User Centered Adaptive Streaming of Dynamic Point Clouds with Low Complexity Tiling

Shishir Subramanyam
Irene Viola
s.subramanyam@cwi.nl
irene@cwi.nl
CWI
Amsterdam, the Netherlands

Alan Hanjalic
a.hanjalic@tudelft.nl
TU Delft
Delft, the Netherlands

Pablo Cesar
p.s.cesar@cwi.nl
CWI
Amsterdam, the Netherlands
TU Delft
Delft, the Netherlands

(a) *Acquisition* (b) *Segmentation into tiles* (c) *Encoding for each tile* (d) *User adaptation* (e) *Playout*

Figure (1) Overview of the proposed tiling approach.

ABSTRACT

In recent years, the development of devices for acquisition and rendering of 3D contents have facilitated the diffusion of immersive virtual reality experiences. In particular, the point cloud representation has emerged as a popular format for volumetric photorealistic reconstructions of dynamic real world objects, due to its simplicity and versatility. To optimize the delivery of the large amount of data needed to provide these experiences, adaptive streaming over HTTP is a promising solution. In order to ensure the best quality of experience within the bandwidth constraints, adaptive streaming is combined with tiling to optimize the quality of what is being visualized by the user at a given moment; as such, it has been successfully used in the past for omnidirectional contents. However, its adoption to the point cloud streaming scenario has only been studied to optimize multi-object delivery. In this paper, we present a low-complexity tiling approach to perform adaptive streaming of point cloud content. Tiles are defined by segmenting each point cloud object in several parts, which are then independently encoded. In order to evaluate the approach, we first collect real navigation paths, obtained through a user study in 6 degrees of freedom with 26 participants. The variation in movements and interaction behaviour among users indicate that a user-centered adaptive delivery could lead to sensible gains in terms of perceived quality. Evaluation of the performance of the proposed tiling approach against state of the art solutions for point cloud compression, performed on the collected navigation paths, confirms that considerable gains can be obtained by exploiting user-adaptive streaming, achieving bitrate gains up to 57% with respect to a non-adaptive approach with the same codec. Moreover, we demonstrate that the selection of navigation data has an impact on the relative objective scores.

CCS CONCEPTS

• **Information systems** → **Multimedia streaming**; • **Human-centered computing** → **Virtual reality**.

KEYWORDS

Adaptive streaming; 6 DoF VR; 3D Point Clouds

ACM Reference Format:
Shishir Subramanyam, Irene Viola, Alan Hanjalic, and Pablo Cesar. 2020. User Centered Adaptive Streaming of Dynamic Point Clouds with Low Complexity Tiling. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413535

1 INTRODUCTION

In recent years, the availability of low cost sensors, affordable Head Mounted Displays (HMD) and the computational power of commodity hardware have allowed content providers to serve a broad range of users with immersive Virtual Reality (VR) experiences. In particular, photorealistic volumetric reconstructions of dynamic real world objects have emerged as a popular representation for exploring scenes in VR, enabling the user to navigate the virtual space in 6 Degrees of Freedom (DoF) [13][1]. Among different volumetric representations, such as animated meshes and time-varying meshes, point clouds have emerged as a popular format for real time reconstructions, due to the relative low complexity in acquisition and rendering. However, point cloud contents generally require

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413535

a large amount of data to properly represent volumetric contents, making them unsuitable for bandwidth-limited networks. Several compression solutions have been recently designed to effectively reduce the amount of data that needs to be transmitted, while maintaining an acceptable level of visual quality, such as the upcoming MPEG point cloud compression standard V-PCC [32].

Current compression solutions are generally optimized based on the global quality of the point cloud content to be transmitted, without considering how it will be rendered and presented to the users. However, as users navigate the virtual scene in which the volumetric content is placed, they are able to decide which part of the content they want to visualize. Thus, notable gains in compression efficiency can be achieved when taking into consideration the portion of the content that will be visible to the users. In particular, by assigning a larger portion of the available bandwidth to parts of the content that are visible to the users, one can achieve user-centered bandwidth adaptation, that maximizes the user Quality of Experience (QoE) while adhering to network limitations.

Network adaptation for media streaming is a well-known concept for traditional media; in particular, HTTP-Adaptive Streaming (HAS) is commonly achieved for 2D videos using the Dynamic Adaptive Streaming over HTTP (DASH) standard [33]. In recent years, user-centered adaptation strategies have been proposed for 360-degree video streaming in previous research, for 3DoF [40][21][9] or for discrete 6DoF [8]. The 360 video is spatially divided into independently decodable portions, i.e., tiles. The client predicts the user's navigation pattern and requests the tiles that are likely to fall in the user's visible portion of the scene (viewport). Such tiles are retrieved in high quality, whereas the remaining tiles are retrieved in descending order of quality as the distance from the viewport increases. Adaptive streaming strategies have also been tested to optimize streaming of multiple point cloud objects in a scene [18][37]. The approach allows to achieve network gains by assigning a larger portion of the available bandwidth to the contents that fall within the user's viewport. However, no adaptation is performed within the point cloud content, meaning that the same bandwidth is assigned to points that, while belonging to the content within the viewport, are not directly visible to the user. Moreover, while the approach was extensively tested, the navigation paths used to simulate the users' behaviour were arbitrarily selected by the experimenters, and thus may not reflect the actual users' behaviour.

In our work, we aim at extending the previous literature in user-centered adaptive streaming to point cloud content, aiming at real-time tiled streaming of volumetric media for social VR. We focus on users wearing an HMD to view and interact with the content rather than watching on a monitor. While previous work focused on assigning point cloud objects to single tiles, thus performing optimization in a multi-object streaming scenario, we focus on optimizing the delivery of a single point cloud object, by assigning different parts of the content to tiles with diversified utility. Our tiling approach is based on inferring the different surfaces on a point cloud using a low-complexity centroid-to-point vector in order to split an object into tiles. The tiles are independently decodable and allow for adaptive streaming based on the location and orientation of the user's viewport. The relative simplicity of our tiling approach makes it suitable for real-time systems, whose compatibility with adaptive streaming approaches, such as DASH, has been recently demonstrated [19]. To test whether our tiling approach would yield gains in terms of perceived quality, we first collect a dataset of navigation paths, obtained by letting 26 users visualize point cloud contents in a 6DoF VR scenario. The difference we observe in the interaction behavior among participants suggests that a user-adaptive streaming strategy could lead to a gain in rate-distortion performance, and serves as a motivation for our tiling approach. We use the obtained navigation data in order to compare the visual quality of the tiling approaches with respect to state-of-the-art compression solutions, such as MPEG V-PCC, using popular video quality metrics. Our results show that our tiling approach can effectively improve the QoE with respect to the baseline, while underlining the importance of using real user data to perform the analysis.

