Full UHD 360-Degree Video Dataset and Modeling of Rate-Distortion Characteristics and Head Movement Navigation

Jacob Chakareski
New Jersey Inst. of Tech.

Ridvan Aksu
Univ. of Alabama

Viswanathan Swaminathan
Adobe Research

Michael Zink
Univ. of Massachusetts

ABSTRACT

We investigate the rate-distortion (R-D) characteristics of full ultra high definition (UHD) 360° videos and capture corresponding head movement navigation data of virtual reality (VR) headsets. We use the navigation data to analyze how users explore the 360° look-around panorama for such content and formulate related statistical models. The developed R-D characteristics and modeling capture the spatiotemporal encoding efficiency of the content at multiple scales and can be exploited to enable higher operational efficiency in key use cases. The high quality expectations for next generation immersive media necessitate the understanding of these intrinsic navigation and content characteristics of full UHD 360° videos.

CCS CONCEPTS

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

KEYWORDS

Virtual Reality, Full UHD 360-Degree Video, Head Navigation, Rate-Distortion Characteristics, 360° Video Tiling, Streaming Systems.

ACM Reference Format:
Jacob Chakareski, Ridvan Aksu, Viswanathan Swaminathan, and Michael Zink. 2021. Full UHD 360-Degree Video Dataset and Modeling of Rate-Distortion Characteristics and Head Movement Navigation. In *12th ACM Multimedia Systems Conference (MMSys '21), September 28-October 1, 2021, Istanbul, Turkey.* ACM, New York, NY, USA, 7 pages.
https://doi.org/10.1145/3458305.3478447

1 INTRODUCTION

Advances in VR augment the perceived immersion fidelity and quality of experience (QoE) of the user. Wearing a VR headset, a user can experience a 360° video for remote scene immersion and virtual teleportation. Present use cases include education and training, telepresence and telecommuting, healthcare, environmental monitoring, entertainment, and first responders [9].

360° video is a new video format that has emerged recently and is captured by an omnidirectional camera that records incoming light rays from every direction (see Figure 1, top left). It enables a 3D 360° look-around of the surrounding scene for a remote user, virtually placed at the camera location, on his/her VR headset (see Figure 1, right). After capture, the raw spherical or 360° video frames are first mapped to a wide equirectangular panorama (illustrated in Figure 1, bottom left) and then compressed using state-of-the-art (planar) video compression such as HEVC. The former intermediate step is introduced, as compression techniques operating directly on spherical data are much less mature and performing relative to traditional video compression operating on 2D video frames.

Relative to traditional video, 360° video is only partially experienced by a viewer. In particular, at any point of time, the viewer observes only a small portion of the entire 360° view sphere denoted as viewport V_c (see Figure 1, right). The viewport is selected according to the user's head orientation, as detected by the headset. To achieve a high level of immersion, very high pixel quality should be provided due to the close proximity of the headset to the user's eyes. Otherwise, any visual artifact could easily be recognized by the user and degrade the user's QoE. MPEG suggests a 12K resolution for the entire 360° panorama, 40 pixels/degree of pixel density, 100 fps video frame rate, 360° surround sound, and a maximum latency of 20 ms [13]. Full UHD (8K) 360° video represents one relevant advance towards meeting these objectives.

Due to their doubled horizontal/vertical resolution, the data rate and network bandwidth requirements of full UHD 360° videos are at least 4-fold bigger relative to their more common 4K counterparts. Simultaneously, the larger spatial resolution of 8K videos enables exploiting broader R-D coding/streaming efficiency tradeoffs. Having such knowledge, together with statistical knowledge of user navigation, can play a critical role towards optimizing the delivered quality of immersion in VR systems. In particular, instead of uniform spatial compression/streaming rate allocation, optimal 360° video quality distribution can be enabled by exploiting the uneven spatial R-D characteristics of the 360° panorama and respective user navigation patterns, as shown later. Moreover, the higher resolution and broad panoramic aspect of full UHD 360° videos can facilitate easier resource provisioning in upcoming temporal instances, by enabling a more accurate near-future content and navigation action prediction. These benefits motivate even further exploring the spatiotemporal R-D and navigation characteristics of full UHD 360° content to facilitate such advanced operations.

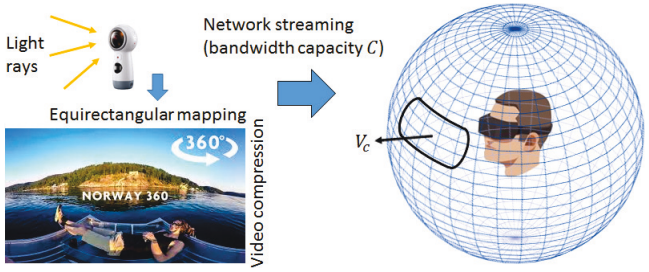


Figure 1: 360° streaming setup: [Top left] 360° **camera; [Bottom left] Equirectangular 360° panorama; [Bottom right] User viewport V_c on the 360° sphere.**

Viewport adaptive streaming has been proposed to overcome the inefficiencies of present monolithic 360° video streaming practices [12, 15, 19]. As a user only experiences the viewport region of the content, streaming the entire 360° panorama in high quality is not only unnecessary, but also unfeasible for present end-user and backbone network capabilities [6]. Streaming strictly the user viewport

MMSys 21, September 28-October 1, 2021, Istanbul, Turkey

ACM ISBN 978-1-4503-8434-6/21/09…$15.00
https://doi.org/10.1145/3458305.3478447

is an ideal approach that cannot be deployed in practice, however, due to inter-frame prediction used in video compression and the inherent round-trip latency of traditional server-client architectures that dramatically exceeds the response time requirement of VR applications. Predicting the user viewport and delivering its content ahead of time helps overcome this major challenge. To facilitate developing an accurate user navigation prediction model and carrying out effective resource allocation, having user navigation datasets and related modeling for full UHD 360° videos is critical.

