Demo Abstract: Narwhal: a DASH-based Point Cloud Video Streaming System over Wireless Networks

Jie Li†, Cong Zhang†, Zhi Liu‡, Wei Sun∓, Wei Hu§, Qiyue Li∓
†School of Computer and Information, Hefei University of Technology, China
‡Department of Mathematical and Systems Engineering, Shizuoka University, Japan
∓ School of Electrical Engineering and Automation, Hefei University of Technology Hefei, China
§ Wangxuan Institute of Computer Technology, Peking University, China

***Abstract*—Hologram video is expected to be the next generation video by providing immersive viewing experiences with 6 degrees of freedom. How to efficiently transmit hologram video is one fundamental research issue to promote its applications. As one of the most popular ways to represent hologram video, point cloud video draws more and more attentions. Point cloud video streaming faces many challenges due to large source video rate and high encoding/decoding complexity. To this end, we propose a novel DASH-based point cloud video streaming system: *Narwhal*, which aims to maximize the user's viewing experiences by efficiently allocating the computational and communication resources. We prototype this system and verify its performance over state-of-the-art wireless networks.**

***Index Terms*—hologram, point cloud, video streaming, immersive video, DASH**

I. Introduction

Hologram video is expected to be the next generation video by providing immersive viewing experiences with 6 degrees of freedom, and a typical application of 6G cellular networks. As one of the most popular way to represent hologram, point cloud video draws more and more attentions.

How to efficiently transmit point cloud video over wireless networks is one fundamental research issue, which faces many challenges due to large source video rate and high decoding complexity. In particular, the number of 3D points is about 2.8M for commonly used point cloud sequence *Longdress* [1] with moderate quality and file size of a single frame is approximately 78Mbits. Moreover, although the current point cloud video encoding and decoding technology has high efficiency, and supports multi-core multi-thread operations [2], its complexity is high and thus can not support real-time processing. Finally, the quality of experience (QoE) model of point cloud video streaming has not been well established yet, making it hard to find an optimization method for a practical streaming system. Few attempts have been made for high quality hologram video streaming [3]. To this end, we aim to demonstrate our DASH-based point cloud video steaming system called *Narwhal*, which utilizes the state-of-the-art point cloud encoding and decoding tool, and optimally allocates both the computational and communication resources to maximize the viewing experience. We prototype this system using off-the-shelf devices and obtain satisfactory performance.

The work of Zhi Liu was supported by JSPS KAKENHI Grants 18K18036, 19H0409, 20H04174 and ROIS NII Open Collaborative Research 2020(20FA02).

II. System Overview

As shown in Fig. 1, *Narwhal* consists of the server side and client side. At the server side, the point cloud video is divided into 3D tiles evenly and each tile is encoded into different representations with different quality levels. The uncompressed tiles have the best quality with the largest data size, while compressed tiles have the same quality, smaller data size and require computational resources to decode. The information of all tiles such as the file size, computational resource required to decode, number of points in a frame and the URL of each file is recorded in a Media Presentation Description (MPD) file, as in MPEG DASH.

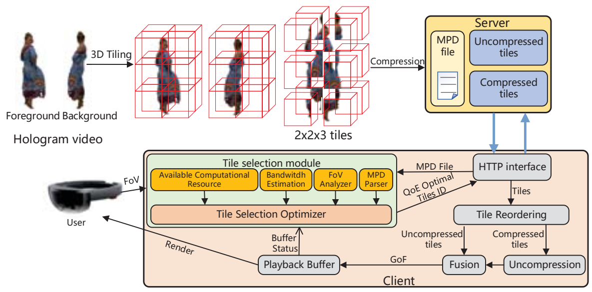


Fig. 1. System Architecture of *Narwhal*.

The client side is composed of HTTP interface, tile reordering module, decoding module, fusion module, playback buffer and tile selection module. The key of *Narwhal* is tile selection, which calculates tiles residing in user's filed of view (i.e. FoV, the information of which can be obtained via the playback device) to avoid unnecessary transmission. This process is achieved by jointly allocating both the computational and communication resources based on the information of buffer status, user's FoV, estimated bandwidth and available computational resources. Due to the page limit, we omit the detailed explanation of the resource optimization used in this paper. Please refer to [4] for an example of such optimization[1].