The contributions of this paper are twofold:

- We follow a user-centered approach to collect a dataset of the navigation paths of users viewing the point cloud with 6DoF, which can be used to evaluate adaptive streaming approaches.
- We demonstrate the validity of user-centered, real-time adaptive streaming to optimize the delivery of a single dynamic point cloud object, through independently decodable tiles.

The approach we propose is codec-agnostic, in that any point cloud compression solution can be used to obtain the adaptation set, provided that real-time compression can be achieved. While we follow a DASH-compliant framework to perform the user adaptive evaluation, the tiling approach can be easily extended to other media delivery mechanisms as well. The dataset comprising the collected navigation paths and the evaluation scores can be found here: https://github.com/cwi-dis/6DoF-HMD-UserNavigationData.

2 RELATED WORK

2.1 360° Video streaming

User adaptive streaming of 360° videos has received significant research and industrial interest in recent years. While legacy video codecs can be used to encode the planar representation of the 360° videos, when the content is being played out, only a small fraction of the whole video is visible from the users viewport at any given moment. In order to exploit user interactions and the corresponding visible portions of the sphere to optimize delivery, a popular approach is to differentiate the quality of different regions in the frame. This approach is included in the Motion Controlled Tile Sets (MCTS) coding included in the High Efficiancy Video Coding (HEVC) codec [16]. In this approach, the projected video is split into independently decodable, non-overlapping spatial regions called tiles. The tiles are encoded at multiple quality levels, and the client is able to select a quality for each tile in the frame based on visibility from the users' viewport. While additional bandwidth overhead is expected, due to a loss of compression efficiency and additional metadata [6], the approach allows for a more flexible delivery based on user interactions, which has led to it being successfully implemented and integrated into standards[26] [24]. 360° video navigation trace datasets have been previously proposed to analyze user interactions[22] [15] [39] [30], in order to optimize streaming based on saliency maps. Abreu et al. [10] record a dataset

of navigation trajectories and saliency maps for users viewing 360° images. Upenik et al. [35] derive viewing probabilities during subjective tests. Corbillon et al. [7] provide a dataset of user head movements for longer video sequences of 70 seconds. Duanmu et al. [12] compare the movements of users watching 360° videos on head mounted displays and on monitors and analyze clusters of users and videos based on rotation movements. In this work, we make available a dataset of navigation patterns in 6 DoF and extend previous work on 360° video to 6DoF.

2.2 Point Cloud Streaming

Streaming volumetric data has previously been addressed by Collet et al. [5]. The authors collect point clouds from multiple depth sensors and reconstruct a sequence of watertight dynamic meshes. They encode the mesh textures maps using a standard H.264 video codec. Park et al. [28] present a streaming framework based on 3D tiles. They define a utility function per tile based on the user's proximity, underlying quality of the tile and the user's display device resolution. They present a greedy algorithm to maximize utility and propose a window based approach to the client buffer manager. Hosseini et al. [18] propose DASH-PC, a dynamic, adaptive view-aware point cloud streaming system. They present three algorithms to spatially sub-sample point cloud objects to create multiple representations. They signal the density or point count and use human visual acuity to select the appropriate representation for the end user. However, they do not account for the orientation of the underlying surface. Hooft et al. [37] propose PCC DASH, a standards-compliant means for HTTP adaptive streaming of scenes comprising multiple dynamic point cloud objects. They present a number of rate adaptation heuristics, based on a user's location, focus, available bandwidth and buffer status, which are used to adaptively set the quality of different point cloud objects in the scene. However, there is no differentiation in quality within each point cloud object. He at al. [17] propose view dependent streaming over hybrid networks. Each point cloud frame is projected onto the six faces of a bounding cube to create a color and depth video for each face of the cube. They transmit low quality videos to the user using digital broadcasting. The user can request particular faces of the cube in high quality from the edge node of a bidirectional broadband network, and reconstructs the point cloud from the downloaded depth and color videos. Qian et al.[29] propose Nebula, a streaming system to deliver volumetric video to mobile devices. They vary the spatial density of the volumetric video stored as point clouds, and present two rate adaptation heuristics.

In this work, we propose an approach to tile individual point cloud objects into independently decodable, non-overlapping spatial regions, and select a quality level for each tile based on the current position and orientation of the user's viewport.

2.3 Segmentation

Segmentation is the division of the point cloud into clusters of points that are homogeneous with respect to a selected characteristic. In order to adaptively stream a point cloud object based on the orientation of the user's viewport, we need to cluster points based on the orientation of their underlying surface. There are two classes of methods to estimate normals to the surface for each point [31]. The most accurate method is to reconstruct the surface and create a watertight mesh from the point cloud frames. The normals to the surface can then directly be associated with each point. The second approach is based on inferring the underlying surface based on the point local neighborhood. Nguyen et al. [25] present a taxonomy of five classes of segmentation algorithms: edge based, region based, attribute based, model based, and graph based. Attribute-based segmentation methods can be used to tile point clouds, and account for surface orientation. These methods rely on estimating additional attributes for each point, e.g., normals, to obtain the surface orientation before clustering. Thus, these solutions are not suitable for a real-time system, as inferring the surface normal requires repeated eigendecomposition for the local neighbourhood of each point.

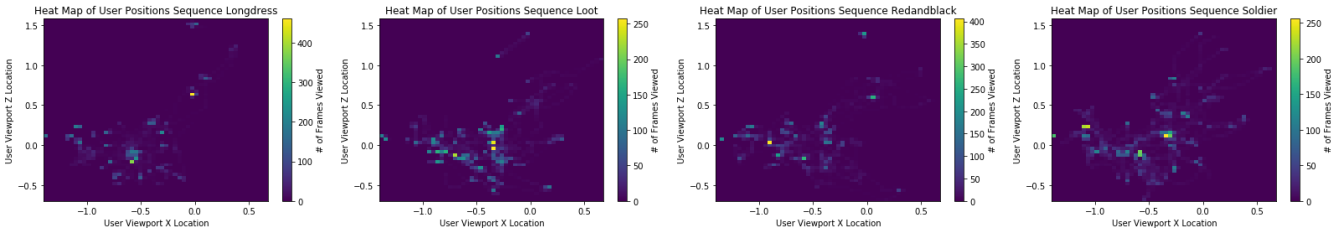
In this work, we propose a simple low complexity approach to estimate surfaces suitable for real time applications. We draw a vector from the centroid to every point on the surface, and compare these vectors to 4 virtual cameras placed around the object on the XZ plane. In this way, we assign points to each camera, and allow for adaptive streaming based on the orientation of the user's viewport, as shown in Figure 1 (b).