The rest of the paper is organized as follows. Section 2 discusses the literature on 360° video datasets made publicly available. Our methodologies for developing data comprising R-D and head navigation characteristics of full UHD 360° video are discussed in Section 3. We then analyze the developed data with respect to various properties and formulate related models in Section 4. Subsequently, we discuss important use cases of the developed data and modeling in Section 5. Finally, we conclude in Section 6.

2 RELATED WORK

Despite the recent popularity of 360° video, only low quality 4K 360° videos are widely available and have been considered in research studies. Corbillon et al. [14] and Fremerey et al. [18] have published 360° video navigation datasets and open source recording software. David et al. [17] supplied gaze movements in addition to head navigation movements. A few studies have considered the R-D characteristics of monolithic 360° videos. Sun et al. study the quality-rate dependency of two-layer scalable encoding of 360° videos [27]. Li et al. study spherical R-D optimization and related quality metrics for 360° content. Yu et al. analyze the R-D dependency of compressed 360° videos under diverse sphere-to-planar-shape projection methods [28]. Chakareski et al. explore the spatiotemporal R-D trade-offs of tiled 360° video for end-to-end optimized streaming [4, 12]. Presently, there are no publicly available datasets and modeling of the spatiotemporal R-D and head movement navigation characteristics of tiled full UHD 360° videos.

3 DATA ACQUISITION METHODOLOGY

3.1 Full UHD 360° Sequences and Encoding

For our analysis, we have gathered 15 raw 360° video sequences, recorded at 30 frames per second. The first nine sequences stem from SJTU [21, 25] (Fig. 2 shows a snapshot of Runner). Each of these sequences is 36 seconds long with a pixel depth of 8 bits. The remaining six sequences stem from GoPro [10] (two) and InterDigital [20] (four), each of duration 10 seconds. The pixel depth of these sequences is 10 bits (GoPro) and 8 bits (InterDigital).

We opted not to include the raw 360° video sequences and their compressed instances as part of the contributed dataset. First, they are very voluminous in terms of data size and this would make publicly sharing the dataset challenging. Second, our focus is on developing spatiotemporal R-D/navigation characteristics and modeling for full UHD 360° video. Still, we have included here online links to all raw video sequences we studied. By facilitating them and the material presented therein, researchers can then reproduce all our findings and even go beyond in their investigations.

The gathered data comprise diverse content covering different use cases of 360° video. The sequences from SJTU and InterDigital are mostly static in nature, showing multiple objects moving in the background with no particular foreground object. Their static nature induces diverse head navigation movements across users and a slow temporal variation of spatial R-D characteristics. The GoPro sequences on the other hand are mostly action oriented and more dynamic in nature. They include a primary object moving with the camera and a dynamically changing background. Due to having a primary object, these sequences indicate higher correlation among users' head movements. Alike, the dynamic background suggests more sudden temporal changes of spatial R-D characteristics.

The gathered sequences are recorded as equirectangular panoramas, onto which the original spherical video data have been projected (see Figure 1, left bottom). Here, the viewport's azimuth and polar angles (φ, θ) on the sphere correspond to respective horizontal and vertical positions on the equirectangle.

We compressed each sequence using a fast application library of HEVC [3, 26], using its tiling option and an 8×8 layout. The latter has been empirically shown to provide good performance in terms of compression efficiency and processing complexity. Figure 2 shows this tiling for a video frame of the Runner sequence from the SJTU dataset. The tiles are indexed left-to-right, top-to-bottom, in a raster scan fashion. Tiling has been originally introduced to facilitate parallel processing of the video data in multi-core processor systems. Here, we use tiling to facilitate analysis of the spatiotemporal rate-distortion and head navigation characteristics of a 360° video and development of related models. These advances can then enable diverse key application use cases, as described in Sect. 5.

Figure 2: 8×8 tiling of the Runner sequence (#5).

Let a GOP-tile denote the set of tiles across the consecutive panoramic 360° video frames of a GOP at the same location in the tiling layout. Using HEVC, we compressed each GOP-tile independently from each other, across the duration of a video sequence. We used the Quantization Parameter (QP) of the video encoder to control the level of compression induced upon the data. To enable multiple qualities, we compress each 360° video multiple times, setting QP to a progressively increasing value, in each subsequent encoding run. The set of QP values is selected such that it results in a well sampled range of low-high video quality. We set the Group of Pictures (GOP) size to 30 in all our encodings.

3.2 Head Movement Recording

Table 1 shows the step by step procedure for recording a user's head movement navigation actions. We used the Oculus Rift VR headset and the Whirligig player [2] to enable navigation of a 360° video and display of the user's viewport during a recording session. We used the OpenTrack software to record the navigation actions

in real time on the computer to which the headset was linked [1]. OpenTrack captures the rotation angles yaw, pitch, and roll around three canonical axes centered on the user's head. They uniquely identify the user's head orientation at that time.

