

360-Degree Video Bandwidth Reduction Techniques

Subjects: [Computer Science](#), [Software Engineering](#)

Contributor: En Sing Wong , Nur Haliza Abdul Wahab , Faisal Saeed , Nouf Alharbi

The usage of 360-degree videos has prevailed in various sectors such as education, real estate, medical, entertainment and more. However, various challenges are faced to provide real-time streaming due to the nature of high-resolution 360-degree videos such as high bandwidth requirement, high computing power and low delay tolerance. To overcome these challenges, streaming methods such as dynamic adaptive streaming over HTTP (DASH), tiling, viewport-adaptive and machine learning (ML) are discussed.

Virtual Reality (VR)

360-degree video

bandwidth reduction

metaverse

DASH

tiling

viewport-adaptive

1. Introduction

A 360-degree video is a video filmed in all directions by an omnidirectional camera or numerous cameras simultaneously, encompassing a whole 360-degree 3D sphere view, hence creating a Virtual Reality (VR) environment. When played back on a 2D flat screen (mobile or computer), viewers may alter the viewing direction and view the film from whichever angle they like, similar to a panorama. It can also be played on a display like a head-mounted display or projectors organized in the shape of a sphere or a portion of a sphere. The potential of 360-degree video and VR is enormous. The development of VR, AR and 360-degree video could be seen in education, real estate, medical, economics, and more.

The superiorities of 360-degree video can be concluded as: (a) Boost interest and creativity in education; (b) Generate various business and job opportunities in Metaverse; (c) Providing a virtual communication platform highly similar to face-to-face interaction; (d) Enabling a supreme experience in entertainment: games, concerts, etc.

Although lots of benefits can be listed on 360-degree video, there are a few problems such as lack of tools and network barriers. Due to the extremely high bandwidth demands, providing a great Quality of Experience (QoE) to viewers while streaming 360 videos over the Internet is particularly difficult. Both academics and businesses are currently looking for more effective ways to bridge the gap between the user experience of VR apps and the VR networking issues such as high bandwidth requirements.

Four categories of solutions proposed by various research are Dynamic adaptive HTTP streaming (DASH), tiling, viewport-adaptive, and Machine learning (ML), as illustrated in **Figure 1**.

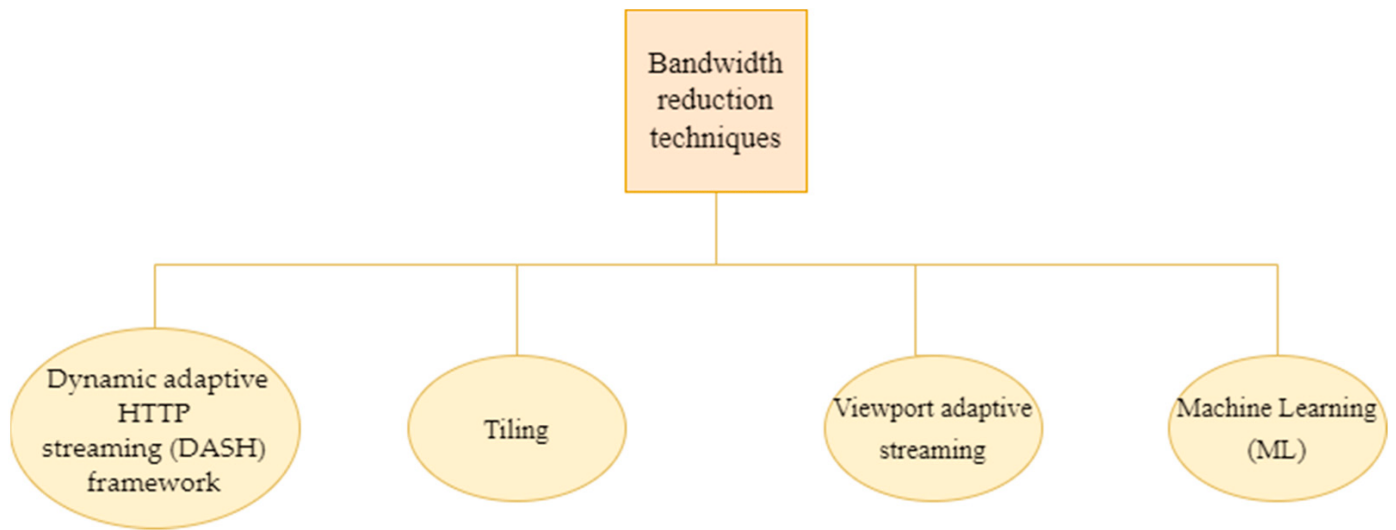


Figure 1. Bandwidth reduction techniques.

2. Dynamic Adaptive HTTP Streaming (DASH) Framework

Dynamic adaptive HTTP streaming (DASH) is an MPEG standard that provides a multimedia style and specification for sending material over HTTP using an adjustable bitrate method [1]. DASH is extremely compatible with the existing internet infrastructure due to its minimal processing burden and transparency to middleboxes, and the ability to apply alternative adaption methods makes it adaptable to diverse network conditions standard is generally extensively utilized for two-dimensional video streaming over the world wide web recently. DASH streaming works by splitting videos into short segments, each segment on the DASH server maintains a number of video streams with varying bitrates [2]. By requesting the proper HTTP resource, based on the view on the streaming client, the main viewpoint segment stream with higher resolution and the other viewpoint segment stream with lower resolution. A video player can switch from one quality level to another in the middle of the video playback without interruption. **Table 1** demonstrates the major steps in the DASH streaming process:

Table 1. Major steps in the DASH streaming process.

Step	Process
Stitching	Stitch videos collected by many cameras/an omnidirectional camera onto diverse planar models such as cubic and affine transformation models match up the various camera images, merging and distorting the views to a sphere's surface [3]. For successful coding and transmission, the 360-degree sphere is projected to a 2D planar format such as Cubic Mapping Projection (CMP) and Equirectangular Projection (ERP).
Encoding and segmentation	The video file is segmented into smaller parts of a few seconds in length by the origin server. Each section is encoded in numerous bitrate or quality level variants.

Step	Process
Delivery	The encoded video segments are sent out to client devices over a content delivery network (CDN).
Decoding, rendering and play	Decodes the streamed data. With adaptive bitrate streaming, it plays the video and automatically adjusts the quality of the picture according to the network condition/user's views at the client device.