3 PROPOSED APPROACH

In this section we provide an overview of the delivery system proposed in this work. The DASH streaming system is designed to operate efficiently over large distributed HTTP networks. It is based on the server providing the same content at multiple quality levels by encoding the same content at multiple bit rates. The client then selects a quality level to receive content from the server based on the available network throughput. This reduces buffering events and improves the user experience by adapting to varying network conditions [4]. For 3 DoF and 6 DoF VR/AR content, streaming can be further improved by adapting to user interactions and movements by streaming only the content currently visible to the user at high quality [8].

3.1 Tiling Strategy

3D dynamic point clouds are usually captured using a multi camera setup surrounding the object to be scanned. In order to consistently align each of the 3D views into a complete model, a registration step is required in order to identify a transformation matrix for each of the cameras. In a real time capture scenario with multiple depth sensors the camera visibility of each point in the point cloud can be recorded. This can serve as a proxy for the orientation at each point without explicitly performing a surface reconstruction. We can assign an orientation vector to each camera location using the transformation matrix for each camera from the registration step. The assumption here is that the camera locations are a proxy for the end user's viewport location. This allows us to assign an orientation to each segment or tile without any additional computational overhead. We then treat each tile as a separate point cloud track. The server creates multiple representations by encoding each tile at multiple qualities. We also include additional metadata to each tile in the adaptation set such as the orientation vector of each tile. The client then uses this information to request each representation at the optimal quality based on orientation of a tile as shown in



(a) ***Long dress*** **(b)** ***Loot*** **(c)** ***Red and black*** **(d)** ***Soldier***

Figure (2) Heatmaps of user positions on the XZ (floor) plane during playout of each of the sequences

Figure 1. This allows the client to adapt to user interactions such as moving the viewport.

In this work we propose a simple low complexity tiling approach based on drawing a vector from the object centroid to the surface and comparing this to four virtual cameras placed around the object in the XZ plane in order to divide the object spatially by assigning points to each of the virtual cameras as shown in Figure 1 (b). We restrict the number of tiles to four for compression efficiency and lower metadata. Here we assume that the point cloud object presents a smooth convex hull. In the Media Presentation Description (MPD) document we include the tile metadata in the adaptation set for each tile. This can be used by the Client Buffer Manager to optimize the representation per tile based on the user's viewport. The tile metadata includes the components of the camera orientation vector and the coordinates of the tile centroid.

3.2 Tile Selection

In order to compare different tile selection algorithms and study the potential impact of tiling on optimizing delivery we assume that the location and orientation of the user's viewport for the next frame is always known. In practice a probability density function can be modelled based on past navigation patterns and the probability of each possible next viewport can be computed. In future we will apply this approach to a telepresence system such as [19].

The viewport of the user is represented by a location V_l and an orientation V_o. We define the utility U of a tile $T - i$ as $U(T_i) = \vec{T}_i \cdot \vec{V}_o$. The bit rate budget is divided amongst available tiles, based on this utility. A representation for each tile can then be selected, and the final representation vector for a frame is retrieved from the server, as shown in Figure 1 (c). In this work, we use three allocation schemes to select representations for each of the tiles, as originally proposed by Hooft et al. [37]. We first sort the tiles based on their utility, defining visible tiles as the ones having a positive $U(T_i)$.

(1) **Greedy** bit rate allocation: The highest quality representation is first set for the highest utility tile and then we move on the next highest ranked tile until the bit rate budget is spent.
(2) **Uniform** bit rate allocation: The representation of tiles are increased one step at a time starting with the highest utility tile.
(3) **Hybrid** bit rate allocation: The representations of visible tiles are first uniformly increased in order of utility. The representations of the remaining tiles are then uniformly increased until the bit rate budget is spent.

4 USER NAVIGATION PATTERNS

4.1 Dataset Preparation

Point cloud dataset For this experiment we use the 8i Voxelised Full Body Dataset [11] provided by MPEG for testing. This dataset comprises of four dynamic point cloud sequences (*Long dress*, *Loot*, *Red and black*, *Soldier*) captured using photogrammetry. Each sequence is recorded at 30 frames per second (fps). We clip the sequences to the first 5 seconds. The uncompressed original point clouds are then rendered in the unity game engine in an empty room with a grey background.

Playback conditions To render a point cloud frame, we store the points in a vertex buffer and draw procedural geometry on the GPU. The points are rendered by placing a camera facing quadrilateral with a fixed offset of 0.008 units (this corresponds to a side length of approximately 2 mm) centered around each point. All sequences were rendered at a constant 30 fps.

Experimental setup We used a workstation with two GeForce GTX 1080 Ti in SLI for the GPU and an Intel Core i9 Skylake-X 2.9GHz CPU. All point clouds were rendered using the Unity 2018.4.23f1 game engine. We recruited 26 participants for this test (10 Female, 16 Male) with an average age of 27.88. Participants were asked to wear an Oculus Rift HMD to view each of the point cloud sequences and they were free to move around the virtual space and inspect each of the sequences. We oriented the rendered point clouds to face the user's starting position. At every rendered frame we collected and logged the camera position represented using x,y,z coordinates and rotation as three Euler angles about the x, y and z axis. The playback was looped and participants were allowed to watch the content until they indicated to move on to the next content.

4.2 Results

To explore the motion differences across the navigation dataset we first look at the movements of participants on the floor plane (XZ plane) while viewing all the sequences shown in Figure 2. The centroids of each of the dynamic point cloud sequences were approximately at (0.01,1.7,-0.04) during playback. User movements vary considerably across the four datasets used in the experiment. After the experiment, participants reported that they preferred the soldier sequence as the facial features were clearer and the reconstruction has unitoned clothes and slower movements. We observe a greater spread of user viewport locations around the object for this sequence.

To explore the angular movements of the participants we use the angular distance in degrees covered by each participant for every

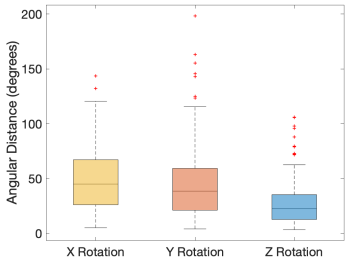


Figure (3) Total angular distance in degrees travelled by participants for every completed playback loop

Table (1) Number of frames in the first playback loop where each tile is facing the user, for all 26 participants

	Tile1	Tile2	Tile3	Tile4
Long dress	110	3531	3772	351
Loot	53	3694	3847	206
Red and black	235	3730	3665	170
Soldier	109	3587	3765	287

completed 5 second playback loop. Figure 3 shows the box plot of the angular distance in degrees covered by participants in every completed playback loop. We see a significant dispersion in the angular movements across playback loops especially about the X and Y axis. From the participants starting position, while facing the content these correspond to components of yaw head movements. The angular distance and positions are based on the location and orientation of the user's viewport in the Unity world coordinate system. The variations in movements across the sequences and participants indicate a possibility to optimize delivery of dynamic point clouds based on user interactions.