Table 1: Procedure for recording user head movement navigation data.

A user's demographic information is collected
Test sequence is shown to the user for familiarity and adjustment
Videos are selected in a random fashion
Attending researcher manually starts playback and recording
$\Rightarrow$ *Capture (yaw, pitch, roll) for each 360° video frame displayed*
After each playback the user is asked for any discomfort

To familiarize a user with the headset, prior to a recording session, we showed to the user a one minute test sequence. During this training phase, the user is asked to find a comfortable posture, calibrate the headset, and familiarize himself with the 360° look-around. During a session, a user is shown a number of the full UHD sequences. Each recording run is started manually by the attending researcher. A user is asked if he/she feels comfortable after experiencing each sequence and the recording process is terminated if the user experiences a simulator sickness [22].

3.3 Dataset Formatting

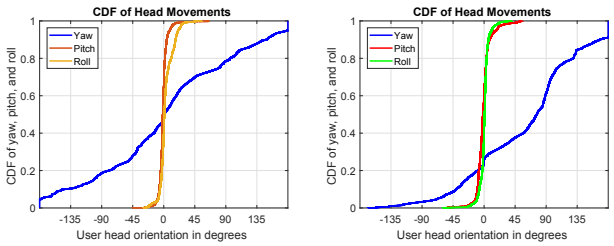
The dataset comprises five multidimensional structure arrays in Matlab. For each user and 360° video pair, the first array comprises a temporally ordered sequence of triplets of values of head rotation angles yaw, pitch, and raw, recorded at time instance t_j of every video frame j comprising the 360° video sequence, indicating the navigation actions of the user. For each 360° video and QP value pair, the second and third arrays comprise the respective encoding distortion and data rate values, for the 64 tiles of each GOP comprising that 360° video. Finally, for each 360° video, the fourth and fifth arrays comprise the exponential and power law models (the two coefficients of each model) for the 64 tiles of each GOP comprising that 360° video, for the respective rate-distortion and rate-QP dependencies. A full description of the contributed data arrays is provided in the included Readme file [11].

4 DATASET ANALYSIS AND MODELING

4.1 Head Movement Traces Analysis

We recorded a total of 121 head navigation traces with 5-12 traces per 360° video. We performed a brief demographic analysis of the users. 25% were female. In terms of VR familiarity, 17% of users have never tried it before, 66% have had little experience, and the rest were experienced users. 50% of the subjects were standing up during a session, while the rest were sitting on a rotating chair. 8% experienced a simulator sickness and terminated the recording.

In Figure 3, we examine the CDF of the Yaw, Pitch, and Roll angles recorded across the users, for the Runner and Basketball sequences. Similar trends are observed in each case. Yaw has a wider variation than both Pitch and Roll, due to the extended range of Yaw (−180°, 180°), relative to the latter two angles (−90°, 90°), as recorded by OpenTrack. This is also induced by the tendency of users to navigate the content by rotating their heads/bodies to the right/left instead of looking up or down. This characteristic, together with the observed low variation of the Pitch angle, indicates that the equatorial regions of the two 360° videos are strictly more viewed



(a) Runner sequence. **(b) Basketball sequence.**

Figure 3: Cumulative Distribution Function (CDF) of head navigation data.

than the polar regions. Moreover, the low variation of Roll indicates that users tend to make very minimal head tilting movements. Finally, comparing the Yaw CDF across the two 360° videos shows that users tend to navigate around few angles for Basketball (0°, 90°, 180°), while for Runner users tend to explore all angles evenly.

4.2 Head Navigation Likelihood Modeling

Let $\{(\varphi_j, \theta_j)\}$ denote a navigation trace for a given 360° video and user. For notational convenience, we use here the spherical coordinate system counterparts to yaw and pitch, and disregard roll, since users rarely and only marginally tilt their heads sideways, as captured by the respective CDF in Figure 3. Let $S_j^{V_c}$ denote the set of pixels in the 360° panorama occupied by V_c at time instance t_j (temporal video frame j). Similarly, let S_j^{nm} denote the set of pixels in the 360° panorama associated with tile (n, m), for $n = 1, \ldots, N$, and $m = 1, \ldots, M$. Now, let $S_j^{nm,V_c} = S_j^{V_c} \cap S_j^{nm}$ denote the set of pixels in tile (n, m) present in the user's viewport at that time instance. That is, S_j^{nm,V_c} represents the spatial area in the 360° panorama shared by tile (n, m) and V_c at time t_j.

To account for the unequal shape and size of V_c on the 360° panorama, depending on its latitude (the viewport's polar angle), we formulate next the fractions of the spatial areas of every tile, present in the user viewport V_c at t_j, as $w_j^{nm} = \frac{|S_j^{nm,V_c}|}{\sum_{n,m} |S_j^{nm,V_c}|}$. Here, $|S|$ denotes the size of a set S, in this case in number of pixels. Thus, $\{w_j^{nm}\}$ represents the normalized distribution of the spatial area of the user viewport across every tile in the 360° panorama, at time instance t_j. Given the above, we can formulate the probability (likelihood) of the user navigating tile (n, m) over a time interval spanned by the time instances $[t_i, t_j]$, as $P_{nm}^{(t_i,t_j)} = \frac{\sum_{k=i}^{j} w_k^{nm}}{j-i+1}$. In other words, $P_{nm}^{(t_i,t_j)}$ indicates how often tile (n, m) appears (at least in part) in the user viewport during navigation of the 360° video from its temporal instance t_i to t_j.

