network

MDPI

Review

Adaptive 360° Video Streaming: Prediction, Tiling, and Transport Trade-Offs

Muhammad Farooq, Gioacchino Manfredi, Luca De Cicco * and Saverio Mascolo

Dipartimento di Ingegneria Elettrica e dell'Informazione, Politecnico di Bari, 70125 Bari, Italy;
m.farooq1@phd.poliba.it (M.F.); gioacchino.manfredi@poliba.it (G.M.); saverio.mascolo@poliba.it (S.M.)
* Correspondence: luca.decicco@poliba.it

Abstract

The growing demand for virtual reality and immersive applications has increased interest in 360° video streaming. When viewing omnidirectional content through a head-mounted display, users observe only a limited portion of the content, i.e., the viewport, at any given time. Consequently, transmitting the complete panoramic frame at uniformly high quality is bandwidth-inefficient. This review presents a system-level analysis of viewport-adaptive three-degree-of-freedom (3DoF) 360° video streaming, focusing on the coupled roles of viewport prediction, tile-based multi-rate encoding and bitrate allocation, transport mechanisms, and edge-assisted processing. The reviewed literature is examined to identify the design dependencies and trade-offs among these components. Viewport-adaptive approaches seek to reduce the bandwidth allocated to regions outside the instantaneous viewport while preserving the quality of the visible region. The analysis shows that their effectiveness cannot be attributed to prediction accuracy alone: the resulting Quality of Experience (QoE) depends jointly on tile granularity, bitrate allocation, buffer occupancy, transport delay, and whether prioritized tiles arrive before their playback deadlines. Finer tiling can improve spatial selectivity but increases coding, signaling, and request overhead. Moreover, HTTP/2, HTTP/3/QUIC, RTP/RTSP, and WebRTC present different reliability, latency, congestion-control, and scalability trade-offs across buffered video-on-demand, low-latency live streaming, and interactive immersive applications. Based on this synthesis, the review formulates a unified closed-loop cross-layer framework that coordinates prediction, tiling, bitrate allocation, request timing, transport configuration, buffering, and edge processing under bandwidth, latency, and resource constraints.

Keywords: Virtual Reality (VR); adaptive 360° video streaming; viewport prediction; Quality of Experience (QoE); transport protocols

Check for updates

Academic Editor: Christos Bouras

Received: 29 May 2026
Revised: 11 August 2026
Accepted: 15 August 2026
Published: 17 August 2026

1. Introduction

VR systems have evolved from experimental platforms to commercially deployed solutions for immersive media delivery. Among immersive-media formats, 360° video is widely used because it is compatible with head-mounted displays (HMDs) and existing video-compression standards. Unlike conventional 2D video, which employs a fixed rectangular frame, 360° video captures an omnidirectional view through an array of cameras and maps it onto a spherical surface around the viewer. The user's head orientation determines the visible viewport within this scene [1]. Consequently, the visible region changes dynamically with the user's viewing direction rather than remaining fixed.

Commercial HMDs provide a horizontal Field-of-View (FoV) of approximately 100–110° and a vertical FoV below 100°. Therefore, users only observe a limited portion of the spherical content at any given time, typically less than 20% of the complete projected frame. Nevertheless, conventional streaming approaches transmit the entire panoramic frame at uniform quality. As a consequence, as shown in Figure 1, a substantial fraction of the transmitted data belong to regions outside the instantaneous viewport when viewport adaptation is not employed [2].

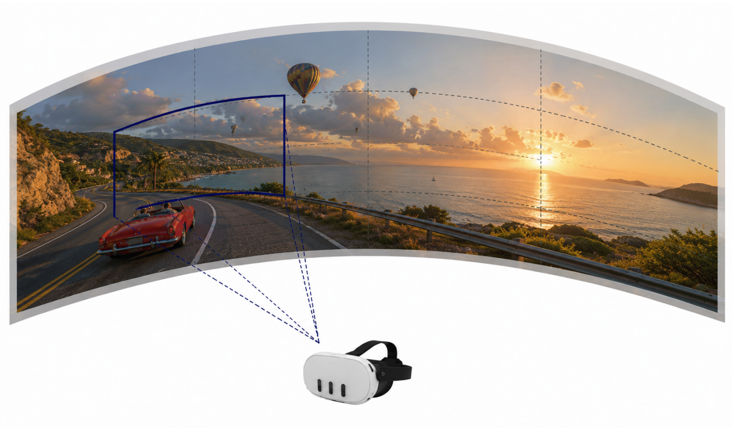

Figure 1. Viewport in tile-based 360° video streaming using ERP projection with 3 × 4 tile grid.

The projection of spherical content onto planar representations introduces additional constraints. Since standard video codecs operate on 2D frames, spherical content must first be mapped to a planar representation, such as *Equirectangular Projection* (ERP) or *Cubemap Projection* (CMP), before encoding. ERP is often used because of its simplicity and compatibility with existing pipelines. However, it produces a non-uniform sampling density and redundant pixel representation near the poles, which can reduce coding efficiency [3]. Moreover, 360° videos must be encoded at high spatial resolution to maintain adequate quality in the viewport, which increases bandwidth demand [4]. These requirements are particularly demanding in bandwidth-limited or mobile environments.

To address the high bandwidth demand of 360° video with the fact that users observe only a limited viewport, recent studies employ tile-based viewport-adaptive streaming [5,6]. The projected frame is partitioned into equal- or unequal-sized tiles using configurations such as 4 × 4 [7] or 16 × 9 [8]. Each tile is encoded at different quality representations and the client selects among them based on the predicted viewport and the available bandwidth. In a typical allocation strategy, tiles within the predicted viewport are requested at high quality, neighboring tiles at intermediate quality, and non-visible tiles at low quality.

The effectiveness of viewport-adaptive streaming depends on viewport-prediction accuracy, tile configuration, bitrate allocation, buffer state, and delivery timing. Although the reviewed studies report improvements in viewport quality, bandwidth utilization, and QoE, their numerical results cannot be compared directly because they are obtained using different baselines, head-motion traces, network conditions, and QoE definitions.

1.1. Motivation and Scope

Viewport-adaptive streaming cannot be evaluated by considering viewport prediction, tiling, bitrate adaptation, and transport independently. Viewport prediction determines which regions should be prioritized, while the tile configuration determines the spatial granularity at which this prioritization can be applied and the associated coding and signaling overhead. Bitrate allocation determines the quality assigned to the selected tiles, whereas transport and network conditions determine whether those tiles arrive before their playback deadlines. Consequently, an improvement in one component does not necessarily improve the overall Quality of Experience (QoE) unless its effects on the other components are also considered [4,9]. Existing reviews generally examine these dimensions separately, whereas their combined effect on end-to-end streaming performance has received limited systematic analysis, as discussed in Section 1.3. Motivated by these dependencies, this review examines viewport-adaptive 360° video streaming as a coupled system-design problem. It analyzes how viewport-prediction uncertainty, tile granularity, bitrate allocation, buffer state, delivery timing, transport behavior, and edge processing jointly affect viewport quality, bandwidth utilization, latency, rebuffering, and QoE. The review focuses on three-degree-of-freedom (3DoF) systems, in which the user's viewing position remains fixed and the visible viewport changes according to head orientation.

Six-degree-of-freedom (6DoF) systems are excluded because translational movement requires depth- or geometry-aware scene representations, novel-view synthesis, and volumetric or multi-view delivery. These requirements introduce acquisition, rendering, coding, and transport problems that differ substantially from the 3DoF streaming systems considered in this review.

1.2. Challenges in Adaptive 360° Video Streaming

The key technical challenges in adaptive 360° video streaming are summarized as follows:

- Viewport prediction errors directly degrade QoE by causing high-quality tiles to be unavailable within the user's actual FoV. Although short-term prediction is generally more reliable, long-term forecasting remains difficult because of unpredictable head movements.
- Fine-grained tiling improves spatial adaptation, while increasing encoding overhead, storage, signaling cost and server-side management. Coarse tiling reduces overhead; however, it limits viewport-level adaptation precision.
- Selecting bitrate levels for viewport and non-viewport tiles requires a trade-off among spatial quality, temporal quality stability, bandwidth constraints and rebuffering risk.
- Projection formats may cause non-uniform distortion of the video frame, which makes saliency estimation and bitrate allocation more difficult.
- Conventional Dynamic Adaptive Streaming over HTTP (DASH) is scalable, yet retransmission delay and head-of-line (HoL) blocking limit latency-sensitive viewport delivery. QUIC can reduce these effects through multiplexing and prioritization, with gains depending on the employed congestion control algorithm, reliability mode and deployment complexity.
- VR systems are highly sensitive to motion-to-photon latency. Increased latency in such systems may result in cybersickness and user discomfort.

1.3. Limitations in Existing Research

Several recent surveys have examined different dimensions of 360° video streaming. For instance, learning-based viewport prediction techniques have been extensively reviewed in [1], where approaches are categorized into user HMD movement data and saliency-driven and hybrid ML/DL frameworks. Similarly, adaptive streaming taxonomies and tile-based delivery mechanisms are discussed in [10,11], while system-level acquisition, transmission, and display pipelines are analyzed in [12]. Another survey focused primarily on coding efficiency, projection formats, and QoE modeling [13].

Although these studies provide specific insights in this domain, several limitations remain.

1. Viewport prediction studies often assess accuracy in isolation [1], with limited analysis of how uncertainty affects tile selection and bitrate allocation across different tiling granularities.

2. Surveys covering adaptive streaming and tiling mechanisms [10,11] categorize viewport-dependent versus viewport-independent schemes, yet they do not systematically analyze the trade-off between tile granularity, encoding overhead, and rate distortion efficiency. Quantitative modeling of bitrate–latency trade-offs and bitrate overhead is generally not considered.
3. Transport-layer implications are not sufficiently addressed. While DASH-based adaptive streaming is frequently mentioned [10], emerging transport protocols such as HTTP/3 and QUIC are not comparatively analyzed in the context of 360° viewport-adaptive delivery. The latency sensitivity of motion-to-photon pipelines is discussed in [12]; however, this is without protocol-level benchmarking.
4. Cross-layer optimization between application-layer viewport prediction and network-layer transport mechanisms is not treated within a unified analytical framework. Surveys tend to focus either on prediction algorithms [1], system architectures [12], or coding/QoE aspects [13], without integrating these components into a unified trade-off model.

Therefore, despite the breadth of the existing literature, there remains a need for a comprehensive and quantitatively grounded review that jointly analyzes viewport prediction, tiling strategies, bitrate adaptation, and transport-layer mechanisms within a unified system-level framework. Table 1 compares the selected review articles and shows that prediction uncertainty, tiling overhead and transport behavior are rarely examined together.

Table 1. Comparison of existing review articles on viewport-driven 360° video streaming.

Ref.	Year	Viewport	Streaming	Transport	Evidence	Main Limitation
[1]	2025	ML/DL prediction taxonomy	Tile prioritization	Not addressed	Prediction accuracy trends	No integrated cross-layer analysis
[14]	2024	AR/360° video streaming	High-level 360° AR streaming	Not addressed	No comparative streaming metrics	Descriptive review; no benchmarking
[15]	2024	Head-movement prediction	VR adaptive streaming models	DASH-level discussion	Qualitative discussion of bandwidth, latency and QoE	No system-level benchmarking
[3]	2023	Brief VP discussion	Adaptive streaming overview	Not addressed	No bitrate–distortion analysis	Limited analytical depth
[11]	2022	Viewport-adaptive categories	DASH, tiling, and ML allocation	5G/6G and MEC	Limited numerical comparison	No unified VP network model
[13]	2021	FoV and saliency prediction	Tiling and QoE trade-offs	Network support briefly reviewed	QoE metrics and coding analysis	Transport delay not integrated
[16]	2020	System constraints	Projection, tiling and streaming	MPEG standards	No comparative system metrics	Broad coverage with limited critical analysis
[10]	2020	Viewport-dependent taxonomy	Tile-based adaptive streaming	DASH and multicast	No bitrate–latency trade-off	Strong taxonomy; limited quantification
[12]	2019	Pipeline-level review	Acquisition, transmission, and display	System-level protocol discussion	No standard benchmarks	Limited focus on prediction-driven adaptation

Note: Existing reviews rarely examine prediction, tiling, and transport within one framework.

1.4. Contributions of This Review

Compared with the existing surveys discussed in Section 1.3 and summarized in Table 1, the original contribution of this review is its unified prediction–tiling–transport perspective on viewport-adaptive 360° video streaming.

The specific contributions are as follows:

- A unified closed-loop analytical framework that relates the observed system state, including viewport-prediction outputs, network conditions, buffer occupancy, playback deadlines, and available edge resources, to joint decisions concerning prediction horizon, tile configuration, bitrate allocation, request timing, transport configuration, and processing location. Section 6 formalizes this state–decision–outcome relationship.
- A structured synthesis of quantitative results reported across individual studies, covering prediction horizon, tile granularity, segment duration, buffer size, delivery latency, viewport quality, bandwidth utilization, and rebuffering risk.
- A transport-aware and scenario-oriented comparison of HTTP/1.1 over TCP, HTTP/2 over TCP, HTTP/3 over QUIC, RTP/RTSP, and WebRTC. The comparison distinguishes buffered video-on-demand, low-latency live streaming, and interactive immersive applications and considers congestion control, stream scheduling, reliability, playback deadlines, scalability, and deployment constraints.
- A unified evaluation perspective that distinguishes viewport-prediction metrics, such as angular error, FoV overlap, and tile accuracy, from streaming outcomes, such as viewport PSNR/VMAF, viewport misses, bandwidth utilization, quality variation, rebuffering, and QoE. This distinction clarifies why improvements in prediction accuracy do not necessarily translate directly into better end-to-end streaming performance.
- Identification of open research challenges concerning uncertainty-aware bitrate allocation, long-horizon viewport prediction, tile-granularity and coding-overhead trade-offs, transport evaluation under volatile network conditions, edge-assisted correction, multi-user viewport modeling, and reproducible cross-layer evaluation.

1.5. Literature Selection Methodology

The database search was conducted in Scopus, IEEE Xplore, and the ACM Digital Library and was completed on 30 April 2026. It covered English-language, peer-reviewed journal and conference papers published from January 2016 to the final search date. The search was applied to title, abstract and keyword fields in Scopus and ACM Digital Library, and to metadata and abstract fields in IEEE Xplore, using the following Boolean expression:

> ("360-degree video" OR "360 degree video" OR "360 video" OR "omnidirectional video") AND ("adaptive streaming" OR "viewport prediction" OR "field-of-view prediction" OR "FoV prediction" OR "head-motion prediction" OR "head movement prediction" OR "saliency prediction" OR "tile-based streaming" OR tiling OR "viewport-adaptive streaming" OR "adaptive bitrate" OR "bitrate allocation" OR "rate adaptation" OR DASH OR "HTTP/2" OR "HTTP/3" OR QUIC OR RTSP OR RTP OR WebRTC).

Papers were included if they addressed adaptive 360° or omnidirectional video streaming and reported quantitative results directly relevant to viewport-adaptive streaming. Conventional 2D streaming, unrelated 6DoF or volumetric delivery, preprint-only publications, prediction or coding studies without a clear connection to viewport-adaptive streaming decisions, and studies without quantitative evaluation were excluded.

Records were exported and screened first by title and abstract and then by full text. Duplicates were identified by DOI, followed by title, author, and publication-year matching and manual checking; where conference and journal versions reported the same study, the more complete peer-reviewed version was retained. The search identified 450 records (Scopus: 300; IEEE Xplore: 100; ACM Digital Library: 50). After removing 50 duplicates, 400 records were screened and 100 were excluded. Of the 300 full-text articles assessed, 220 were excluded as outside the review scope ($n = 80$), lacking streaming-level outcomes ($n = 70$), lacking quantitative evaluation ($n = 50$), or non-English/preprint-only papers ($n = 20$). After completion of the database search, a targeted update was conducted to identify recently accepted peer-reviewed publications and technical standards relevant to the final discussion. These sources were used for contextual updating and were not included in the PRISMA counts. The database-screened evidence base comprised 80 primary studies, as summarized in Figure 2 and reported in Table S1 in the Supplementary Material.