Next we consider the tiles visible to the user as defined in section 3.2. The number of frames where each tile is facing the user in the first playback loop is shown in Table 1. Across all sequences tile 2 and tile 3 are visible in 95% of the frames. In figure 1 (b) the tiles 1 to 4 are shown from left to right. Tiles 2 and 3 correspond to the frontal view of the point cloud object. This indicates an opportunity to optimize the user's quality of experience by allocating a larger portion of the available bandwidth for tiles 2 and 3, as shown in Figure 1 (b).

5 USER ADAPTIVE STREAMING

5.1 Dataset Preparation

Content selection and tiling In order to perform our evaluation, we select the same 8i Voxelised Full Body Dataset, as described in section 4.1, and place four virtual cameras around the object on the XZ (floor) plane (at (1,0,0), (0,0,1), (-1,0,0) and (0,0,-1)). We assume a smooth convex hull with no occlusions, similar to real time capture with multiple depth sensors. We draw a vector from the centroid of the point cloud to every point on the surface, and we use the vector dot product of these vectors with the 4 virtual cameras to assign a tile number to each point in the cloud.

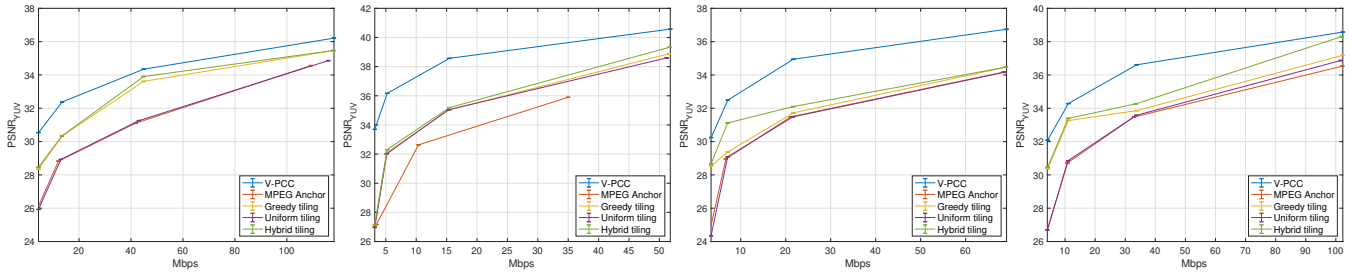
Codec selection To evaluate the impact of user adaptation, we select the MPEG anchor codec proposed by Mekuria et al. [23] which uses the popular octree space partitioning structure to encode point cloud geometry. Attributes like color are then encoded by mapping them to a 2D grid and applying JPEG compression. This codec design allows for low delay encoding and decoding, making it suitable for real time applications. Thus, it represents a viable solution for evaluating our tiling strategies. The adaptation set is prepared using the MPEG anchor codec in an all-intra configuration. Each tile is encoded at octree depths from 6 to 11, with the JPEG quantization parameter varying from 55 to 95 in increments of 10. To measure the performance of viewport adaptive streaming we encode the source point clouds with the same codec configuration on the MPEG anchor codec. Additionally, we compare the performance of our tiling approaches against the upcoming MPEG VPCC standard [32]. The codec is based on extending legacy video compression techniques by mapping the point cloud geometry and attributes to a 2D grid, and using video compression to encode both the geometry and the attributes separately. The current implementation of this approach has high encode complexity, making it unsuitable for real time applications. Moreover, its compression performance is expected to decrease when using tiling approaches, which reduce the amount of data that can be packed in the 2D grid. We use the VPCC codec to provide a baseline, indicating the state-of-the-art rate-distortion performance. We encode the source point clouds using Release 7.0 of the MPEG VPCC codec, using the configurations provided in the Common Test Conditions for Category 2 All Intra (C2AI) encoding. We selected the rate points 1, 3 and 5 and extend it to an additional rate point, using a Texture QP of 9, a geometry QP of 12 and an occupancy precision of 2.

Rate selection To measure the impact of user-adaptive streaming, we use static bitrates targets, to remove the effect of network adaptation. We define the maximum bit allocation budget based on the encoded bitstream size for the Common Test Condition compression profiles supplied with the MPEG V-PCC codec, for each sequence and rate point.

Tile selection The navigation patterns recorded during the experiment described in the last section were then used to set the camera position and rotation in Unity. To render the tiles, we first compute the utility of each tile by comparing the recorded user viewport orientation with the four virtual cameras used to create the tiles. We then follow the approach described in section 3.2 to select a representation for each tile. An example of a point cloud rendered using tiles encoded at different octree depths is shown in Figure 1.

5.2 Metrics

In order to obtain an assessment of the visual distortion of a given point cloud content, several point cloud objective quality metrics have been developed, and serve as a benchmark of compression solutions [2]. Most of the state-of-the-art objective metrics, however, evaluate the visual quality of the entire point cloud content, thus including parts that would likely not be observed by users. The tiling approach aims at exploiting user attention in order to dedicate a larger portion of the bandwidth to parts of the point cloud that are actively visualized, while assigning less bits to parts that are effectively hidden to the users. Thus, common point clouds metrics, such as point-to-point or point-to-plane approaches, would not be



(a) *Long dress* (b) [*Loot* (c) *Red and black* (d) *Soldier*

Figure (4) PSNR computed on the YUV channels against achieved bit-rate, expressed in Mbps, averaged across frames and navigation paths.

suitable to assess the bit-rate gains brought by user-centered tiling approaches. In this paper, we thus decided to evaluate the visual quality of the point cloud contents as they would be seen from the users, using common video metrics to estimate their quality. In particular, we use the camera positions recorded in the experiment detailed in Section 4.1, in order to simulate users visualizing point cloud contents. Each frame of each content is rendered in Unity, and the scene is captured in a resolution of 1920x1080, to be as close as possible to the resolution of the HMD. The background color of the scene is set to $RGB = [0, 177, 85]$. The color was selected to provide maximal contrast with respect to the content. We used the same playback conditions defined in Section 4.1. For encoded contents and tiles with an octree depth less or equal to 7, we increase the offset to 0.016 units, to obtain watertight surfaces.

