Advanced Predictive Tiles Selection Using Dynamic Tiling for Prioritized 360° Video VR Streaming

ABID YAQOOB* and GABRIEL-MIRO MUNTEAN, School of Electronic Engineering, Dublin City University, Ireland

The widespread availability of smart computing and display devices such as mobile phones, gaming consoles, laptops, and tethered/untethered head-mounted displays has fueled an increase in demand for omnidirectional (360°) videos. 360° video applications enable users to change their viewing angles while interacting with the video during playback. This allows users to have a more personalized and interactive viewing experience. Unfortunately, these applications require substantial network and computational resources that the conventional infrastructure and end devices cannot support. Recent-proposed viewport adaptive fixed tiling solutions stream only relevant video tiles based on user interaction with the virtual reality (VR) space to use existing transmission resources more efficiently. However, achieving real-time accurate viewport extraction and transmission in response to both head movements and bandwidth dynamics can be challenging, which can impact the user's Quality of Experience (QoE). This paper proposes innovative dynamic tiling-based adaptive 360° video streaming solutions in order to achieve high viewer QoE. First, novel and easy-to-scale tiling layout selection methods are introduced, and the best tiling layouts are employed in each adaptation interval based on the prediction-assisted visual quality metric and the observed viewport divergence. Second, a novel proactive tile selection approach is presented, which adaptively extracts tiles for each selected tiling layout based on two low-complex viewport prediction mechanisms. Finally, a practical dynamic tile priority-oriented bitrate adaptation scheme is introduced, which uniformly distributes the bitrate budget among different tiles, during 360° video streaming. Extensive trace-driven experiments are conducted to evaluate the proposed solutions using head motion traces from 48 VR users for five 360° videos with tiling layouts of 4x3, 6x4, and 8x6 and segment durations of 1s, 1.5s, and 2s. The experimental evaluations show that the dynamic video tiling solutions achieve up to 11.2% more viewport matches and an average improvement in QoE of 9.7%-18% compared to state-of-the-art 360° streaming approaches

CCS Concepts: • **Virtual Reality → Multimedia Streaming**; • **Information Processing**; • **Communication**;

Additional Key Words and Phrases: 360° video streaming, Dynamic Tiling, Tiles selection, Bitrate adaptation, QoE.

ACM Reference Format:
Abid Yaqoob and Gabriel-Miro Muntean. 2022. Advanced Predictive Tiles Selection Using Dynamic Tiling for Prioritized 360° Video VR Streaming. *ACM Trans. Multimedia Comput. Commun. Appl.* 1, 1 (May 2022), 28 pages. https://doi.org/XXXXXXX.XXXXXXX

Authors' address: Abid Yaqoob, abid.yaqoob2@mail.dcu.ie; Gabriel-Miro Muntean, gabriel.muntean@dcu.ie, School of Electronic Engineering, Dublin City University, Dublin, Ireland, D09 Y5N0.

This work was supported by the Science Foundation Ireland (SFI) via the Frontiers Projects grant 21/FFP-P/10244 (FRADIS) and Research Centres grant 12/RC/2289_P2 (INSIGHT). (Corresponding author: A. Yaqoob.)

XXXX-XXXX/2022/5-ART $15.00
https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

Recently, 360° virtual reality (VR) video has improved the traditional streaming format by allowing the viewer to feel fully immersed in the video by providing a complete spherical field of view (FoV). This is achieved by capturing video from all directions using multiple cameras and then stitching the video together into a single, seamless sphere. Users can have an incredibly immersive experience, especially when using high-resolution head-mounted display (HMD) devices [53]. However, remote transmission and rendering of ultra-high-resolution panoramic content significantly exceeds the capacity of conventional infrastructure. However, the emerging 5G and beyond wireless network technologies are expected to bridge the current performance gap by offering higher network flexibility, transmission capacity, and mobility support [3].

Currently, a standard way to mitigate the transmission of ever-increasing 360° video services is through viewport-based adaptive streaming frameworks (i.e., monolithic streaming [7, 64] and tile-based streaming [37, 54]). Multiple versions of pre-defined viewports are prepared on the server-side in monolith streaming. The entire spherical frame provides higher viewport quality and gradually lower outside quality for each viewing feedback. Contrarily, tile-based streaming lower these requirements by spatially partitioning the video frames into independently encodable rectangular video parts known as tiles [21, 59]. The VR user can envision the FoV tiles in higher quality levels [31, 65] compared to the other tiles which are delivered in lower resolution [12, 38] or even discarded [49]. The user's head motion patterns are an essential measurement for quality-efficient remote transmission. However, it is limited in many cases. Viewport prediction can help to reduce the time it takes for new tiles to be loaded as the viewer changes their viewing angle, improving the overall streaming experience. The client can allocate more bits to these tiles based on visual visit information [28].

The spatial partitioning structure of tiles plays a vital role in balancing viewport availability and bandwidth utilization. Existing fixed tiling layout solutions [15, 16, 36] stream variable quality views in order to reduce data transmission. However, this can still lead to poor visual boundaries and inefficient use of bandwidth. In contrast, a dynamic tiling-based streaming framework reduces redundant data and provides improved FoV availability for different viewing behaviours of users. However, it is challenging to support dynamic tiling-based streaming under complex viewing patterns. Similarly, identifying and selecting prioritized views is necessary but not simple. Using traditional bitrate adaptation heuristics [42, 52] for tile-based streaming in the presence of various uncertainties (such as connection speed, user movements, segment sizes, etc.) is not practical due to the spatial and temporal separation of 360° content. Suppose learning-based [22, 45, 46], or controlled adaptation technique [60, 61] can correctly calculate the bitrate for the next segment in real-time. Still, it is strenuous to best match the quality scores due to the instantaneous short-term viewport updates.

This paper introduces two novel Dynamic video Frames Tiling-based (DFT) 360° video streaming solutions involving a three-tier adaptation in terms of tiling layout adaptation, streaming tiles selection, and bitrate adaptation. In an end-to-end remote 360° video transmission, the first solution DFT1 decides an optimal tiling layout based on a newly proposed priority-assisted weighted visual quality metric. The second solution referred to as DFT2 intelligently adapts the tiling version based on the head movement prediction accuracy for each video segment. The proposed DFTs solutions perform prioritized tiles selection by classifying streaming regions into the following cases: (1) Case 1: Fixed viewport with no marginal tiles; (2) Case 2: Fixed viewport with marginal tiles; (3) Case 3: Extended viewport with no marginal tiles. Finally, a DFT bitrate adaptation heuristic is designed in such a way as to support the dynamic tiling-based streaming framework by implementing

prioritized bitrate budget distribution between different tile groups. This paper has the following main contributions:

(1) **Adaptive Tiling Layouts Switching based on Visual Quality and Prediction Relevance:** Two innovative solutions that dynamically determine tiling layouts, taking into account both visual quality prioritization (DFT1) and viewport prediction accuracy (DFT2) during each segment playback are introduced. In particular, DFT1 selects the highest-quality tiling layout to deliver an optimal viewing experience, effectively addressing the complexity and scalability issues faced by existing solutions. The second strategy DFT2 tailors tiling layouts based on viewport prediction performance, thereby enhancing viewport availability across a variety of motion content.
(2) **Efficient Computation of Streaming Regions:** A low-complexity, precise solution for determining the optimal arrangement of streaming tiles, utilizing a combination of two viewport prediction mechanisms, where the viewport is defined in terms of 110° angles in both horizontal and vertical directions is described. This approach employs advanced tiles classification, i.e., dynamic viewport and marginal regions, in order to improve the displayed viewport's adaptability in response to non-native head movements.
(3) **Region-based Uniform Bitrate Adaptation:** A dynamic tiling-based uniform bitrate adaptation algorithm that incorporates diverse adaptation policies, including aggressive, weighted, and conservative is proposed. This novel algorithm proactively allocates the available bandwidth to specific spatial regions and optimizes viewer experience according to the desired adaptation strategy.

We present extensive experimental evaluations using real head motion traces of 48 VR users considering five 4K videos prepared in three tiling layouts (4x3, 6x4, 8x6) and with three segment durations (1s, 1.5s, 2s). Experimental results show that DFT improves the streaming performance measured in terms of viewport overlap (8.6%-11.2%) and QoE (9.70%-18%) under dynamic bandwidth conditions in comparison to popular fixed tiling-based and dynamic tiling-based solutions.

This work presents significant new contributions compared to our previously proposed solutions, CFOV [55] and DVS [56]. In comparison to [55] and [56], the proposed solutions have the following new points. First, two novel options for tiling layout selection are proposed which can improve viewport availability and reduce the transmission of redundant pixels under variable head movement prediction accuracy. Secondly, the DFT tiles selection mechanisms are comprehensively different from those proposed before. DFTs employ adaptive marginal and extension region selections, which are fine-grained and help with highly dynamic viewing patterns. DVS considered visual complexity and circular distance between viewpoints to classify viewport, marginal, and background tiles sets, while CFOV considered fixed and extended FoV scenarios and adopts a wider marginal region based on prediction results. Thirdly, DFTs solutions introduce a novel bitrate adaptation algorithm designed to handle dynamic adaptation decisions for multiple tiling layouts, which is a significant new contribution in contrast with the previously introduced fixed tiling-based solutions. DVS specifically switches between uniform (per-region) and non-uniform (per-tile) quality allocation strategies, while DFT considers per-region uniform bitrate adaptation. Finally, a significantly expanded testing setup is used to evaluate comparatively the streaming behaviours of both fixed and dynamic tiling-based solutions.

Paper Organization: Section 2 discusses the most recent literature on 360° tile-based streaming. Section 3 details the structure of the proposed 360° adaptation framework and problem formulation. The details of tiling layout selection, tiles selection, and tiles bitrate adaptation are introduced in Section 4. Section 5 presents the experimental settings, results, and performance analysis. Finally, Section 6 offers conclusive remarks.

2 BACKGROUND AND RELATED WORKS

This section presents the important technical background linked to our research and provides a comprehensive overview of the most recent streaming techniques, applications, and limitations.

2.1 Fixed Viewport-based Streaming

In this streaming approach, the size of the viewer window (the "viewport") is fixed. The system delivers a higher-quality version of the video to the portion of the video that is within the viewport. This approach takes into account the viewer's dynamic motion patterns, as the viewport is adjusted to follow their movements.

Hosseini et al. [16] proposed a priority-based bitrate adaptation (PBA) algorithm for 360° video streaming that takes into account the location of different tiles within the video (central, surrounding, and outside). The algorithm starts by assigning the lowest quality version of the video to the entire segment, and then gradually increases the quality of the central tile to the highest level, followed by the surrounding and outer tiles. However, the PBA algorithm was evaluated using a VR setup with a 2K resolution and videos encoded using H.264/AVC, which may not be optimal for enriched 360° videos. Similarly, Chen et al. [4] proposed a system for adapting the quality of 360° video based on the location of different tiles within the viewport, with higher priority given to tiles in the centre and lower priority given to tiles in the corners. However, this system does not take into account viewer motion or use any prediction mechanism and was evaluated using fixed network connections. Nasrabadi et al. [28] employed a cube map projection-based scalable video coding scheme where each face of the cube was divided into two horizontal and two vertical tiles and encoding was performed using one base layer and two enhancement layers. The experimental evaluations using four streams of different spatiotemporal complexities demonstrate that compared to the non-scalable coding, layer-assisted tiles coding results in fewer rebuffering events while offering improved quality. Hooft et al. [15] proposed Uniform ViewPort quality (UVP) solution that is designed for use with a fixed viewport. UVP divides the video into two regions: the viewport, which is the portion of the video that is currently being displayed to the viewer, and the non-viewport, which is the rest of the video. The tiles in both regions are arranged using a prediction approach that extrapolates the viewer's head motion to anticipate their upcoming viewing points. However, this method was only tested using three videos with a single segment duration. Wei et al. [45] proposed a hybrid adaptation solution to control viewpoint prediction and adaptation decisions by leveraging a deep reinforcement learning (DRL) method to continuously compute first the segment bitrate and then the per-tile bitrate based on predicted fixed viewport maps and use them in a cooperative bargaining game theory approach. The proposed solution processes head movement and eye fixation information to adjust the prioritized quality decisions within the spatial and temporal domains.

2.2 Marginal Region-based Streaming

In this streaming approach, a spatial extension, known as the "marginal area," is defined around the viewport. The purpose of the marginal region is to provide a buffer around the viewport to account for possible errors in head movement prediction. Petrangeli et al. [36] proposed an adaptive virtual reality (AVR) streaming approach which divides the tiles of the 360° video into viewport, adjacent, and outside groups. The authors collected viewport traces using the Gear VR framework while ten users watched a single 360° video. However, the evaluation was limited to a single 60-second long 360° video clip. Ben Yahia et al. [2] divided the equirectangular frame into viewport, marginal, immediate background, and far background regions. The proposed model involves two viewport prediction intervals, i.e., before and during the delivery of the same segment. The client assigns

variable weights to different priority regions and can update the resource allocation based on updated prediction results. Zou et al. [65] introduced a convolutional neural network (CNN)-based prediction mechanism and then distributed the communication resources for the quality selection of predicted tiles. The proposed solution maps the spherical representation to the planer projection to calculate the viewing probability of each tile. The tiles are then divided into viewport, marginal, and background tiles groups. The marginal tiles surround the viewport in all directions, similar to [36]. However, CNN-based viewport prediction models are computationally expensive and are difficult to extend for different videos. Zhang et al. [57] proposed a simple yet effective buffer-based quality-aware bitrate adaptation algorithm to allocate different quality levels to the viewport, marginal, and outside tiles. The experimental evaluations using three 4K test sequences prepared in a 6x4 tiling layout under staged bandwidth variations show that the proposed solution favours the high visible quality levels with considerable navigation smoothness. However, concise simulations were performed for each video content (about 10s). Yadav and Ooi [50] modelled the per-tile bitrate allocation problem as a multiclass knapsack problem based on a dynamic profit function of the current FoV, buffer level, and per-tile representation level. The proposed tile-rate allocation solution based on the previously proposed non-tiled ABR algorithm [51], achieves good results in terms of reducing playback interruptions and quality switches while improving the overall quality and bandwidth savings. However, this approach may lead to higher spatial quality variance within the viewport, and the use of a separate buffer for each tile can cause the playback of the entire video to stall if one of the tiles is not downloaded in time.