Another extension of DASH or other streaming systems is the Omnidirectional Media Format (OMAF) standard specifying the spatial information of video segments [4]. For the DASH OMAF scheme, storage space is sacrificed to increase the bandwidth of the VR video streaming [5]. **Figure 2** shows the technical framework of the DASH OMAF architecture network. Furthermore, OMAF specifies several requirements for users, bringing the standard specification for omnidirectional streaming one step closer to completion. Players based on OMAF have already been implemented and demonstrated [6].

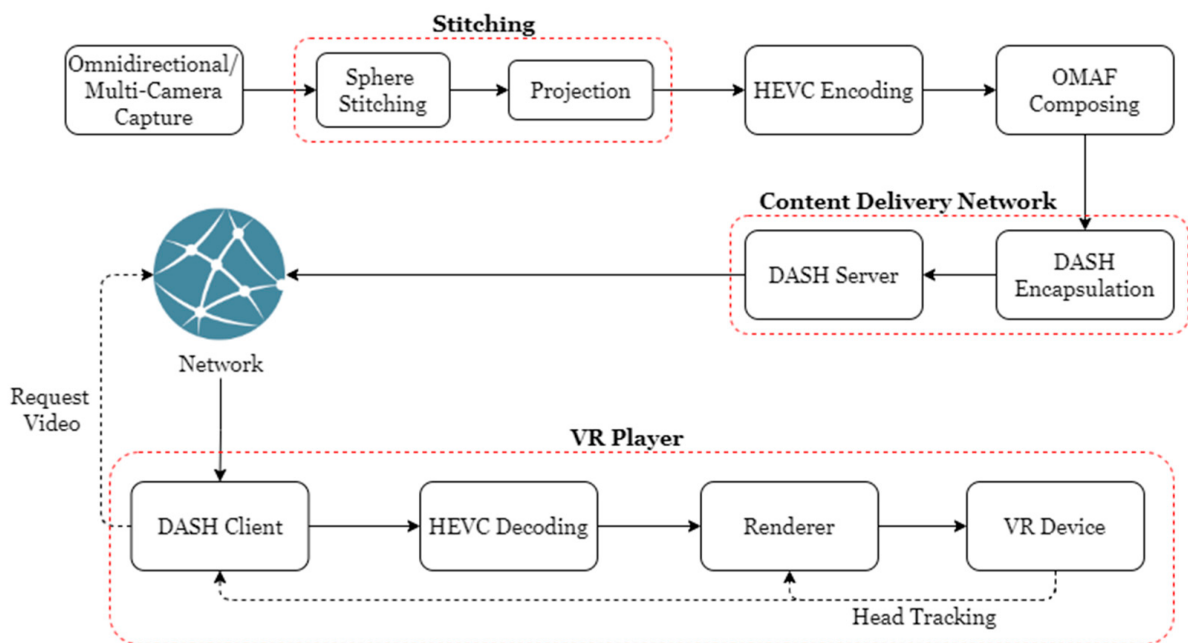


Figure 2. DASH-OMAF architecture network.

OMAF also defines tile-based streaming and Viewport-Based Streaming approaches where the Field of View (FoV) is downloaded at the highest quality possible, along with the lower quality of the other viewable region. This enables the client to download a collection of tiles with varying encoding qualities or resolutions, with the visible region prioritized to improve the quality of experience (QoE) while consuming less bandwidth.

Next, OMAF also specifies video profiles based on the High-Efficiency Video Coding (HEVC) coding standard, as well as HEVC-based or older Advanced Video Coding (AVC), AVC-based viewport-dependent profiles that support Equirectangular Projection (ERP), Cubic Mapping Projection (CMP), and tile-based streaming [7]. The comparison of ERP and CMP is shown in **Figure 3**.



Equirectangular



CubeMap

Figure 3. Equirectangular projection (EMP) and cube map projection (CMP) comparison.

Clients can stream omnidirectional video from a DASH SRD or OMAF compliant server. The server will deliver segments with different viewport-dependent projections or independent tiles based on the choices of the client. The client then downloads the appropriate segments, potentially discarding low viewing probability segments or downloads with lower quality to save bandwidth. Next, the features of HEVC of fast Field of View (FoV) switching allow the client to request the segments based on users' head movements in high quality [8], users can even zoom into the region of interest within the 360-degree video [9], providing a smooth user experience with minimal server-side changes.

In recent years, some researchers have enhanced the Quality of Experience (QoE) of 360 videos streaming with the DASH architecture [10]. At any one point in a VR 360-degree movie, the user can at most see a portion of the 360-degree film. As a result, sending the entire picture wastes bandwidth and processing power. With the DASH-based viewpoint of adaptive transmission, these problems may be resolved. The client must pre-download the video material to ensure seamless playing, which needs the client predicting the user's future viewpoint.

Based on HTTP 2.0, a real-time video streaming technology with low latency has been developed by Huang, Ding [11]. The MPEG DASH prototype implements HTTP 2.0 server push functionality to actively deliver live video from the server to the client with low latency whereas Nguyen, Tran [12] suggested an efficient adaptive VR video stream approach based on the DASH transport architecture via HTTP/2 that implements stream prioritization and stream termination.

3. Tiling

Tiling is one of the typical solutions proposed by various researchers in order to overcome the bandwidth issues of 360-degree videos by projecting and splitting video frames into numerous sections known as tiles. In general, this technique divides a frame into several sections known as tiles, focusing on the quality of the Region of interest (RoI)/Quality Emphasis Region (QER)/Field of View (FoV) while reducing the others to overcome the bandwidth issue. Most of the solutions are based on the DASH framework as discussed earlier.

Figure 4 illustrates the small region of FoV in an equirectangular mapped 2K picture. Following that, the most popular HMDs have a small FoV. For example, Google Cardboard [13] and Samsung Gear VR [14] have an FoV of 100 degrees whereas Oculus Rift and HTC Vive [15] have wider 110 degrees of FoV as demonstrated in **Figure 5**.