4.3 Tile Rate-Distortion Analysis

As the R-D dependency of a 360° video is convex, examining extremely high or extremely low QP values, does not lead to insightful observations. Thus, we empirically selected the QP range of 15-35 as suitable for the gathered full UHD 360° video sequences. Table 2 compiles bitrate and distortion statistics for the collected 360° video corpus, for 2 QP values. We measured distortion as the MSE of the luminance (Y) component of the 360° video frames. The reported values pertain to the first GOP and all 64 tiles of a 360° panorama. We observed consistent relative outcomes across the two QP values (quality levels) examined in Table 2, for each GOP of a 360° video.

Table 2: Raw bitrate and distortion statistics of compressed tiles.

#	Video name	Bitrate (Mbps)				Y-MSE			
		20 QP		35 QP		20 QP		35 QP	
		avg	var	avg	var	avg	var	avg	var
1	Academic	0.47	0.24	0.06	0.02	0.65	0.37	4.28	3.08
2	Basketball	1.37	2.06	0.19	0.29	0.62	0.52	4.27	5.52
3	Bridge	1.01	0.39	0.04	7.9e-3	0.50	0.20	1.18	0.63
4	Gate Night	0.90	0.41	0.04	0.01	0.48	0.22	1.46	0.96
5	Runner	0.85	0.63	0.07	0.06	0.61	0.36	2.92	2.90
6	Siyuan	0.37	0.20	0.05	0.02	0.50	0.31	2.46	1.80
7	South Gate	0.56	0.38	0.09	0.05	0.77	0.54	6.22	5.49
8	Studyroom	0.29	0.14	0.05	0.02	0.43	0.23	1.92	1.62
9	Sward	1.59	1.26	0.22	0.15	1.27	0.54	12.7	6.04
10	Chairlift	1.87	1.26	0.21	0.21	4.65	2.32	17.5	8.33
11	Skateboard	2.87	2.13	0.41	0.37	4.36	2.41	14.5	8.59
12	Gaslamp	0.51	0.31	0.06	0.04	0.59	0.24	2.91	1.86
13	Harbor	0.83	0.84	0.10	0.13	0.60	0.26	3.07	2.36
14	KiteFlite	1.95	1.33	0.26	0.23	0.83	0.37	5.51	3.49
15	Trolley	0.88	0.55	0.11	0.07	0.90	0.49	6.18	4.43

In terms of bitrate, between the two quality levels, a 10-20 times ratio is observed. A generally lower bitrate variance value is observed for the lower quality level (QP=35), yet, across the two quality levels a similar variance ratio is observed across the different videos. We observed that a full UHD 360° video is expected to be compressed on average at 30-120 Mbps bitrate for QP=20. For the lower quality level in Table 2, this range reduces to 3-15 Mbps. We also observed that depending on its content complexity, a tile compressed at QP=35 can still exhibit a higher bitrate relative to another tile compressed at QP=20.

When we examine the tile distortion values, we observe a smoother distribution for the higher quality level. This indicates that using QP=20 enables good visual quality across the video corpus. We note here that the higher bit-depth of the GoPro sequences induces a broader range of pixel intensity values, which in turn increases the range of prospective distortion values, as evident from Table 2. Moreover, the average and variance of tile distortion values increase for the lower quality level, as expected. We observed that in this case these quantities are impacted by multiple factors, indicating that a tile's R-D characteristics can more dramatically vary here, depending on its content complexity.

According to the statistics above, we selected an average example to analyze the impact of the content on bitrate and distortion. We opted for the Runner sequence (Figure 2) as such an example, as its bitrate and distortion variances are close to the median values for all four cases (columns) examined in Table 2.

The top portion of Figure 4 compares the bitrates of various tiles for the two QP cases. In the high quality case (Fig. 4, upper left), the majority of tiles with high bitrate are the tiles comprising the most distinct views. In particular, the pixel intensity diversity of Tiles 33, 39, and 40 (cf. tile indices in Figure 2) results in the highest bitrate values. In the low quality case (Fig. 4 upper right), high bitrates are observed in areas with highest content complexity. Tiles 25, 26, and 27 have the highest bitrate values and their content comprises complex shapes of trees that result in low encoding efficiency relative to other tiles, thus leading to high bitrate values.

When we observe the respective distortion values in the bottom portion of Figure 4, we can observe that the tiles with high bitrate

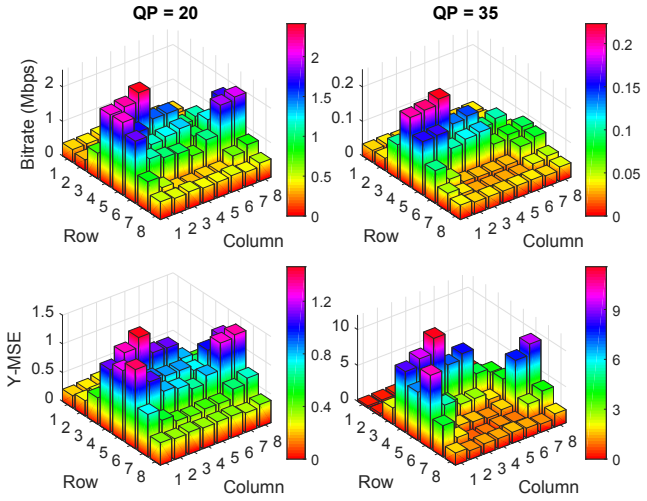


Figure 4: Comparison of tile bitrate and distortion values for Runner.

in the low quality case also demonstrate high distortion values (Fig. 4 lower right), as expected. While a similar trend can be observed in the high quality case, we observe here a smaller distortion variance among the tiles due to the low overall distortion (Fig. 4 lower left). Especially the tiles centered around tile 38 have similar distortion levels while having a smaller data volume, as well, and overall a better encoding efficiency. Still, even here, the content complexity considerably impacts the encoding quality, which motivates the advantage of using unequal QP values across the 360° panorama.