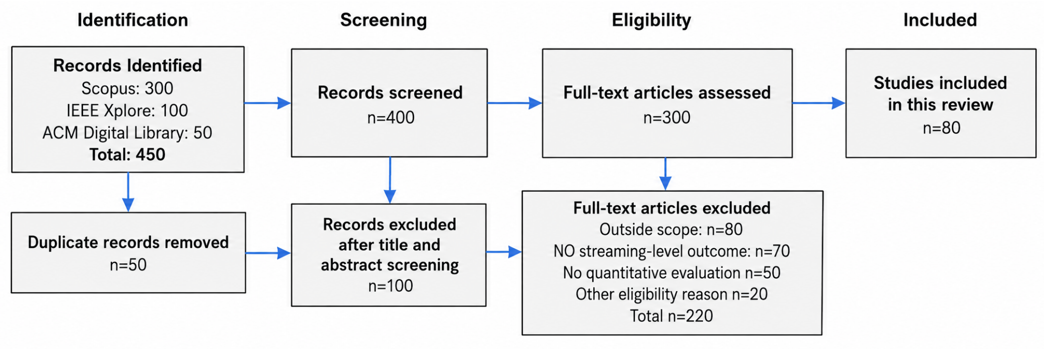

Figure 2. PRISMA-style flow diagram of the literature-selection process.

1.6. Organization

This review is organized as follows: Section 2 introduces the background of 360° video representation, projection formats, viewport modeling, tiling, and QoE assessment metrics. Section 3 analyzes viewport prediction methods including head-motion-based, saliency-based, and hybrid learning approaches. Section 4 reviews tile-based adaptive streaming strategies and bitrate allocation mechanisms. Section 5 discusses transport-protocol trade-offs for low-latency 360° video delivery. Section 6 formulates a unified cross-layer framework for joint prediction, tiling, bitrate, transport, and edge decisions. Section 7 concludes the review and outlines future research directions.

2. Background of Adaptive 360° Video Streaming

This section introduces the spatial, temporal, and delivery abstractions required to analyze viewport-adaptive 360° video streaming. It first describes spherical-video representation, projection, and viewport geometry; next, it examines temporal alignment and the prediction–delivery horizon, followed by tiled and segment-based delivery, bitrate adaptation, and QoE assessment.

2.1. Spherical Representation, Projection, and Viewport Geometry

A 360° video represents an omnidirectional scene surrounding the viewer. Camera-captured content is typically acquired using omnidirectional or multi-lens camera systems and stitched into a spherical representation, whereas computer-generated content can be rendered directly in this form. For geometric analysis, the visual content is modeled on the inner surface of a unit sphere centered at the viewer's position. The viewer's orientation at time t is represented in spherical coordinates as

$$(\theta(t), \phi(t)), \tag{1}$$

where $\theta(t)$ represents the yaw, or azimuth, angle and $\phi(t)$ represents the pitch, or elevation, angle. Because conventional video codecs operate on rectangular two-dimensional frames, spherical content must be projected onto a planar domain before encoding. This transformation introduces geometric distortion that affects spatial sampling density, compression efficiency, and subsequent tile partitioning. Consequently, the projection format should be treated as a rate–distortion and tiling-design variable rather than as a neutral preprocessing choice [17].

Multiple projection formats have been proposed for representing spherical video on a planar domain. ERP and CMP are first considered as the principal baseline formats because they are widely used in viewport-adaptive streaming and tile-based evaluation studies. Distortion-aware alternatives are then examined in terms of their mapping operations and principal computational overheads. ERP provides a simple rectangular representation that is compatible with existing codec and DASH workflows, although this compatibility is obtained at the cost of non-uniform spherical sampling [18,19].

Let longitude $\lambda \in [-\pi, \pi]$ and latitude $\phi \in [-\pi/2, \pi/2]$. For a projected frame of width W and height H, ERP performs the following mapping:

$$\begin{aligned} x &= W\frac{\lambda + \pi}{2\pi} \\ y &= H\frac{\phi + \frac{\pi}{2}}{\pi}. \end{aligned} \tag{2}$$

Here, W and H denote the projected frame width and height, respectively. Although ERP is computationally simple, it introduces substantial area distortion near the poles. Sampling density increases as latitude approaches $\pm$ 90°, leading to oversampling near the poles. When user viewing trajectories are concentrated around equatorial regions, this oversampling reduces rate-distortion efficiency because bits are spent on regions with lower viewing probability. To address this limitation, region-based rate-distortion modeling strategies have been proposed to better account for the non-uniform spatial characteristics of ERP content [20,21]. Figure 3 summarizes the common projection formats used to map spherical 360° video content onto 2D frames.

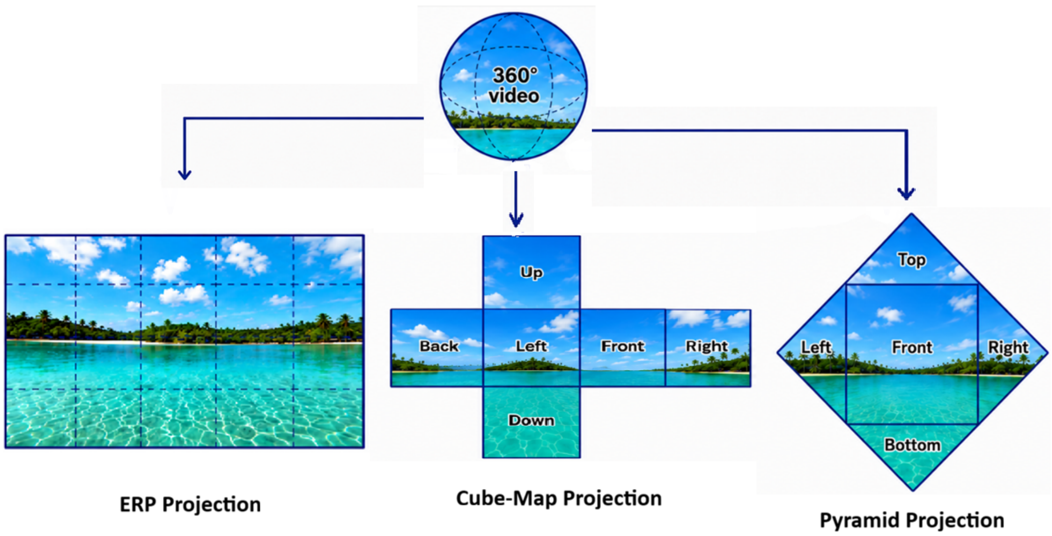

Figure 3. Common projection formats for mapping spherical 360° video to planar frames.

CMP maps the spherical video onto six square faces of a cube and offers a more uniform spatial sampling density than ERP. This property can improve the spatial correspondence between projected content and viewport-driven tile selection, particularly when tile boundaries align well with the visible region and support more efficient bitrate allocation [22,23]. However, these benefits are not unconditional. CMP introduces geometric discontinuities at cube-face boundaries, which can complicate inter-face motion estimation, reduce coding efficiency, and increase the complexity of tile stitching during viewport reconstruction. In practice, the extent of bitrate reduction depends on the projection format, tile granularity, viewport-prediction accuracy, and the QoE-aware rate-allocation policy [20].

Recent studies have investigated distortion-aware projection formats designed to improve spherical sampling uniformity or coding efficiency [24–27]. Compared with ERP and conventional CMP, these formats introduce additional operations such as nonlinear coordinate transformations, interpolation, face selection, face warping, or orientation-dependent reprojection [24,26–28]. Because the available studies do not report computational performance under standardized implementations and hardware configurations, Table 2 provides a qualitative comparison of their principal processing overheads.

Table 2. Qualitative comparison of distortion-aware projection formats and their principal computational overheads.

Ref.	Year	Map	Distortion Strategy	Core Operations	Main Computational Overhead	Findings
[24,26–28]	2018, 2021, 2023, 2024	EAC	Angular uniformity	Face selection; angular remapping; interpolation	Nonlinear six-face remapping	Low distortion; high WS-PSNR
[24,27,28]	2018, 2021, 2024	ACP	Sampling uniformity	Face warping; inverse mapping; interpolation	Nonlinear six-face processing	High coding efficiency
[25,27,28]	2018, 2021, 2024	ECP	Latitude balancing	Cylindrical remapping; interpolation	Nonlinear pixel remapping	High coding efficiency
[24,27,28]	2018, 2021, 2024	AEP	Area preservation	Equal-area transform; interpolation	Nonlinear pixel remapping	Low area distortion
[24,25,27]	2021, 2024	HEC	Hybrid sampling	Cube–cylinder conversion; face packing	Multiple projection stages	Improved coding efficiency
[25,27]	2021, 2024	RSP	View alignment	Sphere rotation; coordinate remapping	Rotation and reprojection	View-oriented sampling

Note: Standardized runtime and memory results are not reported consistently across the cited studies.

Table 2 shows that ECP and AEP avoid polyhedral face handling; however, they require nonlinear pixel-wise remapping. EAC and ACP add six-face processing. In addition, RSP adds orientation-dependent rotation and reprojection [27,28]. Thus, practical complexity varies mainly through nonlinear transforms, interpolation, face handling and dynamic reprojection.

The viewport is the spherical region visible through the HMD at time t and determines which tiles should be prioritized for timely transmission and decoding [29].

Let θ_h and θ_v denote the horizontal and vertical FoV in radians, respectively, and let ϕ_c denote the latitude of the viewport center. To derive an analytical approximation of the visible fraction of the spherical viewing domain, the immersive content is modeled on a unit sphere and the FoV is represented as an ERP-aligned spherical rectangle bounded by constant longitude and latitude. This representation is a simplified geometric approximation of the visible viewport. Lens distortion, display cropping, projection-dependent boundary curvature, pole crossing, and longitude wrap-around are neglected. The approximation is therefore restricted to viewports that do not intersect the poles, i.e.,

$$-\frac{\pi}{2} \leq \phi_c - \frac{\theta_v}{2} < \phi_c + \frac{\theta_v}{2} \leq \frac{\pi}{2}.$$

For longitude limits $\lambda_c \pm \theta_h/2$ and latitude limits $\phi_c \pm \theta_v/2$, integration of the unit-sphere surface element $\mathrm{d}A = \cos\phi \, \mathrm{d}\phi \, \mathrm{d}\lambda$ over the approximated viewport and normalization by the total sphere area 4π gives

$$\alpha = \frac{\theta_h \left[\sin\left(\phi_c + \frac{\theta_v}{2}\right) - \sin\left(\phi_c - \frac{\theta_v}{2}\right) \right]}{4\pi}, \tag{3}$$

For an equator-centered viewport, $\phi_c = 0$, with $\theta_h = 100^\circ$ and $\theta_v = 90^\circ$, this gives

$$\alpha \approx 0.196, \tag{4}$$

which corresponds to approximately 19.6% of the spherical surface.

2.2. Temporal Alignment and Prediction Horizon

The temporal sequence of orientation samples forms the head-motion trajectory used for viewport prediction and tile-priority estimation. In commercial HMDs, head orientation is commonly sampled at 90 Hz or higher, whereas 360° video is often encoded at 30–60 fps. The tracking rate, video frame rate, keyframe interval, and segment duration therefore operate on different time scales. Viewport observations must consequently be aligned with the encoding and delivery boundaries at which tile-selection and bitrate decisions can be applied [30].

Let Δt_f denote the video-frame interval and Δt_s the head-motion sampling interval. For video encoded at 60 fps, Δt_f is approximately 16.7 ms, whereas head tracking at 90 Hz gives $\Delta t_s \approx 11.1$ ms. The two sampling grids therefore do not generally coincide. Shortening the keyframe interval can improve the responsiveness of viewport-dependent representation switching; however, it may also increase encoding overhead and bitrate requirements [30].

Streaming decisions rely on head-motion observations aligned with frame, keyframe, or segment boundaries. A mismatch between the predicted and actual FoVs can assign high quality to tiles outside the viewport while leaving actually viewed tiles at insufficient quality. The resulting effect on QoE depends on whether the required tiles are already available or can be delivered before their playback deadlines [29,30]. The prediction horizon T_p must therefore be selected in relation to the tile-delivery and processing lead time. A useful feasibility condition is

$$T_p \gtrsim T_{\mathrm{request}} + T_{\mathrm{download}} + T_{\mathrm{decode}} + T_{\mathrm{margin}}. \tag{5}$$

Here, T_{request} includes request-generation and RTT-related delay, T_{download} is the tile-download time, T_{decode} includes decoding and stitching delay, and T_{margin} is an additional playback-safety margin. The inequality expresses a delivery-feasibility requirement: if T_p is shorter than the combined delivery and processing lead time, the prediction may use more recent viewing information, but the selected tiles may not arrive before playback. Increasing T_p provides more time for prefetching and delivery, but it also increases the probability that the predicted and actual FoVs diverge. The prediction horizon must therefore balance prediction reliability against the time required for tile delivery and processing.

2.3. Tiled and Segment-Based Delivery

Viewport-aware streaming requires spatially selective delivery. Towards this end, the projected frame is partitioned into spatial tiles that can be represented, packaged, and requested separately. The degree of coding independence among tiles depends on the codec configuration and packaging format. Tile-based FoV-adaptive streaming is used when the system requires spatially differentiated bitrate allocation within the projected frame [31–33].

If R is the total number of frame pixels and N is the number of tiles, then, under uniform partitioning and neglecting boundary padding, the number of pixels per tile is approximated as

$$R_{\text{tile}} \approx \frac{R}{N}. \tag{6}$$

Encoding tiles independently, or constraining coding dependencies to allow separate tile extraction, can reduce motion-compensation efficiency and increase bitrate overhead [29,34]. Finer grids improve viewport matching by reducing the spatial area covered by each tile, but they also increase the number of tile representations that must be encoded, stored, selected, and requested. Tile grouping and clustering can help balance this trade-off, particularly in multi-user scenarios [31,32]. Layered delivery provides another alternative by transmitting enhancement layers only for tiles associated with the predicted or buffered viewport [29,33]. Table 3 organizes representative spatial-adaptation strategies according to the design variable they control, while Section 4.3 examines the tiling-granularity and coding-overhead trade-off in detail.

Table 3. Comparison of spatial adaptation strategies in 360° video streaming.

Ref.	Adaptation Domain	Spatial Control	Evaluation Focus	Reported Gain	Main Limitation
[35]	Projection and FoV	ERP/Cubemap weighting	Viewport quality robustness	Lower degradation under unstable wireless links	No tile-level request control
[36]	Tile bitrate	Fixed tiles with bitrate versions	Viewport rate allocation	Rate-quality-based tile selection	Fixed boundaries limit FoV matching
[37]	Projection switching	RSP/EAC switching	Bitrate–distortion trade-off	26% bitrate saving	Projection switching overhead
[38]	Prediction-based tiling	Fixed tiles with MPC	QoE under FoV uncertainty	16% QoE improvement	Prediction error and MPC complexity
[39]	Projection control	TSP truncation control	Projection-quality trade-off	Up to 1.1 dB PSNR gain	No tile-level delivery control
[40]	Edge-assisted tiling	Tile/QER version selection	MEC-assisted latency control	Improved delay guarantee	Cache and edge-resource dependent
[41]	Adaptive tiling	Optimized tile segmentation	QoE-resource trade-off	Improved segmentation-download balance	Higher tiling-management complexity
[42]	Projection control	Orientation-aware offset projection	View-aligned quality	Improved dominant-view quality	Less robust to random head motion
[43]	Tile-size optimization	Tile size with QP control	Rate–distortion trade-off	Improved tile-size/QP allocation	Higher encoding cost from tile versions
[2]	Prediction-based tiling	Pyramid bitrate allocation	QoE and bandwidth saving	30% QoE improvement	Sensitive to abrupt attention shifts
[44]	Probability-based tiling	Visibility-probability classes	Expected distortion reduction	Improved server-side tile optimization	Limited by visibility prediction accuracy
[5]	RL-based tiling	Three tile granularities	Long-term QoE control	Improved QoE through DRL	Limited generalization beyond training
[45]	Dynamic tiling	Chunk-level spherical tiles	Redundancy–QoE trade-off	14% QoE improvement	Higher chunk-level tiling complexity

The comparison in Table 3 shows that fixed-grid tiling is simpler to encode, package, and manage, whereas adaptive layouts can improve viewport alignment at the cost of greater encoding, storage, signaling, and management complexity.

Spatial tiling is combined with temporal segmentation. In the reviewed MPEG-DASH-based implementations, 360° video is commonly divided into segments with durations on the order of 1–4 s [46,47]. Tile priority alone does not determine the selected representation level: the ABR logic must also decide how much bitrate to allocate to viewport, neighboring, and non-viewport tiles under the current throughput and buffer constraints. Shorter video segments allow tile and bitrate selections to be updated more frequently, but they increase request, signaling, and packaging overhead. Longer segments reduce request frequency and may improve compression efficiency, but they also increase adaptation latency and require viewport predictions over a longer horizon [46].

https://doi.org/10.3390/network6030066

Spatial Relationship Description (SRD) metadata identify the positions and dimensions of spatial regions within the projected source representation. They allow a client to associate requested tiles with their positions in the projected frame and to reconstruct the spatial layout required for decoding and rendering [32,47]. Tile granularity and segment duration define the spatial and temporal resolution at which the ABR controller can respond to changes in viewport orientation, throughput, and buffer state.