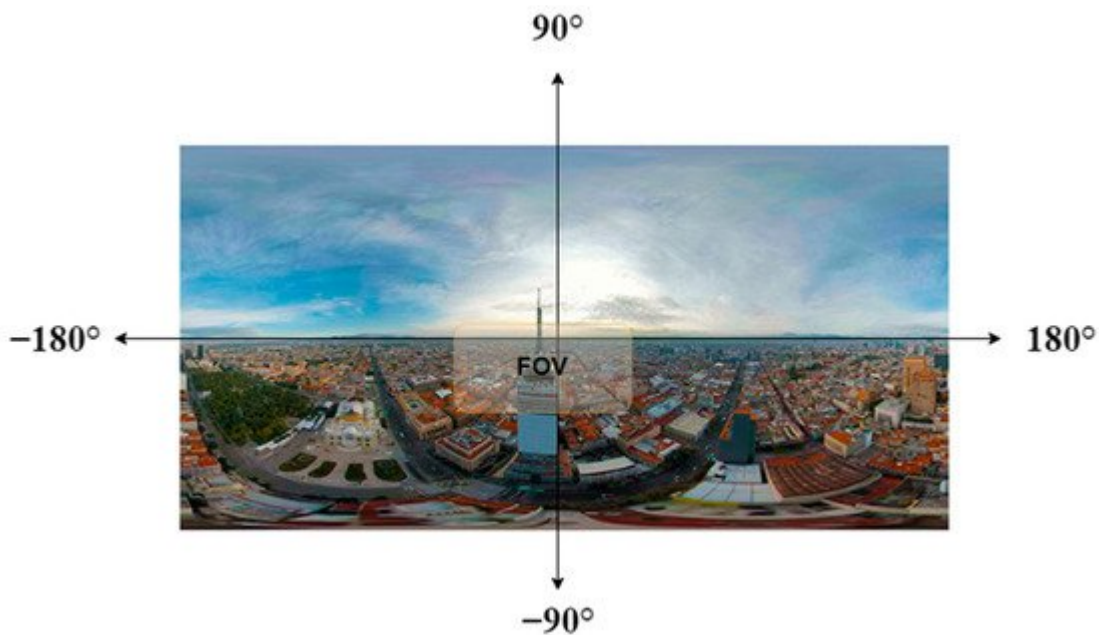


Figure 4. FoV in a full 360-degree video frame.

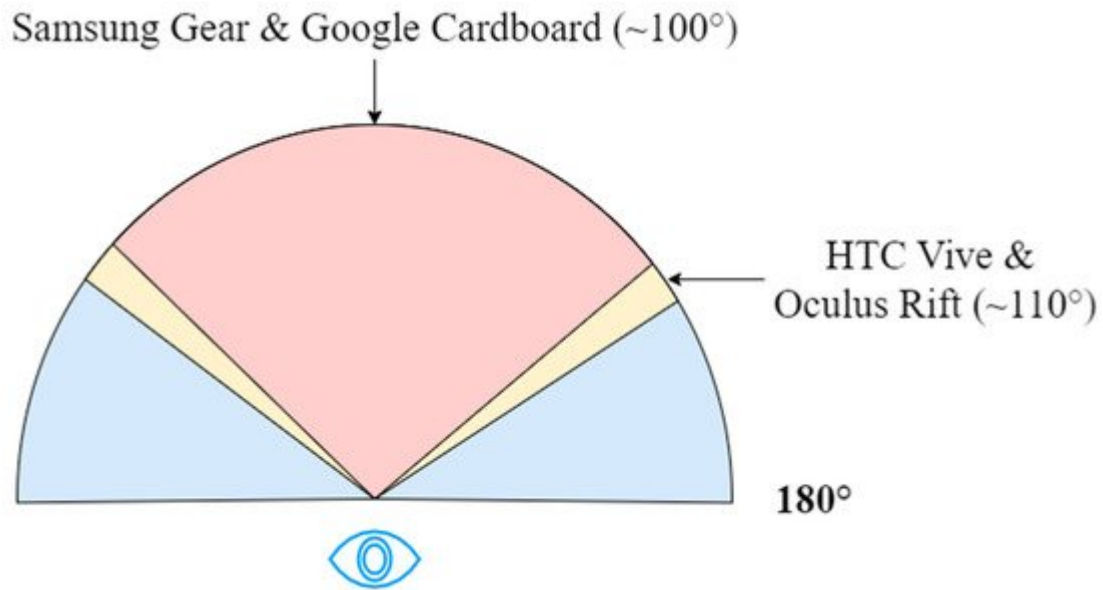


Figure 5. FoV associated with the human eye.

Figure 6 shows the methods using the tiling technique whereas **Table 2** summarizes and compared the characteristics of each tiling scheme.

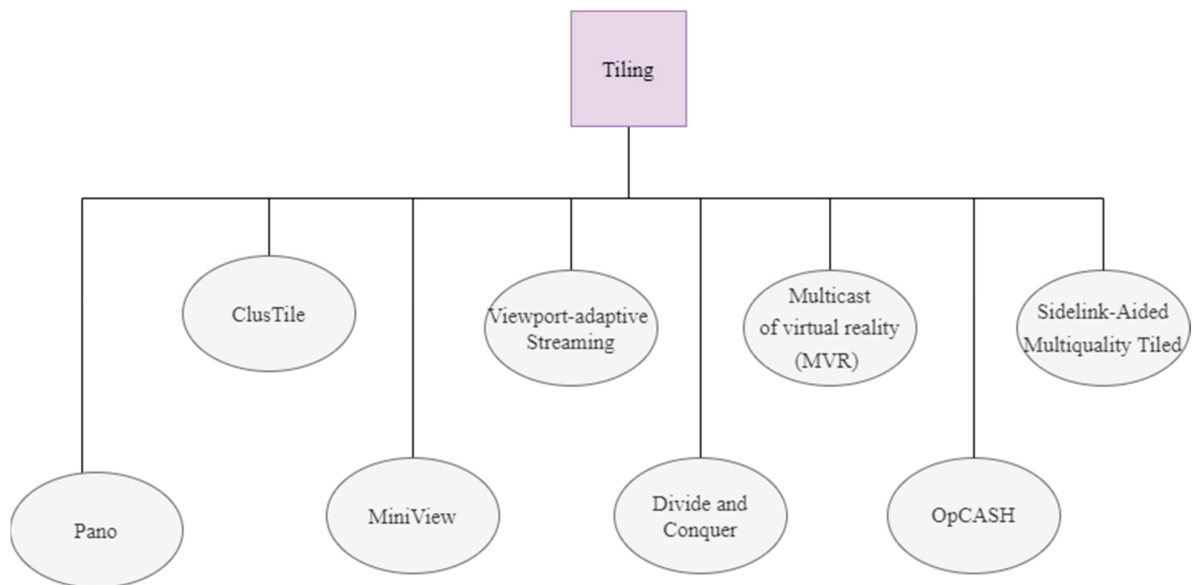


Figure 6. Methods using the tiling technique.