4.4 Tile Rate-Distortion Modeling

We accurately model the rate-distortion dependency of compressed GOP tiles to enable the envisioned use cases and beyond. We focus on two key dependencies here: (i) Video bitrate (R) vs. video distortion (D), and (ii) Video bitrate (R) vs. quantization parameter (QP). We explore two models, exponential ($y = c_1 e^{-d_1 x}$) and power law ($y = c_2 x^{d_2}$), to capture these two dependencies. Moreover, we analyze which model provides the best fit for each relationship.

Figure 5(a)-(c) show QP-R graphs for three compressed GOP-tiles of the Runner sequence. These tiles are selected to capture the entire range of R-D values exhibited by the respective 64 GOP-tile set examined earlier in Table 2. Concretely, Tile 5 (recall Figure 2 for the indexing of tiles and their location in the tiling layout) captures the low-end of R-D values, Tile 38 captures the mid-range, and Tile 27 the high-end. It should be noted that in all three figures the exponential model accurately fits the data. Occasional small differences are observed infrequently, yet relative to the power law model, they are negligible. In Table 3, we provide a comprehensive assessment of the accuracy of these models across the entire video corpus. These outcomes highlight the accuracy of the exponential model in capturing the QP-R dependency for compressed GOP-tiles.

Figure 6(a)-(c) show R-D graphs for these three GOP-tiles. We observe three cases here for the studied models: (i) low rate, low accuracy, (ii) medium rate, high accuracy, and (iii) high rate, high accuracy. Concretely, as the exhibited empirical R-D range for Tile 5 is very small, it is challenging to capture its R-D dependency accurately using either of the two models. For Tile 38, we observe the medium rate, high accuracy case. We can see that the power law

Table 3: Average absolute and relative prediction errors for the R-D and QP-R models.

Video Name	Rate-Distortion								QP-Rate							
	Power law				Exponential				Power law				Exponential			
	20 QP		35 QP		20 QP		35 QP		20 QP		35 QP		20 QP		35 QP	
	avg	rel(%)	avg	rel(%)	avg	rel(%)	avg	rel(%)	avg	rel(%)	avg	rel(%)	avg	rel(%)	avg	rel(%)
Academic	0.08	12.9	0.06	3.45	0.49	64.9	0.19	8.39	0.45	97.2	0.24	363	0.11	25.6	0.05	78.7
Basketball	0.08	10.5	0.07	3.37	0.35	47.4	0.17	8.59	0.39	75.8	0.63	394	0.09	17.1	0.09	62.2
Bridge	0.04	6.46	0.05	4.39	0.08	15.8	0.17	12.6	0.73	71.3	0.62	1522	0.12	12.7	0.03	82.5
Gate Night	0.04	7.25	0.05	4.19	0.13	25.8	0.19	13.0	0.72	82.0	0.55	1297	0.13	15.8	0.03	68.7
Runner	0.04	6.52	0.04	2.95	0.36	48.8	0.22	12.0	0.72	100	0.57	921	0.19	21.5	0.04	66.0
Siyuan	0.05	10.0	0.06	4.97	0.32	52.7	0.17	11.3	0.39	118	0.19	395	0.08	28.4	0.04	82.2
South Gate	0.11	17.5	0.07	5.01	0.61	67.2	0.21	9.47	0.56	95.6	0.36	331	0.10	18.7	0.05	68.1
Studyroom	0.09	15.9	0.12	8.39	0.31	60.5	0.17	14.0	0.50	156	0.22	435	0.11	34.2	0.04	77.1
Sward	0.12	10.1	0.05	0.61	1.00	80.3	0.28	3.03	0.64	45.9	0.72	338	0.21	18.2	0.09	46.4
Chairlift	0.36	7.48	0.55	2.69	0.88	20.4	1.47	9.36	0.73	45.3	1.08	655	0.29	17.8	0.11	70.4
Skateboard	0.28	7.52	0.53	3.88	0.46	13.2	1.03	9.71	0.52	45.8	1.36	613	0.21	20.6	0.20	73.8
Gaslamp	0.10	16.2	0.10	4.88	0.44	67.3	0.20	9.44	0.70	161	0.39	670	0.18	42.2	0.03	46.4
Harbor	0.10	16.2	0.09	5.45	0.42	59.2	0.19	8.34	0.72	130	0.54	783	0.20	39.3	0.03	44.1
KiteFlite	0.07	9.81	0.06	2.24	0.65	76.6	0.25	6.67	0.71	59.9	0.89	548	0.17	14.3	0.06	28.2
Trolley	0.14	14.5	0.09	3.68	0.70	68.4	0.26	8.53	0.69	102	0.65	517	0.19	30.7	0.05	38.6

model accurately captures the actual R-D values for most of the distortion levels exhibited in the graph, and relative to the exponential model, has a smaller error margin. Finally, Tile 27 exhibits the high rate, high accuracy case. We observe here that the power law model again exhibits a much higher accuracy. Moreover, though the actual R-D points in the graph are closer to each other, relative to those in Figure 6(c), a slightly lower accuracy of the power law model is observed in this setting relative to Figure 6(c). This is due to the higher range of actual R-D values observed here.