2.4. Bitrate Adaptation, QoE, and End-to-End Pipeline

In viewport-adaptive systems, the ABR controller selects tile representations using information such as estimated throughput, buffer occupancy, viewport-importance weights, and prediction confidence. Buffer-aware bitrate adaptation has been formulated using Lyapunov optimization and prediction-assisted control to maintain QoE under bandwidth fluctuations [31,46]. Similarly, tile-based viewport-adaptive streaming can be formulated as a QoE-maximization problem subject to bandwidth and buffer constraints [48]. A commonly used QoE structure combines viewport quality, stall duration, and quality variation:

$$\mathrm{QoE} = w_Q Q_v - w_S S - w_V V, \tag{7}$$

where Q_v denotes viewport quality, S denotes stall duration, and V denotes quality variation. The non-negative weights w_Q, w_S, and w_V determine the relative importance assigned to these components. The definition of Q_v varies across studies and may be based on viewport PSNR, weighted tile bitrate, VMAF, or subjective quality scores. Similarly, the relative importance of stalls and quality variation depends on the application scenario: stall penalties are often dominant in low-buffer live streaming, whereas quality variation may receive greater weight in buffered VoD. QoE results must therefore be interpreted together with the adopted quality metric, segment duration, buffer model, network conditions, and viewport-prediction assumptions.

Whether the selected tile representations arrive before the playback deadline depends jointly on application-layer bitrate allocation, transport behavior, path variability, and edge-processing delay.

PRISM-XR [46], for example, reports different QUIC and TCP congestion dynamics under mmWave channel fluctuations. This result does not establish that either transport is universally preferable; performance depends on loss patterns, congestion-control behavior, stream scheduling, and wireless-channel variability. Multipath delivery [47] and edge-assisted processing [49] introduce additional opportunities to reduce delivery or processing delay, but their benefit depends on path coordination, edge-resource availability, and the remaining playback-deadline margin. Edge-assisted processing is examined in Section 4.6, while transport behavior is analyzed in Section 5.

Figure 4 summarizes the end-to-end pipeline of a viewport-adaptive 360° video streaming system. On the server side, the omnidirectional content is projected into a planar representation such as ERP or CMP, partitioned into tiles, and encoded in multiple quality representations. The encoded representations are stored on the content server, while the Media Presentation Description (MPD) identifies their locations, properties, and temporal and spatial relationships. On the client side, viewport, throughput, and buffer information guide tile and representation selection. The requested content is then downloaded, decoded, spatially composed, and rendered for display.

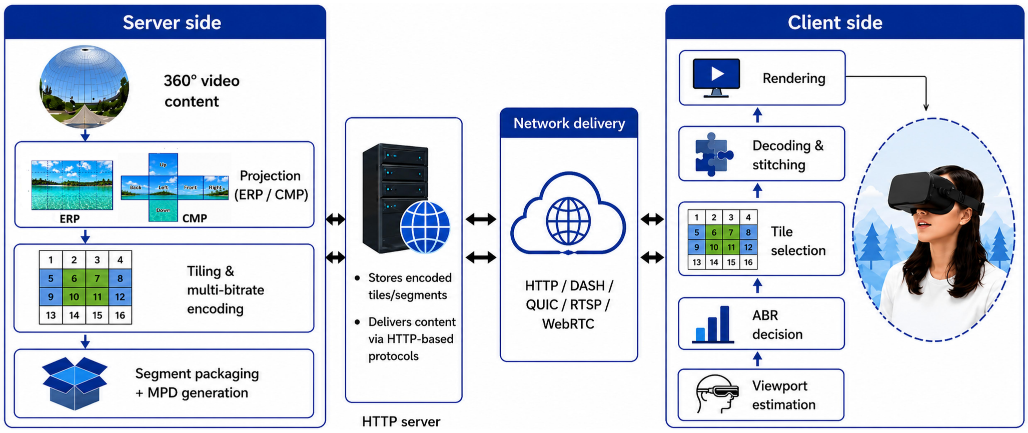

Figure 4. End-to-end architecture of a viewport-adaptive 360° video streaming system.

3. Viewport Prediction in 360° Video Streaming

Viewport prediction estimates a user's future FoV over a prediction horizon T_p, expressed in seconds. The prediction output may be represented using spherical coordinates (θ, ϕ), Cartesian orientation, or tile indices.

In adaptive 360° video streaming, the predicted FoV indicates which tiles should be prioritized for high-quality delivery and which can be delivered at lower quality. Therefore, viewport prediction operates both as an estimation task and as an input to bitrate-allocation, prefetching, and tile-scheduling decisions within a bandwidth-constrained and latency-sensitive streaming system.

Viewport prediction should be evaluated using two complementary categories of metrics. The first one measures *prediction accuracy*, including angular error, FoV overlap, tile-level accuracy, and viewport hit rate. The second one measures the resulting *streaming performance*, including viewport miss ratio, viewport PSNR/VMAF, bandwidth savings, stall duration, and quality variation. This distinction is important because a reduction in angular error does not necessarily improve QoE if the selected tiles arrive after the playback deadline or if the tiling grid is too coarse to exploit the improved prediction.

The methodological evolution of viewport prediction can be organized into five broad and partially overlapping families: (i) statistical and trajectory-extrapolation models; (ii) recurrent DL models for temporal sequence learning; (iii) content-aware and hybrid models combining head motion with visual information; (iv) attention-, transformer-, and graph-based models for longer prediction horizons and cross-user dependencies; and (v) probabilistic and deployment-aware models addressing prediction uncertainty and computational constraints.

3.1. Statistical and Trajectory-Based Prediction

The earliest methodological family relies on trajectory extrapolation using linear regression or velocity-based heuristics, and such approaches continue to appear in recent systems because of their low computational cost.

The 360EAVP player, for example, integrates linear and ridge regression for viewport prediction in dynamic editing scenarios [50]. Trajectory-based long-horizon prediction using clustered pitch, yaw and roll functions has been reported to improve predicted viewport area by 13% for horizons up to 10 s [51]. Nevertheless, broader evaluations show that linear regression performs only marginally better than reusing the last viewport position [52]. In addition, statistical methods degrade strongly beyond a 1 s prediction horizon [52]. These results indicate that purely kinematic models have limited ability to capture abrupt or content-driven head movements, particularly at medium and long prediction horizons.

3.2. Recurrent DL Prediction

To address these limitations, recurrent neural networks such as LSTM- and GRU-based models, as well as encoder–decoder architectures, were introduced. The GLVP model reports a 9.5–20% improvement compared with trajectory-based baselines under the evaluated prediction horizons, showing that recurrent modeling improves temporal dependency learning; however, it does not eliminate long-horizon uncertainty [53]. Attention-based encoder-decoder models further improve prediction performance while also converging faster than earlier recurrent approaches [54]. Comparative studies indicate that LSTM provides a favorable trade-off between prediction accuracy and redundancy among classical learning-based models [52].

When integrated with live-streaming ABR, LiveCL reports an approximately 50% bandwidth reduction relative to non-viewport-adaptive delivery [55]. This is an end-to-end streaming result that depends on prediction performance, segment timing, and server-side computational capacity.

3.3. Content-Aware and Hybrid Prediction

Trajectory-based models depend only on historical head motion and therefore neglect scene semantics, making them less reliable when viewport shifts are driven by content. Content-aware approaches address this limitation by incorporating saliency and motion features extracted from video frames. For example, the spherical convolution-based method in [56] improves distortion invariance for ERP-projected content, enabling more consistent saliency estimation over the spherical domain. Hybrid approaches combine trajectory information with content-derived features to improve robustness while attempting to preserve real-time feasibility. For live 360° video streaming, the study in [57] reports that joint modeling of user trajectory and content motion improves viewport prediction with limited additional latency under its evaluated conditions.

3.4. Attention, Transformer, and Graph-Based Prediction

More recently, the work in [7] proposed a cross-modal multiscale transformer that jointly processes saliency and trajectory features and reports improved stability for longer prediction horizons. Despite these advances, content-aware and transformer-based models introduce additional feature-extraction and inference costs, which may increase latency and limit deployment on resource-constrained devices [7].

Recent work has further explored cross-user and multi-modal fusion to improve robustness. The GNN-based method in [58] integrates tile-level features derived from user behavior, video content, and cross-user traces, enabling the model to capture both spatial and behavioral dependencies. Although these models capture richer multimodal and cross-user dependencies, they mainly produce deterministic predictions and require greater computational resources. These limitations motivated probabilistic and deployment-aware approaches.

3.5. Probabilistic and Deployment-Aware Prediction

Probabilistic methods represent multiple possible future viewports instead of producing only a single deterministic estimate. The multi-window stochastic method in [59] predicts viewport distributions over short time horizons and integrates these predictions with model-predictive adaptation. This formulation allows streaming decisions to account for uncertainty across different prediction windows.

Furthermore, personalized approaches account for differences among users. The federated-learning framework in [60] constructs user profiles in a distributed manner and reports a high average prediction accuracy while avoiding the direct centralization of

https://doi.org/10.3390/network6030066

users' raw data. For longer prediction horizons, eye-gaze heatmaps can be combined with cross-user trajectory patterns to improve temporal stability [61]. Although this approach strengthens medium- and long-horizon prediction, it also increases system complexity and introduces scalability and privacy concerns.

The deployment of viewport prediction reflects a fundamental trade-off among latency, computational capacity, and scalability. Client-side methods such as MFVP [62] prioritize low delay and limited signaling by constraining model complexity to mobile hardware, which limits long-horizon prediction capability. By contrast, server-side frameworks, including the hybrid architecture [57], exploit centralized computation to support deeper multi-modal fusion, at the cost of additional signaling latency and reduced scalability with large user populations.

Edge-assisted hierarchical offloading [63] provides an intermediate design by dynamically splitting prediction tasks between client and edge resources. This reduces bandwidth usage and inference delay relative to fully centralized solutions, but it depends on edge availability and incurs coordination overhead. Deployment architecture therefore directly affects the achievable prediction horizon, model complexity, and system responsiveness. Because the reviewed viewport-prediction studies use different datasets, prediction horizons, viewport definitions, baseline models, and evaluation metrics, their reported results are not directly comparable. Thus, Table 4 reports the original metric and results obtained by each study. For bounded accuracy-like metrics, it also reports the Relative Error Reduction (RER) with respect to the baseline evaluated under the same experimental conditions. RER quantifies the within-study reduction in prediction error; it does not make results from different studies directly comparable and must not be interpreted as a cross-study ranking.

Table 4. Study-reported results for viewport-prediction models and within-study relative error reduction.

Ref.	Model	Horizon	Metric	Proposed Result	Baseline Result	Relative Error Reduction (RER)
[51]	Trajectory clustering	Up to 10 s	Viewport area	13% reported gain	Trajectory heuristics	N/A
[53]	GLVP GRU–LSTM	First 6 s	Accuracy	94.28%	GRU: 85.33%	61.0%
[64]	C-GhostNet + GRU-ECA	2 s	Accuracy	94.40%	Motion: 92.59%	24.4%
[7]	CMMST	1 s	Tile accuracy	68.0%	PanoSalNet: 31.8%	53.1%
[58]	Multi-feature GNN	1 s	Accuracy	84.6%	LiveDeep: 73.5%	41.9%
[59]	Multi-window stochastic	1–3 s	Accuracy	64.7%	Prob-TBAS: 51.4%	27.4%
[61]	Gaze-based SE-Unet	1–10 s	Hit rate	72.10%	SM: 68.19%	12.3%
[62]	MFVP	1 s	Accuracy	91–96%	MobileNetV2: 74–83%	65.4–76.5%

Note: For bounded accuracy-like metrics, relative error reduction is calculated as RER $= (P - B)/(1 - B) \times 100$, where P and B denote the proposed and baseline results, respectively, expressed as proportions in $[0, 1]$. For Ref. [51], only a study-reported relative gain in viewport area is available; therefore, RER cannot be calculated. The MFVP [62] range is obtained from the corresponding proposed–baseline pairs reported under different experimental conditions. Because the studies use different datasets, metrics, prediction horizons, viewport definitions, and baselines, RER values across rows are not directly comparable and do not imply a universal ranking.

Table 5 examines viewport-prediction systems from a deployment perspective, focusing on latency, computational complexity, real-time capability, and scalability. Lightweight client-side models can reduce signaling and inference latency but are constrained by the computational capacity of the HMD or mobile device. Server- and edge-assisted approaches can support more complex models, although they introduce communication, coordination, and infrastructure costs that may increase with the number of users. Multi-window stochastic viewport prediction further incorporates uncertainty modeling into MPC-based bitrate allocation and reports QoE improvements of 16.75–18.91% under the evaluated conditions [59]. Such methods allow bitrate-allocation decisions to account explicitly for multiple possible future viewports, although their online optimization and uncertainty modeling introduce additional computational cost.

Table 5. System-level role of viewport prediction in adaptive 360° video streaming deployment.

Ref.	Deployment Layer	Target	Role	Suitability	Constraint	Trade-Off
[65]	Client	Near-future FoV	Tile request	Local low-latency prediction	HMD/mobile budget	Suitable for short-term use; limited under abrupt motion.
[66]	Client	Short-/medium-term FoV	Tile selection	Multiscale motion capture	Inference deadline	Useful only when inference finishes before tile request.
[67]	Server	Saliency map	Tile ranking	Lower client computation	Server scalability	Supports lightweight clients; server load grows with users.
[68]	Edge	Viewport	Prediction and offloading	Shorter delivery path	Queuing delay	Effective only when edge delay remains low.
[69]	Server	ROI/viewport	Viewport-aware encoding	Content-aware bitrate allocation	3D-CNN complexity	Improves content sensitivity; costly for live use.
[70]	Hybrid	Attention/gaze	QoE-driven delivery	Better attention modeling	Feature synchronization	Improves targeting, requires synchronized sensing.
[6]	Server/Client	Long-horizon FoV	Assisted prefetching	Richer temporal and context features	Transformer complexity	Improves long-range modeling; costly for HMD execution.
[71]	Edge	Future FoV	Predictive prefetching	Earlier tile preparation	Horizon uncertainty	Supports prefetching; errors increase with prediction horizon.
[72]	Client	Head sequence	Real-time request	Lightweight execution	Short look-ahead	Practical for HMDs, mainly for short horizons.
[73]	Client	Bitrate + viewport	RL-based ABR	Joint quality–buffer control	Generalization	Couples ABR and viewport prediction; generalization remains limited.
[74]	Client	Tile utility	Online ABR	Low-complexity selection	Simplified assumptions	Simple deployment; less content-adaptive.
[75]	Edge	Tile geometry	Edge tile generation	Less non-visible delivery	Generation delay	Reduces spatial waste, increases coordination delay.
[76]	Server	User cluster	Popularity adaptation	Shared-content benefit	Trace availability	Effective with repeated views; limited for sparse content.
[77]	Hybrid	Prediction + network	End-to-end control	Cross-layer optimization	Benchmark complexity	Strong formulation; difficult to compare fairly across baselines.

Note: Deployment suitability depends on the model complexity, prediction horizon, communication delay, device capacity, and availability of edge or server resources.

Viewport prediction does not operate alone. Live viewport-driven ABR optimization frameworks show 50% bandwidth reduction when prediction is tightly integrated with adaptive bitrate control [55].

Fine-grained tiling can localize the spatial effect of a prediction error and reduce the amount of unnecessarily transmitted high-quality content; however, it also requires more precise tile selection and introduces additional encoding, signaling, and request overhead. Longer prefetch horizons increase prediction uncertainty and may cause a larger mismatch between prefetched tiles and the viewport observed at playback time. Transport latency and offloading delay further influence the effective QoE. Prediction accuracy can improve QoE only when bitrate, buffer and transport conditions allow the selected tiles to be delivered before their playback deadlines. This qualitative cross-layer dependency can be expressed as

$$\text{QoE} = f(\epsilon, G, R, B, T), \tag{8}$$

where ϵ denotes the viewport-prediction error, G the tile granularity, R the per-tile bitrate allocation, B the playback-buffer state, and T the end-to-end tile-delivery latency. This expression represents a qualitative dependency, since the functional relationship and relative importance of these variables depend on the streaming architecture and the adopted QoE definition.

Because the reviewed studies use heterogeneous datasets, prediction horizons, viewport definitions, and evaluation metrics, the available evidence does not identify a single prediction model that consistently outperforms the others. Head-motion models are computationally lightweight and suitable for short-horizon client-side prediction, but their performance can degrade under abrupt or content-driven viewport shifts [52,62]. Saliency-based and hybrid models incorporate visual information and can better capture content-driven viewing behavior, although feature extraction increases computational cost and deployment complexity [7,56,57]. Probabilistic and MPC-coupled methods explicitly propagate viewport uncertainty into bitrate-allocation decisions [58], but their benefits must be evaluated jointly with buffering, transport latency, congestion control, and tile-request scheduling. Thus, future benchmarks should report prediction accuracy together with viewport miss ratio, delivered viewport quality, bandwidth savings, stall duration, and inference latency under clearly specified prediction horizons, tiling configurations, and network conditions [55,59,77].