2.3 Extended Viewport-based Streaming

Extended viewport-based streaming is a technique of delivering 360° video in which the viewport is virtually extended by a certain percentage, typically 10-30%, in order to provide a buffer around the viewport to account for viewer movements. Hooft et al. [15] proposed a quality adaptation approach by considering the extended viewport (full-frame) region. This approach, called Centre Tile First (CTF), focuses on improving the quality of the centre or viewpoint tile, and then gradually increases the quality of the remaining tiles. CTF was evaluated considering the weighted viewport quality metric, which assigns higher weights to the centre tile quality and gradually lowers the weights towards the end tiles. It was shown to outperform the uniform viewport quality allocation solution UVP for the weighted viewport quality metric. However, when tested using average viewport quality, UVP performs better than CTF.

He et al. proposed [14] a joint adaptation solution that adjusts both the size of the viewport and the bitrate of the video based on network conditions. The algorithm measures the round trip time (RTT) of the network connection and uses this information to determine the viewport size and the necessary bitrate for smooth streaming. Simulation results using the Network Simulator (NS)-3 tool showed that this adaptable viewport coverage approach can improve the quality of the streaming experience. However, the details of this work, such as the viewport prediction mechanism, the dataset and tiling layout used, and the content resolution, are not provided. Similarly, Hu et al. [17] proposed a system called MELiveOV for live streaming high-resolution 360° video using 5G-enabled edge servers to distribute processing tasks. This edge-based live streaming system adjusts the size of the viewport based on network conditions, with a smaller viewport (90°) requested in higher bitrates under poor network conditions and a larger viewport (120°) selected for streaming under ideal conditions. However, the performance of this work was only compared to a viewport-independent streaming approach. Guo et al. [13] proposed a solution for 360° video streaming that takes into account random motion patterns and variable network conditions for each viewer, and tries to use multicast opportunities to reduce redundant data transmissions. The proposed solution computes the actual viewport tiles for the current user and adds more tiles to the viewing region based on the

common interest of other users. The authors considered 100° viewport coverage and an extra 15° in both horizontal and vertical directions. Similarly, Long et al. [24] optimized the overall utility of multiple users in a wireless network environment with a single server. The proposed solution takes into account factors such as transmission time, video quality smoothness, and power constraints in order to maximize the aggregated utility of the users.

proposed a method for optimizing the aggregated utility of multiple users in a single-server multi-user wireless network environment by considering transmission time, video quality smoothness, and power constraints.

2.4 Dynamic Tiling-based Streaming

In dynamic tiling-based adaptive streaming, multiple tiling layouts are prepared on the server-side in order to optimize the delivery of a 360° video to a viewer. The tiling layout that is used for a particular viewer may be changed dynamically in order to adapt to their viewing and network conditions. Khiem et al. [39] investigated the impact of tiling layouts on interactive zoomable video streaming by employing the dynamic cropping of regions of interest (RoI). The authors compared the performance of regular monolithic streaming and tile-based streaming using two HD videos and found that larger tiles can improve compression efficiency, but at the cost of transmitting redundant pixels. In this work, we attempt to reduce the transmission bits and provide improved viewport availability but with an unmodified decoder. In the follow-up work [30], the authors employed user access patterns to encode the different streaming regions with different encoding parameters. Our DFTs solutions also assign variable uniform bitrates to different streaming regions, but with more profound viewing region selection and dynamic bandwidth distribution. Nguyen et al. [32] proposed an adaptive tiling selection (ATS) solution for 360° video streaming. The authors evaluated four different tiling layouts (4x3, 6x4, 8x4, and 8x8) and divided the selected tiles into viewport and non-viewport groups for each layout. During each adaptation interval, the tile sets that resulted in the minimum viewport distortion or the maximum viewport bitrate were chosen for streaming. However, this approach did not incorporate any viewport prediction mechanism and was tested using fixed network connections. Xiao et al. [48] proposed an optimal tiling solution by partitioning a 360° segment into variable-size sub-rectangles to minimize the storage cost on the server side. The proposed solution estimates the storage and transmission cost by extracting the motion vectors and sizes of all basic sub-rectangles. An integer linear program (ILP) is then used to output the optimal tiling version that covers possible views of the segment. The proposed solution achieves interesting results, but at the cost of increased computational complexity. We attempt to achieve a similar goal of balancing storage size and data transmission, but with reduced server-side storage overhead and by utilizing standard computing and streaming components. The proposed solutions are essential for viewers who want to take advantage of the immersive and interactive VR experience, without having to invest in additional hardware.

Kattadige et al. [20] proposed a method for selecting the tiling layout of each segment of a video based on the visual attention of the user. The approach involves analyzing the frames of the video, creating visual attention maps for the user, and dividing the frames into three regions based on the user's attention. The proposed solution was compared to three fixed tiling layouts (4x6, 6x6, and 10x20) and was found to be more efficient in terms of pixels and bandwidth usage. Ozcinar et al. [34] employed visual attention maps to improve the network capacity planning for different tile groups. Variable-sized non-overlapping tiles are adaptively selected for each segment. However, real-time visual attention map computation and transmission require extensive resources, which is not in favour of this proposed solution. In a follow-up work [35], the authors extended their visual attention aware variable size non-overlapping tile mapping to benefit from the dynamic tiling structure. Each 360° video frame was split into two fixed-sized polar tiles (1/4th of the frame

Table 1. Summary of Tile-based Viewport Adaptive 360° Video Streaming Solutions

Streaming Techinque	Works	Design	Dataset	Tile Layouts	Resolution	Segment Duration	Experimental Duration
Fixed Viewport	[16]	Non-Uniform VP	5 Videos, 1 Users	6 tiles	720p-4K	-	Video duration
	[4]	Non-Uniform VP	5 Videos [23]	3x3, 4x4, 5x5	2K	1s	20s
	[15]	Uniform and Non-Uniform VP	3 Videos, 48 Users [47]	1×1, 2×2, 4×2, 4×4, 8×4, 8×6, 8x8, 16×12/16	4K	1.067s	Video duration
	[49]	Probability-based	1 Video, 5 Users	6x12	2K	1s	3m
	[28]	Layer-assisted	4 Videos, 5 Users	6 and 24 tiles	4K	32 frames	Video duration
Marginal Region	[36]	Fixed Margin	1 Video, 10 Users	6 tiles	8K	1s, 2s, 4s	60s
	[2]	Fixed Margin	3 Videos, 3 Users [6]	6x4	4K	1s	1m
	[65]	Fixed Margin	3 Videos, 10 Users [1]	8x8	4K	1s	Video duration
	[57]	Dynamic Margin	3 Videos, 1 Trace	4x6	4K	2s	10s
Extended Viewport	[14]	Dynamic Extension	-	-	-	-	-
	[17]	Dynamic Extension	4 Videos, 1 User	4x6	4K	Live	Video duration
	[13]	Fixed Extension (15°)	1 Video	36x2	-	0.1s	Video duration
	[24]	Fixed Extension (10°)	1 Video	18x36	-	-	Video duration
Dynamic Tiling	[32]	Visual distortion	1 Video, 10 Users	4x3, 6x4, 8x4, 8x8	4K	1s	60s
	[35]	Visual-attention	7 Videos, 25 Users	Multiple	8K	-	10s
	[20]	Region-based	30 Videos, 30 Users	Multiple	HD-4K	-	60s
	[48]	Variable rectangles	5 Videos, 58 Users	Multiple	2K & 4K	-	-

from the top and 1/4th from the bottom). The remaining equator region was horizontally divided into 1 and 2 tiles and then each part was divided into 1, 2, 4, 8, and 16 vertical tiles. Numerous dynamic tiling combinations can be considered using this division. The authors employed seven different spatial and temporal motion content types, but all with a duration of 10s. However, this type of tiling structure is not feasible in real-time streaming scenarios, as the two fixed size polar tiles (half of the frame) need to be transmitted in full quality if any part of the viewport is predicted to be in that region.

Table 1 illustrates the most significant streaming techniques for tile-based adaptive 360° video streaming. These algorithms use user-specific viewing preferences to improve the user's QoE by establishing a stable background. Most of the fixed viewport-based solutions [4, 15, 16] define variable quality levels within the viewport, which can lead to severe spatial quality oscillations even for perfect prediction results. Several solutions [2, 12, 36, 65] simply employ a fixed marginal area around the viewport in all directions. It can compensate for the highly dynamic viewing nature of the user; however, a significant waste of the bandwidth can be observed under medium to high prediction accuracy. Similarly, always extending the viewport region by 15° [13] and 10° [24] can lead to unnecessary transmission under perfect predictions. Different from previous works, in our approach the viewport and marginal region are considered special cases in the quest to overcome viewing uncertainty. Dynamic tiling solutions [20, 30, 34, 35] are theoretically effective in terms of increasing the picture quality and users' QoE. However, some of these solutions require real-time visual mapping which makes them difficult to implement in traditional on-demand scenarios. Mixing different resolution tiles [44] to provide a non-redundant viewport transmission [20, 34, 35] can result in users sensing quality variations and degradation for high and relatively static motion content. These solutions are difficult to be extended to consider different content types and are associated with additional coding and reconstruction overheads.

3 PROPOSED DYNAMIC TILING-BASED ARCHITECTURE

3.1 Dynamic Tiling-based System Architecture

Fig. 1 illustrates the workflow of DFTs solutions. On the server side, the 360° video is pre-processed by dividing it into a number of segments, i.e., $\mathcal{S} = \{\mathcal{S}(1), \mathcal{S}(2), ..., \mathcal{S}(i), ..., \mathcal{S}(I)\}$. Each segment is then divided into l tiling layouts, i.e., $\mathcal{T}_l(i),\ \forall\ l \in \{x, y, z\}$, containing small, medium, and large

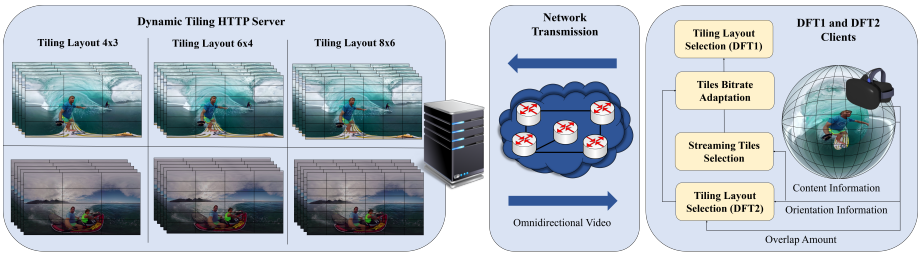


Fig. 1. The proposed 360° client-server streaming architecture.

number of tiles, respectively. Each tiling layout is further divided into a number of tiles, i.e., $\mathcal{T}_l = \{\mathcal{T}_l^1(i), \mathcal{T}_l^2(i), ..., \mathcal{T}_l^k(i), ..., \mathcal{T}_l^K(i)\}$. These tiles are then encoded at a number of different bitrates, i.e., $\mathcal{L}_l = \{\mathcal{L}_{l,1}^k(i), \mathcal{L}_{l,2}^k(i), ..., \mathcal{L}_{l,j}^k(i), ..., \mathcal{L}_{l,J}^k(i)\}$. Let $\mathcal{L}_{l,j}^k(i)$ represent the jth bitrate of the kth tile in the lth tiling layout of the ith segment.

The DFTs clients, which control the adaptive streaming operations, need to know in advance about the available tiling layouts on the server side. DFT2 performs tiling layout selection before determining the streaming tiles and bitrate allocations during each adaptation interval. The *tiling layout selection* module in DFT2 checks the overlap between the actual and predicted viewport areas during the previous segment. The *streaming tiles selection* module selects sets of tiles for different priority regions (i.e., viewport ($\mathcal{T}_l^v(i)$), marginal ($\mathcal{T}_l^m(i)$), and background ($\mathcal{T}_l^b(i)$)) based on the predicted viewport coordinates for each segment. This helps to ensure that the video is able to adapt to the viewer's movements and maintain a high level of quality by pre-downloading tiles that are most likely to be watched. The *tiles bitrate adaptation* unit then selects appropriate bitrates for each tile based on the associated region and the available network capacity. DFT1, on the other hand, first calculates the streaming regions and relevant bitrates for each tiling layout. It then selects the tiling layout that results in the highest weighted-area-based visual quality score in each adaptation interval. The segment request is then sent, and upon receiving the segments, the client decodes and reconstructs the requested views similar to fixed tiling-based views in the post-processing phase with no additional decoding overhead. The requested content is then presented to the user.

3.2 Problem Definition

In 360° adaptive video streaming, it is important to consider the user's quality expactations which depend largely on the quality of the visible area. Even if the viewport tiles are played at higher quality levels, the intra- and inter-segments quality oscillations may not satisfy the user. The QoE metric used in this context includes viewport quality and spatial and temporal smoothness factors, as well as the risk of playback buffer issues.

- **Viewport Quality:** The user is able to visualize only certain tiles during 360° video playback. The viewport quality reflects how much a user is satisfied with the visual perception. The client can be presented with any visual quality representation, but the average quality levels of the viewport tiles are highly correlated with the average bitrate that is actually consumed by the viewer. Therefore, by averaging the quality of the actual viewport tiles in segment (i), for lth tiling layout, the viewport quality is given as follows [37, 63]:

$$f_1(i) = \frac{\sum_{k \in \mathcal{T}_l^{\hat{v}}(i)} \sum_{j \in \mathcal{L}_l} Q(\mathcal{L}_{l,j}^k(i))}{|\mathcal{T}_l^{\hat{v}}(i)|} \tag{1}$$

where $\mathcal{T}_l^{\hat{v}}(i)$ represents the actual viewport tiles set in the (i)th segment and $|\mathcal{T}_l^{\hat{v}}(i)|$ indicates the cardinality of the set. $Q(\mathcal{L}_{l,j}^k(i))$ maps the jth bitrate of kth tile to the particular video quality level.