The image metric PSNR is used to provide an estimation of the quality of the distorted point clouds with respect to the uncompressed reference, as rendered in Unity. The computation is performed on the YCbCr space, as it was proven to be better correlated with human perception, and it is averaged across the channels using the weights proposed in [27]. To reduce the impact of the background on the metric computation, we decide to perform the metric computation only on the parts of the acquired image which contain the point cloud content. In particular, we define a Region of Interest (RoI) by excluding points whose *RGB* values are equal to the background. Reference and distorted contents will likely have different number of points and occupancy grids, resulting in potentially mismatching RoI. Thus, we decided to use the intersection of the RoIs to compute our metrics, as suggested in [2]. To avoid biasing the results, we exclude from the computation frames whose RoI covers less than 0.1% of the entire frame.

5.3 Results

Figure 4 shows the weighted PSNR computed on the YUV channels against the achieved bitrate, averaged across frames and navigation paths, separately for each content. While, as expected, the V-PCC codec achieves the best overall performance, it can be observed that, among the tiling approaches, the hybrid tiling approach yields the best results, closely followed by the greedy approach. Both outperform the MPEG Anchor by a notable margin. Among the approaches, the uniform approach is the one leading to the smallest gains in terms of PSNR. In fact, its performance is comparable with the MPEG anchor, with the notable exception of content *Loot*, for which the obtained results are in line with the other tiling approaches. Due to the restrictions imposed by the maximum bit budget, which was defined based on the V-PCC bitrates, for content *Loot* at the highest bitrate target, it was not possible to select a better representation for the MPEG Anchor, as it would have led to overshooting the bandwidth allocation. This leads to a loss in performance, as a large portion of the bit budget is left unused. The tiling approaches, on the other hand, are able to exploit the available bandwidth in a more efficient way, conducting to a better performance. Table 2 reports the Bjontegaard rate savings for each tiling approach, computed with respect to the MPEG Anchor. It can be seen that the largest bitrate saving is obtained with the hybrid approach, closely followed by the greedy approach, while the uniform approach yields more modest gains.

Table (2) Bjontegaard rate savings for each tiling approach, with respect to the MPEG Anchor.

	Greedy	Uniform	Hybrid
Long dress	-54.39%	1.87%	**-57.15%**
Loot	-47.75%	-46.76%	**-50.43%**
Red and black	-12.84%	3.57%	**-42.85%**
Soldier	-35.15%	-2.77%	**-46.04%**
Average	-37.53%	-11.02%	**-49.12%**

In order to understand whether the collected PSNR scores indicated significant differences among the conditions under test, we performed statistical tests on the data. Before conducting our analysis, we performed a Kolmogorov-Smirnov normality test on the entire set of PSNR scores ($N = 782760$), which rejected the null hypothesis that our data was normally distributed ($p < .001$). Thus, non-parametric tests were selected to analyse our data. Friedman's rank test performed on the scores revealed a significant effect of the codec selection on the final set of scores ($\chi^2 = 570782.56$, $p < .001$). To confirm that our results were not biased by the large number of scores involved, we performed random sampling on the data, selecting $N = 1000$ samples per codec across 1000 sampling runs (seeds) to reduce the dimensionality. We combined probabilities using Fisher's method [14, 20], concluding that the codecs have a significant effect on the scores ($\chi^2 = 944340$, $p < .001$). Results of the post-hoc Wilcoxon signed-rank test with Bonferroni correction ($\alpha = .05/10$) are reported in Table 3. Results confirm that the codecs all show statistical difference with respect to each other, with a sizable effect size (r 0.60), indicating that the effect of the codec selection on the obtained scores is considerable.

Table (3) Pairwise post-hoc test on the codecs under test, using Wilcoxon signed-rank test with Bonferroni correction.

	Z	p	r
V-PCC - MPEG Anchor	342.65	<.001	0.61
V-PCC - Greedy tiling	342.23	<.001	0.61
V-PCC - Uniform tiling	342.65	<.001	0.61
V-PCC - Hybrid tiling	338.20	<.001	0.60
MPEG Anchor - Greedy tiling	-342.65	<.001	0.61
MPEG Anchor - Uniform tiling	-333.21	<.001	0.60
MPEG Anchor - Hybrid tiling	-342.66	<.001	0.61
Greedy tiling - Uniform tiling	340.25	<.001	0.61
Greedy tiling - Hybrid tiling	-328.72	<.001	0.59
Uniform tiling - Hybrid tiling	-342.66	<.001	0.61

Table (4) Pairwise post-hoc test on the contents, using Mann-Whitney U test with Bonferroni correction.

	Z	p	r
Long dress - Loot	-153.61	<.001	0.25
Long dress - Red and black	28.05	<.001	0.05
Long dress - Soldier	-162.55	<.001	0.26
Loot - Red and black	182.41	<.001	0.29
Loot - Soldier	33.79	<.001	0.05
Red and black - Soldier	-195.63	<.001	0.31

Similarly, we performed statistical analysis on the scores to understand whether the content selection had a significant effect on the final set of scores. Due to the difference in navigation paths among different contents, we selected the unpaired Kruskal-Wallis test. Results showed statistical significance of the content selection ($\chi^2 = 61297.24, p < .001$), which was confirmed by random sampling of the data ($N = 1000$, with 1000 seeds) and combining probabilities with Fisher's method ($\chi^2 = 307470, p < .001$). Table 4 reports the results of post-hoc Mann-Whitney U test with Bonferroni correction ($\alpha = .05/6$). Statistical significance can be found between all the pairs; however, effect sizes for the pairs *Long dress - Red and black* and *Loot - Soldier* ($r = 0.05$ in both cases) indicate that effect, while apparently existing, is small. Similar results were reported in [34] for subjective tests on the same contents, indicating a difference in quality between the two groups of contents.

To assess whether statistical significance could be seen among the selected bit-rates, we ran a Friedman rank test on the scores. As expected, results confirmed that the bit-rates have a significant effect on the PSNR values ($\chi^2 = 587068.8, p < .001$), even when random sampling with $N = 1000$ and 1000 seeds was applied on the data (combined probabilities: $\chi^2 = \infty, p < .001$)[1] . Post-hoc analysis using Wilcoxon signed-rank test with Bonferroni correction ($\alpha = .05/6$) showed statistical significance, with large effect sizes, for all pairs ($Z = -383.10, p < .001, r = 0.61$ for all pairs).

