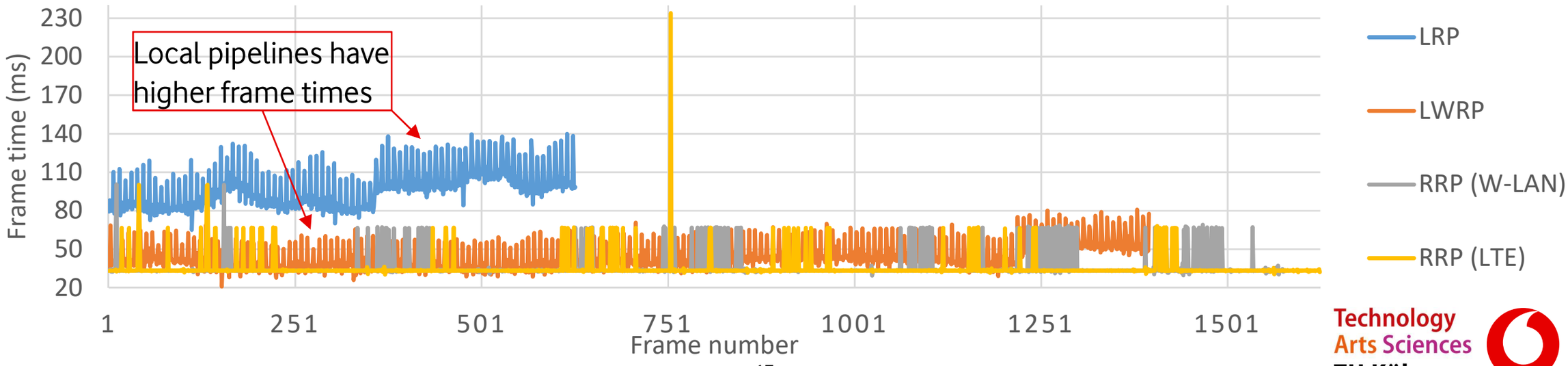
- Local pipelines
  - **LWRP**: Lightweight rendering pipeline
  - **LRP**: Legacy rendering pipeline
- RRP: Remote-Rendering
  - Server = Legacy rendering
  - Client = Lightweight rendering
- Spotlight tunnel scene
- Fullscreen app (1:10)
- 1440x2960
- Virtual camera movement synchronized