- **Temporal Quality Oscillations:** The inter-segment quality switches can reduce the "sense of being there" in an immersive environment. This may happen not only because of the network fluctuations but also due to the differences in head movement predictions. The user's experience can be impaired by physiological symptoms such as dizziness and headache when observing frequent visual disparity [41]. Therefore, the inter-segment quality fluctuations should not be drastic and can be calculated as the difference between the observed viewport quality levels of two consecutive segments [37, 63]:

$$f_2(i) = |\, f_1(i) - f_1(i-1)| \tag{2}$$

- **Spatial Quality Oscillations:** The visual tiles having different quality levels leads to complex perception. Cybersickness, viewing irritation, nausea, fatigue, and aversion [11], can be driven by inconsistent quality levels within the viewport. Compared to regular 2D videos, if the perceived quality of 360° tiles is not smooth, it will reduce the overall QoE. Following [19], we measured the spatial quality oscillations according to the coefficient of variation (CV) of viewport tiles quality.

$$f_3(i) = \frac{\sigma(Q(\mathcal{L}_{l,j}^k(i)))}{\mu(Q(\mathcal{L}_{l,j}^k(i)))}, \quad \forall k \in \mathcal{T}_l^{\hat{v}}(i), \forall j \in \mathcal{L}_l \tag{3}$$

  The standard deviation of the viewport quality samples is in the numerator, and the mean of the samples is in the denominator.

- **Playback Buffer Risk:** A large buffer capacity may not be efficient for 360° video streaming because of the constantly changing FoV during playback [9, 33]. Pre-buffering high-quality tiles can be risky, as the user's FoV may shift at the time of playback. Instead of relying on the traditional playback discontinuity under short-term viewport prediction, it is more beneficial to assess directly risky buffer events based on the available connection bandwidth and the selected video bitrates. This can be expressed as follows [45]:

$$f_4(i) = \begin{cases} 1, & \textbf{if } (\widehat{B}(i) < \sum_{k \in \mathcal{T}_l(i)} \mathcal{L}_{l,j}^k(i)) \\ 0, & Otherwise \end{cases} \tag{4}$$

  where $\widehat{B}(i)$ represents the available bandwidth budget for (i)th segment.

Following the principle behind the QoE metric for traditional video [26], some works [37, 62, 63] consider video quality, quality variations, rebuffering events, etc. to model a QoE metric for 360° videos. The user-perceived QoE for each 360° segment is defined by a weighted summation formulation:

$$QoE(i) = \alpha \times f_1(i) - \beta \times f_2(i) - \gamma \times f_3(i) - \delta \times f_4(i) \tag{5}$$

where α, β, γ, and δ are the parameters indicating how much importance a user gives to video bitrate, temporal and spatial quality variances, and rebuffering risk, respectively. As users do not want to experience quality fluctuations and rebuffering events, the functions $f_2(i)$, $f_3(i)$, and $f_4(i)$ are set to negative.

Accurate evaluation of QoE is essential for optimizing the performance of traditional, multimedia [58], and immersive video content. The level of satisfaction a user experiences while watching a VR video is determined by how long they feel immersed in the scene. The proposed clients aim to

select optimal bitrates for each segment in a dynamic tiling streaming system in order to maximize the user's long-term QoE reward. The mathematical problem formulation is as follows:

Problem:

$$max \sum_{i \in \mathcal{S}} QoE(i) \tag{6}$$

The proposed solutions solve this problem by implementing a three-tier adaptation mechanism. Firstly, they select a relevant tiling layout for each segment. Next, DFTs solutions dynamically perform the viewing area selection based on the two viewport prediction mechanisms to predict the most likely to-be-watched tiles. Finally, the tiles bitrate adaptation mechanism improves the bitrate budget distribution between different tiles groups. These mechanisms are elaborated on in the next section.

4 PROPOSED DYNAMIC TILING-BASED ADAPTATION ALGORITHMS

This section presents the adaptation algorithms for DFT1 and DFT2 streaming clients.

4.1 DFT Tiling Layout Selection Algorithms

Tile-based encoding brings several opportunities such as efficient video coding [40], improved quality distribution, parallel [25], and partial decoding [5], etc., for VR video applications. The choice of the appropriate tiling layout, which reflects the spatial partitioning of frame areas, impacts the overall video compression performance. In 360° video, the polar regions have higher viewing distortions and less viewing probability than the equator regions when transforming a spherical representation into a two-dimensional planer format, i.e., equirectangular projection. Therefore, encoding polar areas with more pixels consume the user's limited bandwidth to transmit data related to less relevant image regions. Fixed tiling solutions encode polar and equator regions at similar bitrate levels leading to unattractive viewport boundaries and losing positive compression opportunities. Employing a smaller number of tiles (i.e., larger resolution tiles) can improve the compression performance in some cases. Yet, at the same time, it may include unnecessary higher-quality portions outside the viewport [35]. Contrary, smaller resolution tiles can reduce the number of redundant pixels [45]; however, it may also cause visual distortions such as flickering, floating, and blurring at the edges of the tiles [8]. Finding ways to dynamically select the most appropriate tiling layout for a given viewing scenario and preferences is an important area of research. By developing smart techniques that can take these factors into account and adjust the tiling layout accordingly, it may be possible to improve the overall viewing experience. Therefore, the proposed solution considers two tiling layout selection solutions to lower redundant data transmission and facilitate a fine-grained visual perception for different motion content.

DFT1: The proposed DFT1 solution decides an optimal tiling layout during each adaptation interval based on the observed visual quality scores. Since the user gaze point is mostly located around the centre of the viewport [27, 43], the viewpoint quality should have a higher priority compared to other tiles. Therefore, we design a priority-assisted visual quality metric to attentively select the suitable tiling layouts during 360° video streaming. In this context, DFT1 assigns different priority weights to the viewport tiles in such a way that the tiles closer to the viewpoint should have a higher priority compared to other tiles. The tiles are arranged based on how far they are located from the viewpoint. The priority weights are assigned such that the most important parts of the image, as determined by their proximity to the centre of the viewer's focus, are rendered with the highest quality, while less important parts of the image are rendered with lower quality. In this context, the highest and lowest weights are allocated for the mapped quality of the viewpoint

Algorithm 1: Tiling Layout Selection Algorithm in DFT2

Input : $\mathcal{O}(i-1)$: Tiles overlap percentage during the $(i-1)$th segment
Output: $\mathcal{T}_l(i)$: Tiling layout selected for the (i)th segment

1 **if** $(i == 1)$ **then**
2 $\quad \mathcal{T}_l(i) \leftarrow \mathcal{T}_z(i)$
3 **else if** $\mathcal{O}(i-1) == 0$ **then**
4 $\quad \mathcal{T}_l(i) \leftarrow \mathcal{T}_x(i)$; // No overlap
5 **else if** $\mathcal{O}(i-1) == 100$ **then**
6 $\quad \mathcal{T}_l(i) \leftarrow \mathcal{T}_z(i)$; // Perfect overlap
7 **else**
8 $\quad \mathcal{T}_l(i) \leftarrow \mathcal{T}_y(i)$; // Partial overlap

and the last tile, respectively, in the sorted tiles set. The weighted quality metric is given in Eq. (7):

$$\mathcal{WQ}_l^v(i) = \frac{\sum_{k=1}^{|\mathcal{T}_l^v(i)|} \sum_{j=1}^{J} (2)^{|\mathcal{T}_l^v(i)|-k} \times Q(\mathcal{L}_{l,j}^k(i))}{(2)^{|\mathcal{T}_l^v(i)|} - 1} \tag{7}$$

where the quantity $|\mathcal{T}_l^v(i)|$ represents the number of tiles in the set of tiles predicted to be within the viewport, and $Q(\mathcal{L}_{l,j}^k(i))$ maps the video bitrate to a specific quality level. Since we consider extended viewport case, elaborated in section 4.2, where the visual area can be different for different tiling layouts, for instance, an extended viewport with $\mathcal{T}_x^v(i)$ could cover more region as compared to an extended viewport with $\mathcal{T}_z^v(i)$. Therefore, we define the visual area-based weighted video quality metric which tries to balance the visual area and the weighted quality and is given in Eq. (8):

$$\mathcal{VQ}_l^v(i) = \frac{|\mathcal{T}_l^v(i)|}{|\mathcal{T}_l(i)|} \times \mathcal{WQ}_l^v(i) \tag{8}$$

where $|\mathcal{T}_l(i)|$ represents the total number of tiles in the tiling layout l. The tiling layout selection procedure for DFT1 is given as follows:

(1) For each tiling layout:
- Perform streaming tiles selection and identify the streaming case using Algorithm 2.
- Perform bitrate adaptation for the tiles groups of the selected case using Algorithm 3.
- Compute the prioritized visual area-based quality scores using Eq. 7 and Eq. 8.

(2) Stream the tiles from the tiling layout that results in the highest visual levels.

DFT2: DFT2 decides an optimal tiling layout based on the viewport prediction performance. Unlike DFT1 which is based on visual area, DFT2 measures the closeness between actual and predicted viewport tiles sets in terms of viewport overlap to select the appropriate tiling layout for the next segment. Let $\mathcal{O}(i-1)$ denote the overlap percentage of the actual and predicted viewport tiles for the $(i-1)$th segment, and is given as [29]:

$$\mathcal{O}(i-1) = \frac{|\mathcal{T}_l^{\hat{v}}(i-1) \cap \mathcal{T}_l^v(i-1)|}{|\mathcal{T}_l^{\hat{v}}(i-1)|} \times 100 \tag{9}$$

Algorithm 1 details the tiling layout selection procedure in DFT2. As no information is available at the start, the tiling layout with a larger number of tiles ($\mathcal{T}_z(i)$) is selected for the first segment (lines 1-2). If there is no overlap between actual and predicted viewing tiles, then the tiling layout with a smaller number of tiles ($\mathcal{T}_x(i)$) is selected for the (i)th segment to deal with fast head rotations (lines

Algorithm 2: Tiles Selection Algorithm in DFT

Input : $\mathcal{T}_l(i)$: Tiles set with tiling layout l for the (i)th segment; $\mathcal{T}_l^{vn}(i)$: Primary predicted viewport tiles set; $\mathcal{T}_l^{vs}(i)$: Secondary predicted viewport tiles set

Output : $\mathcal{T}_l^{v}(i)$: Estimated viewport tiles set for the (i)th segment; $\mathcal{T}_l^{m}(i)$: Estimated marginal tiles set for the (i)th segment; $\mathcal{T}_l^{b}(i)$: Estimated background tiles set for the (i)th segment

1

$$\mathcal{T}_l^{v}(i) = \begin{cases} \mathcal{T}_l^{vn}(i) \cup \mathcal{T}_l^{vs}(i) & \text{if } \mathcal{T}_l^{vn}(i) \cap \mathcal{T}_l^{vs}(i) = \varnothing \\ \mathcal{T}_l^{vn}(i) & \text{otherwise} \end{cases}$$

$$\mathcal{T}_l^{m}(i) = \begin{cases} \varnothing & \text{if } \mathcal{T}_l^{vn}(i) \cap \mathcal{T}_l^{vs}(i) = \varnothing \\ \mathcal{T}_l^{vs}(i) \setminus \mathcal{T}_l^{vn}(i) & \text{otherwise} \end{cases}$$

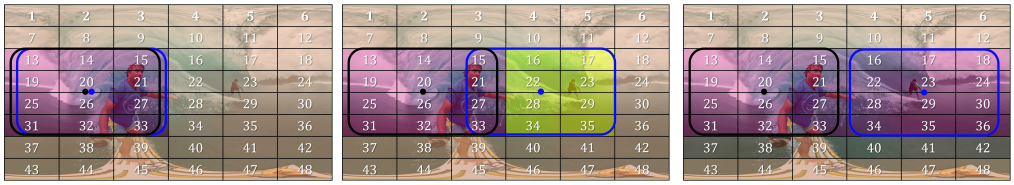
$$\mathcal{T}_l^{b}(i) = \Big\{ \mathcal{T}_l(i) \setminus (\mathcal{T}_l^{vn}(i) \cup \mathcal{T}_l^{vs}(i))$$

3-4). If actual and predicted viewports perfectly overlap during the previous segment, the smallest resolution tiles are selected to lessen the abundance of unnoticeable pixels outside the viewport region (lines 5-6). If the actual and predicted viewports partially overlap during the playback of the previous segment, medium-resolution tiles represented as $\mathcal{T}_y(i)$ are streamed for the next segment (lines 7-8). DFTs solutions do not involve complex frame partitioning and ensure a flexible uniform tiling structure without any modifications of existing video coding and stream processing tools, which makes them attractive to be adopted in on-demand and live streaming scenarios. DFT1 is a scalable solution that can work with any number of tiling layouts. It is also practical for both simulation and real-time environments.

4.2 DFTs Streaming Tiles Selection Algorithm

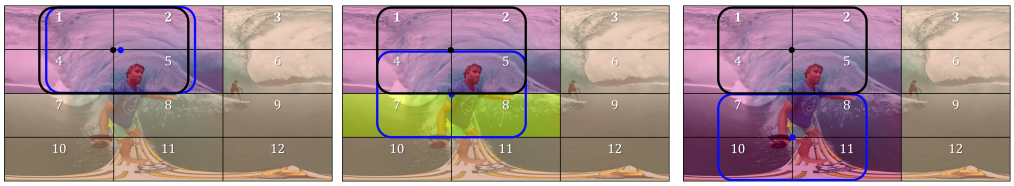
The ability to choose the best-fit tiles in response to the user's unpredictable head movements is one of the fundamental criteria for 360° video applications. The prediction accuracy of current streaming solutions based on a single viewport prediction technique can decrease when predicting longer in the future. To adaptively encompass the real viewing region, this work employs two viewpoint/viewport prediction techniques. It's interesting to note that, in the majority of cases, the naive prediction model (using the current coordinates as predicted points) outperforms more sophisticated models [10]. The primary viewport tiles set ($\mathcal{T}_l^{vn}(i)$) contains the viewport tiles actually watched by the user during the previous segment. The secondary viewport tiles set ($\mathcal{T}_l^{vs}(i)$) is computed using a spherical walk approach described in [15].