Next, in Figure 5d (QP-R) and Figure 6d (R-D), we examine the cumulative distribution function (CDF) of the relative prediction (model) error, for the two models and empirical dependencies, in the case of three select 360° videos. In both figures, the solid line indicates the Runner sequence, the dashed line indicates the Basketball sequence, and the dotted line indicates the GateNight sequence.

We can see from Figure 5d that the exponential and power law QP-R models perform very distinctly. These outcomes align well with the earlier analysis of the graphs from Figure 5(a)-(c). Concretely, we observe from Figure 5d that two out of three CDF graphs for the power law model's relative prediction error in the case of QP=35 lie outside the considered x-axis range. We also observe that all three videos show very similar trends here in the case of the exponential model, though their actual QP-R values are distinct. We can see from Figure 6d that the relative performance of the two models has been reversed in the case of the R-D dependency, with the power law model providing a much more accurate prediction now, as also supported by our earlier analysis of the graphs from Figure 6(a)-(c). Moreover, an opposite trend is consistently observed here with respect to the two QP values examined, across both models and three videos considered. Concretely, the relative prediction error appears smaller and features a steeper CDF for the low quality level now. Finally, a much higher divergence is observed among the three CDF graphs for the exponential model and QP=20.

In Table 3, we examine the average absolute and relative prediction errors for the two models across the entire video corpus. We can see from the left half of the table that the power law model consistently outperforms the exponential model in the case of the R-D dependency. For instance, for Runner and QP=25, the power law model exhibits an average absolute error of 0.04 Y-MSE. This quantity is 0.36 for the exponential model (nine times higher). On the other hand, we can see from the right half of Table 3 that the exponential model consistently fits the data more accurately in the case of the QP-R dependency. All the outcomes observed in Table 3 are consistent with our earlier analysis.

5 KEY USE CASES

5.1 Streaming System Rate Allocation

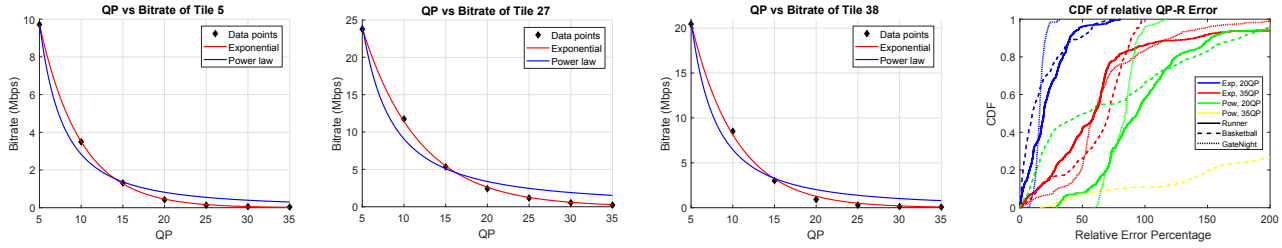
The developed rate-distortion and navigation modeling can be used to enable effective resource allocation in future streaming systems delivering full UHD 360° video content. Concretely, let P_{ij} denote the likelihood that tile (i, j) will appear in the viewport of a user over the next GOP of the content to be streamed to the user. Similarly, let $D_{ij}(R_{ij})$ denote the rate-distortion dependency that encoding the content associated with this GOP-tile exhibits. These modeling concepts were introduced and formulated in Section 4.

Let $\sum_{ij} P_{ij} D_{ij}(R_{ij})$ denote the expected user viewport distortion, given that the streaming system allocated data rates $\{R_{ij}\}$ to the compressed tiles of that GOP. Let C denote the available streaming data rate that the server can use to deliver the content to the user. The server can aim to find the allocation $\{R_{ij}\}$ that will minimize the expected viewport distortion such that the aggregate streaming rate does not exceed C. We formally write this optimization as:

$$\min_{\{R_{ij}\}} \sum_{ij} P_{ij} D_{ij}(R_{ij}), \quad \text{subject to: } \sum_{ij} R_{ij} \leq C. \tag{1}$$

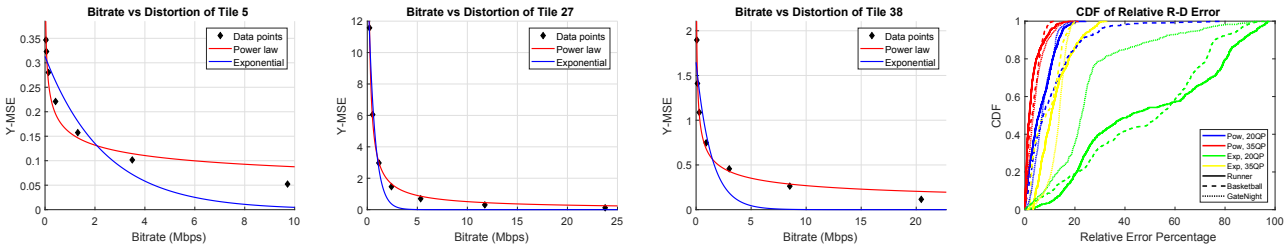
This problem is convex, as its objective and constraint are concave functions. In particular, the dependencies $D_{ij}(R_{ij})$ are concave functions and the multipliers P_{ij} are smaller than one. Thus, the objective function is also concave and the linear constraint is convex/concave at the same time. Hence, (1) can be solved effectively using convex optimization methods to produce the optimal allocation $\{R^*_{ij}\}$ [8] that will maximize the delivered immersion quality.