III. Implementation

A. Experimental setup

As shown in Fig. 2, we prototype *Narwhal* with one laptop as the server and one PC with with i5-7500 CPU, RTX2060 GPU, 16GB RAM and 1TB SSD, as the client, where we select the PC to simulate the playback device and the same FoV parameters ($43^\circ \times 29^\circ$) with Hololens 2 are used. With a Xiaomi Network Router 4A WiFi AP, the intranet interconnection network supports up to 1200Mbps bandwidth. We use $VS2017$+$PCL1.9.1$+ $Qt5.12$ (PCL is a point cloud library and Qt is a C++ graphical user interface application development framework) to design a point cloud video player on the client. VPCC-TMC2-v7, officially recommended by MPEG, is employed as the encoding/decoding tool in the player. We deploy a HTTP web server *nginx1.16* at the server side to implement file transmission. In our demo, we select *Longdress* as the test video, which has a total of 300 frames and each group of frames (GoF) contains 10 frames. There are about 780K of points in each frame and each GoF is divided into $2 \times 2 \times 3$ tiles. In our demo, the player has to buffer up to 5 GoFs before it can start playing to avoid stalls.

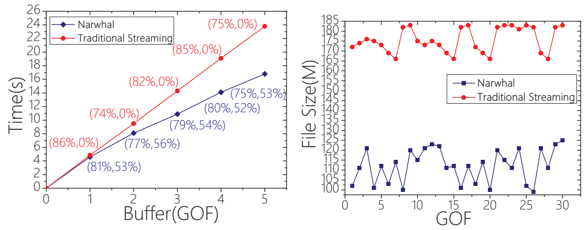
Fig. 2. Overview of the prototype: server, client and WiFi AP.

B. System Performance

We first show the start-up delay and data rate reduction in comparison with the case that no compression is adopted and all the tiles are sent (i.e., *traditional streaming*). Note that most of the existing schemes [3] do not consider the encoding/decoding complexity and hence can not be directly used. From Fig. 3(a), we can observe that *Narwhal* can start playing earlier, where the numbers in brackets indicate the utilization ratio of the communication and computational resources, respectively. Besides, *Narwhal* can greatly reduce the bandwidth consumption and thus lead to better performance, as shown in Fig. 3(b). Overall, *Narwhal* can achieve lower start-up delay time by 29% and reduction of retrieved file size by 35% in comparison with *traditional streaming*.

We show more demo results in Fig. 4. In particular, Fig. 4(a) is the view watched by *Narwhal*, while Fig. 4(c) shows

[1]We implement the whole system and design details in this demo, as compared to [4], which mainly solves the inherent communication and computational resource optimization problem to maximize the QoE.



(a) (b)

Fig. 3. *Narwhal* vs *Traditional Streaming*: (a) Buffering time and resource utilization (b) Retrieved file size of each GoF.

the original frame, both of which show the video from the user's viewing perspective. We can observe that their qualities are almost the same. The PSNR achieved by *Narwhal* reaches 71.33dB with the aforementioned setup, which is quite satisfactory. To further show its effectiveness, we also show the viewing angle in Fig. 4(b) and only the tiles intersect the cone are delivered to the user. As a reference, the original video is presented in Fig. 4(d). For more details, please watch the demo video at: https://www.youtube.com/watch?v=A67vHlwe3js

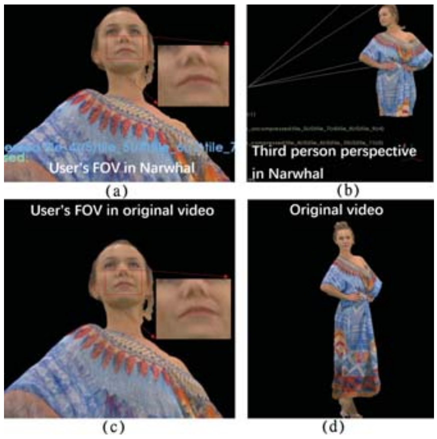


Fig. 4. Point cloud video streaming with Narwhal

IV. Conclusion

This work demonstrated a DASH-based adaptive point cloud video streaming system *Narwhal*. We prototyped this system and experimental results showed that Narwhal can achieve satisfactory viewing performance.

References

[1] E. D'Eon, B. Harrison, T. Myers, and A. Chou, Philip, *8i Voxelized Full Bodies-A Voxelized Point Cloud Dataset*, ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document WG11M40059/WG1M74006, Geneva, January 2017.

[2] D. C. Garcia, T. A. Fonseca, R. U. Ferreira, and R. L. de Queiroz, "Geometry coding for dynamic voxelized point clouds using octrees and multiple contexts," *IEEE TIP*, vol. 29, pp. 313–322, 2020.

[3] J. Park, P. A. Chou, and J. Hwang, "Rate-utility optimized streaming of volumetric media for augmented reality," *IEEE JETCAS*, vol. 9, no. 1, pp. 149–162, March 2019.

[4] J. Li, C. Zhang, Z. Liu, W. Sun, and Q. Li, "Joint communication and computational resource allocation for qoe-driven point cloud video streaming," in *2020 IEEE International Conference on Communications*.