Algorithm 2 aims to find appropriate tiles for the viewport, marginal, and background regions, respectively. The tiles identification and selection are dynamically performed for each adaptation interval. Algorithm 2 takes as input the tiles set $\mathcal{T}_l(i)$ with tiling layout l for the (i)th segment, the primary predicted viewport tiles set $\mathcal{T}_l^{vn}(i)$, and the secondary predicted viewport tiles set $\mathcal{T}_l^{vs}(i)$. It outputs the estimated viewport tiles set $\mathcal{T}_l^{v}(i)$, the estimated marginal tiles set $\mathcal{T}_l^{m}(i)$, and the estimated background tiles set $\mathcal{T}_l^{b}(i)$. The algorithm first determines the viewport tiles set based on the intersection between the primary and secondary predicted viewport tiles sets. If the primary and secondary predicted viewport tiles sets are disjoint sets, then the viewport tiles set is the union of the primary and secondary predicted viewport tiles sets. Otherwise, the primary predicted viewport tiles set is assigned to the viewport tiles set. Next, the algorithm determines the



(a) Case 1 (b) Case 2 (c) Case 3

Fig. 2. Tiles selection cases in DFT2 for $\mathcal{T}_z(i)$ tiling layout of (i)th segment.



(a) Case 1 (b) Case 2 (c) Case 3

Fig. 3. Tiles selection cases in DFT2 for $\mathcal{T}_x(i+1)$ tiling layout of $(i+1)$th segment.

marginal tiles set, such that if the intersection of primary and secondary viewport sets is empty, then the marginal tiles set is empty. Otherwise, the marginal tiles set is the difference between the secondary predicted viewport tiles set and the primary predicted viewport tiles set. Finally, the algorithm determines the background tiles set following a check between tiles set and the primary and secondary predicted viewport tiles sets. Specifically, all the tiles which do not belong to the viewport or marginal tiles sets are added to the background tiles set. Fig. 2 and Fig. 3 illustrate the tiles selection cases in DFT2 based on the output of Algorithm 1 for two consecutive segments. The black rectangle represents the primary predicted viewport, while the blue rectangle represents the secondary predicted viewport. The potential viewport tiles are represented by a purple window, whereas the marginal and background tiles are marked in light green and brown, respectively.

4.3 DFT Tiles Bitrate Adaptation Algorithm

Adaptive streaming players usually maintains a large buffer space for regular 2D videos to absorb the uneven motions in video scenes and playback interruptions. However, for 360° videos, a large buffer capacity is not encouraged due to FoV dynamics. In practice, for 360° tiled video streaming, the buffer should be as small as possible (usually 2 segments [15]) to accommodate the new chunks in response to the user movements within the immersive video. Algorithm 3 takes into account both the predicted tiles and network conditions to more accurately adjust the video quality for smoother viewing experience. This algorithm is specifically designed for dynamic tiling-based 360° video streaming. Both DFT1 and DFT2 clients employ the same bitrate adaptation algorithm to decide the suitable bitrates for tiles.

In the absence of buffer consideration, accurate bandwidth estimation is crucial to achieving higher playback performance [53]. An over/under-estimation of the available bandwidth can result in frequent rebuffering/lower quality playback. Following [28], the bandwidth for (i)th segment is computed as follows:

$$\widehat{B}(i) = \frac{\sum_{\forall k,j} \mathcal{L}^k_{l,j}(i-1) * \tau}{\mathcal{D}(i-1)} \tag{10}$$

where $\mathcal{L}^k_{l,j}(i-1)$ represents the bitrate of previous segment, τ is the playback duration of the segment, and $\mathcal{D}(i-1)$ represents the download time of the $(i-1)$th segment. The proposed bitrate

Algorithm 3: Tiles Bitrate Adaptation Algorithm in DFT

Input : $\mathcal{L}_l$: Video bitrates set of 360° segments; $\mathcal{T}_l^v(i), \mathcal{T}_l^m(i), \mathcal{T}_l^b(i)$: Viewport, marginal, and background tiles sets; $|\mathcal{T}_l^v(i)|, |\mathcal{T}_l^m(i)|$: Number of tiles in viewport and marginal regions; $\widehat{B}(i)$: Available bandwidth for the (i)th segment; $w^{\mathcal{T}_l^v}(i) \leftarrow 1; w^{\mathcal{T}_l^m}(i) \leftarrow 0$: Initialize priority weights of viewport and marginal tiles; $B^{\mathcal{T}_l^v}(i), B^{\mathcal{T}_l^m}(i), B^{\mathcal{T}_l^b}(i)$: Region-based bandwidth;

Output : $\mathcal{L}^{\mathcal{T}_l^v}(i), \mathcal{L}^{\mathcal{T}_l^m}(i), \mathcal{L}^{\mathcal{T}_l^b}(i)$: Video bitrates selected for the tiles of (i)th segment

1 **if** $(\widehat{B}(i) \leq \sum_{k \in \mathcal{T}_l(i)} \mathcal{L}_{l,1}^k(i))$ **then**
2 $\quad \mathcal{L}^{\mathcal{T}_l}(i) = \mathcal{L}_{l,1}^k(i), \forall\, k \in \mathcal{T}_l(i)$
3 **else if** $(\widehat{B}(i) \geq \sum_{k \in \mathcal{T}_l(i)} \mathcal{L}_{l,J}^k(i))$ **then**
4 $\quad \mathcal{L}^{\mathcal{T}_l}(i) = \mathcal{L}_{l,J}^k(i), \forall\, k \in \mathcal{T}_l(i)$
5 **else**
6 $\quad \mathcal{L}^{\mathcal{T}_l}(i) = \mathcal{L}_{l,1}^k(i), \forall\, k \in \mathcal{T}_l(i)$
7 $\quad B(i) = \widehat{B}(i) - \sum_{k \in \mathcal{T}_l(i)} \mathcal{L}_{l,1}^k(i)$
8 $\quad$ **if** $(\mathcal{T}_l^m(i) \neq \emptyset)$ **then**
9 $\quad\quad w^{\mathcal{T}_l^m}(i) = \frac{|\mathcal{T}_l^m(i)|}{2*|\mathcal{T}_l^v(i)|+|\mathcal{T}_l^m(i)|}$
10 $\quad\quad w^{\mathcal{T}_l^v}(i) = 1 - w^{\mathcal{T}_l^m}(i)$
11 $\quad B^{\mathcal{T}_l^v}(i) = B(i) \times w^{\mathcal{T}_l^v}(i)$
12 $\quad B^{\mathcal{T}_l^m}(i) = B(i) \times w^{\mathcal{T}_l^m}(i)$
13 $\quad \mathcal{L}^{\mathcal{T}_l^v}(i) = \max_{j \in [2:J]} \{\mathcal{L}_{l,j}^k(i) | \sum_{k \in \mathcal{T}_l^v(i)} \mathcal{L}_{l,j}^k(i) \leq B^{\mathcal{T}_l^v}(i)\}$
14 $\quad \mathcal{L}^{\mathcal{T}_l^m}(i) = \max_{j \in [2:J]} \{\mathcal{L}_{l,j}^k(i) | \sum_{k \in \mathcal{T}_l^m(i)} \mathcal{L}_{l,j}^k(i) \leq B^{\mathcal{T}_l^m}(i)\}$
15 $\quad B^{\mathcal{T}_l^b}(i) = B(i) - (\sum_{k \in \mathcal{T}_l^v(i)} \mathcal{L}^k l, j(i) + \sum_{k \in \mathcal{T}_l^m(i)} \mathcal{L}^k l, j(i))$
16 $\quad \mathcal{L}^{\mathcal{T}_l^b}(i) = \max_{j \in [2:J]} \{\mathcal{L}_{l,j}^k(i) | \sum_{k \in \mathcal{T}_l^b(i)} \mathcal{L}_{l,j}^k(i) \leq B^{\mathcal{T}_l^b}(i)\}$

allocation algorithm considers aggressive, weighted, and conservative quality adjustments for different tiles selection cases to improve the corresponding bitrate choice for each tile that the network can support. For tiles selection *Case 1*, an aggressive quality adjustment is performed for viewport tiles. The algorithm performs a weighted quality adjustment if the marginal region is non-empty (*Case 2* of Algorithm 2). A relatively conservative bitrate selection is performed for *Case 3*, where the viewport region is extended to lower the viewport mismatch while sacrificing the quality.

Algorithm 3 determines the bitrate selection for the tiles belonging to different priority regions calculated in Section 4.2. The input to the algorithm consists of various sets of video tiles (viewport, marginal, and background tiles), the number of tiles in the viewport and marginal regions, the available bandwidth for each segment of the video, and initial priority weights for the viewport and marginal tiles. The output of the algorithm is the selected bitrates for each tile in each segment of the video. The playback adaptation is performed for each segment after the previous segment has been fully downloaded. The algorithm begins by checking if the available bandwidth is less than or equal to the sum of the lowest bitrate options for all tiles in the current segment. If this is the case, the lowest bitrate is selected for all tiles (lines 1-2). If the available bandwidth is greater than or

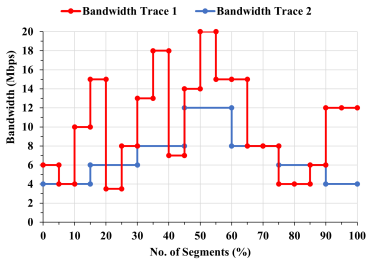


Fig. 4. Bandwidth traces employed in experiments.

equal to the sum of the highest bitrate options for all tiles, the highest bitrate is selected for all tiles (lines 3-4). In other cases, the algorithm sets the bitrate for all tiles to the lowest bitrate option and calculates the remaining available bandwidth (lines 6-7). If there are tiles in the marginal region (i.e., $\mathcal{T}_l^m(i) \neq \emptyset$), the algorithm updates the priority weights for the viewport and marginal tiles (lines 9-10). The priority weights are determined based on the number of tiles in the viewport and marginal regions, with the viewport tiles being given higher priority. The viewport and marginal tiles (only possible in Case 2) are then allocated bandwidth based on the computed weights (lines 11-12). Next, the highest possible bitrates for the viewport and marginal tiles are chosen based on the available bandwidth for each region (lines 13-14). This ensures the weighted quality adaptation for viewport and marginal tiles. If there are no marginal tiles, then for *Case 1* or *Case 3* of Algorithm 2, an aggressive or relatively conservative quality allocation is considered for viewport tiles to ensure visual smoothness. After determining the bitrates for the viewport and marginal tiles, the bandwidth for the background tiles is calculated by subtracting the sum of these bitrates from the revised overall bandwidth budget (line 15). Finally, the bitrate of the background tiles is also increased, as long as it does not exceed the available bandwidth budget (line 16).

5 EXPERIMENTAL EVALUATION

This section presents the experimental evaluations of our proposed solutions using a diverse range of content and network conditions.

5.1 Experimental Setup

The proposed solution evaluation is performed by modifying a VR player provided by [15], on a machine with an Intel Core i7-7500U CPU and 16 GB of memory running Ubuntu 16.04. In the experiments, the VR player retrieves 360° video segments from an HTTP server while the connection speed between the VR player and HTTP server was varied, as illustrated in Fig. 4. Bandwidth trace 1 has more irregular increasing and decreasing trends compared to bandwidth trace 2. The maximum connection speed for trace 1 is 20 Mbps, while for trace 2, the maximum bandwidth value is 12 Mbps.

5.1.1 Content Pre-processing. This work employs a highly cited open-source video and head movement dataset captured by Wu et al. [47]. The dataset contains real head movement patterns of 48 unique VR users viewing 18 long-duration videos in two learning-based testing sessions using an HTC Vive headset with a field of view of 110°. In the first experiment, participants were asked to explore the content without paying too much attention to the specifics of what they were looking at. In the second experiment, on the other hand, they were asked to focus on the content and pay close attention to it, simulating certain behaviours or habits. We choose five videos, namely, **LOSC Football**(experiment 1), **Weekly Idol-Dancing**(experiment 2), **Google Spotlight-HELP**(experiment 1), **GoPro VR-Tahiti Surf**(experiment 1), and **Rio Olympics VR**

Table 2. Content Characteristics

Videos	Category	Duration	Resolution	FPS
Football	Sport	2′44″	3840x2160	25
Performance	Performance	4′38″	3840x1920	29
Spotlight	Film	4′53″	3840x2160	30
Surfing	Sport	3′25″	3840x1920	29
VR-Interview	Talkshow	3′07″	3840x1920	25

Interview(experiment 2) from this dataset. This is in line with the recommendations of ITU-T R. P.913 [18] and is typical for research and development solutions evaluations. The five different duration immersive clips in this dataset can be classified into four categories: Sport (**LOSC Football** and **GoPro VR-Tahiti Surf**), Performance (**Weekly Idol-Dancing**), Film (**Google Spotlight-HELP**), and Talkshow (**Rio Olympics VR Interview**). These videos are referred to as **Football**, **Performance**, **Spotlight**, **Surfing**, and **VR Interview** throughout the remaining chapter. Table 2 summarizes the content features of five videos. All of the videos were resized to 4K resolution using FFmpeg[1] software. Following [12], we spatially split 360° videos into 4x3, 6x4, and 8x6 tiling layouts. This work suggests that the 6x4 tilling structure results in an optimal trade-off between viewport availability, bitrate overhead, and bandwidth requirements. The video tiles were encoded using an open-source encoder called Kvazaar[2], with five different quantization parameter (QP) values: 22, 27, 32, 37, and 42. Considering the experimental recommendations for selecting segment duration for viewport adaptive streaming [7, 38], three different duration, i.e., 1s, 1.5s, and 2s, MPEG-DASH video segments were generated using GPAC MP4Box[3]. The playback buffer was set to two segments for each experiment. The average segment sizes for each video are shown in Table 3. The simulation length was set according to the duration of each video.

5.1.2 Comparative Approaches. **DFT** solutions are compared with dynamic tiling-based (ATS) and fixed tiling-based (UVP, CTF, PBA, AVR) solutions.

(1) **ATS [32]:** This solution performs adaptive tiles selection based on weighted viewport distortions. The tiling layout resulting in minimum viewport distortion or maximum viewport bitrate is selected for streaming during each decision interval.
(2) **UVP [15]:** A straightforward per-region uniform quality adaptation approach for different frame areas classified by considering the user's walk on a spherical surface prediction mechanism.
(3) **CTF [15]:** This scheme is an extended version of UVP but takes into consideration the entire frame as a potential viewing area. Rather than dividing the frame into regions and assigning bitrates evenly across them, this method increases the quality of the video in a per-tile fashion, beginning with the centre tiles and working outward towards the edges.
(4) **PBA [16]:** The highly cited approach divides tiles into three zones, Z_1 (viewport centre tile), Z_2 (surrounding tiles), and Z_3 (background tiles). In this system, priority-based bitrate adaptation is applied to tiles within certain regions, while also considering the available bandwidth budget.
(5) **AVR [36]:** One of the early approaches which allows for efficient use of resources while maintaining a high quality of playback by dividing 360° frames into viewport, adjacent, and outside regions.