Similarly, in receiver-driven DASH streaming, HTTP/2 Push techniques have been explored towards low-latency operation [23]. Here, the server can benefit from the navigation likelihoods $\{P_{ij}\}$ to anticipate accurately which GOP-tiles the user is likely to request next, and preemptively push them to the client, to save the



(a) Tile 5 (b) Tile 27 (c) Tile 38 (d) CDF of Error

Figure 5: ***(a), (b), and (c):*** **Comparison of QP-R models for Runner.** ***(d):*** **CDF of the relative error for the two models (Runner, Basketball, GateNight).**



(a) Tile 5 (b) Tile 27 (c) Tile 38 (d) CDF of Error

Figure 6: ***(a), (b), and (c):*** **Comparison of R-D models for Runner.** ***(d):*** **CDF of the relative error for the two models (Runner, Basketball, GateNight).**

round-trip latency induced by the respective client request, otherwise. Moreover, in recent neural adaptive streaming [16], a client employs a deep learning super-resolution model to generate higher resolution video from a streamed lower-resolution version. The respective server can employ the same model to anticipate the client's behavior and prospectively improve the delivered video quality by allocating the remaining available network bandwidth towards minimizing the residual error after the super-resolution process at the client. Our dataset can be employed to train, simulate, and improve such neural streaming approaches.

5.2 Distribution System Caching/Transcoding

To meet the growing popularity, caching and prefetching mechanisms will have to be deployed in distribution systems to ensure scalable streaming of full UHD 360° video. Also, DASH-compliance of such streaming methods would enable interoperability between the different components (server, caches, clients) of such systems. Our dataset and modeling can contribute to both aspects.

Concretely, the developed tile navigation likelihoods $\{P_{ij}\}$ capture the likeliness that a user will direct his viewport towards a specific set of tiles during a given time interval. This can be used to enable preprocessing and prefetching of tiles that are very likely to be navigated by the viewer. In a distribution system where a cache is serving a large number of users, this information can be used to prefetch to the cache highly popular tiles for the users. Moreover, benefiting from the developed R-D modeling and analysis in (1), the cache can then transcode these tiles to minimize the delivered viewport distortion for each user. These two strategies advance also the scalability of the system, as dynamic encoding can be carried out at the cache for a subset of clients, thus avoiding the need for central operation at the back-end server for all clients.

Similarly, the R-D modeling can benefit a DASH streaming method with minor adaptations. In DASH, a client already determines its download bitrate for the most recent GOP-tile set it has requested. Using its most recent download bitrate values and a prediction method [7], the client can then predict the expected download bitrate C for the set of tiles $\mathcal{T}$ requested for the next GOP/segment. This information can be signaled back to the server that can then employ an equivalent analysis to (1) (summation is carried out only over tiles in $\mathcal{T}$ and the respective P_{ij} values are set to one) to determine the optimal compression rates $\{R^*_{ij}\}$ for each such tile.

5.3 Perceptual Studies and Immersion Saliency

The developed dataset and modeling can benefit diverse studies that explore the perceptual interaction and quality of experience of users in immersive environments. For instance, the navigation information and R-D characteristics can help understand the viewing behavior of users of full UHD 360° video content and develop related behavioral models of viewing navigation and fixation points in the immersive environment [24]. Similarly, salient aspects of the content can be identified to investigate saliency prediction methods and facilitate diverse applications such as 360° video synopsis, compression, and streaming [5].

6 CONCLUSION

We studied the R-D characteristics of full UHD 360° videos and captured corresponding head movement navigation data of VR headsets. We used the navigation data to analyze how users explore the 360° look-around panorama for such content and formulated related statistical models. The developed R-D characteristics and modeling capture the spatiotemporal encoding efficiency of the content at multiple scales and can be exploited to enable higher operational efficiency in key use cases, in synergy with the formulated navigation models. The high-quality expectations of future immersive media motivate the understanding of these intrinsic navigation and content characteristics of full UHD 360° videos.

ACKNOWLEDGMENT

The work of J. Chakareski and R. Aksu was supported by NSF Awards CCF-1528030, ECCS-1711592, CNS-1836909, and CNS-1821875. The work of M. Zink was supported by NSF Award CNS-1901137.

REFERENCES

[1] [n.d.]. OpenTrack: Head Tracking Software. https://github.com/opentrack/opentrack

[2] [n.d.]. Whirligig VR Media Player. https://www.whirligig.xyz/

[3] [n.d.]. x265 HEVC Encoder / H.265 Video Codec. https://www.x265.org/

[4] Ridvan Aksu, Jacob Chakareski, and Viswanathan Swaminathan. 2018. Viewport-driven Rate-distortion Optimized Scalable Live 360 Video Streaming. In *Proc. IEEE ICME Workshops* (San Diego, CA).

[5] Duin Baek, Hangil Kang, and Jihoon Ryoo. 2020. SALI360: Design and implementation of saliency based video compression for 360-degree video streaming. In *Proc. ACM MMSys Conf.*

[6] Bo Begole. 2016. Why The Internet Pipes Will Burst When Virtual Reality Takes Off. Forbes Magazine.