Finally, a Kruskal-Wallis test revealed a significant effect of the navigation paths on the PSNR scores ($\chi^2 = 4621.5, p < .001$). To account for the difference in sample length, we performed random sampling ($N = 1000$), with 1000 seeds, and we aggregated the probabilities using Fisher's method. Results confirmed the significant effect of the navigation paths ($\chi^2 = 561800, p < .001$). Results of the post-hoc test using Mann-Whitney tests with Bonferroni correction ($\alpha = .05/325$) are reported in Figure 5. Out of 325 possible pairs, 246 of them (75.69%) presented statistical significance. Results indicate that the choice of navigation path has a significant impact on the PSNR scores. The effect size (reaching max $r = 0.19$, cfr Figure 5 (c)) suggests that the effect is not big, as it is expected that the choice of navigation path will not have same impact on the collected scores as the codec or rate selection. However, the statistical significance of a large combination of the pairs reveal that particular care should be put in differentiating navigation paths when performing evaluation of compression solutions, as it appears to have an impact on the collected PSNR scores. While the difference in sample length might explain some of the statistical significance, it is noteworthy that the significant effect is maintained when sampling the same number of frames for each navigation path. The varying length is an important feature of the collected navigation paths, as different users had variable experiences in visualizing the contents. Yet, even when reducing the paths to the same length, statistical differences are observed among the paths, indicating that the varying ways of consumption among the users lead to significantly different scores.

[1] As $p = 0$ for all the sampled pairs, the aggregated χ^2 results equal to infinity.

Table (5) Results of ANOVA on the linear model PSNR ~ Path*Content*Codec*Rate.

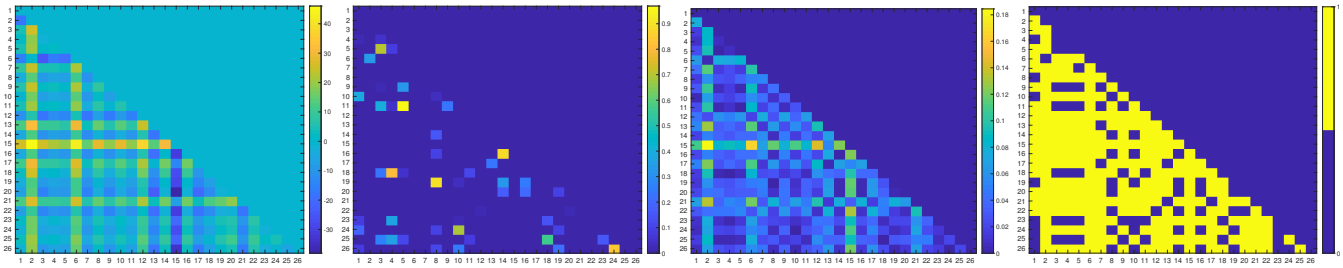
	df	MS	F	p
Path	25	1973.40	3897.27	<.001
Content	3	302310.70	597034.58	<.001
Codec	4	436070.93	861198.19	<.001
Rate	1	7574782.73	14959468.11	<.001
Path:Content	75	582.54	1150.47	<.001
Path:Codec	100	27.62	54.54	<.001
Content:Codec	12	16083.88	31764.11	<.001
Path:Rate	25	511.29	1009.74	<.001
Content:Rate	3	47865.63	94530.01	<.001
Codec:Rate	4	41393.25	81747.70	<.001
Path:Content:Codec	300	15.66	30.94	<.001
Path:Content:Rate	75	99.50	196.50	<.001
Path:Codec:Rate	100	15.96	31.52	<.001
Content:Codec:Rate	12	9058.22	17889.11	<.001
Path:Content:Codec:Rate	300	6.42	12.69	<.001
Error	781720	0.51	1	0.5

A linear regression was applied on the data to understand the ability of navigation paths, contents, codecs and rates to predict the PSNR scores, using a full interaction model. Navigation paths, contents and codecs were treated as categorical variables. The results of the ANOVA conducted on the fitted model are reported in Table 5. The adjusted R^2 for the fitted model is 0.965, indicating that our independent variables are able to account for 96.5% of the variance of the model. Moreover, results of the ANOVA confirm that the main effects and the interactions all contribute significantly to the model.

6 DISCUSSION

6.1 Limitations

We have presented a low-complexity tiling approach intended real-time systems, due to the relative simplicity with which the tiles are created and their utility is computed. In particular, our system



(a) Z (b) p (c) r (d) $p < \alpha$

Figure (5) Results of the pairwise post-hoc test on the navigation paths, using Mann-Whitney U test with Bonferroni correction. Results are shown on the lower triangular matrix.

aims at exploiting information related to the acquisition system, such as the positions of the cameras used to capture the point cloud contents, to optimize delivery with respect to the user's point of view. However, due to the lack of point cloud datasets with labelled acquisition information, we decided to emulate our approach on a widely-used point cloud dataset. The assumption that the point cloud object can be approximated by a convex hull, while it appears to be working in our case, might not generally be true, which would lead to a difference in performance. A larger point cloud dataset, complete with acquisition information, is needed to assess our approach in a more realistic scenario.

The evaluation of our proposed tiling approach for point cloud streaming has been conducted under controlled conditions. In particular, the available bandwidth for each time instance was modeled after the MPEG Test Conditions, which are defined to be representative of different degradation levels for the models under exam. Thus, in our evaluation scenario, the bit budget was a) constant for the duration of the dynamic sequence, and b) known in advance. Real network conditions, however, seldom follow such an ideal scenario. Further analysis in more adverse network conditions is needed in order to evaluate the gains a tiling approach can bring when the rate allocation can change unpredictably over time.

6.2 User Navigation Data

As adaptive streaming for traditional 2D video aims at delivering the best possible quality in varying network conditions, a great effort has been spent in the literature for testing adaptive streaming algorithms in a variety of network settings [4, 21, 36]. Similarly, user-adaptive streaming for omnidirectional content has been tested using users' interaction as the driving force [8]. However, due to the lack of representative datasets of user navigation behavior in visualizing point clouds in unconstrained 6DoF, recent work on evaluating point cloud compression [32] and adaptive streaming [37] has relied on predefined navigation paths, which might not represent how users actually consume point cloud contents. The difference we observed among the collected navigation paths, in terms of positions and angular distance, as well as in the corresponding objective scores, confirm that the selection of the appropriate navigation path has an impact on the performance of the compression and tiling approaches. In making the dataset publicly available, we hope to help the research community tailor their efforts towards a true user-centered approach, fostering new research in the field.

The dataset we collected demonstrates how users engage with uncompressed content at the highest possible quality. However, it has been shown that interactivity is affected by the level of distortion of the content under exam [3, 38]. Moreover, whereas our dataset focuses on exploring 6DoF through physical locomotion, other means of movement in VR, such as teleportation, might lead to varying patterns in user behavior, for which different tiling solutions might be more appropriate. Additional datasets should be released, exploring users' behavior beyond what has been presented in this paper.