[1] https://ffmpeg.org/
[2] http://ultravideo.fi/
[3] https://gpac.wp.imt.fr/mp4box/

Table 3. Average and Standard Deviations of Segment Bitrates [Mbps] for the **Football**, **Performance**, **Spotlight**, **Surfing**, and **VR Interview** videos.

Video	QP	1s			1.5s			2s		
		4x3	6x4	8x6	4x3	6x4	8x6	4x3	6x4	8x6
Football	22	6.9±2.3	7.0±2.3	7.2±2.3	10.5±5.1	10.6±5.1	10.9±5.2	13.8±4.6	14.1±4.6	14.4±4.6
	27	3.5±1.3	3.6±1.4	3.8±1.4	5.3±2.8	5.5±2.8	5.7±2.9	7.1±2.7	7.3±2.7	7.6±2.7
	32	1.9±0.8	2±0.8	2.2±0.8	2.9±1.6	3.1±1.6	3.3±1.6	3.9±1.5	4.1±1.5	4.5±1.6
	37	1.1±0.4	1.2±0.4	1.4±0.4	1.7±0.9	1.8±0.9	2.1±1	2.3±0.9	2.4±0.9	2.8±0.9
	42	0.7±0.2	0.7±0.2	0.9±0.2	1±0.5	1.1±0.5	1.4±0.6	1.3±0.5	1.5±0.5	1.8±0.5
Performance	22	8.5±2.9	8.6±2.9	8.9±3.0	12.8±5.9	13.0±5.9	13.4±6.0	17.0±4.7	17.3±4.7	17.8±4.8
	27	4.6±1.7	4.7±1.7	5.0±1.7	6.9±3.3	7.1±3.3	7.5±3.4	9.3±2.7	9.5±2.7	10.0±2.7
	32	2.6±0.9	2.7±0.9	2.9±0.9	4.0±1.9	4.1±1.9	4.5±2.0	5.3±1.5	5.5±1.5	6.0±1.5
	37	1.6±0.5	1.7±0.5	1.9±0.5	2.4±1.1	2.5±1.1	2.8±1.2	3.2±0.9	3.4±0.8	3.8±0.9
	42	0.9±0.3	1.0±0.3	1.2±0.3	1.4±0.6	1.6±0.6	1.9±0.7	1.9±0.5	2.1±0.5	2.5±0.5
Spotlight	22	13.6±8.8	13.9±8.8	14.3±8.9	20.4±15.0	20.9±15.2	21.5±15.4	27.1±17.1	27.7±17.2	28.5±17.3
	27	7.2±5.3	7.4±5.3	7.7±5.4	10.8±8.9	11.1±9.0	11.6±9.1	14.3±10.3	14.8±10.4	15.5±10.5
	32	4.0±3.1	4.2±3.1	4.5±3.2	6.1±5.2	6.3±5.3	6.7±5.4	8.1±6.1	8.4±6.1	9.0±6.2
	37	2.3±1.8	2.4±1.8	2.7±1.8	3.5±2.9	3.7±3.0	4.1±3.1	4.7±3.5	4.9±3.5	5.4±3.5
	42	1.3±0.9	1.4±0.9	1.6±0.9	2.0±1.5	2.2±1.6	2.5±1.6	2.7±1.7	2.9±1.8	3.3±1.8
Surfing	22	22.7±11.2	23.0±11.3	23.5±11.4	34.0±21.0	34.5±21.2	35.3±21.4	45.3±22.2	45.9±22.3	46.9±22.5
	27	12.8±6.7	13.0±6.8	13.4±6.8	19.2±12.4	19.5±12.5	20.2±12.7	25.5±13.3	26.0±13.4	26.8±13.5
	32	7.2±3.9	7.4±3.9	7.7±3.9	10.8±7.1	11.1±7.2	11.6±7.3	14.4±7.7	14.7±7.8	15.4±7.8
	37	4.0±2.2	4.1±2.2	4.4±2.2	6.0±3.9	6.2±4.0	6.6±4.1	7.9±4.3	8.2±4.3	8.8±4.3
	42	2.1±1.1	2.2±1.1	2.5±1.1	3.2±2.0	3.4±2.1	3.7±2.2	4.2±2.2	4.5±2.2	5.0±2.2
VR Interview	22	7.6±1.0	7.7±1.1	7.8±1.1	11.4±4.1	11.5±4.2	11.8±4.3	15.2±2.0	15.4±2.0	15.7±2.1
	27	3.7±0.7	3.8±0.7	3.9±0.7	5.5±2.1	5.7±2.2	5.9±2.3	7.4±1.3	7.6±1.3	7.9±1.4
	32	1.7±0.3	1.8±0.4	2.0±0.4	2.6±1.0	2.8±1.1	3.0±1.2	3.5±0.7	3.7±0.7	4.0±0.8
	37	0.9±0.2	1.0±0.2	1.2±0.2	1.4±0.6	1.6±0.6	1.8±0.7	1.9±0.4	2.1±0.4	2.5±0.4
	42	0.6±0.1	0.7±0.1	0.8±0.1	0.9±0.3	1.0±0.4	1.3±0.4	1.2±0.2	1.4±0.2	1.7±0.2

5.1.3 Evaluation Metrics. The performance of the proposed and comparative schemes is assessed in terms of the following metrics:

(1) **Streaming Behavior:** We evaluate how the DFT1 and DFT2 switch to different tiling layouts and behave in terms of adopting tiles selection and bitrate adaptation scenarios. We also show how the ATS client switches between available tiling layouts for each streaming session.
(2) **Tiles Overlap:** This metric measures the real and predicted viewport tiles overlap as defined in eq. (9).
(3) **Average QoE:** It reflects the average quality score of all the users for each video for the QoE metric defined in eq. (5).

5.2 Experimental Results

This subsection presents the results of experiments and a thorough analysis of the performance of each solution in a variety of testing conditions.

5.2.1 Streaming Behavior. Table 4 provides insight into how the **DFT1** solution performs interms of tiling layout selection, tiles selection, and bitrate adaptation for five different motion 360° videos. **DFT1** supports the larger visual area with higher quality streaming; therefore, for all the videos, larger resolution tiles (i.e., 4x3 and 6x4) are predominantly selected. However, The use of the 6x4 tiling layout decreases while the use of the 4x3 tiling layout slightly increases (by 5.38%) when the segment duration is increased from 1s to 2s for all the videos. Overall, a small percentage of smaller resolution tiles (i.e., 8x6) is selected for all videos. **DFT1** selects a 4x3 tiling layout for more than 67% for **VR Interview** video and mostly performs aggressive bitrate selection for selected tiling layouts. **DFT1** fetches the segments of **Football**, **Performance**, **Spotlight**, **Surfing**, and **VR Interview** videos by performing aggressive bitrate selection by up to 59.14%, 75.33%, 66.27%, 58.75%, and 78.43%, respectively, averaged across three segment durations. **DFT1** performs

Table 4. Streaming Behavior of **DFT1** Client in terms of Tiling Layouts Selection, Tiles Selection, and Bitrate Adaptation Scenarios. The Percentage Results are Averaged for Five Videos Watched by 48 VR Users.

Videos	Segment Duration	Tiling Layout [%]			Tiles Selection: Case 1 Bitrate: Aggressive			Tiles Selection: Case 2 Bitrate: Weighted			Tiles Selection: Case 3 Bitrate: Conservative		
		8x6	6x4	4x3	8x6	6x4	4x3	8x6	6x4	4x3	8x6	6x4	4x3
Football	1	17.73	52.27	30.00	8.84	37.39	20.45	8.70	14.36	6.45	0.19	0.53	3.09
	1.5	17.53	50.10	32.38	7.42	32.59	17.74	9.54	16.02	7.91	0.57	1.49	6.73
	2	17.48	47.97	34.55	6.43	30.06	16.51	9.88	15.19	9.07	1.17	2.72	8.97
Performance	1	4.96	66.74	28.30	2.68	55.40	21.41	2.16	11.08	5.42	0.12	0.25	1.47
	1.5	5.42	63.77	30.81	2.80	51.00	21.32	2.48	12.18	6.48	0.14	0.59	3.02
	2	6.71	60.49	32.79	3.75	46.66	20.97	2.67	13.08	7.69	0.30	0.75	4.14
Spotlight	1	21.00	44.49	34.51	12.99	32.75	27.48	7.71	11.13	5.09	0.30	0.61	1.93
	1.5	19.01	42.00	39.00	10.30	27.67	27.24	8.10	12.86	7.47	0.61	1.46	4.28
	2	18.06	38.90	43.04	8.57	24.14	27.66	8.65	12.50	9.41	0.84	2.27	5.97
Surfing	1	20.46	42.23	37.32	11.12	28.75	27.63	9.10	12.89	7.00	0.24	0.59	2.68
	1.5	19.30	39.81	40.89	8.67	23.62	24.68	10.04	14.32	9.87	0.59	1.87	6.34
	2	17.84	36.67	45.49	7.12	19.72	24.96	9.59	14.02	11.17	1.13	2.93	9.36
VR Interview	1	6.46	26.42	67.11	3.93	19.11	59.60	2.46	6.87	5.99	0.07	0.45	1.52
	1.5	7.29	24.14	68.57	4.33	16.15	57.48	2.67	7.17	7.64	0.29	0.82	3.44
	2	8.60	23.16	68.23	4.61	14.45	55.65	3.20	7.53	8.00	0.78	1.19	4.59

weighted quality adjustments for segments of these videos by up to 32.37%, 21.08%, 27.64%, 32.66%, and 17.18%. Interestingly, there is a decrease in the percentage of aggressive quality adjustments and an increase in the percentage of weighted quality adjustments when the segment duration is increased. In addition, a tiny percentage of conservative bitrate selection is observed for all the videos in the **DFT1** solution.

The streaming behaviour of the **DFT2** client is presented in Table 5. **DFT2** achieves a perfect viewport match (by up to 65.50%), a partial viewport match (by up to 27.71%), and a complete viewport mismatch (by up to 6.77%) by selecting on average 8x6, 6x4, and 4x3 tiling layouts, respectively. In particular, **DFT2** observes a perfect viewport match (i.e., 57.35% for the **Football** video, 73.88% for the **Performance** video, 64.95% for the **Spotlight** video, 55.46% for the **Surfing** video, and 75.86% for the **VR Interview** video) averaged across three prediction horizons. The lower values of perfect viewport match for the sports videos, i.e., **Football** and **Surfing**, reflect the fast-moving objects within these videos. Therefore, the client observes a lower percentage of aggressive bitrate adaptation, 34.03%, 31.31% for the **Football** and **Surfing** videos, with an 8x6 tiling layout in comparison to other videos. For content with minimal movements, such as the **Performance** and **VR Interview** videos, there is only a small percentage of viewport mismatch even when the segment duration is set to 2s. **DFT2** requests a 4x3 tiling layout by up to 3.36% and 5.94% for the **Performance** and **VR Interview** videos, respectively, for 2s segment duration. Therefore, these videos observe a limited percentage of extended viewport and conservative bitrate adaptation cases compared to the other videos. Interestingly, the percentage of the 4x3 and 6x4 tiling layouts selection increases with the increase in segment duration. Additionally, as the segment duration increases, the viewer tends to experience a higher percentage of weighted and conservative quality adjustments. Conversely, the percentage of fixed viewport cases that come with aggressive bitrate adjustments tends to decrease with longer segment durations. This is because the accuracy of predictions tends to decline when attempting to predict further into the future.

Fig. 5 represents the streaming behaviour of the ATS algorithm in terms of selecting the average tiling layouts for the entire video data set. ATS selects tiling layouts based on the minimum weighted viewport distortions measured to achieve maximum viewport bitrate. ATS results in selecting a 4x3 tiling layout mostly for **Football** video, followed by 8x6 and 6x4 tiling grids. ATS requests 6x4 and 8x6 tiling layouts for about 15.06% and 50.94% of the streaming session for **Performance** video with 1s, 1.5s, and 2s. The 6x4 tiling layout is mostly requested for **VR Interview** video

Table 5. Streaming Behavior of **DFT2** Client in terms of Tiling Layouts Selection, Tiles Selection, and Bitrate Adaptation Scenarios. The Percentage Results are Averaged for Five Videos Watched by 48 VR Users.