3.1. ClusTile

Research as Zhou, Xiao [\[16\]](#) proposed ClusTile, a tiling approach that schemes each tile represents a DASH segment covering a portion of the 360-degree view with typically fixed time intervals, formulated by solving the set of integer linear programs (ILPs). Although this work mentions a decrease of such a high percentage in bandwidth

reduction (76%), it does not allow varying the solution of representations but only their bitrate. The increasing number of tiles in the process is not sufficient for the segments downloaded and uploaded.

3.2. PANO

Guan, Zheng [17] propose a quality model named Pano for 360° videos that capture the factors that affect the QoE of 360° video including difference in depth-of-field (DoF), relative viewpoint-moving speed and change in scene luminance. The proposed tiling scheme with variable-sized tiles aims to find the tradeoff between the video quality and efficiency of video encoding. Pano achieves 41–46% less bandwidth consumption than Zhou, Xiao [16] with the same Peak Signal-to-Perceptible-Noise Ratio (PSPNR) [17].

3.3. MiniView Layout

To reduce the bandwidth requirement of 360-degree video streaming, Xiao, Wang [18] proposed the MiniView Layout which has saved up to 16% of the encoded video without downgrading the visual qualities. In this method, the video was projected into equalized tiles with each MiniView independently encoded into segments. It increases the number of segments and higher in the number of requests parallelly to the streaming client. Plus, Ref. [18] showed improvements in projection efficiency as it created a set of views with the rectilinear projection referred to as “miniview”, which has smaller FOVs than cube faces, hence able to save encoded 360-degree videos’ storage size without quality loss. Each miniview has its parameters which include FOV, orientation and pixel density [18].

3.4. Viewport Adaptive Streaming

In [19], The adaption algorithm initially chooses the video’s Quality Emphasized Region (QER) based on the viewport center and the Quality Emphasis Center (QEC) of the available QERs. Each QER-based video is composed of a pre-processed collection of tile representations that are then encoded at various quality levels. This allows for faster server maintenance (fewer files, resulting in a smaller media presentation description (MPD) file), a simpler selection procedure for the client (through a distance computation), and no need to reconstruct the video prior to viewport extraction. However, improved adaption algorithms are required to predict head movement, as well as a new video encoding approach to do quality-differentiated encoding for high-resolution videos.

3.5. Divide and Conquer

Research by Hosseini and Swaminathan [20] proposed a divide and conquer approach to increase the bandwidth efficiency of the 360 VR video streaming system. The hierarchical resolution degrading enables a seamless video quality-switching process hence providing a better user experience. Compared to the other method which uses equirectangular projection [20], implements hexaface sphere projection as illustrated in ([20] Figure 4), and significantly saved 72% bandwidth compared to other tiling approaches without viewport awareness. To improve the performance of this approach, an adaptive rate allocation method for tile streaming based on available bandwidth is needed.

3.6. Multicast Virtual Reality (MVR)

In [21], the Multicast Virtual Reality (MVR) streaming technique, which is a basic rate adaptation mechanism, serves all members in a multicast group with the same data rate to ensure that all members can receive the video. The data rate is selected based on the member with the poorest network conditions. However, a better tile weighting technique with data-driven probabilistic and an improved rate adaption algorithm is required to improve the user experience.

3.7. Sidelink-Aided Multiquality Tiled

Dai, Yue [22] adapt sidelink is a modification of the basic LTE standard that enables device-to-device (D2D) communication in 360-degree streaming without the use of a base station. Allocate tile weight based on long-term weight (how often the tile was visited) and short-term weight (tile distance from the FOV). To find suboptimal solutions with minimal computational cost, a two-stage optimization technique is used to pick sidelink Receivers and Senders in stage 1 and allocate bandwidth and select tile quality level in stage 2.

3.8. OpCASH

In [23], a tiling scheme with variable-sized tiles is proposed. To deliver optimal cached tile coverage to user viewports (VP), Mobile Edge Computing (MEC) cache usage is used. Next, an ILP-based technique is used to determine the best cache tile configuration to decrease the redundancy of stored variable tiles at a MEC server while limiting queries to faraway servers, lowering delivery delay, and increasing cache utilization. OpCASH successfully reduces data fetched from content servers by 85% and overall content delivery time by 74% with MEC.

Table 2. Comparison of existing tiling approaches.

Source	Technique	Result	Limitation
[16]	<ul style="list-style-type: none"> Dynamic adaptive HTTP streaming (DASH). Integer linear programs (ILPs). Artificial Neural Networks (ANN). 	<ul style="list-style-type: none"> Saved 76% bandwidth in comparison to the non-tiling scheme. Saved 52% downloaded volume in comparison to fixed tiling schemes. 	A fixed tiling scheme requires tile selection algorithms.
[19]	<ul style="list-style-type: none"> Dynamic adaptive HTTP streaming (DASH). 	<ul style="list-style-type: none"> Enable High-quality service with high interactivity with 	Improved adaption algorithms are required to predict head movement, as well as a new video encoding approach to do

Source	Technique	Result	Limitation
	<ul style="list-style-type: none"> An adaptation algorithm first chooses the Quality Emphasized Region (QER) of the viewport's video-based left and the Quality Emphasis Center (QEC) of the available QERs. 	HMD with low management required from VR providers.	quality-differentiated encoding for high-resolution videos.
[21]	<ul style="list-style-type: none"> Dynamic adaptive HTTP streaming (DASH). Heuristic algorithm: Multicast Virtual Reality streaming algorithm (MVR). 	<ul style="list-style-type: none"> Increased video bitrates ($\leq 46\%$) for video tiles in users' viewports. 	Require a better tile weighting approach with data-driven probabilistic as well as an improved rate adaption algorithm.
[20]	<ul style="list-style-type: none"> MPEG-DASH SRD. hierarchical resolution degrading hexaface sphere projection. 	72% bandwidth savings.	Improve performance with an adaptive rate allocation method for tile streaming based on available bandwidth.
[17]	<ul style="list-style-type: none"> Variable-sized tiling scheme. A new quality model for 360° videos captures the factors that affect the QoE of 360° video including Difference in depth-of-field (DoF), Relative viewpoint-moving speed and Change in scene luminance. 	The same PSPNR was obtained with 41–46 percent reduced bandwidth consumption than [16].	The 360JND model is based on the results of a survey in which the values of 360° video-specific characteristics were varied individually.