Technology Arts Sciences TH Köln

Performance Comparison Fullscreen HD1080 & HD1440

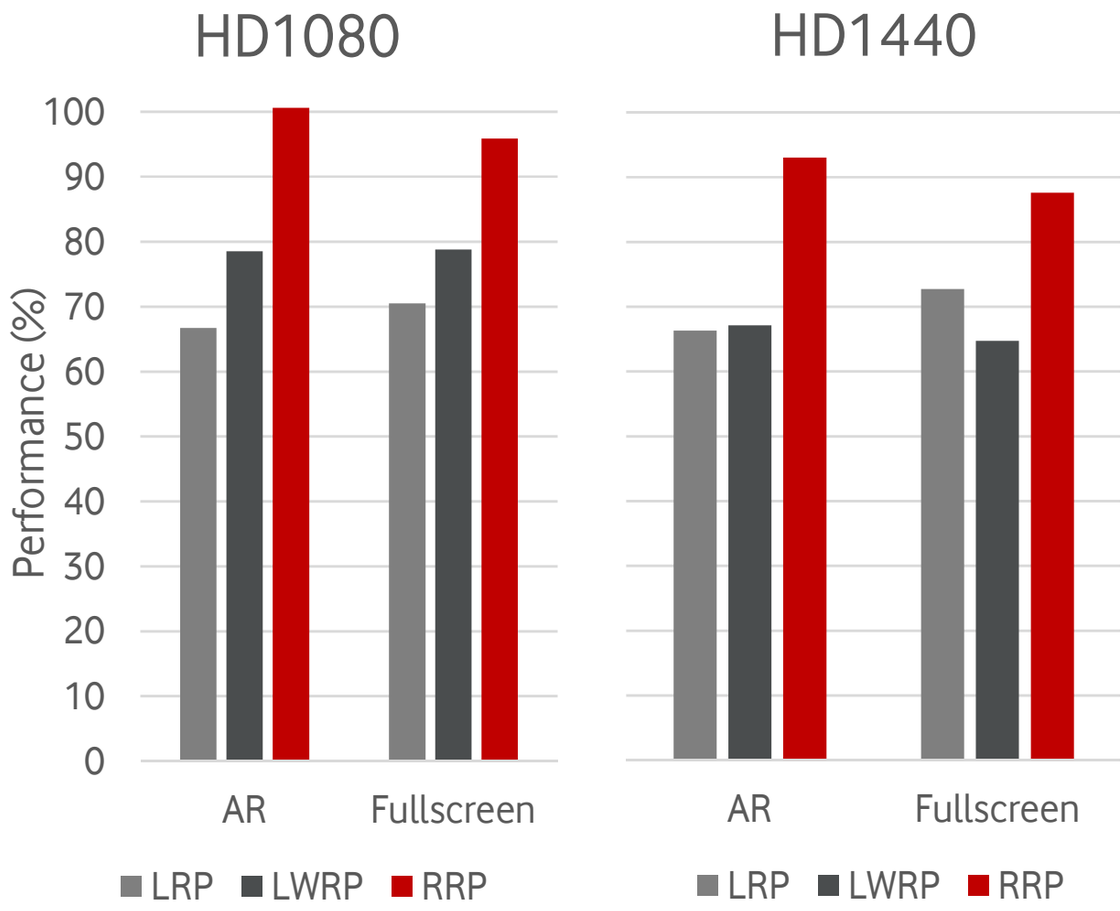




Technology
Arts Sciences
TH Köln

Performance Throttling

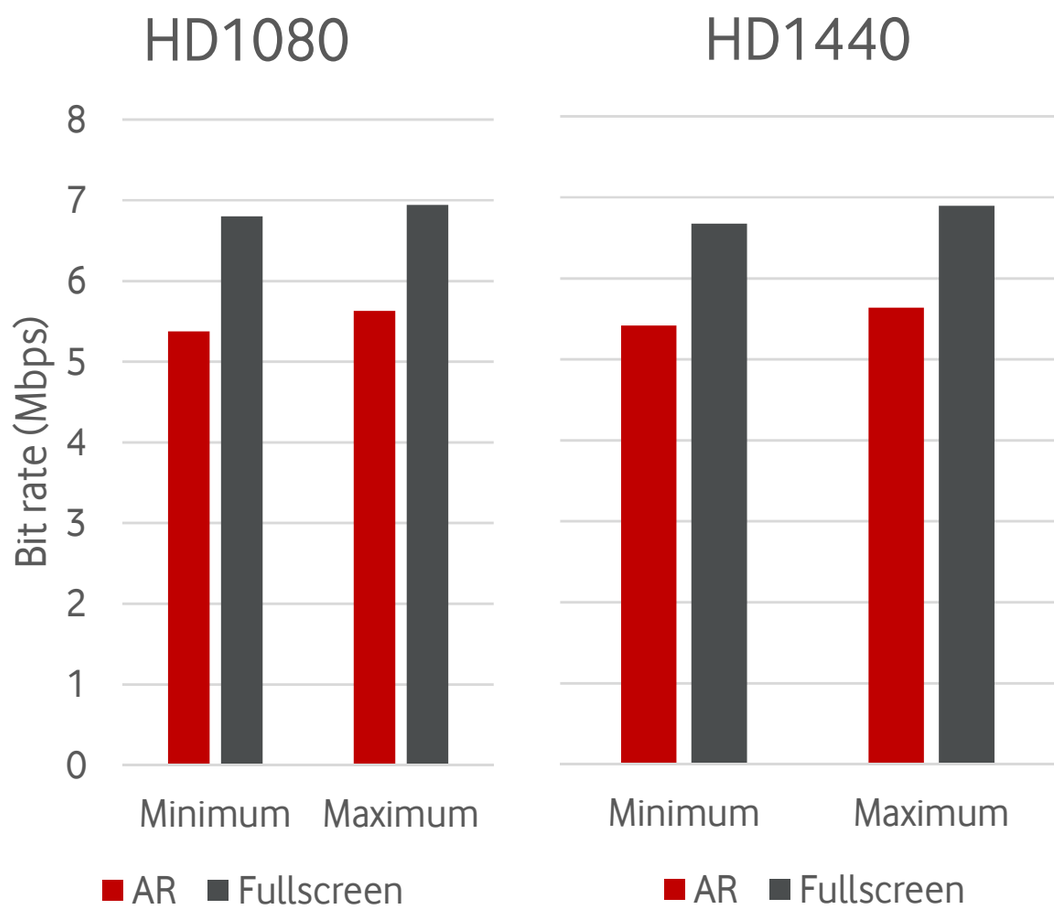
- Modern devices throttle their performance under certain circumstances
- Effect occurred in the applications during five test runs
- RRP has better long-use performance