[7] Divyashri Bhat, Amr Rizk, Michael Zink, and Ralf Steinmetz. 2018. SABR: Network-Assisted Content Distribution for QoE-Driven ABR Video Streaming. *ACM Trans. Multimedia Comput. Commun. Appl.* 14, 2s, Article 32 (April 2018), 25 pages. https://doi.org/10.1145/3183516

[8] S. Boyd and L. Vandenberghe. 2004. *Convex Optimization.* Cambridge University Press.

[9] Jacob Chakareski. 2019. UAV-IoT for next generation virtual reality. *IEEE Transactions on Image Processing* 28, 12 (2019), 5977–5990.

[10] Jacob Chakareski and Ridvan Aksu. 2021. GoPro 8K 360-degree video sequences. https://alabama.app.box.com/s/h9qiy5rsq5ukg69q6gl0jn0gj66znbon (Courtesy of Jill Boyce, Intel, Inc.).

[11] Jacob Chakareski and Ridvan Aksu. 2021. NJIT Full UHD 360-Degree Video Dataset and Modeling of Rate-Distortion Characteristics and Head Movement Navigation. https://alabama.app.box.com/v/8k-360-dataset

[12] Jacob Chakareski, Ridvan Aksu, Xavier Corbillon, Gwendal Simon, and Viswanathan Swaminathan. 2018. Viewport-driven Rate-distortion Optimized 360-degree Video Streaming. In *Proc. IEEE Int'l Conf. Communications* (Kansas City, MO).

[13] M. Champel, T. Stockhammer, T. Fautier, E. Thomas, and R. Koenen. 2016. Quality Requirements for VR. In *Proc. 116th MPEG Meeting of ISO/IEC JTC1/SC29/WG11.*

[14] Xavier Corbillon, Francesca De Simone, and Gwendal Simon. 2017. 360-Degreee Video Head Movement Dataset. In *Proc. ACM MMSys Conf.* Taipei, Taiwan.

[15] X. Corbillon, G. Simon, A. Devlic, and J. Chakareski. 2017. Viewport-adaptive navigable 360-degree video delivery. In *Proc. IEEE ICC.* Paris, France.

[16] Mallesham Dasari, A. Bhattacharya, Santiago Vargas, Pranjal Sahu, A. Balasubramanian, and S. Das. 2020. Streaming 360-Degree Videos Using Super-Resolution. In *Proc. IEEE INFOCOM.*

[17] Erwan J. David, Jesús Gutiérrez, Antoine Coutrot, Matthieu Perreira Da Silva, and Patrick Le Callet. 2018. A Dataset of Head and Eye Movements for 360 Videos. In *Proc. ACM MMSys Conf.* Amsterdam, Netherlands.

[18] 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 *Proc. ACM MMSys.* Amsterdam, Netherlands.

[19] M. Hosseini and V. Swaminathan. 2016. Adaptive 360 VR Video Streaming: Divide and Conquer. In *Proc. IEEE International Symposium on Multimedia (ISM).* San Jose, CA, USA, 107–110.

[20] InterDigital, Inc. 2021. 8K 360-degree video sequences. https://www.interdigital.com/download_resource/FileName-360-degree-video-sequence (Instruction: Replace "FileName" in URL with {Gaslamp, Harbor, KiteFlite, or Trolley}).

[21] X. Liu, Y. Huang, L. Song, R. Xie, and X. Yang. 2017. The SJTU UHD 360-Degree Immersive Video Sequence Dataset. In *Proc. ICVRV2017.* Zhengzhou, China.

[22] J. D. Moss and E. R. Muth. 2011. Characteristics of headmounted displays and their effects on simulator sickness. *Human Factors: The Journal of the Human Factors and Ergonomics Society* 53, 3 (June 2011), 308–319.

[23] Stefano Petrangeli, Viswanathan Swaminathan, Mohammad Hosseini, and Filip De Turck. 2017. Improving Virtual Reality Streaming Using HTTP/2. In *Proc. Multimedia Systems Conference* (Taipei, Taiwan). ACM, 225–228.

[24] Vincent Sitzmann, Ana Serrano, Amy Pavel, Maneesh Agrawala, Diego Gutierrez, Belen Masia, and Gordon Wetzstein. 2018. Saliency in VR: How do people explore virtual environments? *IEEE Trans. Visualization and Computer Graphics* 24, 4 (April 2018), 1633–1642.

[25] SJTU Media Lab. 2017. SJTU 8K 360-degree video sequences. http://medialab.sjtu.edu.cn/vr8K/index.html

[26] G. J. Sullivan, J.R. Ohm, W.J. Han, and T. Wiegand. 2012. Overview of the High Efficiency video coding (HEVC) standard. *IEEE Trans. Circuits and Systems for Video Technology* 22, 12 (Dec. 2012), 1649–1668.

[27] Liyang Sun, Fanyi Duanmu, Yong Liu, Yao Wang, Yinghua Ye, Hang Shi, and David Dai. 2018. Multi-path Multi-tier 360-degree Video Streaming in 5G Networks. In *Proc. ACM MMSys.* ACM, Amsterdam, Netherlands.

[28] M. Yu, H. Lakshman, and B. Girod. 2015. A Framework to Evaluate Omnidirectional Video Coding Schemes. In *Proc. IEEE ISMAR.* Fukuoka, Japan.