Source	Technique	Result	Limitation
[18]	<ul style="list-style-type: none"> Higher sphere-to-2D projection efficiency. The ffmpeg360 program transcodes 360-degree videos and assesses the quality of 360-degree videos based on user head movement patterns. Created collection of views with the rectilinear projection referred to as “miniview”, which have smaller FOVs than cube faces, hence able to save encoded 360-degree videos’ storage size while maintaining the quality. 	Saved up to 16% encoded video size without much quality loss.	Fixed tiles, each miniview might well be encoded into segments individually, and the streaming client could request these segments as needed.
[22]	<ul style="list-style-type: none"> Adapt sidelink. Weighted tile allocation. Two-stage optimization technique. 	Dai, Yue [22] formulated optimization problems based on the interaction between tile quality level selection, sidelink sender selection, and bandwidth allocation to optimize the overall utility of all users.	When the number of groups is increased from 10 to 50, the tile quality degrades because less bandwidth can be provided to each group as the number of groups grows.
[23]	<ul style="list-style-type: none"> Variable-sized tiling scheme. Adapt MEC cache usage. ILP-based technique for determining the best cache tile configuration on the MEC server. 	OpCASH obtained more than 95 percent VP coverage from cache after only 24 views of the video. When compared to a baseline that illustrates standard tile-based caching, OpCASH reduces data fetched from content servers by 85% and overall content delivery time by 74%.	Improve real-time tile encoding features on content servers by including tile quality selection in the ILP formulation and increasing the variable quality level tiles streaming in. Next, in a lab scenario, interact with many edge nodes using real-world user testing to achieve the biggest benefit at the edge layer.

transmission efficiency could be improved by identifying and transmitting the current viewport content and the predicted viewport corresponding to the head movement of users. Similar to the tiling technique in the previous section, the server contains a number of video representations that range not just in bitrate but also in the quality of various scene areas. Then, the region of the viewport is dynamically selected and streams in the best quality while

the other regions are in lower quality or not being delivered at all to reduce the bandwidth transmission. In other words, the highest bitrate is assigned to tiles in users' viewports, while some other tiles possess bitrates that are proportionate to the likelihood that users may switch viewports, which is also similar to DASH. However, the number of adaption variants of the same content increases dramatically to smooth the viewport-switching due to the sudden head movements. As a result, storage is sacrificed, and the transmission rate increases.

Ribezzo, De Cicco [24] proposed a DASH 360° Immersive Video Streaming Control System which consists of control logic with two cooperating components: quality selection algorithm (QSA) and view selection algorithm (VSA) to dynamically select the demanded video segment. The QSA functions similarly to traditional DASH adaptive video streaming algorithms whereas VSA aims to identify the proper view representation based on the current head position of the users. Ref. [24] reduced segments bitrate around 20% with improved visual quality. In [19], the adaptation algorithm first selects the Quality Emphasized Region (QER) of the video based on the viewport center and the Quality Emphasis Center (QEC) of the available QERs, hence providing high interactive service to head-mounted device (HMD) users with low management. However, improved adaption algorithms are required to predict head movement, as well as a new video encoding approach to do quality-differentiated encoding for high-resolution videos.

High responsiveness and processing power are required to adapt to rapid changes in viewports and viewport prediction to ensure smooth viewport switching with accurate prediction. Many viewport prediction approaches have been developed to cover the demands, such as historical data-driven probabilistic, popularity-based, deep content analysis, and so on as summarized in **Table 3**.

Table 3. Viewport prediction scheme of the viewport adaptive streaming approach.

Source	Viewport Prediction Scheme	Descriptions
[25]	Historical viewport movement	Prediction with Linear Regression (LR) and Ridge Regression (RR) using viewing data collected from 130 users.
[26]	Cross-user similarity	Cross-Users Behaviors (named CUB360) based on k-NN and LR take into account both the user's specific information and cross-user behavior information to forecast future viewports.
[27]	Popularity-based model	Predict based on the popularity of the tiles where they are visited with a higher frequency at a certain time, might be due to the nature of the video like interesting content along with the evaluation of the rate-distortion curve for each tile.
[28]	Popularity-based model	Similar to [27] and provide the popularity of each shown viewport (heatmap) and rate-distortion function for each tile-representation for the interested segments periodically to clients during each downloading.
[29]	Content Analysis + Popularity	Sensor- and content-based predictive mechanisms, similar to [30] with linear regression (LR). When a transition due to insufficient bandwidth occurs, the tile

Source	Viewport Prediction Scheme	Descriptions
		popularity is solely used to determine the tile quality levels.
[31]	k-Nearest Neighbors (k-NN)	Improve the accuracy of traditional linear regression (LR) with cross-users watching behaviors that take advantage of prior users' data by identifying common scan paths and allocating a higher chance to future FoVs from those users.
[30]	Deep content analysis	Concurrently leverage sensor characteristics (HMD orientations) and content-related information (image saliency maps and motion maps) with LSTM to predict the viewer fixation in the future. The estimated viewing probability for each equirectangular tile may then be used in the quality optimization based on probability.
[32]	3D-CNN (convolutional neural networks)	3D-CNN to extract the Spatio-temporal features (saliency, motion, and FoV info) from the videos, has better performance than [30].
[33]	Content Analysis + Cross-user similarity	PARIMA, which is a hybrid of Passive Aggressive (PA) Regression and Auto-Regressive Integrated Moving Average (ARIMA) times series models to predict viewports based on users' behavior and the YOLOv3 algorithm on the stitched image to recognize the objects and retrieve their bounding box coordinates in each frame.
[34]	Content Analysis + Cross-user similarity	2 dynamic viewport selection (DVS) which changes the streaming areas depending on content complexity and user head movements to assure viewport accessibility and non-delay visual views for virtual reality users. To achieve higher accuracy, DVS1 focuses on the adjusted prediction distance between two prediction mechanisms whereas DVS2 selects the tiles for the following segment based on the modified prediction difference between actual and predicted perspectives based on content complexity variations.