18

Video Bandwidth

- AR has smaller number of used pixels
- Unused pixels can be compressed more effectively
  - No major changes
- Encoder is exhausted in full screen applications
  - Used configured 7 Mbps completely
  - Higher bit rate required to avoid coding artifacts and dropouts
- The AR version can be transferred without coding artifacts

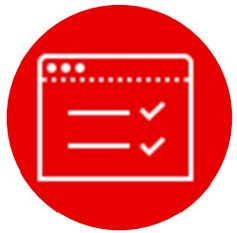


19

5

Conclusions

Conclusions



Proof-of-Concept works

- ✓ Better frame times at high resolution
- ✓ Less throttling
- ✓ Further work is necessary

Only a slight difference in the video latency over LTE & WiFi

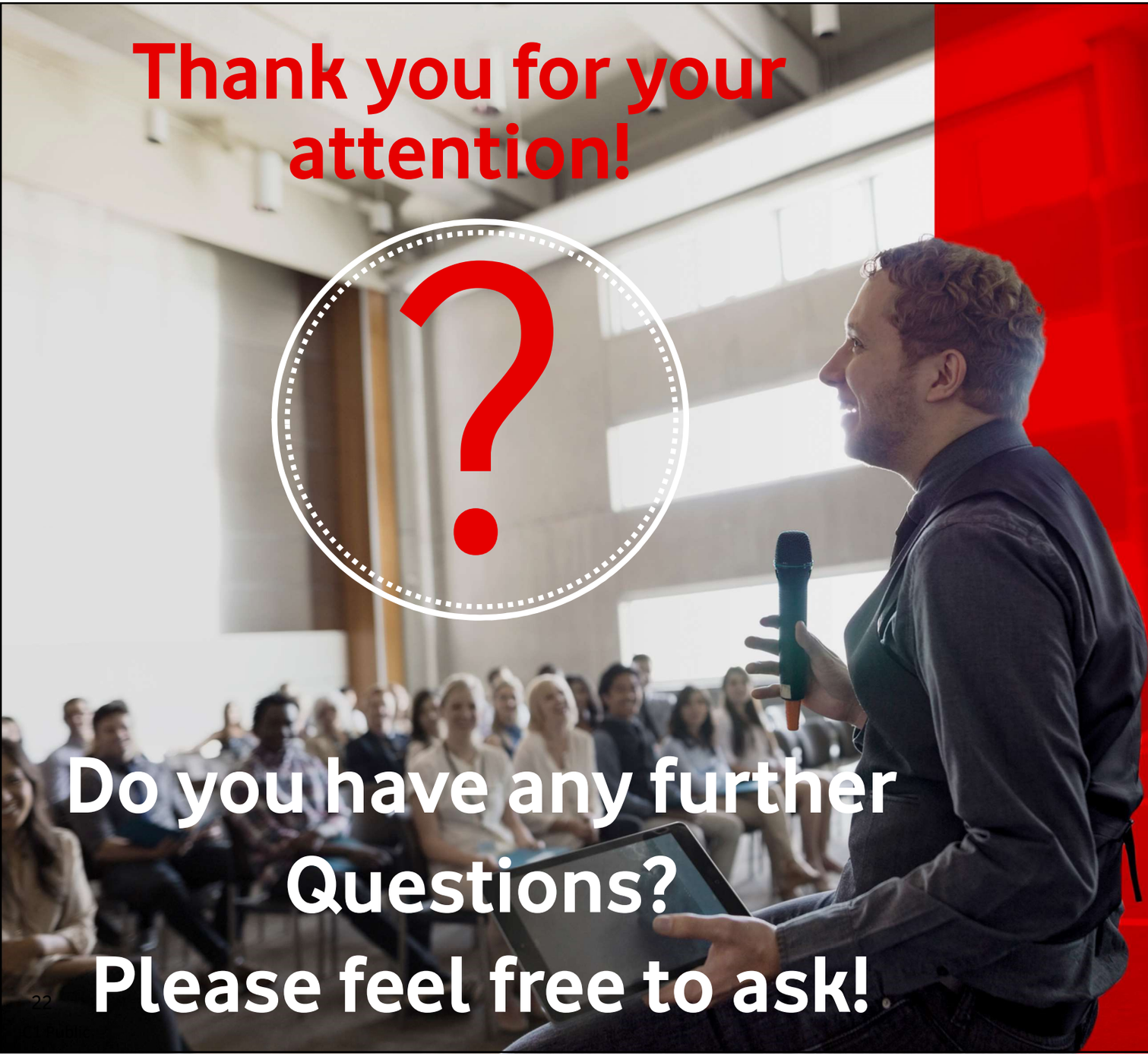
- Optimization of the ultra-reliable and low-latency communication capability of the integrated media player is necessary