4. Adaptive Streaming Strategies

Adaptive streaming in 360° video systems is used to improve QoE under bandwidth, latency and viewport estimation constraints. Compared with conventional streaming, adaptive strategies provide more flexibility in tile selection, bitrate allocation and buffering. This section examines the evolution of these approaches, from non-adaptive full-frame delivery to tiling-based, prediction-aware and system-level optimized schemes. Table 6 summarizes representative tile-based methods and highlights the use of HTTP-DASH and viewport-aware bitrate adaptation for improving visual quality and bandwidth efficiency. The comparison also shows that performance of immersive streaming systems remains highly dependent on viewport prediction accuracy, bandwidth estimation and implementation strategy.

Table 6. Comparison of tile-based adaptive 360° video streaming approaches.

Ref.	Tile Layout	Viewport Model	Key Findings	Transport	Constraints
[78]	Fixed ERP grid	Probabilistic viewport model	Lower quality variation; higher viewport PSNR	HTTP-DASH	Sensitive to prediction errors
[79]	Fixed grid	Crowdsourced FoV probability	Higher QoE under viewport uncertainty	HTTP-DASH	Requires large viewing dataset
[36]	Multi-version tiles	Viewport heuristic	Higher viewport visual quality	HTTP-DASH	Depends on bitrate estimation
[80]	Adaptive tiles	Viewport prediction model	Higher PSNR than fixed tiling	HTTP-DASH	Higher storage overhead
[81]	Viewport and margin tiles	Viewport-overlap weighting	Better FoV bitrate allocation	HTTP-DASH	Heuristic weight design
[82]	Adaptive tiles	Prediction-reliability model	Higher QoE in mobile VR	HTTP-DASH	Higher computational cost
[44]	Visibility-based tiles	CNN viewport prediction	Higher multi-user QoE under bandwidth limits	HTTP-DASH	Complex server optimization
[83]	ERP tiled layout	Viewport-distance heuristic	Higher viewport quality than non-tiled streaming	HTTP-DASH/HTTP/2	Heuristic prioritization
[84]	Cubemap tiling	Viewport-overlap estimation	Higher viewport coverage efficiency	HTTP-DASH	Projection conversion overhead
[4]	Tile layers with SVC	Tile-priority model	Fewer stalls during bandwidth drops	HTTP/2	Requires SVC support
[85]	Variable-size tiles	Viewport-partition model	Higher bandwidth efficiency and QoE	HTTP-DASH	Complex encoding management

Takeaway: Viewport-adaptive tiling improves bandwidth allocation and viewport quality, but its benefit remains dependent on prediction accuracy and is obtained at the cost of additional encoding, storage, or computational overhead.

4.1. Non-Tiled and Full-Frame Streaming Approaches

Full-frame approaches transmit the complete panoramic video at uniform quality and avoid viewport-prediction complexity [86,87]. However, 360° videos require four to six times more data than conventional videos at comparable perceptual quality [88,89], increasing rebuffering risk and limiting scalability in mobile networks at 6K–8K resolutions.

4.2. Fixed Grid Tiling Strategies

In tile-based adaptive streaming, the ERP-projected frame is partitioned into spatially independent tiles and encoded at multiple bitrate levels [78,83]. A representative example is provided by 360ProbDASH, which combines 360° video tiling, multi-bitrate encoding, rate adaptation and viewport-aware delivery [78]. This design enables higher bitrate allocation to tiles within the predicted user viewport, while lower bitrates are assigned to neighboring or non-viewport regions [78,83].

However, fixed-grid tiling systems exhibit several limitations. The first is high sensitivity to viewport prediction errors. Because tiles must be requested sufficiently early to meet their segment playback deadlines, tile selection commonly relies on a viewport predicted over a horizon determined by the segment duration, buffer state, and delivery delay. Prediction inaccuracies can therefore result in missing regions or visible quality degradation. This limitation is emphasized in [79], where a robust optimization framework shows that naive tile selection can substantially reduce QoE under head-movement uncertainty. Their stochastic optimization method improves QoE by at least 30% and achieves up to 50% higher video bitrate under high prediction error [79].

A second limitation is that fixed rectangular tiling does not account for projection-induced distortion. In ERP-based representations, geometric distortion increases toward the poles, creating an imbalance in the distribution of visual information across tiles [13,90]. Although fixed-grid tiles may have equal dimensions in the projected frame, their encoded sizes and rate–distortion characteristics can differ substantially because of content complexity and projection-induced sampling distortion [90]. Their size-aware tile allocation method improves perceptual quality, measured by V-VMAF, by up to 39% compared with state-of-the-art bitrate adaptation schemes [90].

Figure 5 contrasts full-frame and tiled ERP delivery. The effect of tile granularity on viewport precision and encoding overhead is examined quantitatively in Section 4.3 [91,92].

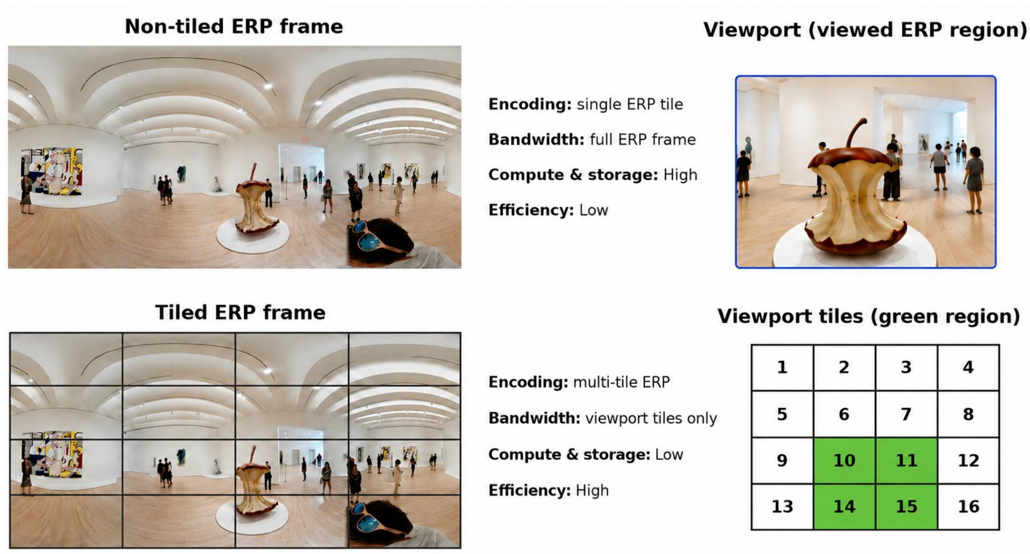

Figure 5. Comparison of non-tiled and tiled ERP 360° video frames for viewport-based delivery.

4.3. Variable and Adaptive Tiling Schemes

Fixed-grid tiling achieves spatial selectivity but introduces coding overhead because motion prediction and other coding operations are constrained at independently encoded tile boundaries. Increasing the number of tiles generally improves the ability to approximate the viewport and to allocate bitrate selectively, but it also increases encoding, storage, signaling, and request overhead. Variable and adaptive tiling schemes attempt to shift this operating point by selecting tile sizes and layouts according to viewport geometry, content characteristics, or expected user behavior.

The TRAS scheme [85] uses convex decomposition to reduce the number of independently encoded tiles while preserving viewport-oriented spatial control. Under the experimental conditions considered in that study, TRAS reports an approximately 18% reduction in encoding overhead and an improvement of about 5 dB in PSNR. By comparison, the tile-size-aware allocation method proposed in [90] allocates bitrate among unequal-size tiles and reports an improvement of up to 39% in V-VMAF relative to the evaluated state-of-the-art bitrate-allocation schemes. Other adaptive approaches jointly optimize tile layout, viewport prediction, and bitrate allocation to reduce the transmission of regions that are unlikely to be viewed [92–94].

These quantitative findings cannot be combined into a common cross-study comparison because the reviewed studies use different datasets, codecs, projection formats, tile layouts, viewport definitions, prediction assumptions, network conditions, baselines, and evaluation metrics. Some studies report viewport PSNR or V-VMAF, while others report QoE, subjective quality, bandwidth wastage, or encoding overhead.

Additional adaptive-tiling studies report study-specific quality improvements. TBRA reports an approximately 14.8% improvement in its subjective quality measure [82], while the optimal tile-based approach in [94] reports an approximately 3.8 dB viewport-PSNR gain. Results concerning prediction-aware bitrate allocation, including 360ProbDASH and VATP360, are discussed separately in Section 4.4.

Figure 6 provides a study-specific illustration of the relationship between spatial granularity and coding overhead using the Flare configurations reported in [95]. For the 2×4, 4×4, and 4×6 configurations, the number of tiles increases from 8 to 16 and 24, respectively. The corresponding average tile area, calculated as $A_{\text{tile}} = 100/N$, decreases from 12.5% to 6.25% and 4.17% of the projected frame. Smaller average tile areas provide finer spatial selectivity and can reduce the amount of high-quality content delivered outside the viewport. However, over the same configurations, the reported median encoded-size overhead increases from 7.3% to 9.3% and 10.4%, respectively. This increase shows the loss of compression efficiency caused by introducing additional independently encoded tile boundaries.

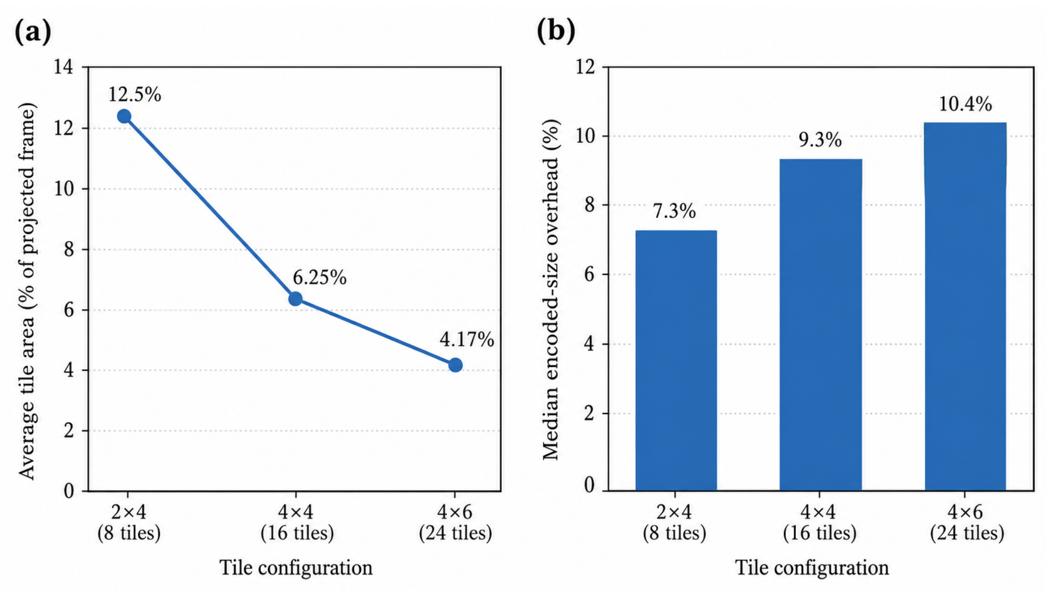

Figure 6. Tile-granularity and coding-overhead trade-off for the Flare configurations: (**a**) average tile area, calculated as $A_{\text{tile}} = 100/N$, as a proxy for spatial granularity; and (**b**) reported median encoded-size overhead. Data source: [95].

Average tile area is used here as a proxy for spatial granularity instead of a direct measurement of viewport precision. The correspondence between tile boundaries and the actual viewport also depends on the tile layout, projection format, viewport geometry, and prediction error. Moreover, the Flare [95] results represent one encoding and evaluation setting and should not be interpreted as a universal quantitative relationship between tile count and coding overhead.

To summarize, the reviewed literature indicates that variable tiling does not remove the trade-off between viewport-oriented spatial control and coding efficiency. Instead, it shifts the operating point by adapting tile sizes and layouts to reduce unnecessary viewport-external delivery while limiting encoding, storage, signaling, and transmission overhead. The preferred configuration therefore depends not only on tile count but also on content characteristics, prediction reliability, client and server resources, and the timing constraints of the streaming system.

4.4. Prediction-Aware Bitrate Allocation

Modern systems integrate viewport prediction into bitrate allocation decisions. In 360ProbDASH [78], probabilistic viewport modeling reduces spatial variance by 46% and improves viewport PSNR by approximately 39%. VATP360 [93] introduces tile-priority classification and reinforcement learning-based bitrate allocation. However, the use of reinforcement learning increases computational overhead.

Prediction-aware allocation schemes commonly model QoE using terms such as the minimum bitrate within the FoV, inter-chunk quality variation, and stall penalty.

4.5. Error Resilience and Misprediction Mitigation

Viewport prediction errors become more severe as the prediction horizon increases. Transformer-based results [9] show approximately 96% accuracy at 1 s, decreasing to about 71% at 2 s. This degradation directly affects streaming decisions. To address this issue, several mechanisms have been proposed, including probabilistic prefetching [96], robust rate guarantees [79] and server-side tile merging [97]. TVSR-OR [97] reduces synchronization delay and lowers CPU usage under high concurrency by merging tiles on the server side, thereby improving disk efficiency and reducing parallel read overhead. These findings show that resilience requires the joint consideration of prediction confidence, buffer size constraints and transport protocol latency [98].

4.6. Multi-Access Edge Computing for Prediction-Aware Streaming

Multi-access Edge Computing (MEC) can mitigate the consequences of viewport-prediction errors by shortening the processing and content-delivery path. Edge-assisted prediction frameworks offload computationally intensive visual or content analysis to an edge server while recent head-orientation or gaze information can continue to be processed at the client [68,99]. This division of computation can allow viewport estimates to be updated closer to the playback deadline than would be possible with a remote cloud server.

MEC also supports prediction-aware caching and prefetching. Cache-aided delivery places viewport-relevant tile representations closer to the user and can reduce transmission and queuing delay relative to retrieval from a remote content server [40]. In Explora-VR, the output of the client-side viewport prediction and rate-adaptation process is advertised in advance to a cache-enabled edge server, which prefetches the tile representations likely to be requested in subsequent segments [100]. Learning-based caching can further update the stored tile representations according to user demand, network conditions, and replacement cost [101].

These mechanisms have the potential to reduce the impact of a misprediction through a shorter correction loop. An initial prediction is used to prefetch or prioritize likely viewport tiles; when more recent viewing information becomes available, the selected viewport can be updated. If the correction changes the required tile set, a nearby edge server may deliver the replacement tiles more quickly than a remote origin server. The correction is effective only if the combined client-edge communication, edge queuing, processing, and tile-delivery delay remains smaller than the available playback-deadline margin.

MEC benefits depend on edge-server load and processing capacity, cache availability, client-to-edge latency, backhaul conditions, user mobility, and the remaining playback slack [40,68,100]. Under edge congestion or frequent cache misses, offloading and coordination overhead may limit the latency advantage.

4.7. Comparative System-Level Trade-Offs

Short, reliable prediction can support finer tiling and focused bitrate allocation, whereas longer, less certain horizons may require broader viewport protection and coarser or grouped tiles to limit misses and request overhead. Transport behavior determines how early tiles must be requested and whether the viewport prediction can be updated before playback. The playback buffer mediates this timing trade-off. A larger buffer provides more protection against bandwidth variation and transport delay, but tiles for buffered segments must generally be selected further ahead of their presentation time, increasing the required prediction horizon and the associated viewport uncertainty. A smaller buffer allows tile selection based on more recent viewport observations, but it leaves less time for retransmission, late viewport correction, or recovery from abrupt capacity reductions. Thus, buffer policy, prediction horizon, and transport delay should be designed jointly [4,46]. For this reason, reported gains should be comparable only under aligned evaluation settings. Table 7 summarizes these cross-layer trade-offs.

Table 7. Optimization strategies and system trade-offs in adaptive 360° video streaming architectures.