Videos	Segment Duration	Tiling Layout [%]			Tiles Selection: Case 1 Bitrate: Aggressive			Tiles Selection: Case 2 Bitrate: Weighted			Tiles Selection: Case 3 Bitrate: Conservative		
		8x6	6x4	4x3	8x6	6x4	4x3	8x6	6x4	4x3	8x6	6x4	4x3
Football	1	62.94	31.97	5.09	39.46	10.66	0.74	23.24	21.16	2.39	0.24	0.15	1.96
	1.5	56.31	33.68	10.02	33.03	9.27	0.73	22.76	24.06	4.05	0.52	0.34	5.24
	2	52.82	34.55	12.63	29.62	8.51	0.76	22.41	25.28	4.07	0.79	0.76	7.80
Performance	1	77.81	20.51	1.68	57.95	8.03	0.20	19.66	12.40	0.85	0.19	0.08	0.62
	1.5	73.85	23.28	2.87	52.48	7.38	0.17	21.07	15.64	1.19	0.30	0.26	1.51
	2	69.99	26.65	3.36	48.91	7.76	0.10	20.65	18.51	0.99	0.43	0.37	2.26
Spotlight	1	71.39	24.18	4.44	46.03	8.26	0.75	25.21	15.73	1.92	0.15	0.19	1.76
	1.5	64.12	28.40	7.48	38.76	7.69	0.56	24.96	20.42	2.69	0.41	0.29	4.23
	2	59.35	30.77	9.88	35.50	9.18	1.50	23.38	21.03	2.95	0.47	0.55	5.43
Surfing	1	62.08	32.17	5.74	37.10	10.25	0.87	24.75	21.78	2.71	0.23	0.14	2.15
	1.5	54.41	34.93	10.66	30.52	8.06	0.81	23.51	26.51	4.43	0.38	0.36	5.43
	2	49.90	36.25	13.86	26.31	7.71	0.85	22.92	27.51	4.53	0.67	1.03	8.47
VR Interview	1	79.18	17.67	3.15	58.25	6.38	0.29	20.68	11.23	1.59	0.25	0.06	1.27
	1.5	74.75	20.33	4.92	53.41	6.15	0.34	20.93	13.93	1.65	0.40	0.25	2.94
	2	73.66	20.41	5.94	50.13	5.62	0.31	22.92	14.14	1.68	0.60	0.65	3.94

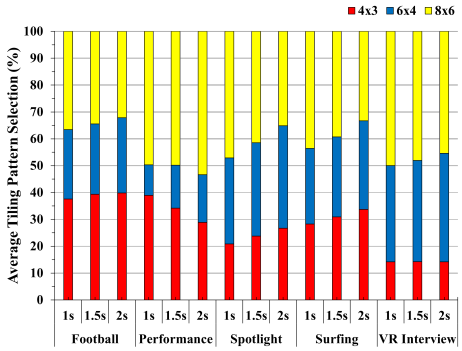
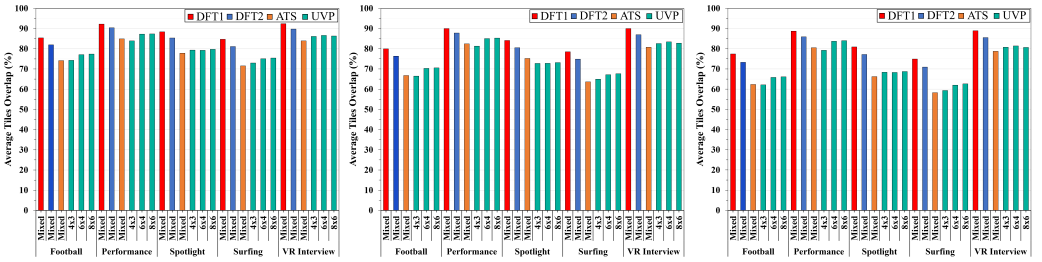


Fig. 5. Average Tiling Layout Selection in the ATS method.

with a 2s segment duration. For **Spotlight** and **Surfing** videos, ATS mostly requests an 8x6 tiling layout (41.23% and 38.71%) followed by the 6x4 (34.97% and 30.29%) and 4x3 (23.79% and 30.98%), respectively. For the entire test dataset, the ATS method achieves 42.61% for selecting 8x6, 29% for 6x4, and 28.39% for 4x3. This is because the larger tiling layout results in relatively larger segment sizes.

5.2.2 Average Tiles Overlap. Fig. 6 summarizes the average tiles overlap results (per video 48 head movement traces) for the **DFT1**, **DFT2**, ATS, and UVP methods under various prediction horizons. The ATS, UVP, CTF, PBA, and AVR streaming algorithms all use the spherical walk prediction method which is used to inform adaptive tiles selection and bitrate selection. According to Fig. 6, it can be seen that the **DFT1** method leads to higher tile overlap for all five videos. This is because the tiles in the dynamic tiling layouts produced by **DFT1** are arranged based on the arc distance between the viewpoint and the centre of each tile. This allows **DFT1** to cover the viewport and reduce the risk of gaps in the visual field. The **Football** and **Surfing** videos tend to elicit more dynamic head movements from viewers because they contain fast-moving outdoor sports-related objects. In contrast, the **Performance** and **VR interview** videos tend to have a higher average tile overlap because they feature slower-moving indoor objects that are the primary focus of attention. This suggests that the nature of the content being watched can impact the amount of head movement and, in turn, the tiles overlap observed in the video. It is notable that **DFT1** and **DFT2** attain higher



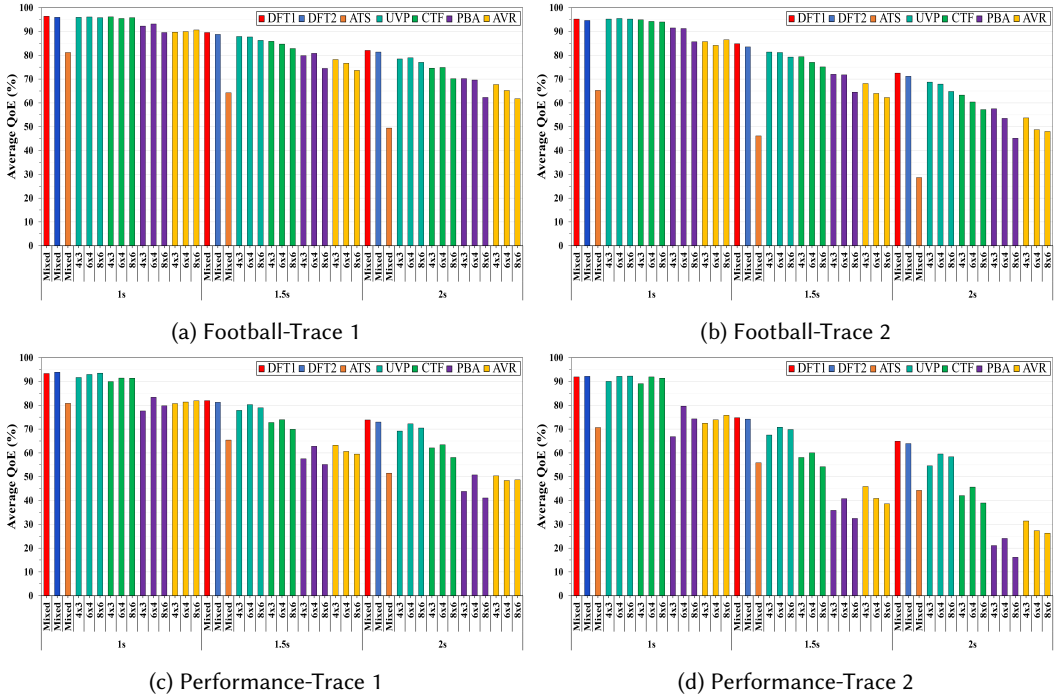
(a) Segment duration: 1s (b) Segment duration: 1.5s (c) Segment duration: 2s

Fig. 6. Average tiles overlap achieved by **DFT1**, **DFT2**, and Spherical Walk methods for the **Football**, **Performance**, **Spotlight**, **Surfing**, and **VR Interview** videos. These videos were prepared in 4x3, 6x4, and 8x6 tiling layouts, which were watched by 48 VR users. The recorded results are for 1s, 1.5s, and 2s segment durations.

matching performance and outperform the ATS and UVP methods for different user behaviours. For all 48 VR users, **DFT1** and **DFT2** experience an average tiles overlap of 85.40% and 81.95% (**Football**), 92.22% and 90.43% (**Performance**) for 1s (Fig. 6a), 84.09% and 80.50% (**Spotlight**), 90.03% and 86.94% (**VR Interview**) for 1.5s (Fig. 6b), 74.93% and 70.91% (**Surfing**) and 88.92% and 85.54% (**Performance**) for 2s (Fig. 6c) prediction windows. The proactive tiles selection methods are able to adapt more effectively to the varied spatial and temporal information present in different motion scenes, which explains their superior performance. Simultaneously, The ATS method exhibits a lower average tile overlap than the UVP method for content with fast and stable head movements. As can be seen in the **Spotlight** video, **DFT1** outperforms the ATS and UVP methods by up to 8.88% and 11.19% for the next 1.5s (Fig. 6b), and by 14.66% and 12.43% for a 2s prediction horizon (Fig. 6c), respectively. Similarly, **DFT2** demonstrates its ability to increase viewport overlap for the **Surfing** video, outperforming other methods by achieving viewport overlap that is about 7.37%, 9.02%, and 10.35% higher for 1s, 1.5s, and 2s prediction times, respectively. For **Spotlight** video, the average gain of **DFT** methods ranges from 6.27%-9.32%, 7.02%-10.61%, and 9.28%-12.98% for different prediction horizons. The tiles overlap for the **DFT2** is reduced by 8.64% (**Football**) and by 10.23% (**Surfing**) when the segment duration is increased from 1s to 2s. In contrast, for the ATS and UVP methods, the tiles overlap is reduced by 11.83% and 11.57% (**Football**) and by 13.29% and 13.17% (**Surfing**), respectively (Fig. 6). This indicates that the **DFT2** method is more effective at maintaining a high level of tile overlap even when the segment duration is increased. As a result, it can be concluded that employing two prediction mechanisms (as in **DFT**) leads to better viewing probability than employing a single prediction mechanism for fixed (UVP) and dynamic (ATS) tiling-based streaming.

5.2.3 Average QoE. Next, the performance of the proposed solutions is tested against five tile-based methods by employing bandwidth trace 1 and trace 2 for **Football** and **Performance** videos. We normalized the values of the QoE functions defined in eq. 1-4. The QoE weight coefficients are set as $\alpha = 1, \beta = 0.8, \gamma = 0.6, \delta = 0.2$. The weights are selected to emphasize a different combination of QoE objectives. A larger value of α indicates that the user is more concerned with the quality of the viewport, while a smaller value of δ indicates that the user places less importance on playback buffer risk. Increasing the weights of the β, γ, and δ parameters results in negative QoE values for CTF and PBA clients for Surfing videos. Therefore, these values are selected to provide a useful QoE comparison between the proposed and other solutions.

The reference tile-based delivery solutions use viewers' head motion patterns to adaptively select bitrates. Fig. 7 depicts the video quality experienced and averaged across 48 users for 1s, 1.5s, and 2s



(a) Football-Trace 1

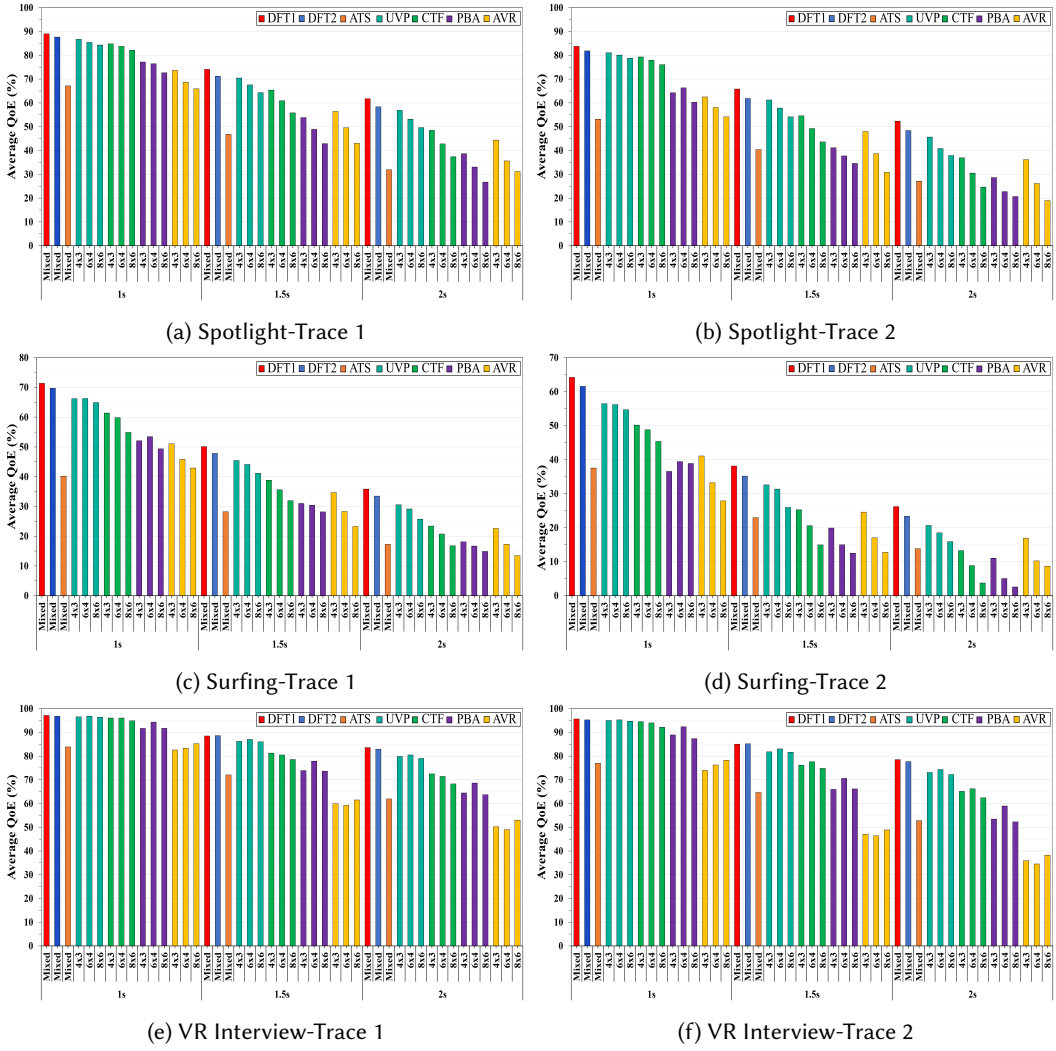
(b) Football-Trace 2

(c) Performance-Trace 1

(d) Performance-Trace 2

Fig. 7. Average QoE achieved by **DFT1**, **DFT2**, ATS, UVP, CTF, PBA, and AVR streaming clients for **Football** and **Performance** videos.