Further investigations with 5G NR and slicing

- Higher data rates
- Guaranteed data rate and network latency

Technology
Arts Sciences
TH Köln

Thank you for your attention!



Do you have any further Questions? Please feel free to ask!

References:

[BTV+13] Steven Bohez, Joeri De Turck, Tim Verbelen, Pieter Simoens, and Bart Dhoedt. Mobile, collaborative augmented reality using cloudlets. In 2013 International Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications. European Alliance for Innovation, nov 2013.

[CHC17] Yu-Jung Chen, Chung-Yao Hung, and Shao-Yi Chien. Distributed rendering: Interaction delay reduction in remote rendering with client-end gpu-accelerated scene warping technique. In 2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW), pages 67–72, July 2017.

[HS98] Gerd Hesina and Dieter Schmalstieg. A network architecture for remote rendering. In Proceedings of the 2nd International Workshop on Distributed Interactive Simulation and Real-Time Applications, pages 88–91, 1998.

[Lev95] Marc Levoy. Polygon-assisted JPEG and MPEG compression of synthetic images. In SIGGRAPH '95, page 21–28, New York, NY, USA, 1995. ACM.

[NVI20] NVIDIA Corporation. Nvidia video codec sdk | nvidia developer. https://developer.nvidia.com/nvidia-video-codec-sdk. Accessed: 01.09.2020.

[SH15] Shu Shi and Cheng-Hsin Hsu. A survey of interactive remote rendering systems. ACM Computing Surveys, 47(4):1–29, may 2015.

[SJNC09] Shu Shi, Won J. Jeon, Klara Nahrstedt, and Roy H. Campbell. Real-time remote rendering of 3D video for mobile devices. In Proceedings of the 17th ACM International Conference

[ZHH18] Wenxiao Zhang, Bo Han, and Pan Hui. Jaguar: Low latency mobile augmented reality with flexible tracking. In Proceedings of the 26th ACM International Conference on Multimedia, MM '18, pages 355–363, New York, NY, USA, 2018. ACM.

Technology
Arts Sciences
TH Köln