Machine learning (ML) is used to predict bandwidth and views as well as increase video streaming bitrate to improve the Quality of Experience (QoE) [35]. **Table 4** summarizes the many papers that use machine learning to increase QoE in video streaming applications. The proposed scheme in [36] significantly reduces bandwidth consumption by 45% with less than a 0.1% failure ratio while minimizing performance degradation with Naïve linear regression (LR) and neural networks (NN). Next, Dasari, Bhattacharya [37] developed a system called PARSEC (PAAnoRamicStrEaming with neural Coding) to reduce bandwidth requirements while improving video quality based on super-resolution, where the video is significantly compressed at the server and the client runs a deep learning model to enhance the video quality. As for this, although Dasari, Bhattacharya [37] successfully reduce the bandwidth requirement and enhance the quality of the video, deep learning is large in models. It also results in the slowest inference rate. Furthermore, Yu, Tillo [38] present a method for adapting to changing video streams with the combination of the Markov Decision Process and Deep Learning (MDP-DL). In Filho, Luizelli [39], a strategy for adapting to fluctuating video streams (the Reinforcement Learning (RL) model) is researched. Next, a Recurrent Neural Network-Long Short-Term Memory(RNN-LSTM) and Logistic Regression-Ridge Regression(LR-RR)) to predict bandwidth and viewpoint is researched by Qian, Han [25] and Zhang, Guan [40]. To increase QoE, Vega, Mocanu [41] suggested a Q-learning technique for adaptive streaming systems. In [42], the deep reinforcement learning (DRL) model uses eye and head movement data to assess the quality of 360-degree videos.

Table 4. Machine learning (ML)-based approaches.

Source	Technique	Scope
[36]	<ul style="list-style-type: none"> Naïve linear regression (LR). Neural networks (NN). 	Motion detection and prediction.
[37]	<ul style="list-style-type: none"> Deep neural networks (DNN). Neural-aware adaptive bitrate (ABR) algorithm (§IV). 	Reduce bandwidth requirement and Improve video quality.
[38]	<ul style="list-style-type: none"> Markov Decision Process-Deep Learning (MDP-DL). 	Improve Variable bitrate (VBR).
[39]	<ul style="list-style-type: none"> Reinforcement Learning (RL) model. 	Improve Adaptive VR Streaming.
[25]	<ul style="list-style-type: none"> Logistic Regression-Ridge Regression (LR-RR). 	Viewpoint prediction and Bandwidth prediction.
[40]	<ul style="list-style-type: none"> Recurrent Neural Network-Long Short-Term Memory (RNN-LSTM). 	Viewpoint prediction and Bandwidth prediction.
[41]	<ul style="list-style-type: none"> Q-Learning Reinforcement Learning (RL). 	Improve constant bitrate (CBR).
[43]	<ul style="list-style-type: none"> Markov decision process (MDP). Deep reinforcement learning (DRL)-based algorithm. 	Viewpoint prediction and Optimal bitrate allocation.
[44]	<ul style="list-style-type: none"> Encoder-Decoder Based LSTM Model. Model Predictive Control (MPC)-based rate adaptation. 	Viewpoint prediction and Rate adaptation.
[45]	<ul style="list-style-type: none"> Markov Decision Process (MDP). 	Reactive caching and Viewport prediction.

Source	Technique	Scope
	<ul style="list-style-type: none"> Deep-Q-Network (DQN). [43] 	ction and nt bitrate

allocation for tiles. Younus, Shafi [44] presents an Encoder-Decoder based Long-Short Term Memory (LSTM) model that transforms data instead of receiving direct input to more correctly capture the non-linear relationship between past and future viewport locations to predict future user movement. To ensure that the 360 films sent to end-users are of the highest possible quality, Maniotis and Thomos [45] propose a reactive caching scheme that uses the Markov Decision Process (MDP) to determine the content placement of 360° videos in edge cache networks and then using the Deep Q-Network (DQN) algorithm, a variant of Q-learning to determine the optimal caching placement and cache the most popular 360° videos at base quality along with a virtual viewport in high quality.

6. Comparison between Techniques

Firstly, the DASH framework, tiling and viewport-adaptive techniques are correlated to each other as most of the tiling and viewport-adaptive techniques are using the DASH framework. Some of the tiling techniques [16][19][20][21] and the viewport-adaptive approach [28][31][32] are all using DASH to stream the areas covered by users' FOV in high quality while some other tiles are streamed in lower quality. The differences between these techniques are the mapping projection, encoding, tiling scheme and tile selection algorithm.

However, there are several limitations to the tiling and viewport-adaptive method. Firstly, more bandwidth is required to stream a screen-size movie at viewport devices as compared to a typical 2D laptop screen at the same quality. As illustrated in **Figure 4**, streaming a viewport region with a width of 110 degrees is still significantly wider than a normal laptop screen with a width of 48 degrees roughly [3][35]. Furthermore, most tiling solutions employ the viewport-driven technique, in which only the viewport that is the viewed area of the viewer is streamed in high resolution, yet it may also suffer from a significant delay due to the switching of the viewport, which might be due to the video content from the other viewports are not being delivered at the moment. So, when the user abruptly switches his/her viewport during the display time of the current video segment, a delay occurs. Next, as human eyes have a low delay and error tolerance, any viewport prediction errors can cause rebuffering or quality degradation and result in a break of immersion and poor user Quality of Experience (QoE). Furthermore, to accommodate users' random head movements, causing the need to increase the number of tiles of the video has and thus the video size increases significantly. Therefore, the implications of smooth viewport switching, minimized delays, with lessened video size and bandwidth should be addressed during 360-degree video delivery.