Ref.	Layer	Mechanism	System Effect	Main Limitation
[78]	Tile delivery	Probabilistic prefetching	Reduces viewport misses	Sensitive to prediction errors
[102]	Server-side processing	On-demand transcoding	Reduces storage and bandwidth use	Transcoding delay under high load
[20]	Rate allocation	Rate–distortion optimization	Improves tile bitrate allocation	High computational complexity
[22]	Cross-layer control	Segment–tile adaptation	Reduces freezing and quality shifts	Complex implementation
[34]	Content enhancement	Super-resolution streaming	Improves quality at low bitrate	High GPU/edge demand
[47]	Transport layer	Multipath scheduling	Reduces mobile buffering	Complex path coordination
[103]	Encoding layer	Adaptive tile granularity	Improves FoV bandwidth efficiency	Encoding and storage overhead
[104]	ABR control	DRL-based bitrate selection	Improves QoE under bandwidth changes	Training and dataset dependence
[105]	Viewport streaming	Viewport-adaptive tiling	Reduces full-frame bandwidth waste	Static tiling limits flexibility
[106]	Region-based encoding	ROI-aware encoding	Improves viewport bitrate efficiency	ROI boundary artifacts

Note: Layer-specific gains remain limited by prediction reliability, delivery timing and resource cost.

As discussed in Section 3 and illustrated in Section 4.5, prediction reliability decreases as the prediction horizon increases, raising the risk of viewport misses [1].

HTTP-based FoV transmission is affected by frequent HTTP GET requests, inefficient parallel disk reads and GOP-level transmission delay [97]. RTSP-based TVSR-OR shows average latency of 80–150 ms at 100 users, while latency remains below 200 ms up to 180 concurrent users. In contrast, the baseline solution ranges from 100 to 500 ms under load [97]. Furthermore, server-side tile merging reduces synchronization complexity and disk utilization variance [97]. These results show that protocol-level optimization directly affects scalability beyond algorithmic prediction gains. Table 8 compares reported study-specific performance gains under the baseline and evaluation conditions of each system.

Table 8. Comparison of optimization scope, outcome type, and study-specific evidence in adaptive 360° video streaming.

Ref.	Optimization Scope	Outcome Type	Evaluation Condition	Within-Study Evidence	Main Limitation
[2]	Prediction + bitrate allocation	QoE	8 × 8 tiles; 1-s chunks; constant bandwidth	35% vs. clustering; 78% vs. PanoSalNet	Baseline-sensitive result
[93]	Prediction + ABR	QoE	4-s VP; 8 × 8 tiles; 1-s chunks; 4G/LTE traces	2.04–9.13% improvement	Higher model complexity
[59]	Prediction + MPC	QoE	1–3-s prediction horizons	16.75–18.91% improvement	Horizon-dependent accuracy
[88]	Robust tile allocation	Study-defined QoE/utility	Head-motion uncertainty	30% improvement	Uncertainty-model dependent
[103]	Tile bitrate control	Study-defined QoE/utility	Baseline tile allocation	25% improvement	Encoding overhead
[22]	Segment–tile control	Study-defined QoE	Adaptive-streaming baseline	20% improvement	Implementation complexity
[104]	DRL-based ABR	QoE	Variable bandwidth and buffer states	15–20% improvement	Training dependence
[95]	VP + tile scheduling + ABR	Quality level; downloaded bytes	Commercial LTE; BBA baseline	22–23% higher quality level; 26–35% fewer bytes	Tile- and trace-dependent
[45]	Dynamic tiling	QoE	Fixed-tiling baseline	14% improvement	Tile-change overhead
[21]	Latency control	Study-defined QoE/latency	Delay-aware condition	18% improvement	Scenario-specific result
[107]	QoE-aware ABR	QoE	Baseline ABR	22% improvement	QoE-model dependent

Note: Reported values are study-specific. Differences in datasets, prediction horizons, tile and segment configurations, network traces, QoE models, baselines, and transport assumptions prevent direct numerical comparison or ranking across studies.

5. Transport Protocol Trade-Offs

Building on the prediction, tiling, and buffering dependencies discussed in Section 4.7, this section examines how transport behavior determines whether selected viewport tiles arrive before their playback deadlines.

The delivery stack should distinguish among DASH adaptation logic, the HTTP version used for request and response delivery, and the underlying TCP or QUIC transport. The transport implementation also determines the employed congestion-control algorithm (CCA). Neither DASH nor HTTP defines a CCA: HTTP/1.1 and HTTP/2 commonly operate over TCP, whereas HTTP/3 operates over QUIC and therefore uses the CCA selected by the QUIC implementation [108,109]. This distinction is critical because fading, blockage, handover and sudden bandwidth reduction affect queue growth and sending-rate recovery before they affect average throughput. For viewport-adaptive delivery, the relevant outcome is whether prioritized tiles meet their playback deadlines. HTTP/TCP remains the deployment baseline [110], while HTTP/2 [111], HTTP/3/QUIC [112], RTP/RTSP [113] and WebRTC [114] should be compared jointly with their CCA, pacing and reliability mechanisms. Figure 7 summarizes key transport trade-offs and suitable streaming scenarios.

DASH adapts the selected media representation, whereas the underlying congestion-control algorithm determines the download rate of requested segments or tiles. For viewport-adaptive streaming, congestion-control performance should therefore be evaluated using viewport-tile deadline misses, tail delay, rebuffering, and quality variation rather than mean throughput alone [4,109,115,116]. Section Congestion Control Under Volatile Network Conditions examines these congestion-control trade-offs under volatile network conditions.

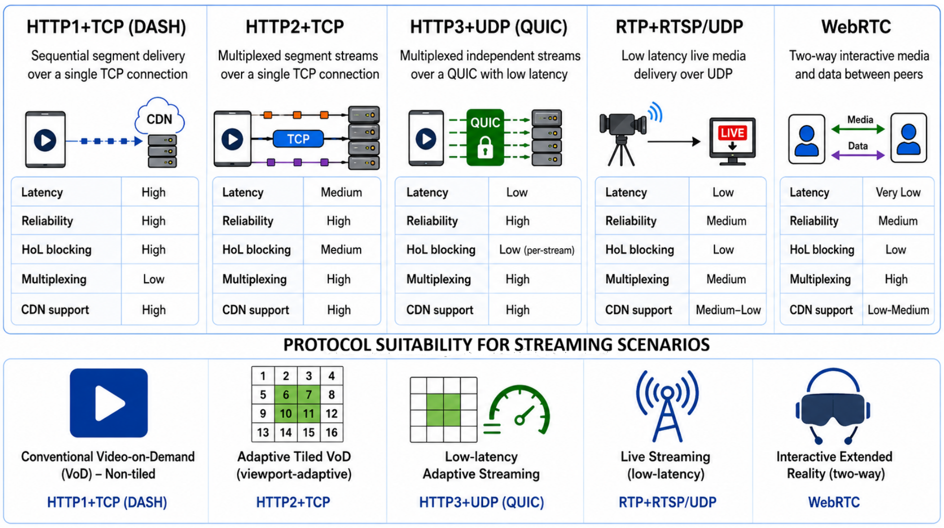

Figure 7. Trade-offs among transport protocols in adaptive 360° video streaming.

Studies using controlled or custom HTTP/2 streaming implementations report benefits from server push and tile prioritization. For instance, the study in [117] uses server push to deliver viewport tiles proactively, reducing request overhead and improving throughput under its evaluated conditions. The study in [118] investigates tile prioritization mechanisms that allow viewport tiles to be delivered earlier than background tiles. Reference [119] proposes dynamic server push to adapt segment delivery to network conditions. The studies in [92,120] focus on optimizing bitrate allocation and video encoding techniques to improve streaming efficiency. Table 9 summarizes the application-layer gains, which remain constrained by transport behavior. Incomplete reporting of the TCP congestion control algorithm and network conditions also limits direct comparison.

Table 9. Comparison of TCP/HTTP-based adaptive streaming mechanisms for viewport-adaptive 360° video delivery.

Ref.	Protocol	Mechanism	Main Role	Outcome	Limitation
[110]	HTTP/TCP	HAS architecture	Overview of adaptive bitrate streaming systems	Improved QoE	HoL blocking
[117]	HTTP/2	Server push tiles	Efficient viewport tile delivery	35% bandwidth saving	TCP dependency
[119]	HTTP/2	Dynamic push	Adaptive segment delivery	Higher delivered bitrate	Playback instability
[121]	HTTP/2	Tile prioritization	Viewport-aware streaming	Improved viewport quality	Sensitive to network variations
[92]	HTTP/TCP	Tile bitrate optimization	Bandwidth efficiency	Reduced stalls	Requires FoV prediction
[120]	HTTP/TCP	HEVC tile streaming	Encoding optimization	Bitrate reduction	Computational complexity
[118]	HTTP/2	Tile prioritization	Dynamic tile scheduling	Reduced bandwidth	Prediction errors
[103]	HTTP/TCP	Non-uniform tile coding	Transmission efficiency	QoE improvement	Encoding overhead
[122]	HTTP/TCP	Edge-assisted streaming	DRL bitrate adaptation	Stable QoE	High complexity

Takeaway: HTTP/2 mechanisms can reduce request overhead and improve viewport-tile delivery order, but their gains remain constrained by the underlying TCP congestion controller, network variability, and viewport-prediction accuracy.

Congestion Control Under Volatile Network Conditions

QUIC stream multiplexing and its congestion control response should be analyzed separately. RFC 9002 defines a NewReno-like reference controller and permits alternative CCAs [108]. CUBIC is a standardized congestion-control algorithm [123], whereas BBRv3 is an implementation and research evolution of model-based congestion control [124]. The NewReno-like controller provides conservative recovery; however, it may underutilize links whose capacity recovers rapidly. CUBIC increases its congestion window more aggressively, potentially supporting higher-bitrate viewport delivery on stable high-capacity paths; however, its aggressive probing may provoke inflated queues and tile-deadline misses after blockage, handover, or sudden capacity reduction [123]. Compared with CUBIC, BBRv3 exhibits different queuing and recovery behavior, which can affect tail delay and tile-deadline compliance. These differences directly affect rate-recovery time, tail delay, and viewport-tile deadline compliance under volatile network conditions.

Congestion control and stream scheduling are complementary. Prioritized independent streams improve viewport-tile delivery order [125], partial reliability avoids retransmitting obsolete background data [126], and tile-to-stream mapping can reduce segment download time [127]. Multipath scheduling further reduces tile delay and rebuffering by exploiting path diversity [128,129]. However, these mechanisms decide which data are transmitted, whereas the connection-level CCA determines the available sending budget. Persistent queuing or sluggish CCA recovery can therefore delay urgent tiles despite effective prioritization. Study-level comparisons should report both the QUIC delivery mechanism and the employed CCA.

L4S is not itself a congestion-control algorithm. It is an architecture combining a scalable sender-side congestion controller, fine-grained ECN feedback, and compatible bottleneck queue management, such as DualQ Coupled AQM [130–132]. When these mechanisms are supported along the path, frequent congestion signaling can maintain a shallow queue while preserving high utilization. This can benefit late viewport corrections and urgent-tile delivery by reducing queuing delay and delay variation. The benefit is deployment-dependent and is not guaranteed when the sender, receiver, or bottleneck lacks the required L4S support.

QUIC-DC is a novel CCA that reacts to one-way queuing-delay growth before buffer overflow and reports lower loss and end-to-end delay than CUBIC, BBRv2, and NewReno while preserving network utilization [133].

Recent evidence shows that QUIC video QoE depends jointly on the CCA implementation and ABR logic [109,134], while protocol-level comparisons show additional dependence on the delivery model and evaluated network conditions [115]. Mixed-reality measurements similarly confirm that CCA selection changes the throughput–latency tradeoff [116]. Table 10 shows the evaluation setups used in recent QUIC/HTTP/3 studies.

Table 10. Evaluation environments reported in QUIC/HTTP/3 studies for viewport-adaptive streaming.

Ref.	Transport Scheme	Evaluation Environment	Main Result
[125]	DASH over QUIC with urgent-tile priority	Custom DASH/QUIC testbed with bandwidth control and HMD-trace replay	Lower tile-miss ratio and higher viewport quality.
[126]	Partially reliable HTTP/3 with viewport priority	Mininet on imec Virtual Wall using Linux `tc`, `quiche`, and real 5G traces	46% higher throughput and 33% fewer interruptions.
[127]	Customized QUIC for a VR network slice	`aioquic`-based simulation with a three-client slice topology and Oculus Rift DK2 traces	Shorter segment downloads and higher segment quality.
[128]	HTTP/3/MPQUIC with stream-aware scheduling	IVSH3/AStream Network emulator using Mahimahi/mpshell and Puffer traces	Up to 25% lower tile-download time.
[129]	QUIC/MPQUIC with cross-layer scheduling	Ubuntu 18.04 testbed using `picohttp`, `picoquic`, PQUIC, 360AStream, and Linux `tc`	Rebuffering halved; median quality increased by 1.5–2.5 levels.
[135]	Multipath HTTP/3 stream scheduling	IVSH3/Stream-P in Vagrant using Mahimahi/mpshell and real HMD traces	Up to 81% lower stall time.

Note: QUIC/HTTP/3 results depend on the protocol implementation, congestion-control algorithm, network-emulation or simulation configuration, trace source, and experimental environment. The table reports the methodological details provided by the respective studies; omitted implementation details were not consistently available.

Before HTTP-based adaptive streaming became dominant, multimedia delivery commonly relied on RTP/RTSP architectures. In this model, RTP supports real-time packet delivery over UDP, while RTSP manages session control. Low-latency immersive delivery using RTSP is discussed in [136], which proposes an RTSP-based architecture for VR services over IPTV platforms. Relevant studies [46,137] further examine network architectures for immersive media delivery in wireless environments.

WebRTC is designed for real-time peer-to-peer communication and is widely adopted in teleconferencing, cloud gaming and interactive VR applications [138]. Compared with DASH-based systems, WebRTC prioritizes low latency over strict reliability, making it suitable for real-time interaction and dynamic viewport updates in immersive environments. This advantage comes with a trade-off: packet loss can directly degrade visual quality through visible artifacts. For RTP/WebRTC delivery, SCReAM provides delay-sensitive congestion control for interactive real-time media and has been evaluated under 5G network conditions [139].

Transport suitability depends on the immersive-streaming scenario. For buffered 360° video on demand, HTTP/2 over TCP provides mature CDN support, reliable delivery, and multiplexed request–response exchanges, making it suitable when the playback buffer can absorb transport and retransmission delay [4,111]. HTTP/3 over QUIC can be advantageous for low-latency adaptive VoD and live 360° streaming because it provides independent streams and low-latency connection establishment. Viewport-tile prioritization can be implemented through stream scheduling, while partial-reliability mechanisms require non-standard or experimental QUIC/HTTP/3 extensions [125,126]. Its performance nevertheless depends on the congestion-control algorithm, scheduling policy, and network variability [46,134]. RTP/RTSP remains suitable for controlled live-delivery environments in which session management and low-delay media transport are more important than CDN-scale HTTP delivery [97,136]. WebRTC is the more natural choice for ultra-low-latency interactive VR and two-way immersive communication because it integrates real-time media transport, congestion control, encryption, and peer-session establishment [114]. For multi-user immersive applications, the appropriate architecture depends on the communication pattern. One-to-many VoD or live distribution can exploit HTTP-based CDN and edge caching, whereas synchronized interactive sessions generally require WebRTC forwarding or relay infrastructure, or RTP-based delivery in managed networks. In this case, scalability depends not only on the transport protocol but also on server-side forwarding, viewport sharing, edge processing, and the degree to which content can be reused across users.

6. Discussion: A Unified Cross-Layer Optimization Framework

Adaptive 360° video streaming can be formulated as a closed-loop cross-layer control problem whose objective is to maximize viewport quality delivered before the playback deadline. The framework combines viewport history and prediction outputs with network conditions, buffer occupancy, tile deadlines, and available client or edge resources. Decisions are coordinated across different timescales: tile selection, per-tile bitrate, and delivery priority may be updated at each adaptation interval, whereas prediction-horizon policies, tile layouts, segment duration, transport configuration, and processing placement are generally selected or revised less frequently.

Figure 8 shows the proposed closed-loop coordination across the streaming layers.

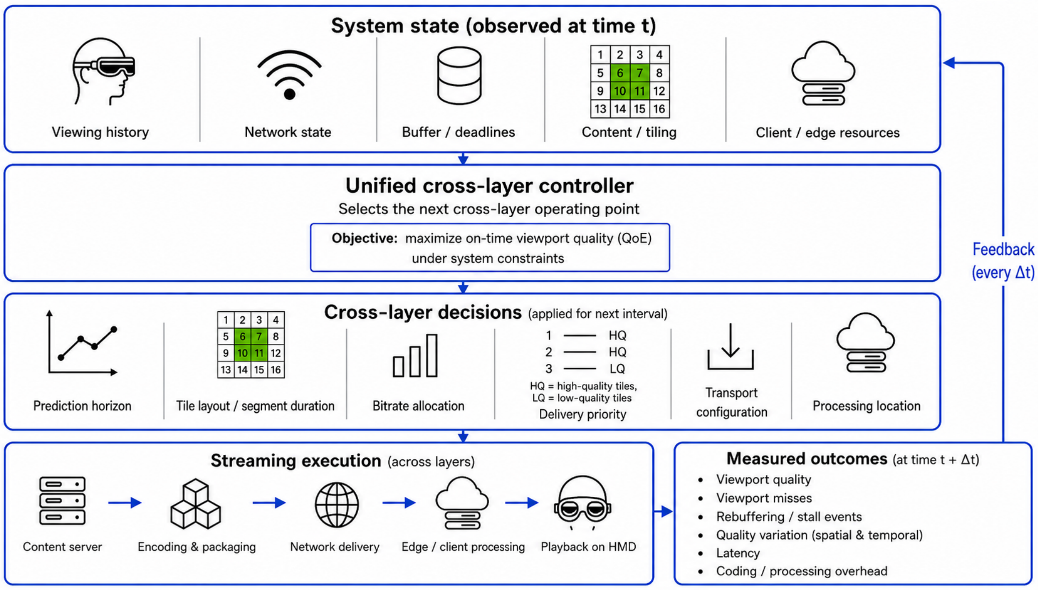

Figure 8. Unified closed-loop framework for adaptive 360° video streaming.