segments. It can be seen that the performance of the algorithms in Fig. 7a and Fig. 7c is higher than that shown in Fig. 7b and Fig. 7d. The average QoE values are lower accordingly with bandwidth decrease for the same QoE weight coefficients. The higher QoE scores of larger tiling layouts (i.e., 6x4 and 8x6) for the 1s **Performance** video (Fig. 7c-Fig. 7d) are due to the higher average tiles overlap. Despite the lower tiles overlap, the UVP, CTF, PBA, and AVR streaming methods achieve higher quality scores for the **Football** video with a 1s segment duration due to the smaller average segment sizes (Fig. 7a and Fig. 7b). Fig. 7a results show that **DFT1** improves the QoE compared to other methods by about 3.96%, 9.29%, and 12.90% for **Football** video with 1s, 1.5s, and 2s segment durations when employing bandwidth trace 1. For both bandwidth traces, **DFT1** outperforms ATS by about 25.31%-38.71%, UVP by about 2.25%-4.25%, CTF by about 5.08%-7.67%, PBA by about 11.16%-15.42%, and AVR by about 13.37%-20.07% for **Football** video with a 1.5s segment duration. Fig. 7b shows that **DFT2** achieves about 5.44% (for 1s), 12.56%(for 1.5s), and 15.98% (for 2s), higher average QoE for **Football** video streaming in comparison to other solutions. The increment in quality with the increase in segment duration reflects that **DFT** solutions have better prediction accuracy with longer segment duration. Similarly, Fig. 7c and Fig. 7d show that **DFT** solutions observe the highest visual quality levels for all segment durations since they better accommodate the user's viewing directions than the other methods. In particular, **DFT1** achieves an average gain of 7.45% (1s), 14.42% (1.5s), and 17.69% (2s) for **Performance** video streaming under bandwidth trace 1, while it is increased to 10.34% (1s), 23.20% (1.5s) and 27.23% (2s) for bandwidth trace 2. Viewport mismatch leads to a drop in quality for tile-based streaming methods for longer segment lengths. In **DFT** methods, the combination of viewport coverage selection and bitrate selection policies favour the higher quality perceptibility of the viewing area. For **Performance** video



(a) Spotlight-Trace 1

(b) Spotlight-Trace 2

(c) Surfing-Trace 1

(d) Surfing-Trace 2

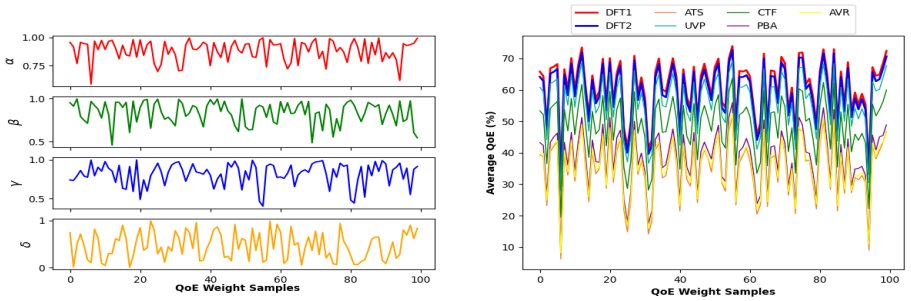
(e) VR Interview-Trace 1

(f) VR Interview-Trace 2

Fig. 8. Average QoE achieved by **DFT1**, **DFT2**, ATS, UVP, CTF, PBA, and AVR streaming clients for the **Spotlight**, **Surfing**, and **VR Interview** videos.

with a 2s segment duration, **DFT2** outperforms fixed tiling-based solutions by about 2.34%-6.39%, 11.76%-21.67%, 27.75%-43.46%, and 23.80%-35.58% for both bandwidth scenarios. The improved performance of **DFT** solutions over CTF and PBA methods is for the reason that they perform a uniform quality allocation for the predictive tiles to favour the higher visual quality levels with a reduced amount of data for the background tiles.

The results of the experiments on the **Spotlight**, **Surfing**, and **VR Interview** videos are shown in Fig. 8. It can be seen that the **Surfing** and **Spotlight** videos require higher bitrates for satisfactory quality scores (as seen in Table 3), making it more difficult to achieve a high QoE with limited network connections and high QoE expectations. On the other hand, the **VR Interview** video has higher QoE scores due to its smaller average segment sizes and higher viewport overlap. Therefore, factors such as segment size, bandwidth capacity, and viewport prediction significantly impact the



(a) QoE Weight Coefficients Samples (b) Average QoE Vs QoE Weight Coefficients

Fig. 9. Average QoE obtained by **DFT1**, **DFT2**, ATS, UVP, CTF, PBA, and AVR streaming clients for the comprehensive dataset (comprising 5 videos, 3 tiling patterns, 3 segment durations, and 2 bandwidth traces) when assessed under varying QoE weight coefficients.

streaming performance of 360° videos. For example, when streaming the **Spotlight** video with a 2s segment duration, the **DFT1** method achieves average QoE improvements of up to 29.8%, 12.15%, 24.36%, 28.7%, and 30.6% compared to ATS, UVP 8x6, CTF 8x6, PBA 6x4, and AVR 8x6, respectively (Fig. 8b). This is because **DFT1** has 14.65% and 12.15% higher average tiles overlap than the ATS and UVP methods for the **Spotlight** video with a 2s segment duration (Fig. 6c). The average quality score for the **Surfing** video with a 1s segment duration under bandwidth trace 2 (Fig. 8d) is 64.21% for **DFT1**, 61.57% for **DFT2**, 37.53% for ATS, 56.45% for UVP 4x3, 48.79% for CTF 6x4, 38.9% for PBA 8x6, and 41.08% for AVR 4x3. For the **VR Interview** video with a 1.5s segment duration, **DFT2** improves the average QoE by up to 20.55% compared to ATS, 3.02% compared to UVP, 9% compared to CTF, 17.61% compared to PBA, and 37.74% compared to AVR for bandwidth trace 2 (Fig. 8f), while the average improvement for **DFT1** is 25.7%, 5.3%, 13.94%, 23.64%, and 42.3% for all tiling layouts of ATS, UVP, CTF, PBA, and AVR, respectively, for the 2s **VR Interview** video (Fig. 8e). The ATS method performs better than the AVR method in only a few cases for the **Performance** and **VR Interview** videos. The poor performance of the ATS method is due to its restriction of the quality of background tiles to minimum levels, which leads to lower quality scores under lower and medium prediction performance. In Fig. 8, it can be seen that when simulated with all tiling layouts, segment durations, and bandwidth profiles, the **DFT1** and **DFT2** methods result in QoE for **Spotlight**, **Surfing**, and **VR Interview** videos, with improvements of 16.53%, 15.56%, and 13.62%, respectively. This is because the QoE metric used favours higher visible quality. The lower QoE values for the PBA algorithm are due to its strategy of assigning different priorities to tiles within the viewport zones (Z_1 and Z_2) and lead to poor user-perceived quality and visual smoothness. The AVR method, meanwhile, performs poorly even under stable head movements because it unnecessarily increases the quality of adjacent tiles. In general, the **DFT1** and **DFT2** solutions lead to average QoE improvements of 9.70%-10.56% for **Football**, 16.33%-16.72% for **Performance**, 15.08%-18% for **Spotlight**, 14.33%-16.79% for **Surfing**, and 13.45%-13.79% for **VR Interview** videos compared to other solutions.

5.2.4 Ablation Study—Impact of QoE Weight Coefficients. We investigated and evaluated the influence of QoE weight coefficients on the streaming performance of adaptive 360° video solutions. For each streaming solution, we collected streaming metrics, presented in eq. (1) - eq. (4), including viewport quality, temporal quality oscillations, spatial quality oscillation, and playback buffer risk, across a comprehensive testing dataset that encompassed five videos, three tiling patterns, three segment durations, and two bandwidth traces. Fig. 9a illustrates the values for QoE weight

coefficients where the values of α, β, γ, and δ are in the range of 0 and 1. Fig. 9b displays the average QoE values for each corresponding weight sample. The findings from Fig. 9b reveal that **DFT1** and **DFT2** solutions consistently outperform the other methods by achieving the highest QoE scores across all combinations of QoE weight samples. The average QoE scores obtained are as follows: **DFT1** (60.82%), **DFT2** (59.17%), ATS (35.08%), UVP (56.06%), CTF (48.45%), PBA (38.23%), and AVR (36.02%). In general, **DFT1** and **DFT2** surpass ATS by up to 24-25.74%, UVP by up to 3.11-4.76%, CTF by up to 10.71-12.36%, PBA by up to 20.94-22.59%, and AVR by up to 23.14-24.79% in terms of improved QoE performance. The average QoE weights coefficients are α = 0.885, β = 0.835, γ = 0.817, and δ = 0.466.

5.3 Discussion

Existing fixed tiling-based adaptive streaming solutions aim to improve visual quality while reducing variations in spatial and temporal quality and the risk of playback interruptions. However, the proposed dynamic tiling-based streaming solutions result in more accurate viewport prediction and higher QoE levels since they systematize the best resolution tiles for static and dynamic motion scenes. The ATS and UVP solutions allocate bitrate uniformly to tiles in the same classification to improve the visual smoothness objectives defined in Eq. 2-3. However, ATS limits the background quality to the minimum level with lower viewport matching performance and achieves the lowest average QoE values for the entire test dataset. The UVP solution, on the other hand, increases the quality of the whole video to the highest possible level and produces better quality scores even under difficult-to-predict head movements. For CTF and PBA solutions, the primary focus is on improving the quality of the centre tile, which significantly leads to degraded quality levels for poor viewport prediction and spatial quality variations even for stable viewport prediction results. The underperformance of the AVR streaming method under drastic and stable viewport switches is due to the inefficient tiles' arrangement to consume an essential share of the network bandwidth. In contrast, **DFT** solutions consume a much larger bandwidth share for the most likely to be watched tiles and result in higher QoE scores than the comparative methods for all tested datasets. **DFT1** provides a useful trade-off between visual area and visual quality, and **DFT2** works to minimize the viewport mismatch ratio. Both proposed solutions work reasonably well under different testing settings and try to avoid unacceptable viewport deviations for end-users. The **DFT** solutions allocate a fair share of the bandwidth to tiles in the viewport, marginal, and background regions, resulting in lower spatial and temporal quality variations for different viewport prediction results. Under stable or variable motions of experienced or naive VR users, the dynamic selection of the tiling layouts and coverage of the visible region (Fixed/Extended) along with the aggressive, weighted, and/or conservative quality adjustment policies provide improved QoE for different bandwidth settings, segment sizes, and motion trends. Therefore, the proposed solutions have demonstrated their potential to offer superior quality of experience compared to other approaches for delivering 360° video.

6 CONCLUSIONS AND FUTURE WORKS

This paper proposed and evaluated two novel dynamic video frames tiling-based solutions, **DFT1** and **DFT2**, for advanced predictive tiles selection during adaptive 360° video streaming. **DFT** solutions achieve an appropriate balance between viewport availability and perceived visual quality. **DFT1** performs an interactive tiling layout selection by leveraging the visual area and associated weighted quality with overcoming the attention field's dynamics. **DFT2** observes the potential viewport prediction errors to best accommodate different tiling layouts. **DFT** solutions extract the user attention fields by leveraging two viewport prediction mechanisms to select the best-fit dynamic size regions for transmission over bandwidth-limited networks. The proposed solutions

consider the level of interest in each region when deciding how much bitrate it should receive in order to simplify the process of selecting the appropriate bitrate for each tile. The effectiveness of the **DFTs** algorithms was evaluated through extensive trace-driven experiments. The experimental results on publicly available dataset under different segment lengths and bandwidth settings demonstrate that the proposed solutions achieve up to 8.6%, 9.77%, and 11.2% improved viewport availability for 1s, 1.5s, and 2s segment duration. At the same time, **DFT** solutions can improve QoE (9.7%-18%) for different motion VR videos compared to other alternative solutions. In the future, we aim to develop a guidance-enhanced fuzzy reinforcement learning (FRL) solution to control the continuous tile selection and bitrate adaptation for equirectangular, cubemap, and truncated squared pyramid projected 360° videos under more complex network and head movement datasets. Using advanced QoE metrics, we will evaluate the effectiveness of our FRL-based solution and identify any potential optimization opportunities.

REFERENCES

[1] Yanan Bao, Huasen Wu, Tianxiao Zhang, Albara Ah Ramli, and Xin Liu. 2016. Shooting a Moving Target: Motion-Prediction-based Transmission for 360-Degree Videos. In *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 1161–1170.

[2] M. Ben Yahia, Y. Le Louedec, G. Simon, and L. Nuaymi. 2018. HTTP/2-Based Streaming Solutions for Tiled Omnidirectional Videos. In *2018 IEEE International Symposium on Multimedia (ISM)*. 89–96. https://doi.org/10.1109/ISM.2018.00023

[3] Tengfei Cao, Changqiao Xu, Mu Wang, Zhongbai Jiang, Xingyan Chen, Lujie Zhong, and Luigi Alfredo Grieco. 2019. Stochastic Optimization for Green Multimedia Services in Dense 5G Networks. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 15, 3 (2019), 1–22.

[4] Xiaolei Chen, Di Wu, and Ishfaq Ahmad. 2021. Optimized Viewport-adaptive 360-degree Video Streaming. *CAAI Transactions on Intelligence Technology* (2021).

[5] Cyril Concolato, Jean Le Feuvre, Franck Denoual, Frédéric Mazé, Eric Nassor, Nael Ouedraogo, and Jonathan Taquet. 2017. Adaptive Streaming of HEVC Tiled Videos Using MPEG-DASH. *IEEE transactions on circuits and systems for video technology* 28, 8 (2017), 1981–1992.

[6] 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*. 199–204.

[7] Xavier Corbillon, Gwendal Simon, Alisa Devlic, and Jacob Chakareski. 2017. Viewport-Adaptive Navigable 360-Degree Video Delivery. In *Communications (ICC), 2017 IEEE International Conference on*. IEEE, 1–7.

[8] R. G. d. A. Azevedo, N. Birkbeck, F. De Simone, I. Janatra, B. Adsumilli, and P. Frossard. 2019. Visual Distortions in 360-degree Videos. *IEEE Transactions on Circuits and Systems for Video Technology* (2019), 1–1. https://doi.org/10.1109/TCSVT.2019.2927344

[9] Pingping Dong, Rongcheng Shen, Xiaowei Xie, Yajing Li, Yuning Zuo, and Lianming Zhang. 2022. Predicting Long-term Field of View in 360-degree Video Streaming. *IEEE Network* (2022), 1–8. https://doi.org/10.1109/MNET.106.2100449

[10] Miguel Fabian Romero Rondon, Lucile Sassatelli, Ramon Aparicio Pardo, and Frederic Precioso. 2019. Revisiting Deep Architectures for Head Motion Prediction in 360° Videos. *arXiv e-prints*, Article arXiv:1911.11702 (Nov. 2019), arXiv:1911.11702 pages. arXiv:1911.11702 [cs.CV]

[11] Ajoy S Fernandes and Steven K Feiner. 2016. Combating VR Sickness Through Subtle Dynamic Field-of-View Modification. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE, 201–210.