DASH, tiling and viewport-adaptive are focused on improving the streaming efficiency of the 360-degree video with lower bandwidth by streaming the demanded region of the 360-degree video with higher quality. On the other hand, Machine Learning (ML) techniques not only focus on lowering the bandwidth but also focusing on the improvement of QoE of the streaming. The proposed scheme ML also improves video quality, improves bitrate and predicts viewpoint in real-time which as is also effectively reduces bandwidth consumption while minimizing performance degradation. Some of the tiling and viewport-adaptive methods also use some algorithms such as Artificial Neural

Network [16], Heuristic algorithm [21] and adaptive algorithm [19] to optimize tile selection and predict the users' viewpoint.

References

1. Gohar, A.; Lee, S. Multipath Dynamic Adaptive Streaming over HTTP Using Scalable Video Coding in Software Defined Networking. *Appl. Sci.* 2020, 10, 7691.
2. Doppler, K.; Torkildson, E.; Bouwen, J. On wireless networks for the era of mixed reality. In *Proceedings of the 2017 European Conference on Networks and Communications (EuCNC)*, Oulu, Finland, 12–15 June 2017; IEEE: New York, NY, USA, 2017.
3. Shafi, R.; Shuai, W.; Younus, M.U. 360-Degree Video Streaming: A Survey of the State of the Art. *Symmetry* 2020, 12, 1491.
4. Hannuksela, M.M.; Wang, Y.-K.; Hourunrant, A. An overview of the OMAF standard for 360 video. In *Proceedings of the 2019 Data Compression Conference (DCC)*, Snowbird, UT, USA, 26–29 March 2019; IEEE: New York, NY, USA, 2019.
5. Monnier, R.; van Brandenburg, R.; Koenen, R. Streaming UHD-Quality VR at realistic bitrates: Mission impossible? In *Proceedings of the 2017 NAB Broadcast Engineering and Information Technology Conference (BEITC)*, Las Vegas, NV, USA, 22–27 April 2017.
6. Skupin, R.; Sanchez, Y.; Podborski, D.; Hellge, C.; Schierl, T. Viewport-dependent 360 degree video streaming based on the emerging Omnidirectional Media Format (OMAF) standard. In *Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP)*, Beijing, China, 17–20 September 2017; IEEE: New York, NY, USA, 2017.
7. Chiariotti, F. A survey on 360-degree video: Coding, quality of experience and streaming. *Comput. Commun.* 2021, 177, 133–155.
8. Song, J.; Yang, F.; Zhang, W.; Zou, W.; Fan, Y.; Di, P. A fast fov-switching dash system based on tiling mechanism for practical omnidirectional video services. *IEEE Trans. Multimed.* 2019, 22, 2366–2381.
9. D'Acunto, L.; Van den Berg, J.; Thomas, E.; Niamut, O. Using MPEG DASH SRD for zoomable and navigable video. In *Proceedings of the 7th International Conference on Multimedia Systems*, Klagenfurt, Austria, 10–13 May 2016.
10. Xie, L.; Xu, Z.; Ban, Y.; Zhang, X.; Guo, Z. 360 probdash: Improving qoe of 360 video streaming using tile-based http adaptive streaming. In *Proceedings of the 25th ACM International Conference on Multimedia*, Mountain View, CA, USA, 23–27 October 2017.

11. Huang, W.; Ding, L.; Wei, H.Y.; Hwang, J.N.; Xu, Y.; Zhang, W. Qoe-oriented resource allocation for 360-degree video transmission over heterogeneous networks. arXiv 2018, arXiv:1803.07789.
12. Nguyen, D.; Tran, H.T.; Thang, T.C. A client-based adaptation framework for 360-degree video streaming. *J. Vis. Commun. Image Represent.* 2019, 59, 231–243.
13. Google Cardboard. 2021. Available online: <https://arvr.google.com/cardboard/> (accessed on 30 December 2021).
14. Samsung Gear VR. 2021. Available online: <https://www.samsung.com/global/galaxy/gear-vr/> (accessed on 30 December 2021).
15. HTC Vive VR. 2021. Available online: <https://www.vive.com/> (accessed on 30 December 2021).
16. Zhou, C.; Xiao, M.; Liu, Y. Clustile: Toward minimizing bandwidth in 360-degree video streaming. In Proceedings of the IEEE INFOCOM 2018-IEEE Conference on Computer Communications, Honolulu, HI, USA, 16–19 April 2018; IEEE: New York, NY, USA, 2018.
17. Guan, Y.; Zheng, C.; Zhang, X.; Guo, Z.; Jiang, J. Pano: Optimizing 360 video streaming with a better understanding of quality perception. In Proceedings of the ACM Special Interest Group on Data Communication, Beijing China, 19–23 August 2019; pp. 394–407.
18. Xiao, M.; Wang, S.; Zhou, C.; Liu, L.; Li, Z.; Liu, Y.; Chen, S. Miniview layout for bandwidth-efficient 360-degree video. In Proceedings of the 26th ACM international Conference on Multimedia, Seoul, Korea, 22–26 October 2018.
19. Corbillon, X.; Simon, G.; Devlic, A.; Chakareski, J. Viewport-adaptive navigable 360-degree video delivery. In Proceedings of the 2017 IEEE International Conference on Communications (ICC), Paris, France, 21–25 May 2017.
20. Hosseini, M.; Swaminathan, V. Adaptive 360 VR video streaming: Divide and conquer. In Proceedings of the 2016 IEEE International Symposium on Multimedia (ISM), San Jose, CA, USA, 11–13 December 2016; IEEE: New York, NY, USA, 2016.
21. Ahmadi, H.; Eltobgy, O.; Hefeeda, M. Adaptive Multicast Streaming of Virtual Reality Content to Mobile Users. In Proceedings of the on Thematic Workshops of ACM Multimedia, Mountain View, CA, USA, 23–27 October 2017; pp. 170–178.
22. Dai, J.; Yue, G.; Mao, S.; Liu, D. Sidelink-Aided Multiquality Tiled 360° Virtual Reality Video Multicast. *IEEE Internet Things J.* 2022, 9, 4584–4597.
23. Madarasingha, C.; Thilakarathna, K.; Zomaya, A. OpCASH: Optimized Utilization of MEC Cache for 360-Degree Video Streaming with Dynamic Tiling. In Proceedings of the 2022 IEEE International Conference on Pervasive Computing and Communications (PerCom), Pisa, Italy, 22–25 March 2022.