The observed system state is translated into coordinated decisions, whose measured outcomes guide the next adaptation interval. The combined evidence identifies the dependencies that form the basis of the proposed joint-control framework. Prediction error changes which tiles receive high-quality delivery. With fine tiling, a small angular error may shift the actual FoV across tile boundaries, leaving visible tiles at low quality while bandwidth is spent outside the viewport. Wider protection or base-layer coverage reduces this spatial miss; however, this increases redundant transmission and download time [54,79]. Two failure modes should therefore be distinguished. A spatial miss occurs when prediction error causes one or more tiles in the actual viewport to be excluded from the high-quality tile set. A tile deadline miss occurs when a required viewport tile is selected or requested, yet arrives after its playback deadline. The first failure primarily depends on prediction uncertainty, viewport protection, and tile layout, whereas the second depends on the available deadline margin, bitrate allocation, transport behavior, and processing delay.

The resulting QoE impact depends on whether tile selection can be revised after more recent viewport information becomes available. With sufficient deadline margin, newly identified viewport tiles may still be requested or upgraded before playback. Otherwise, the spatial miss remains visible. Independently, even correctly selected viewport tiles may incur a delivery miss if transport or processing delay prevents them from arriving before their playback deadlines. Both failures can reduce viewport quality and increase quality variation, while sufficiently severe delivery delays may also cause rebuffering [4,46]. Viewport prediction accuracy should guide tile coverage and transport scheduling jointly to reduce spatial and deadline misses.

As a conceptual synthesis, these interactions can be represented by the following single-interval objective:

$$\max_{a_k \in \mathcal{A}(s_k)} \mathbb{E}\Big[Q^{\mathrm{on}}_{v,k} - \lambda_{\mathrm{sp}} M^{\mathrm{sp}}_k - \lambda_{\mathrm{dl}} M^{\mathrm{dl}}_k - \lambda_s S_k - \lambda_v V_k - \lambda_o O_k - \lambda_c C_k \,\Big|\, s_k\Big], \tag{9}$$

where s_k denotes the observed system state and $\mathcal{A}(s_k)$ is the set of feasible cross-layer decisions. The term $Q^{\mathrm{on}}_{v,k}$ denotes the quality of actual viewport tiles received before their playback deadlines. The penalties M^{sp}_k and M^{dl}_k represent spatial misses caused by tile selection and delivery misses caused by late arrival, respectively. Furthermore, S_k denotes rebuffering, V_k quality variation, O_k coding and signaling overhead, and C_k computational cost. The weighting coefficients determine the relative importance of these outcomes. Feasibility is constrained by the available bandwidth, buffer state, tile deadlines, and client, edge, and server resources.

These interactions define a joint operating policy rather than independent choices. The prediction horizon should reflect end-to-end delivery time, with longer horizons enabling earlier prefetching and a larger buffer margin [71]. Tile refinement is justified only when its viewport-quality gain exceeds coding and storage overhead [43]; greater uncertainty requires wider viewport protection and priority for likely viewport tiles [78,100]. Transport scheduling, reliability mechanisms, and CCA selection or configuration should account for network variability, whereas edge processing should be used when it reduces end-to-end delivery delay [40,134].

Future evaluations should report prediction confidence, tile configuration, segment duration, buffer policy, RTT, loss, CCA and processing platform, together with on-time viewport quality, viewport misses, tile-deadline violations, rebuffering, quality variation, coding overhead, and tail latency. Such reporting would facilitate more transparent attribution of observed gains to prediction, tiling, bitrate control, transport, and their interactions, although rigorous separation would require controlled ablation studies under aligned evaluation conditions.

7. Conclusions

Adaptive 360° video streaming should be designed as a deadline-constrained cross-layer system rather than as a set of independent modules. The reviewed evidence shows that gains in prediction accuracy translate into improved QoE only when the selected tile representations receive sufficient bitrate and arrive before their playback deadlines.

Future systems should select the prediction horizon and viewport-protection level jointly according to prediction uncertainty and the available delivery margin. Greater prediction uncertainty may justify broader viewport protection only when the additional tiles can be delivered before their playback deadlines. When the delivery margin is limited or network capacity is highly uncertain, the system may instead require more conservative bitrate allocation, base-quality coverage, or stricter prioritization of the most probable viewport tiles. Tile granularity should be coordinated with transport and processing decisions so that viewport-quality gains are not offset by additional system overhead. Transport configuration should be selected according to the available deadline margin and the dominant source of network variability rather than according to average throughput alone. Similarly, edge processing or caching should be used only when the combined signaling, queuing, processing, and client–edge delivery delay is lower than that of the corresponding client-side or remote-server alternative.

Future evaluations should report the prediction horizon, tile layout, segment duration, buffer policy, and network conditions explicitly and vary these factors through controlled experiments or ablation studies. They should measure on-time viewport quality alongside playback disruption and system overhead, allowing the contributions of individual components and their cross-layer interactions to be assessed more transparently.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/network6030066/s1: Table S1: Studies Included in the Review (n = 80). The studies listed in Table S1 correspond to Refs. [2,4–7,18–23,29–47,51,53,55–59,61,62,64–85,88,90–107].

Author Contributions: Investigation, M.F. and G.M.; writing—original draft preparation, M.F.; writing—review and editing, G.M. and L.D.C.; visualization, M.F.; supervision, L.D.C. and S.M.; project administration, L.D.C. and S.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially funded by Regione Puglia through "Bando Reti" under the project ARIANNA, CUP B97H24001880007.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

3DoF	Three Degrees of Freedom
6DoF	Six Degrees of Freedom
ABR	Adaptive Bitrate
ACP	Adjusted Cubemap Projection
AEP	Adjusted Equal-Area Projection
AQM	Active Queue Management
CCA	Congestion-Control Algorithm
CDN	Content Delivery Network
CMP	Cubemap Projection
CNN	Convolutional Neural Network
DASH	Dynamic Adaptive Streaming over HTTP
DL	Deep Learning
DRL	Deep Reinforcement Learning
EAC	Equi-Angular Cubemap Projection
ECN	Explicit Congestion Notification
ERP	Equirectangular Projection
FoV	Field of View
GNN	Graph Neural Network
GOP	Group of Pictures
GRU	Gated Recurrent Unit
HEC	Hybrid Equi-angular Cubemap Projection
HEVC	High Efficiency Video Coding
HMD	Head-Mounted Display
HoL	Head-of-Line
HTTP	Hypertext Transfer Protocol
L4S	Low Latency, Low Loss, and Scalable Throughput
LSTM	Long Short-Term Memory
LTE	Long-Term Evolution
MEC	Multi-access Edge Computing
ML	Machine Learning
MPC	Model Predictive Control
MPD	Media Presentation Description
MPEG	Moving Picture Experts Group
PRISMA	Preferred Reporting Items for Systematic Reviews and Meta-Analyses
PSNR	Peak Signal-to-Noise Ratio
QoE	Quality of Experience
QP	Quantization Parameter
QUIC	QUIC Transport Protocol (name, not an acronym)
RL	Reinforcement Learning

RSP	Rotated Sphere Projection
RTP	Real-Time Transport Protocol
RTSP	Real-Time Streaming Protocol
RTT	Round-Trip Time
SRD	Spatial Relationship Description
SVC	Scalable Video Coding
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
V-VMAF	Viewport Video Multi-Method Assessment Fusion
VMAF	Video Multi-Method Assessment Fusion
VoD	Video on Demand
VR	Virtual Reality
WebRTC	Web Real-Time Communication
WS-PSNR	Weighted-to-Spherically-Uniform Peak Signal-to-Noise Ratio

References

1. Wahba, M.Z.A.; Baldoni, S.; Battisti, F. Learning-Based Viewport Prediction for 360-Degree Videos: A Review. *Electronics* **2025**, *14*, 3743. [CrossRef]
2. Chopra, L.; Chakraborty, S.; Mondal, A.; Chakraborty, S. PARIMA: Viewport Adaptive 360-Degree Video Streaming. In *Proceedings of the Web Conference*; WWW '21; ACM: New York, NY, USA, 2021; pp. 2379–2391. [CrossRef]
3. Khan, K. Advancements in 360-Degree Virtual Reality Video Streaming: A Comprehensive Overview. *Int. J. Innov. Sci. Res. Technol.* **2023**, *8*, 91–103. [CrossRef]
4. Nguyen, V.H.; Bui, D.T.; Tran, T.L.; Truong, C.T.; Truong, T.H. Scalable and Resilient 360-Degree-Video Adaptive Streaming over HTTP/2 against Sudden Network Drops. *Comput. Commun.* **2024**, *216*, 1–15. [CrossRef]
5. Kan, N.; Zou, J.; Li, C.; Dai, W.; Xiong, H. RAPT360: Reinforcement Learning-Based Rate Adaptation for 360-Degree Video Streaming With Adaptive Prediction and Tiling. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 1607–1623. [CrossRef]
6. Zhang, Z.; Chen, Y.; Zhang, W.; Yan, C.; Zheng, Q.; Wang, Q.; Chen, W. Tile Classification Based Viewport Prediction with Multi-modal Fusion Transformer. In *Proceedings of the ACM International Conference on Multimedia*; MM '23; ACM: New York, NY, USA, 2023; pp. 3560–3568. [CrossRef]
7. Tian, Y.; Zhong, Y.; Han, Y.; Chen, F. Viewport prediction with cross modal multiscale transformer for 360° video streaming. *Sci. Rep.* **2025**, *15*, 30346. [CrossRef] [PubMed]
8. Nguyen, A.; Yan, Z.; Nahrstedt, K. Your Attention is Unique: Detecting 360-Degree Video Saliency in Head-Mounted Display for Head Movement Prediction. In *Proceedings of the ACM International Conference on Multimedia*; MM '18; ACM: New York, NY, USA, 2018; pp. 1190–1198. [CrossRef]
9. Chao, F.Y.; Ozcinar, C.; Smolic, A. Transformer-based Long-Term Viewport Prediction in 360° Video: Scanpath is All You Need. In *Proceedings of the IEEE International Workshop on Multimedia Signal Processing (MMSP)*; IEEE: Piscataway, NJ, USA, 2021; pp. 1–6. [CrossRef]
10. Yaqoob, A.; Bi, T.; Muntean, G.M. A Survey on Adaptive 360° Video Streaming: Solutions, Challenges and Opportunities. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2801–2838. [CrossRef]
11. Wong, E.S.; Wahab, N.H.A.; Saeed, F.; Alharbi, N. 360-Degree Video Bandwidth Reduction: Technique and Approaches Comprehensive Review. *Appl. Sci.* **2022**, *12*, 7581. [CrossRef]
12. Fan, C.L.; Lo, W.C.; Pai, Y.T.; Hsu, C.H. A Survey on 360° Video Streaming: Acquisition, Transmission, and Display. *ACM Comput. Surv.* **2019**, *52*, 1–36. [CrossRef]
13. Chiariotti, F. A Survey on 360-Degree Video: Coding, Quality of Experience and Streaming. *Comput. Commun.* **2021**, *177*, 133–155. [CrossRef]
14. Khan, K. Advancements and Challenges in 360° Augmented Reality Video Streaming: A Comprehensive Review. *Int. J. Comput. Commun. Netw.* **2024**, *13*, 1–20. [CrossRef]
15. Khan, K.; Goodridge, W. Adaptive Video Streaming in Virtual Reality Environments: Mathematical Models for Immersive Experiences with 360-Degree Content and Head Movement Prediction. *Int. J. Multidiscip. Res. Publ.* **2024**, *6*, 252–257. Available online: https://ijmrap.com/volume-6-issue-7/ (accessed on 21 May 2026).
16. Shafi, R.; Shuai, W.; Younus, M. 360-Degree Video Streaming: A Survey of the State of the Art. *Symmetry* **2020**, *12*, 1491. [CrossRef]
17. Bentaleb, A.; Lim, M.; Hammoudi, S.; Harous, S.; Zimmermann, R. Solutions, Challenges, and Opportunities in Volumetric Video Streaming: An Architectural Perspective. *ACM Trans. Multimed. Comput. Commun. Appl.* **2025**, *21*, 1–35. [CrossRef]

18. Li, Z.; Wang, Y.; Liu, Y.; Li, J.; Zhu, P. JUST360: Optimizing 360-Degree Video Streaming Systems With Joint Utility. *IEEE Trans. Broadcast.* **2024**, *70*, 468–481. [CrossRef]
19. Singhal, C.; Ghosh, M.; Mittal, H.; Dewangan, P. MAIVS: Machine Learning Based Adaptive UHD 360-Degree Immersive Video Streaming. In *Proceedings of the IEEE International Conference on Communications Workshops (ICC Workshops)*; IEEE: Piscataway, NJ, USA, 2023; pp. 1082–1087. [CrossRef]
20. Wei, X.; Zhou, M.; Jia, W. Global Optimization Solution for Dynamic Adaptive 360-Degree Streaming. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–5. [CrossRef]
21. Wei, X.; Zhou, M.; Jia, W. Toward Low-Latency and High-Quality Adaptive 360° Streaming. *IEEE Trans. Ind. Inform.* **2023**, *19*, 6326–6336. [CrossRef]
22. Wei, X.; Zhou, M.; Kwong, S.; Yuan, H.; Jia, W. A Hybrid Control Scheme for 360-Degree Dynamic Adaptive Video Streaming Over Mobile Devices. *IEEE Trans. Mob. Comput.* **2022**, *21*, 3428–3442. [CrossRef]
23. Guo, H.; Wang, F.; Zhang, W.; Zhu, Y.; Cui, L.; Liu, J.; Yu, F.R.; Zhang, L. Joint Adaptation for Mobile 360-Degree Video Streaming and Enhancement. *IEEE Trans. Mob. Comput.* **2025**, *24*, 7726–7741. [CrossRef]
24. Hussain, I.; Kwon, O.J. Evaluation of 360° Image Projection Formats; Comparing Format Conversion Distortion Using Objective Quality Metrics. *J. Imaging* **2021**, *7*, 137. [CrossRef] [PubMed]
25. Hussain, I.; Kwon, O.J.; Choi, S. Evaluating the Coding Performance of 360° Image Projection Formats Using Objective Quality Metrics. *Symmetry* **2021**, *13*, 80. [CrossRef]
26. Pi, H.; Tian, S.; Lu, M.; Liu, J.; Guo, Y.; Zhang, S. A Comprehensive Comparison of Projections in Omnidirectional Super-Resolution. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–5. [CrossRef]
27. Regensky, A.; Brand, F.; Kaup, A. Analysis of Neural Video Compression Networks for 360-Degree Video Coding. In *Proceedings of the Picture Coding Symposium (PCS)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–5. [CrossRef]
28. Ye, Y.; Alshina, E.; Boyce, J. *JVET-E1003: Algorithm Descriptions of Projection Format Conversion and Video Quality Metrics in 360Lib*; Technical Report JVET-E1003; Joint Video Experts Team: Geneva, Switzerland, 2018.
29. Wu, Z.; Lu, L.; Chen, X.; Ma, Y.; Yang, S.; Wang, M.; Zhong, L.; Wu, D.O.; Xu, C. SRA360: Super-Resolution Enhanced Adaptive 360-Degree Video Streaming for Heterogeneous Viewers. *IEEE Trans. Netw. Sci. Eng.* **2025**, *12*, 4531–4543. [CrossRef]
30. Liu, H.; Zhang, X.; Jia, C.; Li, Y.; Xie, G. AKI360: Enabling Highly Interactive 360-degree Video Streaming by Adaptive Keyframe Interval. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; IEEE: Piscataway, NJ, USA, 2025; pp. 1–5.
31. Wang, S.; Tan, X.; Liu, Z.; Tai, Z.; Sun, M.; Yang, J.; Wu, J. Adaptive Computing and Multicasting Optimization for Live 360-Degree Video Streaming. *IEEE Trans. Veh. Technol.* **2026**, *75*, 3190–3204. [CrossRef]
32. Viet, H.N.; Huy, H.B.; Thanh, C.T.; Thu, H.T. COSMN: Clustering-Based Optimization for 360-Degree Live Streaming over Mobile Networks. *EAI Endorsed Trans. Ind. Netw. Intell. Syst.* **2025**, *25*, 1–16. [CrossRef]
33. Zhao, B.; Cao, S.; Zhang, C. Towards Scalable 360-Degree Video Streaming: A Method with an Adaptive Selection Enhancement Mechanism. In *Proceedings of the International Conference on Computer Graphics, Image and Virtualization (ICCGIV)*; IEEE: Piscataway, NJ, USA, 2025; pp. 145–149. [CrossRef]
34. Hong, S.; Wang, R.; Cao, G. Adaptive 360-Degree Video Streaming with Super-Resolution and Interpolation. In *Proceedings of the IEEE Conference Virtual Reality and 3D User Interfaces (VR)*; IEEE: Piscataway, NJ, USA, 2025; pp. 34–43. [CrossRef]
35. Yujun, L.; Fujihashi, T.; Saruwatari, S.; Watanabe, T. 360Cast+: Viewport Adaptive Soft Delivery for 360-Degree Videos. *IEEE Access* **2021**, *9*, 52684–52697. [CrossRef]
36. Nguyen, D.; Tran, H.T.; Thang, T.C. A client-based adaptation framework for 360-degree video streaming. *J. Vis. Comun. Image Represent.* **2019**, *59*, 231–243. [CrossRef]
37. Zhou, C.; Wang, S.; Xiao, M.; Wei, S.; Liu, Y. AdaP-360: User-Adaptive Area-of-Focus Projections for Bandwidth-Efficient 360-Degree Video Streaming. In *Proceedings of the ACM International Conference on Multimedia, New York, NY, USA*; MM '20; ACM: New York, NY, USA, 2020; pp. 3715–3723. [CrossRef]
38. Feng, W.; Dai, Y.; Wang, S. Adaptive 360-Degree Video Streaming with Multi-window and Stochastic Viewport Prediction. In *Proceedings of the IEEE International Conference on Communications*; IEEE: Piscataway, NJ, USA, 2024; pp. 3402–3407. [CrossRef]
39. Qiu, T.; Jain, I.K.; Wu, R.; Bharadia, D.; Cosman, P. Delivering 360-degree Video With Viewport-Adaptive Truncation. In *Proceedings of the International Symposium on Wireless Personal Multimedia Communications (WPMC)*; IEEE: Piscataway, NJ, USA, 2022; pp. 7–11. [CrossRef]
40. Liu, J.; Simon, G.; Corbillon, X.; Chakareski, J.; Yang, Q. Delivering Viewport-Adaptive 360-Degree Videos in Cache-Aided MEC Networks. In *Proceedings of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*; IEEE: Piscataway, NJ, USA, 2020; pp. 1–6. [CrossRef]