[12] Mario Graf, Christian Timmerer, and Christopher Mueller. 2017. Towards Bandwidth Efficient Adaptive Streaming of Omnidirectional Video over HTTP: Design, Implementation, and Evaluation. In *Proceedings of the 8th ACM on Multimedia Systems Conference* (Taipei, Taiwan) *(MMSys'17)*. ACM, New York, NY, USA, 261–271. https://doi.org/10.1145/3083187.3084016

[13] Chengjun Guo, Ying Cui, and Zhi Liu. 2018. Optimal multicast of tiled 360 VR video. *IEEE Wireless Communications Letters* 8, 1 (2018), 145–148.

[14] Dongbiao He, Cedric Westphal, and J Garcia-Luna-Aceves. 2018. Joint Rate and FoV adaptation in immersive video streaming. In *ACM Sigcomm workshop on AR/VR Networks*.

[15] Jeroen Van der Hooft, Maria Torres Vega, Stefano Petrangeli, Tim Wauters, and Filip De Turck. 2019. Tile-based Adaptive Streaming for Virtual Reality Video. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 15, 4 (2019), 1–24.

[16] Mohammad Hosseini and Viswanathan Swaminathan. 2016. Adaptive 360 VR Video Streaming: Divide and Conquer. In *2016 IEEE International Symposium on Multimedia (ISM)*. 107–110. https://doi.org/10.1109/ISM.2016.0028

[17] Xinjue Hu, Wei Quan, Tao Guo, Yu Liu, and Lin Zhang. 2019. Mobile Edge Assisted Live Streaming System for Omnidirectional Video. *Mobile Information Systems* 2019 (2019).
[18] ITU-T Recommendation. 2014. P. 913: Methods for the Subjective Assessment of Video Quality, Audio Quality and Audiovisual Quality of Internet Video and Distribution Quality Television in Any Environment.
[19] Xiaolan Jiang, Yi-Han Chiang, Yang Zhao, and Yusheng Ji. 2018. Plato: Learning-based Adaptive Streaming of 360-Degree Videos. In *2018 IEEE 43rd Conference on Local Computer Networks (LCN)*. IEEE, 393–400.
[20] Chamara Kattadige and Kanchana Thilakarathna. 2021. VAD360: Viewport Aware Dynamic 360-Degree Video Frame Tiling. *arXiv preprint arXiv:2105.11563* (2021).
[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*. ACM, 41.
[22] Weihe Li, Jiawei Huang, Wenjun Lyu, Baoshen Guo, Wanchun Jiang, and Jianxin Wang. 2022. RAV: Learning-Based Adaptive Streaming to Coordinate the Audio and Video Bitrate Selections. *IEEE Transactions on Multimedia* (2022).
[23] 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*. 211–216.
[24] Kaixuan Long, Chencheng Ye, Ying Cui, and Zhi Liu. 2018. Optimal Multi-Quality Multicast for 360 Virtual Reality Video. In *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 1–6.
[25] Pietro Lungaro, Rickard Sjöberg, Alfredo Jose Fanghella Valero, Ashutosh Mittal, and Konrad Tollmar. 2018. Gaze-Aware Streaming Solutions for the Next Generation of Mobile VR Experiences. *IEEE transactions on visualization and computer graphics* 24, 4 (2018), 1535–1544.
[26] Hongzi Mao, Ravi Netravali, and Mohammad Alizadeh. 2017. Neural Adaptive Video Streaming With PENSIEVE. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. ACM, 197–210.
[27] Gebremariam Mesfin, Nadia Hussain, Alexandra Covaci, and Gheorghita Ghinea. 2019. Using Eye Tracking and Heart-Rate Activity to Examine Crossmodal Correspondences QoE in Mulsemedia. *ACM Trans. Multimedia Comput. Commun. Appl.* 15, 2, Article 34 (jun 2019), 22 pages. https://doi.org/10.1145/3303080
[28] Afshin Taghavi Nasrabadi, Anahita Mahzari, Joseph D Beshay, and Ravi Prakash. 2017. Adaptive 360-Degree Video Streaming using Scalable Video Coding. In *Proceedings of the 2017 ACM on Multimedia Conference*. ACM, 1689–1697.
[29] Afshin Taghavi Nasrabadi, Aliehsan Samiei, and Ravi Prakash. 2020. Viewport Prediction for 360° Videos: A Clustering Approach. In *Proceedings of the 30th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video* (Istanbul, Turkey) *(NOSSDAV '20)*. Association for Computing Machinery, New York, NY, USA, 34–39. https://doi.org/10.1145/3386290.3396934
[30] Khiem Quang Minh Ngo, Ravindra Guntur, and Wei Tsang Ooi. 2011. Adaptive Encoding of Zoomable Video Streams based on User Access Pattern. In *Proceedings of the second annual ACM conference on Multimedia systems*. 211–222.
[31] Duc V Nguyen, Huyen TT Tran, Anh T Pham, and Truong Cong Thang. 2019. An Optimal Tile-Based Approach for Viewport-Adaptive 360-Degree Video Streamings. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 9, 1 (2019), 29–42.
[32] Duc V Nguyen, Huyen TT Tran, and Truong Cong Thang. 2019. Adaptive Tiling Selection for Viewport Adaptive Streaming of 360-degree Video. *IEICE Transactions on Information and Systems* 102, 1 (2019), 48–51.
[33] Leandro Ordonez-Ante, Jeroen van der Hooft, Tim Wauters, Gregory Van Seghbroeck, Bruno Volckaert, and Filip De Turck. 2022. Explora-VR: Content Prefetching for Tile-Based Immersive Video Streaming Applications. *Journal of Network and Systems Management* 30, 3 (2022), 1–30.
[34] Cagri Ozcinar, Julián Cabrera, and Aljosa Smolic. 2018. Omnidirectional Video Streaming Using Visual Attention-Driven Dynamic Tiling for VR. In *2018 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 1–4.
[35] C. Ozcinar, J. Cabrera, and A. Smolic. 2019. Visual Attention-Aware Omnidirectional Video Streaming Using Optimal Tiles for Virtual Reality. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 9, 1 (March 2019), 217–230. https://doi.org/10.1109/JETCAS.2019.2895096
[36] Stefano Petrangeli, Viswanathan Swaminathan, Mohammad Hosseini, and Filip De Turck. 2017. An HTTP/2-based Adaptive Streaming Framework for 360 Virtual Reality Videos. In *Proceedings of the 2017 ACM on Multimedia Conference*. ACM, 306–314.
[37] Feng Qian, Bo Han, Qingyang Xiao, and Vijay Gopalakrishnan. 2018. Flare: Practical Viewport-Adaptive 360-Degree Video Streaming for Mobile Devices. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*. ACM, 99–114.
[38] Feng Qian, Lusheng Ji, Bo Han, and Vijay Gopalakrishnan. 2016. Optimizing 360 Video Delivery Over Cellular Networks. In *Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges*. ACM, 1–6.
[39] Ngo Quang Minh Khiem, Guntur Ravindra, Axel Carlier, and Wei Tsang Ooi. 2010. Supporting Zoomable Video Streams with Dynamic Region-of-Interest Cropping. In *Proceedings of the first annual ACM SIGMM conference on*

Multimedia systems. 259–270.

[40] Yago Sánchez, Robert Skupin, and Thomas Schierl. 2015. Compressed Domain Video Processing for Tile based Panoramic Streaming Using HEVC. In *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2244–2248.

[41] Muhammad Shahid Anwar, Jing Wang, Sadique Ahmad, Asad Ullah, Wahab Khan, and Zesong Fei. 2020. Evaluating the Factors Affecting QoE of 360-degree Videos and Cybersickness Levels Predictions in Virtual Reality. *Electronics* 9, 9 (2020), 1530.

[42] Kevin Spiteri, Rahul Urgaonkar, and Ramesh K Sitaraman. 2016. BOLA: Near-Optimal Bitrate Adaptation for Online Videos. In *INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE*. IEEE, 1–9.

[43] Evgeniy Upenik and Touradj Ebrahimi. 2017. A simple method to obtain visual attention data in head mounted virtual reality. In *2017 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*. 73–78. https://doi.org/10.1109/ICMEW.2017.8026231

[44] Hui Wang, Vu-Thanh Nguyen, Wei Tsang Ooi, and Mun Choon Chan. 2014. Mixing tile resolutions in tiled video: A perceptual quality assessment. In *Proceedings of Network and Operating System Support on Digital Audio and Video Workshop*. ACM, 25.

[45] Xuekai Wei, Mingliang Zhou, Sam Kwong, Hui Yuan, and Weijia Jia. 2021. A Hybrid Control Scheme for 360-Degree Dynamic Adaptive Video Streaming over Mobile Devices. *IEEE Transactions on Mobile Computing* (2021).

[46] Xuekai Wei, Mingliang Zhou, Sam Kwong, Hui Yuan, Shiqi Wang, Guopu Zhu, and Jingchao Cao. 2021. Reinforcement Learning-based QoE-oriented Dynamic Adaptive Streaming Framework. *Information Sciences* 569 (2021), 786–803.

[47] 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*. 193–198.

[48] Mengbai Xiao, Chao Zhou, Yao Liu, and Songqing Chen. 2017. Optile: Toward Optimal Tiling in 360-degree Video Streaming. In *Proceedings of the 25th ACM international conference on Multimedia*. 708–716.

[49] 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. *Proceedings of the ACM Multimedia Conference* (2017), 315–323. 10.1145/3123266.3123291

[50] Praveen Kumar Yadav and Wei Tsang Ooi. 2020. Tile Rate Allocation for 360-degree Tiled Adaptive Video Streaming. In *Proceedings of the 28th ACM International Conference on Multimedia*. 3724–3733.

[51] Praveen Kumar Yadav, Arash Shafiei, and Wei Tsang Ooi. 2017. QUETRA: A Queuing Theory Approach to DASH Rate Adaptation. In *Proceedings of the 25th ACM International Conference on Multimedia* (Mountain View, California, USA) *(MM '17)*. Association for Computing Machinery, New York, NY, USA, 1130–1138. https://doi.org/10.1145/3123266.3123390

[52] A. Yaqoob, T. Bi, and G.-M. Muntean. 2019. A DASH-based Efficient Throughput and Buffer Occupancy-based Adaptation Algorithm for Smooth Multimedia Streaming. In *2019 15th International Wireless Communications Mobile Computing Conference (IWCMC)*. 643–649. https://doi.org/10.1109/IWCMC.2019.8766648

[53] A. Yaqoob, T. Bi, and G. M. Muntean. 2020. A Survey on Adaptive 360° Video Streaming: Solutions, Challenges and Opportunities. *IEEE Communications Surveys Tutorials* 22, 4 (2020), 2801–2838. https://doi.org/10.1109/COMST.2020.3006999

[54] Abid Yaqoob and Gabriel-Miro Muntean. 2020. A Weighted Tile-based Approach for Viewport Adaptive 360° Video Streaming. In *2020 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. IEEE.

[55] Abid Yaqoob and Gabriel-Miro Muntean. 2021. A Combined Field-of-View Prediction-Assisted Viewport Adaptive Delivery Scheme for 360° Videos. *IEEE Transactions on Broadcasting* 67, 3 (2021), 746–760. https://doi.org/10.1109/TBC.2021.3105022

[56] Abid Yaqoob, Mohammed Amine Togou, and Gabriel-Miro Muntean. 2022. Dynamic Viewport Selection-Based Prioritized Bitrate Adaptation for Tile-Based 360° Video Streaming. *IEEE Access* 10 (2022), 29377–29392. https://doi.org/10.1109/ACCESS.2022.3157339

[57] Hui Yuan, Shiyun Zhao, Junhui Hou, Xuekai Wei, and Sam Kwong. 2019. Spatial and Temporal Consistency-Aware Dynamic Adaptive Streaming for 360-Degree Videos. *IEEE Journal of Selected Topics in Signal Processing* 14, 1 (2019), 177–193.

[58] Zhenhui Yuan, Shengyang Chen, Gheorghita Ghinea, and Gabriel-Miro Muntean. 2014. User Quality of Experience of Mulsemedia Applications. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 11, 1s (2014), 1–19.

[59] Alireza Zare, Alireza Aminlou, Miska M Hannuksela, and Moncef Gabbouj. 2016. HEVC-Compliant Tile-based Streaming of Panoramic Video for Virtual Reality Applications. In *Proceedings of the 24th ACM international conference on Multimedia*. ACM, 601–605.

[60] Haodan Zhang, Yixuan Ban, Zongming Guo, Ken Chen, and Xinggong Zhang. 2022. RAM360: Robust Adaptive Multi-layer 360 Video Streaming with Lyapunov Optimization. *IEEE Transactions on Multimedia* (2022).

[61] Lei Zhang, Yanyan Suo, Ximing Wu, Feng Wang, Yuchi Chen, Laizhong Cui, Jiangchuan Liu, and Zhong Ming. 2021. TBRA: Tiling and Bitrate Adaptation for Mobile 360-Degree Video Streaming. In *Proceedings of the 29th ACM International Conference on Multimedia*. 4007–4015.

[62] Yuanhong Zhang, Zhiwen Wang, Junquan Liu, Haipeng Du, Qinghua Zheng, and Weizhan Zhang. 2022. Deep Reinforcement Learning Based Adaptive 360-degree Video Streaming with Field of View Joint Prediction. In *2022 IEEE Symposium on Computers and Communications (ISCC)*. 1–8. https://doi.org/10.1109/ISCC55528.2022.9913007

[63] Yuanxing Zhang, Pengyu Zhao, Kaigui Bian, Yunxin Liu, Lingyang Song, and Xiaoming Li. 2019. DRL360: 360-degree Video Streaming with Deep Reinforcement Learning. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 1252–1260.

[64] Chao Zhou, Zhenhua Li, Joe Osgood, and Yao Liu. 2018. On the Effectiveness of Offset Projections for 360-Degree Video Streaming. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 14, 3s (2018), 1–24.

[65] Junni Zou, Chenglin Li, Chengming Liu, Qin Yang, Hongkai Xiong, and Eckehard Steinbach. 2019. Probabilistic Tile Visibility-based Server-Side Rate Adaptation for Adaptive 360-Degree Video Streaming. *IEEE Journal of Selected Topics in Signal Processing* 14, 1 (2019), 161–176.