24. Ribezzo, G.; De Cicco, L.; Palmisano, V.; Mascolo, S. A DASH 360 ° immersive video streaming control system. *Internet Technol. Lett.* 2020, 3, e175.
25. Qian, F.; Han, B.; Xiao, Q.; Gopalakrishnan, V. Flare: Practical viewport-adaptive 360-degree video streaming for mobile devices. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, New Delhi, India, 29 October–2 November 2018.
26. Ban, Y.; Xie, L.; Xu, Z.; Zhang, X.; Guo, Z.; Wang, Y. CUB360: Exploiting Cross-Users Behaviors for Viewport Prediction in 360 Video Adaptive Streaming. In *Proceedings of the 2018 IEEE International Conference on Multimedia and Expo (ICME)*, San Diego, CA, USA, 23–27 July 2018; pp. 1–6.
27. Chakareski, J.; Aksu, R.; Corbillon, X.; Simon, G.; Swaminathan, V. Viewport-Driven Rate-Distortion Optimized 360° Video Streaming. In *Proceedings of the IEEE International Conference on Communications (ICC)*, Kansas City, MO, USA, 20–24 May 2018; pp. 1–7.
28. Rossi, S.; Toni, L. Navigation-Aware Adaptive Streaming Strategies for Omnidirectional Video. In *Proceedings of the IEEE 19th International Workshop on Multimedia Signal Processing (MMSp)*, Luton, UK, 16–18 October 2017; pp. 1–6.
29. Koch, C.; Rak, A.-T.; Zink, M.; Steinmetz, R.; Rizk, A. Transitions of viewport quality adaptation mechanisms in 360° video streaming. In *Proceedings of the 29th ACM Workshop on Network and Operating Systems Support for Digital Audio and Video*, Amherst, MA, USA, 21 June 2019; pp. 14–19.
30. Fan, C.L.; Lee, J.; Lo, W.C.; Huang, C.Y.; Chen, K.T.; Hsu, C.H. Fixation Prediction for 360° Video Streaming in Head-Mounted Virtual Reality. In *Proceedings of the 27th Workshop on Network and Operating Systems Support for Digital Audio and Video*, New York, NY, USA, 20–23 June 2017; Association for Computing Machinery: Taipei, Taiwan, 2017; pp. 67–72.
31. Xu, Z.; Ban, Y.; Zhang, K.; Xie, L.; Zhang, X.; Guo, Z.; Meng, S.; Wang, Y. Tile-Based Qoe-Driven Http/2 Streaming System For 360 Video. In *Proceedings of the 2018 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, San Diego, CA, USA, 23–27 July 2018; pp. 1–4.
32. 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 2019 IFIP Networking Conference (IFIP Networking)*, Warsaw, Poland, 20–22 May 2019; pp. 1–9.
33. Chopra, L.; Chakraborty, S.; Mondal, A.; Chakraborty, S. PARIMA: Viewport Adaptive 360-Degree Video Streaming. In *Proceedings of the Web Conference 2021*, Ljubljana Slovenia, 19–23 April 2021.
34. Yaqoob, A.; Togou, M.A.; Muntean, G.-M. Dynamic Viewport Selection-Based Prioritized Bitrate Adaptation for Tile-Based 360° Video Streaming. *IEEE Access* 2022, 10, 29377–29392.

35. Ruan, J.; Xie, D. Networked VR: State of the Art, Solutions, and Challenges. *Electronics* 2021, 10, 166.
36. Bao, Y.; Wu, H.; Zhang, T.; Ramli, A.A.; Liu, X. Shooting a moving target: Motion-prediction-based transmission for 360-degree videos. In *Proceedings of the 2016 IEEE International Conference on Big Data (Big Data)*, Washington, DC, USA, 5–8 December 2016; pp. 1161–1170.
37. Dasari, M.; Bhattacharya, A.; Vargas, S.; Sahu, P.; Balasubramanian, A.; Das, S.R. Streaming 360-Degree Videos Using Super-Resolution. In *Proceedings of the IEEE INFOCOM 2020—IEEE Conference on Computer Communications*, Toronto, ON, Canada, 6–9 July 2020.
38. Yu, L.; Tillo, T.; Xiao, J. QoE-driven dynamic adaptive video streaming strategy with future information. *IEEE Trans. Broadcasting* 2017, 63, 523–534.
39. Filho, R.I.T.D.C.; Luizelli, M.C.; Petrangeli, S.; Vega, M.T.; Van der Hooft, J.; Wauters, T.; De Turck, F.; Gasparly, L.P. Dissecting the Performance of VR Video Streaming through the VR-EXP Experimentation Platform. *ACM Trans. Multimedia Comput. Commun. Appl.* 2019, 15, 1–23.
40. Zhang, Y.; Guan, Y.; Bian, K.; Liu, Y.; Tuo, H.; Song, L.; Li, X. EPASS360: QoE-aware 360-degree video streaming over mobile devices. *IEEE Trans. Mob. Comput.* 2020, 20, 2338–2353.
41. Vega, M.T.; Mocanu, D.C.; Barresi, R.; Fortino, G.; Liotta, A. Cognitive streaming on android devices. In *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, Ottawa, ON, Canada, 11–15 May 2015; IEEE: New York, NY, USA, 2015.
42. Li, C.; Xu, M.; Du, X.; Wang, Z. Bridge the gap between VQA and human behavior on omnidirectional video: A large-scale dataset and a deep learning model. In *Proceedings of the 26th ACM International Conference on Multimedia*, Seoul, Korea, 22–26 October 2018.
43. 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.* 2021, 32, 1607–1623.
44. Younus, M.U.; Shafi, R.; Rafiq, A.; Anjum, M.R.; Afridi, S.; Jamali, A.A.; Arain, Z.A. Encoder-Decoder Based LSTM Model to Advance User QoE in 360-Degree Video. *Comput. Mater. Contin.* 2022, 71, 2617–2631.
45. Maniotis, P.; Thomos, N. Viewport-Aware Deep Reinforcement Learning Approach for 360° Video Caching. *IEEE Trans. Multimed.* 2022, 24, 386–399.

Retrieved from <https://encyclopedia.pub/entry/history/show/62889>