41. Sheng, D.; Gao, B.; Liu, G.; Qi, Q.; Wang, J. Efficient Tile-Based Adaptive Streaming for 360-Degree Video-on-Demand Systems. In *Proceedings of the ACM Multimedia Systems (MMSys)*; MMSys '24; ACM: New York, NY, USA, 2024; pp. 435–440. [CrossRef]
42. Zhou, C.; Li, Z.; Osgood, J.; Liu, Y. On the Effectiveness of Offset Projections for 360-Degree Video Streaming. *ACM Trans. Multimed. Comput. Commun. Appl.* **2018**, *14*, 1–24. [CrossRef]
43. Gao, W.; Li, C.; Lv, H.; Dai, W.; Zou, J.; Xiong, H.; Pan, X.; Wang, H. Optimal Tile-Based Encoding for 360-Degree Video Streaming. In *Proceedings of the Picture Coding Symposium (PCS)*; IEEE: Piscataway, NJ, USA, 2022; pp. 295–299. [CrossRef]
44. Zou, J.; Li, C.; Liu, C.; Yang, Q.; Xiong, H.; Steinbach, E. Probabilistic Tile Visibility-Based Server-Side Rate Adaptation for Adaptive 360-Degree Video Streaming. *IEEE J. Sel. Top. Signal Process.* **2020**, *14*, 161–176. [CrossRef]
45. Li, Z.; Wang, Y.; Liu, Y. SAD360: Spherical Viewport-Aware Dynamic Tiling for 360-Degree Video Streaming. In *Proceedings of the IEEE International Conference on Visual Communications and Image Processing (VCIP)*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–5. [CrossRef]
46. Shippey, S.; Srinivasan, S.; Aryafar, E.; Chakareski, J. PRISM-XR: Predictive Real-Time Intelligent Streaming for 360-Degree Multimedia. In *Proceedings of the International Conference on Metaverse Computing, Networking and Applications (MetaCom)*; IEEE: Piscataway, NJ, USA, 2025. [CrossRef]
47. Yao, Y.; Gao, Q.; Zhang, G.; Yin, Y.; Shen, L.; Qian, X. Improving 360-degree Video Streaming with Adaptive Path Scheduling. In *Proceedings of the International Conference on Advanced Cloud and Big Data (CBD)*; IEEE: Piscataway, NJ, USA, 2023; pp. 192–199. [CrossRef]
48. Quang, N.L.; Tho, N.T.A.; Minh, T.G.; Nghia, T.D.; Huong, T.T.; Nguyen, D.V. Toward Optimal Viewport Adaptive Volumetric Video Streaming: Problem, Essential Tools, and Datasets. In *Proceedings of the International Conference on Ubiquitous and Future Networks (ICUFN)*; IEEE: Piscataway, NJ, USA, 2025; pp. 262–267. [CrossRef]
49. Hu, Q.; He, Q.; Zhong, H.; Lu, G.; Zhang, X.; Zhai, G.; Wang, Y. VARFVV: View-Adaptive Real-Time Interactive Free-View Video Streaming With Edge Computing. *IEEE J. Sel. Areas Commun.* **2025**, *43*, 2620–2634. [CrossRef]
50. De Castro Araújo, G.; Domingues Garcia, H.; C.Q. Farias, M.; Prakash, R.; M. Carvalho, M. A 360-degree Video Player for Dynamic Video Editing Applications. *ACM Trans. Multimed. Comput. Commun. Appl.* **2025**, *21*, 1–23. [CrossRef]
51. Petrangeli, S.; Simon, G.; Swaminathan, V. Trajectory-Based Viewport Prediction for 360-Degree Virtual Reality Videos. In *Proceedings of the IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*; IEEE: Piscataway, NJ, USA, 2018; pp. 157–160. [CrossRef]
52. Nguyen, D. An Evaluation of Viewport Estimation Methods in 360-Degree Video Streaming. In *Proceedings of the International Conference on Business and Industrial Research (ICBIR)*; IEEE: Piscataway, NJ, USA, 2022; pp. 161–166. [CrossRef]
53. Truong, T.H.; Dang, T.L.; Nguyen, T.D.; Pham, N.S. An Accurate Viewport Estimation Method for 360 Video Streaming using Deep Learning. *EAI Endorsed Trans. Ind. Netw. Intell. Syst.* **2022**, *9*, e2. [CrossRef]
54. Yu, J.; Liu, Y. Field-of-view prediction in 360-degree videos with attention-based neural encoder-decoder networks. In *Proceedings of the ACM Workshop on Immersive Mixed and Virtual Environment Systems*; MMVE '19; ACM: New York, NY, USA, 2019; pp. 37–42. [CrossRef]
55. Peng, S.; Hu, J.; Han, X.; Xu, C. Viewport-Driven Adaptive 360° Live Streaming Optimization Framework. *J. Netw. Netw. Appl.* **2022**, *1*, 139–149. [CrossRef]
56. Peng, S.; Hu, J.; Li, Z.; Xiao, H.; Yang, S.; Xu, C. Spherical Convolution-based Saliency Detection for FoV Prediction in 360-degree Video Streaming. In *Proceedings of the International Wireless Communications and Mobile Computing (IWCMC)*; IEEE: Piscataway, NJ, USA, 2023; pp. 162–167. [CrossRef]
57. Feng, X.; Swaminathan, V.; Wei, S. Viewport Prediction for Live 360-Degree Mobile Video Streaming Using User-Content Hybrid Motion Tracking. *ACM Interact. Mob. Wearable Ubiquitous Technol.* **2019**, *3*, 1–22. [CrossRef]
58. Xu, X.; Tan, X.; Wang, S.; Liu, Z.; Zheng, Q. Multi-Features Fusion based Viewport Prediction with GNN for 360-Degree Video Streaming. In *Proceedings of the IEEE International Conference on Metaverse Computing, Networking and Applications (MetaCom)*; IEEE: Piscataway, NJ, USA, 2023; pp. 57–64. [CrossRef]
59. Feng, W.; Wang, S.; Dai, Y. Adaptive 360-Degree Streaming: Optimizing With Multi-Window and Stochastic Viewport Prediction. *IEEE Trans. Mob. Comput.* **2025**, *24*, 5903–5915. [CrossRef]
60. Haseeb Ul Hassan, S.M.; Brennan, A.; Muntean, G.M.; McManis, J. User Profile-Based Viewport Prediction Using Federated Learning in Real-Time 360-Degree Video Streaming. In *Proceedings of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–7. [CrossRef]
61. Dong, P.; Shen, R.; Xie, X.; Li, Y.; Zuo, Y.; Zhang, L. Predicting Long-Term Field of View in 360-Degree Video Streaming. *IEEE Netw.* **2023**, *37*, 26–33. [CrossRef]
62. Zhang, L.; Xu, W.; Lu, D.; Cui, L.; Liu, J. MFVP: Mobile-Friendly Viewport Prediction for Live 360-Degree Video Streaming. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–6. [CrossRef]
63. Aydin, A.G.; Vikalo, H. Viewport Prediction via Adaptive Edge Offloading. *IEEE Netw. Lett.* **2025**, *7*, 21–25. [CrossRef]

64. Chen, X.; Cao, B.; Ahmad, I.; Xue, X. Lightweight Neural Network-Based Viewport Prediction for Live VR Streaming in Wireless Video Sensor Network. *Mob. Inf. Syst.* **2021**, *2021*, 8501990. [CrossRef]
65. Van Damme, S.; Vega, M.T.; De Turck, F. Machine Learning Based Content-Agnostic Viewport Prediction for 360-Degree Video. *ACM Trans. Multimed. Comput. Commun. Appl.* **2022**, *18*, 1–24. [CrossRef]
66. Zhou, H.; Zhao, F.; Li, C. Multi-scale Historical Trajectory Decomposition for Viewport Prediction in 360-degree Videos. *ACM Trans. Multimed. Comput. Commun. Appl.* **2025**, *21*, 1–22. [CrossRef]
67. Deng, R. Saliency Detection Based Adaptive Streaming Method for Panoramic Videos. In *Proceedings of the International Conference on Communications and Broadband Networking*; ICCBN '23; ACM: New York, NY, USA, 2023; pp. 44–51. [CrossRef]
68. Pan, Z.; Zhang, Y.; Lin, T.; Yan, J. LiveAE: Attention-based and Edge-assisted Viewport Prediction for Live 360° Video Streaming. In *Proceedings of the Workshop on Emerging Multimedia Systems*; EMS '23; ACM: New York, NY, USA, 2023; pp. 28–33. [CrossRef]
69. Feng, X.; Li, W.; Wei, S. LiveROI: Region of interest analysis for viewport prediction in live mobile virtual reality streaming. In *Proceedings of the ACM Multimedia Systems Conference (MMSys)*; MMSys '21; ACM: New York, NY, USA, 2021; pp. 132–145. [CrossRef]
70. Wang, S.; Yang, S.; Li, H.; Zhang, X.; Zhou, C.; Xu, C.; Qian, F.; Wang, N.; Xu, Z. SalientVR: Saliency-driven mobile 360-degree video streaming with gaze information. In *Proceedings of the Annual International Conference on Mobile Computing And Networking*; MobiCom '22; ACM: New York, NY, USA, 2022; pp. 542–555. [CrossRef]
71. Gao, B.; Sheng, D.; Zhang, L.; Qi, Q.; He, B.; Zhuang, Z.; Wang, J. STAR-VP: Improving Long-term Viewport Prediction in 360° Videos via Space-aligned and Time-varying Fusion. In *Proceedings of the ACM International Conference on Multimedia*; MM '24; ACM: New York, NY, USA, 2024; pp. 5556–5565. [CrossRef]
72. Sidhu, J.S.; Bentaleb, A. LCR360: Efficient Head Movement Prediction and Viewport Sharing in 360° Video Streaming. In *Proceedings of the International Conference on Mobile Computing and Networking*; ACM MobiCom '24; ACM: New York, NY, USA, 2024; pp. 2084–2090. [CrossRef]
73. Pan, Y.; Jiang, X.; Quan, W.; Zhang, L. VMP360: Adaptive 360° Video Streaming Based on Optical Flow Estimated QoE. In *Proceedings of the Workshop on Quality of Experience (QoE) in Visual Multimedia Applications*; QoEVMA'20; ACM: New York, NY, USA, 2020; pp. 11–18. [CrossRef]
74. Zeynali, A.; Hajiesmaili, M.H.; Sitaraman, R.K. BOLA360: Near-optimal View and Bitrate Adaptation for 360-degree Video Streaming. In *Proceedings of the ACM Multimedia Systems Conference (MMSys)*; MMSys '24; ACM: New York, NY, USA, 2024; pp. 12–22. [CrossRef]
75. Madarasingha, C.; Thilakarathna, K. VASTile: Viewport Adaptive Scalable 360-Degree Video Frame Tiling. In *Proceedings of the ACM International Conference on Multimedia*; MM '21; ACM: New York, NY, USA, 2021; pp. 4555–4563. [CrossRef]
76. Dharmasiri, A.; Kattadige, C.; Zhang, V.; Thilakarathna, K. Viewport-aware dynamic 360° video segment categorization. In *Proceedings of the ACM Workshop on Network and Operating Systems Support for Digital Audio and Video*; NOSSDAV '21; ACM: New York, NY, USA, 2021; pp. 114–121. [CrossRef]
77. Adhuran, J.; Martini, M.G. Efficient viewport prediction and tiling schemes for 360 degree video streaming. In *Proceedings of the ACM Multimedia Systems Conference (MMSys)*; MMSys '24; ACM: New York, NY, USA, 2024; pp. 374–380. [CrossRef]
78. Xie, L.; Xu, Z.; Ban, Y.; Zhang, X.; Guo, Z. 360ProbDASH: Improving QoE of 360 Video Streaming Using Tile-Based HTTP Adaptive Streaming. In *Proceedings of the ACM International Conference on Multimedia*; MM '17; ACM: New York, NY, USA, 2017; pp. 315–323. [CrossRef]
79. Zhao, L.; Cui, Y.; Liu, Z.; Zhang, Y.; Yang, S. Adaptive Streaming of 360 Videos With Perfect, Imperfect, and Unknown FoV Viewing Probabilities in Wireless Networks. *IEEE Trans. Image Process.* **2021**, *30*, 7744–7759. [CrossRef] [PubMed]
80. Nguyen, D.V.; Tran, H.T.T.; Thang, T.C. Adaptive Tiling Selection for Viewport Adaptive Streaming of 360-degree Video. *IEICE Trans. Inf. Syst.* **2019**, *E102.D*, 48–51. [CrossRef]
81. Yaqoob, A.; Muntean, G.M. A Weighted Tile-based Approach for Viewport Adaptive 360° Video Streaming. In *Proceedings of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*; IEEE: Piscataway, NJ, USA, 2020; pp. 1–7. [CrossRef]
82. Zhang, L.; Suo, Y.; Wu, X.; Wang, F.; Chen, Y.; Cui, L.; Liu, J.; Ming, Z. TBRA: Tiling and Bitrate Adaptation for Mobile 360-Degree Video Streaming. In *Proceedings of the ACM International Conference on Multimedia*; MM '21; ACM: New York, NY, USA, 2021; pp. 4007–4015. [CrossRef]
83. Hooft, J.V.d.; Vega, M.T.; Petrangeli, S.; Wauters, T.; Turck, F.D. Tile-based Adaptive Streaming for Virtual Reality Video. *ACM Trans. Multimed. Comput. Commun. Appl.* **2019**, *15*, 1–24. [CrossRef]
84. TRAN, H.T.T.; NGUYEN, D.V.; NGOC, N.P.; THANG, T.C. A Tile-Based Solution Using Cubemap for Viewport-Adaptive 360-degree Video Delivery. *IEICE Trans. Commun.* **2019**, *E102.B*, 1292–1300. [CrossRef]
85. Su, Q.; Huang, J.; Li, W.; Zhong, P.; Gao, F.; Li, Z.; Liu, J.; Zhou, S.; Lu, B.; Song, Z.; et al. Variable-Sized Tiling and Rate Allocation for 360° Video Streaming. *IEEE Trans. Veh. Technol.* **2025**, *74*, 6979–6990. [CrossRef]