7 CONCLUSION AND FUTURE WORK

In this paper, we present a low-complexity, user-centered approach to adaptively stream dynamic point cloud objects, based on segmenting the point cloud into non-overlapping, independently decodable tiles. We collected a dataset of user navigation paths when viewing dynamic point clouds in 6 DoF, and we evaluated our approach using objective metrics. The results show that there is significant influence of navigation paths on objective quality, indicating a need for evaluating future adaptive streaming solutions using navigation data rather than pre-defined fixed paths. Moreover, we demonstrate significant performance gains in objective quality and bitrate savings compared to non-adaptive streaming. The dataset is publicly available here: https://github.com/cwi-dis/6DoF-HMD-UserNavigationData.

Our evaluation of the tiling approaches assumes the client has omniscient knowledge of user behaviour, and is able to a) request and b) receive the tiles with the highest utility at any time instance. Thus, our evaluation represents an upper bound of the performance of the tiling approaches. In future, we aim at analysing the impact of adverse network conditions and time-series prediction, to offer an evaluation of the tiling approaches in a larger variety of settings. Moreover, we plan to evaluate our approach with data acquired from real-time point cloud capture systems, to extend and apply our approach to multi-user 6DoF telepresence applications [19] and evaluate the QoE of subjects interacting in a shared virtual space.

ACKNOWLEDGMENTS

This work is funded by the European Commission H2020 program, under the grant agreement 762111, *VRTogether*, http://vrtogether.eu/. The authors would also like to thank Dick Broekhuis and the management and support staff at CWI for their help and support during the COVID-19 pandemic.

REFERENCES

[1] MPEG 3DG and Requirements. 2017. Call for proposals for point cloud compression. *ISO/IEC JTC1/SC29 WG11 N16732, Geneva, CH* (January 2017).

[2] Evangelos Alexiou, Irene Viola, Tomás Borges, Tiago Fonseca, Ricardo De Queiroz, and Touradj Ebrahimi. 2019. A comprehensive study of the rate-distortion performance in MPEG point cloud compression. *APSIPA Transactions on Signal and Information Processing* 8 (11 2019). https://doi.org/10.1017/ATSIP.2019.20

[3] Evangelos Alexiou, Nanyang Yang, and Touradj Ebrahimi. 2020. PointXR: A toolbox for visualization and subjective evaluation of point clouds in virtual reality. In *12th International Conference on Quality of Multimedia Experience*. 6.

[4] Alexandru Aloman, A.I. Ispas, Petrica Ciotirnae, Ramon Sanchez-Iborra, and Maria-Dolores Cano. 2015. Performance evaluation of video streaming using MPEG DASH, RTSP, and RTMP in mobile networks. In *2015 8th IFIP Wireless and Mobile Networking Conference (WMNC)*. IEEE, 144–151.

[5] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. 2015. High-quality Streamable Free-viewpoint Video. *ACM Trans. Graph.* 34, 4, Article 69 (July 2015), 13 pages. https://doi.org/10.1145/2766945

[6] Cyril Concolato, Jean Le Feuvre, Franck Denoual, Frédéric Mazé, Eric Nassor, Nael Ouedraogo, and Jonathan Taquet. 2018. Adaptive Streaming of HEVC Tiled Videos Using MPEG-DASH. *IEEE Transactions on Circuits and Systems for Video Technology* 28, 8 (2018), 1981–1992.

[7] Xavier Corbillon, Francesca De Simone, and Gwendal Simon. 2017. 360-Degree Video Head Movement Dataset. In *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*. Association for Computing Machinery, New York, NY, USA, 199âĂŞ204. https://doi.org/10.1145/3083187.3083215

[8] Xavier Corbillon, Francesca De Simone, Gwendal Simon, and Pascal Frossard. 2018. Dynamic Adaptive Streaming for Multi-viewpoint Omnidirectional Videos. In *Proceedings of the 9th ACM Multimedia Systems Conference (MMSys '18)*. ACM, New York, NY, USA, 237–249. https://doi.org/10.1145/3204949.3204968

[9] Xavier Corbillon, Gwendal Simon, Alisa Devlic, and Jacob Chakareski. 2017. Viewport-adaptive navigable 360-degree video delivery. In *2017 IEEE International Conference on Communications (ICC)*. 1–7. https://doi.org/10.1109/ICC.2017.7996611

[10] Ana De Abreu, Cagri Ozcinar, and Aljosa Smolic. 2017. Look around you: Saliency maps for omnidirectional images in VR applications. In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*. 1–6.

[11] Eugene d'Eon, Bob Harrison, Taos Myers, and Philip A. Chou. 2017. 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).

[12] Fanyi Duanmu, Yixiang Mao, Shuai Liu, Sumanth Srinivasan, and Yao Wang. 2018. A Subjective Study of Viewer Navigation Behaviors When Watching 360-Degree Videos on Computers. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*. 1–6.

[13] Touradj Ebrahimi, Siegfried Foessel, Fernando Pereira, and Peter Schelkens. 2016. JPEG Pleno: Toward an Efficient Representation of Visual Reality. *IEEE MultiMedia* (October 2016).

[14] Ronald Aylmer Fisher et al. 1934. Statistical methods for research workers. *Statistical methods for research workers*. 5th Ed (1934).

[15] Stephan Fremerey, Ashutosh Singla, Kay Meseberg, and Alexander Raake. 2018. AVtrack360: An Open Dataset and Software Recording People's Head Rotations Watching 360° Videos on an HMD. In *Proceedings of the 9th ACM Multimedia Systems Conference (MMSys '18)*. Association for Computing Machinery, New York, NY, USA, 403–408. https://doi.org/10.1145/3204949.3208134

[16] Ramin Ghaznavi-Youvalari, Alireza Zare, Huameng Fang, Alireza Aminlou, Qingpeng Xie, Miska M Hannuksela, and Moncef Gabbouj. 2017. Comparison of HEVC coding schemes for tile-based viewport-adaptive streaming of omnidirectional video. In *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*. 1–6.

[17] Lanyi He, Wenjie Zhu, Ke Zhang, and Yiling Xu. 2018. View-Dependent Streaming of Dynamic Point Cloud over Hybrid Networks. In *Advances in Multimedia Information Processing – PCM 2018*. Springer International Publishing, Cham, 50–58.

[18] Mohammad Hosseini and Christian Timmerer. 2018. Dynamic Adaptive Point Cloud Streaming. In *Proceedings of the 23rd Packet Video Workshop (PV '18)*. ACM, New York, NY, USA, 25–30. https://doi.org/10.1145/3210424.3210429

[19] Jack Jansen, Shishir Subramanyam, Romain Bouqueau, Gianluca Cernigliaro, Marc Martos Cabré, Fernando Pérez, and Pablo Cesar. 2020. A Pipeline for Multiparty Volumetric Video Conferencing: Transmission of Point Clouds over Low Latency DASH. In *Proceedings of the 11th ACM Multimedia Systems Conference (MMSys '20)*. ACM, New York, NY, USA.