86. Afzal, S.; Chen, J.; Ramakrishnan, K.K. Characterization of 360-degree Videos. In *Proceedings of the Workshop on Virtual Reality and Augmented Reality Network*; VR/AR Network '17; ACM: New York, NY, USA, 2017; pp. 1–6. [CrossRef]
87. Hannuksela, M.M.; Wang, Y.K.; Hourunranta, A. An Overview of the OMAF Standard for 360° Video. In *Proceedings of the Data Compression Conference (DCC)*; IEEE: Piscataway, NJ, USA, 2019; pp. 418–427. [CrossRef]
88. Der Hooft, J.v.; Torres Vega, M.; Petrangeli, S.; Wauters, T.; Turck, F.D. Optimizing Adaptive Tile-Based Virtual Reality Video Streaming. In *Proceedings of the IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*; IEEE: Piscataway, NJ, USA, 2019; pp. 381–387.
89. Vega, M.T.; van der Hooft, J.; Heyse, J.; Backere, F.D.; Wauters, T.; Turck, F.D.; Petrangeli, S. Exploring New York in 8K: An Adaptive Tile-based Virtual Reality Video Streaming Experience. In *Proceedings of the ACM Multimedia Systems Conference (MMSys)*; MMSys '19; ACM: New York, NY, USA, 2019; pp. 330–333. [CrossRef]
90. Huang, J.; Liu, M.; Liu, J.; Gao, F.; Li, W.; Wang, J. Tile-size aware bitrate allocation for adaptive 360° video streaming. *Multimed. Tools Appl.* **2025**, *84*, 12615–12632. [CrossRef]
91. Park, S.; Bhattacharya, A.; Yang, Z.; Dasari, M.; Das, S.R.; Samaras, D. Advancing User Quality of Experience in 360-degree Video Streaming. In *Proceedings of the IFIP Networking Conference (IFIP Networking)*; IEEE: Piscataway, NJ, USA, 2019; pp. 1–9. [CrossRef]
92. Yadav, P.K.; Ooi, W.T. Tile Rate Allocation for 360-Degree Tiled Adaptive Video Streaming. In *Proceedings of the ACM International Conference on Multimedia*; MM '20; ACM: New York, NY, USA, 2020; pp. 3724–3733. [CrossRef]
93. Yu, L.; Pang, Z.; Yao, C.; Zhao, Y. VATP360: Viewport-Adaptive 360-Degree Video Streaming Based on Tile Priority. *Neural Comput. Appl.* **2026**, *38*, 78. [CrossRef]
94. Nguyen, D.V.; Tran, H.T.T.; Pham, A.T.; Thang, T.C. An Optimal Tile-Based Approach for Viewport-Adaptive 360-Degree Video Streaming. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2019**, *9*, 29–42. [CrossRef]
95. Qian, F.; Han, B.; Xiao, Q.; Gopalakrishnan, V. Flare: Practical Viewport-Adaptive 360-Degree Video Streaming for Mobile Devices. In *Proceedings of the Annual International Conference on Mobile Computing and Networking*; MobiCom '18; ACM: New York, NY, USA, 2018; pp. 99–114. [CrossRef]
96. Murthy Naik, S.K.; Wang, M.; Krishnamurthy, D. TYLE: Tile-based Dynamic Quality Enhancement for 360-degree Video Streaming. In *Proceedings of the IEEE International Performance, Computing, and Communications Conference (IPCCC)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–8. [CrossRef]
97. Zeng, Q.; Yin, Z.; Pan, Q.; Ma, Y.; Yu, Y.; Zhuang, Y.; Liang, J.; Yang, S.; Xiao, H. TVSR-OR: Tile-based 360-degree video streaming over real time streaming protocol with optimized read. *Trans. Emerg. Telecommun. Technol.* **2023**, *34*, e4759. [CrossRef]
98. Yuan, X.; Pu, L.; Yun, R.; Xu, J. Tile-based Multi-source Adaptive Streaming for 360-degree Ultra-High-Definition Videos. In *Proceedings of the International Conference on Mobility, Sensing and Networking (MSN)*; IEEE: Piscataway, NJ, USA, 2020; pp. 441–448. [CrossRef]
99. Kumar, S.; Franklin A, A.; Jin, J.; Dong, Y.N. Seer: Learning-Based 360° Video Streaming for MEC-Equipped Cellular Networks. *IEEE Trans. Netw. Sci. Eng.* **2023**, *10*, 3308–3319. [CrossRef]
100. Ordonez-Ante, L.; van der Hooft, J.; Wauters, T.; Seghbroeck, G.V.; Volckaert, B.; Turck, F.D. Explora-VR: Content Prefetching for Tile-Based Immersive Video Streaming Applications. *J. Netw. Syst. Manag.* **2022**, *30*, 38. [CrossRef]
101. Yu, Z.; Liu, J.; Wang, C.; Yang, Q. Bandit Learning-Based Edge Caching for 360-Degree Video Streaming With Switching Cost. *IEEE Access* **2022**, *10*, 80714–80726. [CrossRef]
102. Yin, G.; Zhang, H.; Li, D.; Zhou, X. FoV Prediction-Based Adaptive Bitrate Streaming with On-Demand Transcoding for 360-Degree Videos. *IEEE Trans. Circuits Syst. Video Technol.* **2025**, *36*, 5581–5594. [CrossRef]
103. Guo, J.; Li, S.; Zhu, J.; Li, X.; Sun, B.; Feng, W. Adaptive Transmission Strategy for Non-Uniform Coding of 360° Videos. *Electronics* **2024**, *13*, 3266. [CrossRef]
104. Zhang, Y.; Wang, Z.; Liu, J.; Du, H.; Zheng, Q.; Zhang, W. Deep Reinforcement Learning Based Adaptive 360-degree Video Streaming with Field of View Joint Prediction. In *Proceedings of the IEEE Symposium on Computers and Communications (ISCC)*; IEEE: Piscataway, NJ, USA, 2022; pp. 1–8. [CrossRef]
105. Hosseini, M.; Swaminathan, V. Adaptive 360 VR Video Streaming: Divide and Conquer. In *Proceedings of the IEEE International Symposium on Multimedia (ISM)*; IEEE: Piscataway, NJ, USA, 2016; pp. 107–110. [CrossRef]
106. Chen, X.; Wu, D.; Ahmad, I. Optimized viewport-adaptive 360-degree video streaming. *CAAI Trans. Intell. Technol.* **2021**, *6*, 347–359. [CrossRef]
107. Valiandi, I.; Pattichis, M.S.; Kyriacou, E.; Panayides, A.S. Adaptive Strategies for Enhancing Quality of Experience in 360° Video Streaming. In *Proceedings of the Asilomar Conference on Signals, Systems, and Computers*; IEEE: Piscataway, NJ, USA, 2024; pp. 945–949. [CrossRef]
108. Iyengar, J.; Swett, I. *QUIC Loss Detection and Congestion Control*; RFC 9002; Internet Engineering Task Force: Fremont, CA, USA, 2021. [CrossRef]

109. Ram, L.K.; Gupta, A.; Chakraborty, S. Impact of Congestion Control Algorithms on the QoE Performance of Video Streaming over QUIC. In *Proceedings of the International Conference on Distributed Computing and Networking*; ICDCN '26; ACM: New York, NY, USA, 2026; pp. 153–157. [CrossRef]
110. Timmerer, C.; Amirpour, H.; Tashtarian, F.; Afzal, S.; Rizk, A.; Zink, M.; Hellwagner, H. HTTP Adaptive Streaming: A Review on Current Advances and Future Challenges. *ACM Trans. Multimed. Comput. Commun. Appl.* **2025**, *21*, 1–27. [CrossRef]
111. Nguyen, D.V.; Van Trung, H.; Dieu Huong, H.L.; Huong, T.T.; Ngoc, N.P.; Thang, T.C. Scalable 360 Video Streaming using HTTP/2. In *Proceedings of the IEEE International Workshop on Multimedia Signal Processing (MMSP)*; IEEE: Piscataway, NJ, USA, 2019; pp. 1–6. [CrossRef]
112. Kimura, B.; Ferlin, S.; Paiva, T.; Mahmoodi, T.; Brunstrom, A.; Alay, O. Evaluating Adaptive Video Streaming over Multipath QUIC with Shared Bottleneck Detection. *ACM Trans. Multimed. Comput. Commun. Appl.* **2025**, *21*, 1–25. [CrossRef]
113. Kalan, R.; Dulger, I. A Survey on QoE Management Schemes for HTTP Adaptive Video Streaming: Challenges, Solutions, and Opportunities. *IEEE Access* **2024**, *12*, 170803–170839. [CrossRef]
114. Wang, S.; Zhang, X.; Xiao, M.; Chiu, K.; Liu, Y. SphericRTC: A System for Content-Adaptive Real-Time 360-Degree Video Communication. In *Proceedings of the ACM International Conference on Multimedia*; MM '20; ACM: New York, NY, USA, 2020; pp. 3595–3603. [CrossRef]
115. Nguyen, M.; Nys, P.; Pham, S.; Silhavy, D.; Arbanowski, S.; Steglich, S. Streaming Face-Off: A Testbed Analysis of Media-over-QUIC and Low-Latency DASH. In *Proceedings of the ACM Multimedia Systems Conference (MMSys)*; MMSys '25; ACM: New York, NY, USA, 2025; pp. 317–321. [CrossRef]
116. Asim, R.; Subramanian, L.; Zaki, Y. Impact of Congestion Control on Mixed Reality Applications. In *Proceedings of the ACM SIGCOMM Workshop on Emerging Multimedia Systems*; EMS '24; ACM: New York, NY, USA, 2024; pp. 21–26. [CrossRef]
117. Petrangeli, S.; Swaminathan, V.; Hosseini, M.; De Turck, F. An HTTP/2-Based Adaptive Streaming Framework for 360° Virtual Reality Videos. In *Proceedings of the ACM International Conference on Multimedia*; MM '17; ACM: New York, NY, USA, 2017; pp. 306–314. [CrossRef]
118. Ben Yahia, M.; Le Louedec, Y.; Simon, G.; Nuaymi, L. HTTP/2-Based Streaming Solutions for Tiled Omnidirectional Videos. In *Proceedings of the IEEE International Symposium on Multimedia (ISM)*; IEEE: Piscataway, NJ, USA, 2018; pp. 89–96. [CrossRef]
119. Huang, S.; Wang, Z.; Zhang, W.; Du, H.; Zheng, Q. Adaptive Video Streaming Using Dynamic Server Push over HTTP/2. In *Proceedings of the IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*; IEEE: Piscataway, NJ, USA, 2021; pp. 673–678. [CrossRef]
120. Ghaznavi-Youvalari, R.; Zare, A.; Fang, H.; Aminlou, A.; Xie, Q.; Hannuksela, M.M.; Gabbouj, M. Comparison of HEVC coding schemes for tile-based viewport-adaptive streaming of omnidirectional video. In *Proceedings of the IEEE International Workshop on Multimedia Signal Processing (MMSP)*; IEEE: Piscataway, NJ, USA, 2017; pp. 1–6. [CrossRef]
121. Nguyen, M.; Nguyen, D.H.; Pham, C.T.; Ngoc, N.P.; Nguyen, D.V.; Thang, T.C. An adaptive streaming method of 360 videos over HTTP/2 protocol. In *Proceedings of the NAFOSTED Conference on Information and Computer Science*; IEEE: Piscataway, NJ, USA, 2017; pp. 302–307. [CrossRef]
122. Bi, S.; Chen, H.; Li, X.; Wang, S.; Lin, X.H. Double-Agent Deep Reinforcement Learning for Adaptive 360-Degree Video Streaming in Mobile Edge Computing Networks. In *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–7. [CrossRef]
123. Xu, L.; Ha, S.; Rhee, I.; Goel, V.; Eggert, L. *CUBIC for Fast and Long-Distance Networks*; RFC 9438; Internet Engineering Task Force: Fremont, CA, USA, 2023. [CrossRef]
124. Zeynali, D.; Weyulu, E.N.; Fathalli, S.; Chandrasekaran, B.; Feldmann, A. Promises and Potential of BBRv3. In *Proceedings of the Passive and Active Measurement*; Springer: Cham, Switzerland, 2024; pp. 249–272. [CrossRef]
125. Yen, S.C.; Fan, C.L.; Hsu, C.H. Streaming 360° Videos to Head-Mounted Virtual Reality Using DASH over QUIC Transport Protocol. In *Proceedings of the ACM Workshop on Packet Video*; PV '19; ACM: New York, NY, USA, 2019; pp. 7–12. [CrossRef]
126. Ravuri, H.K.; Vega, M.T.; Van Hooft, J.D.; Wauters, T.; De Turck, F. Adaptive Partially Reliable Delivery of Immersive Media Over QUIC-HTTP/3. *IEEE Access* **2023**, *11*, 38094–38111. [CrossRef]
127. Wei, Y.; Ye, Q.; Qu, K.; Zhuang, W.; Shen, X. Customized Transmission Protocol for Tile-Based 360° VR Video Streaming Over Core Network Slices. *IEEE Trans. Netw.* **2025**, *33*, 340–354. [CrossRef]
128. Rosa, F.; Ferlin, S.; Brunstrom, A.; Kimura, B. End-to-End 360° Video Streaming over HTTP/3: Architecture and Implementation. In *Proceedings of the Applied Networking Research Workshop*; ANRW '25; ACM: New York, NY, USA, 2025; pp. 9–16. [CrossRef]
129. Park, S.; Das, S.R. Cross-Layer Scheduling in QUIC and Multipath QUIC for 360-Degree Video Streaming. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1–6. [CrossRef]
130. Briscoe, B.; De Schepper, K.; Bagnulo, M.; White, G. *Low Latency, Low Loss, and Scalable Throughput (L4S) Internet Service: Architecture*; RFC 9330; Internet Engineering Task Force: Fremont, CA, USA, 2023. [CrossRef]
131. De Schepper, K.; Briscoe, B. *The Explicit Congestion Notification (ECN) Protocol for Low Latency, Low Loss, and Scalable Throughput (L4S)*; RFC 9331; Internet Engineering Task Force: Fremont, CA, USA, 2023. [CrossRef]

132. De Schepper, K.; Briscoe, B.; White, G. *Dual-Queue Coupled Active Queue Management (AQM) for Low Latency, Low Loss, and Scalable Throughput (L4S)*; RFC 9332; Internet Engineering Task Force: Fremont, CA, USA, 2023. [CrossRef]
133. Mascolo, S.; Balillo, A.V.; Manfredi, G.; D'Agostino, D.; De Cicco, L. QUIC Delay Control: An Implementation of Congestion and Delay Control. In *Proceedings of the International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*; IEEE: Piscataway, NJ, USA, 2025; pp. 152–157. [CrossRef]
134. Sidhu, J.S.; Bentaleb, A. Video Streaming Over QUIC: A Comprehensive Study. *ACM Trans. Multimed. Comput. Commun. Appl.* **2026**, *22*, 1–29. [CrossRef]
135. Rosa, F.; Ferlin, S.; Brunstrom, A.; da Costa, J.B.D.; Kimura, B. Enhancing 360° Video Streaming with Stream Scheduling Policies over HTTP/3. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*; IEEE: Piscataway, NJ, USA, 2026; pp. 1–6. [CrossRef]
136. Zeng, Q.; Yin, Z.; Yu, Y.; Zhuang, Y. A New Architecture of 8K VR FOV Video End-to-End Technology. In *Proceedings of the International Wireless Communications and Mobile Computing (IWCMC)*; IEEE: Piscataway, NJ, USA, 2022; pp. 148–154. [CrossRef]
137. Singhal, C. I2MB: Intelligent Immersive Multimedia Broadcast in Next-Generation Cellular Networks. *IEEE Access* **2022**, *10*, 98882–98895. [CrossRef]
138. Diallo, B.; Ouamri, A.; Keche, M. A Hybrid Approach for WebRTC Video Streaming on Resource-Constrained Devices. *Electronics* **2023**, *12*, 3775. [CrossRef]
139. Jagmagji, A.S.; Zubaydi, H.D.; Molnár, S. Exploration and Evaluation of Self-Clocked Rate Adaptation for Multimedia (SCReAM) Congestion Control Algorithm in 5G Networks. In *Proceedings of the International Conference on Telecommunications and Signal Processing (TSP)*; IEEE: Piscataway, NJ, USA, 2022; pp. 230–237. [CrossRef]