[20] James T Kost and Michael P McDermott. 2002. Combining dependent P-values. *Statistics & Probability Letters* 60, 2 (2002), 183–190.

[21] Jean Le Feuvre and Cyril Concolato. 2016. Tiled-Based Adaptive Streaming Using MPEG-DASH. In *Proceedings of the 7th International Conference on Multimedia Systems (MMSys '16)*. Association for Computing Machinery, New York, NY, USA, Article 41, 3 pages. https://doi.org/10.1145/2910017.2910641

[22] Wen-Chih Lo, Ching-Ling Fan, Jean Lee, Chun-Ying Huang, Kuan-Ta Chen, and Cheng-Hsin Hsu. 2017. 360° Video Viewing Dataset in Head-Mounted Virtual Reality. In *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*. Association for Computing Machinery, New York, NY, USA, 211–216. https://doi.org/10.1145/3083187.3083219

[23] Rufael Mekuria, Kees Blom, and Pablo Cesar. 2016. Design, Implementation and Evaluation of a Point Cloud Codec for Tele-Immersive Video. *IEEE Transactions on Circuits and Systems for Video Technology* (January 2016).

[24] MPEG. 2017. ISO/IEC 23000-20. Omnidirectional media application format (omaf). *ISO/IEC JTC1/SC29 WG11* (November 2017).

[25] Anh Nguyen and Bac Le. 2013. 3D point cloud segmentation: A survey. In *2013 6th IEEE Conference on Robotics, Automation and Mechatronics (RAM)*. 225–230. https://doi.org/10.1109/RAM.2013.6758588

[26] Omar A. Niamut, Emmanuel Thomas, Lucia D'Acunto, Cyril Concolato, Franck Denoual, and Seong Yong Lim. 2016. MPEG DASH SRD: Spatial Relationship Description. In *Proceedings of the 7th International Conference on Multimedia Systems (MMSys '16)*. ACM, New York, NY, USA, Article 5, 8 pages. https://doi.org/10.1145/2910017.2910606

[27] Jens-Rainer Ohm, Gary J Sullivan, Heiko Schwarz, Thiow Keng Tan, and Thomas Wiegand. 2012. Comparison of the coding efficiency of video coding standards - including high efficiency video coding (HEVC). *IEEE Transactions on circuits and systems for video technology* 22, 12 (2012), 1669–1684.

[28] Jounsup Park, Philip A. Chou, and Jenq-Neng Hwang. 2019. Rate-Utility Optimized Streaming of Volumetric Media for Augmented Reality. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 9, 1 (2019), 149–162.

[29] Feng Qian, Bo Han, Jarrell Pair, and Vijay Gopalakrishnan. 2019. Toward Practical Volumetric Video Streaming on Commodity Smartphones. In *Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications (HotMobile '19)*. Association for Computing Machinery, New York, NY, USA, 135–140. https://doi.org/10.1145/3301293.3302358

[30] Yashas Rai, Jesús Gutiérrez, and Patrick Le Callet. 2017. A Dataset of Head and Eye Movements for 360 Degree Images. In *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*. Association for Computing Machinery, New York, NY, USA, 205–210. https://doi.org/10.1145/3083187.3083218

[31] Radu Bogdan Rusu. 2011. 3D is here: Point Cloud Library. *Robotics and Automation (ICRA), 2011 IEEE International Conference* (2011).

[32] Sebastian Schwarz, Marius Preda, Vittorio Baroncini, Madhukar Budagavi, Pablo Cesar, Philip A Chou, Robert A Cohen, Maja Krivokuća, Sebastien Lasserre, Zhu Li, et al. 2019. Emerging MPEG Standards for Point Cloud Compression. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 9, 1 (March 2019), 133–148. https://doi.org/10.1109/JETCAS.2018.2885981

[33] Thomas Stockhammer. 2011. Dynamic Adaptive Streaming over HTTP –: Standards and Design Principles. In *Proceedings of the Second Annual ACM Conference on Multimedia Systems (MMSys '11)*. Association for Computing Machinery, New York, NY, USA, 133–144. https://doi.org/10.1145/1943552.1943572

[34] Shishir Subramanyam, Jie Li, Irene Viola, and Pablo Cesar. 2020. Comparing the Quality of Highly Realistic Digital Humans in 3DoF and 6DoF: A Volumetric Video Case Study. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 127–136.

[35] Evgeniy Upenik, Martin Řeřábek, and Touradj Ebrahimi. 2016. Testbed for subjective evaluation of omnidirectional visual content. In *2016 Picture Coding Symposium (PCS)*. 1–5.

[36] Jeroen Van Der Hooft, Stefano Petrangeli, Tim Wauters, Rafael Huysegems, Patrice Rondao Alface, Tom Bostoen, and Filip De Turck. 2016. HTTP/2-based adaptive streaming of HEVC video over 4G/LTE networks. *IEEE Communications Letters* 20, 11 (2016), 2177–2180.

[37] Jeroen van der Hooft, Tim Wauters, Filip De Turck, Christian Timmerer, and Hermann Hellwagner. 2019. Towards 6DoF HTTP Adaptive Streaming Through Point Cloud Compression. In *Proceedings of the 27th ACM International Conference on Multimedia (MM '19)*. Association for Computing Machinery, New York, NY, USA, 2405–2413. https://doi.org/10.1145/3343031.3350917

[38] Irene Viola and Touradj Ebrahimi. 2017. A new framework for interactive quality assessment with application to light field coding. In *Applications of Digital Image Processing XL*, Vol. 10396. International Society for Optics and Photonics, 103961F.

[39] Chenglei Wu, Zhihao Tan, Zhi Wang, and Shiqiang Yang. 2017. A Dataset for Exploring User Behaviors in VR Spherical Video Streaming. In *Proceedings of the 8th ACM on Multimedia Systems Conference (MMSys'17)*. Association for Computing Machinery, New York, NY, USA, 193–198. https://doi.org/10.1145/3083187.3083210

[40] Lan Xie, Zhimin Xu, Yixuan Ban, Xinggong Zhang, and Zongming Guo. 2017. 360ProbDASH: Improving QoE of 360 Video Streaming Using Tile-Based HTTP Adaptive Streaming. In *Proceedings of the 25th ACM International Conference on Multimedia (MM '17)*. Association for Computing Machinery, New York, NY, USA, 315–323. https://doi.org/10.1145/3123266.3123291