Automatic QoE evaluation for asymmetric encoding of 3D videos for DASH streaming service

Paola Guzmán, Pau Arce, Juan Carlos Guerri*.
Institute of Telecommunications and Multimedia Applications (iTEAM)
Universitat Politècnica de València, Spain
{paoguzc1, paarvi}@iteam.upv.es, jcguerri@dcom.upv.es

ABSTRACT
The paper is based on the study of the performance of a Dynamic Adaptive Streaming over HTTP (DASH) system in the context of 3D video streaming, using different scenarios and network conditions, specifically with bandwidth variations. The objective is the development of a framework for the evaluation of QoE in 3D adaptive video streaming scenarios, which allows to analyze the impact on the user's Quality of Experience (QoE) using different bandwidth variation patterns (switching frequency, range and type of variation), among other aspects. A set of subjective tests will be carried out, with the aim of identifying the correlation between the quality of the user experience and the frequency, type, range and temporal location of the bandwidth switching events. The proposed framework allows performance measurements to be carried out in an automated and systematic way for the evaluation of DASH systems in 2D and 3D video streaming service. We have used Puppeteer, the Node.js library developed by Google, which provides a high-level API, to automate actions on Chrome Devtools Protocol, such as starting playback, causing bandwidth changes and saving the results of quality change processes, timestamps, stalls and so on. From this data, a processing is made to allow the reconstruction of the visualized video, as well as the extraction of quality metrics and the users' QoE assessment using the ITU-T P.1203 recommendation.

Categories and Subject Descriptors
C.2.1 [**Computer Systems Organization**]: Computer Communications Networks– *Network Architecture and Design, wireless communication*

General Terms
Performance, streaming, 3D video, QoE.

Keywords
Performance evaluation, 3D video, DASH, Puppeteer, Quality of Experience, testbed evaluation, ITU-T P.1203.

1. INTRODUCTION

Multimedia content distribution, particularly video streaming, currently dominates global Internet traffic. Worldwide consumption of Internet video traffic is expected to grow 4,3 times from 2017 to 2022, following an average annual growth rate of 34% [1]. Linked to the boom of new applications and immersive services in recent years, a growing interest in the production and transmission of 3D video can be observed again. However, it is well known that 3D video streaming over band-limited and unreliable communications channels can introduce artifacts on the transmitted 3D content and the effect could be much more significant compared to conventional 2D video streaming. This has made the issues related to content production, coding, transmission, Quality of Service (QoS) and Quality of Experience (QoE) perceived by users of 3D video distribution systems a research topic with numerous contributions in recent years.

Regarding transmission, the possibility of offering the best possible subjective quality to the user at all times maximizing their QoE has given rise to adaptive streaming of 2D and 3D video over HTTP (HAS), with the DASH (Dynamic Adaptive Streaming over HTTP) [2][3] standard being its most representative example. DASH enables flexible adaptation of video quality to available network resources and client device capabilities. It thus enables better buffer state management, control of interruptions during playback, and better bandwidth management, generally giving a higher QoE. Content servers offer multiple versions (representations) of the same video recording in order to implement DASH. Each representation is encoded using different parameters and divided into small segments (chunks) of a few seconds. All the information associated with the video segments, such as resolution, duration and average bit rate, is specified in the MPD (Media Presentation Description). Then, clients download the segments in sequential order and can change the representation from one segment to another according to the current estimated bandwidth and/or buffer state, so that stalls can be avoided, and the available bandwidth can be used in the best way. DASH technology has been adopted by a wide range of applications and video content providers, such as YouTube [4] or Netflix [5]. For this reason, many publications related to DASH and its impact on user QoE have recently appeared [6] [7] [8].

The development of the aforementioned studies involves the following stages, as illustrated in Figure 1. First, in stage 1, pre-processing of the data associated with the video-coding and generation of the DASH segments that will be available on the server is identified. Stage 2 concerns to Automation and Network Emulation, in this point, the conditions of bandwidth, delays, losses and performance of the devices that are processing and rendering the video contents are defined. Since a wide variety of DASH implementations are currently available, the selection of the video player in stage 3 represents a significant step of this process. This selection is performed taking into account aspects such as: supported formats, adaptation algorithm, open source or proprietary, etc. Finally, once the video streaming emulation has been carried out, the post-processing stages, which are devoted to the analysis of the information obtained (stage 4), and the reconstruction of the video (stage 5) for its subsequent objective and subjective evaluation (stage 6), are run and the process finishes.

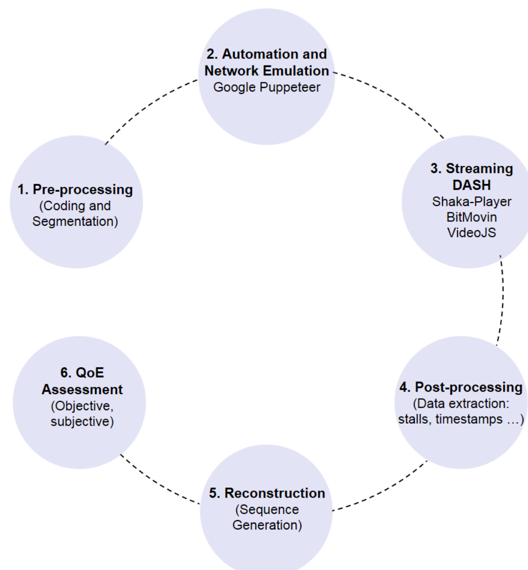

Figure 1. Stages of the QoE evaluation of DASH streaming.

In summary the objective of this work is to:

1. Develop a versatile, easily exportable and scalable testbed system. It will allow to automate and systematize the performance of QoE evaluation tests in an adaptive 3D video transmission scenario, under different bandwidth evolution profiles.
2. Use current, open source and flexible tools such as Puppeteer [9]. This new library developed by Google allows the automation of functional tests in web environments, through the use of the developer tools offered by the application.
3. Ease the processing of the data obtained during the simulations, in order to extract metrics or parameters associated with the QoS and QoE analysis.
4. Compare the asymmetric transform-domain quantization coding scheme with the symmetric coding method and verify the potential coding gains.
5. Carry out subjective 3D quality assessment experiments that allow user evaluation in different situations of video adaptive streaming.
6. Integrate objective and subjective QoE assessment methods and tools.

The rest of the paper is organized as follows. Section II presents a brief summary of the state of the art related to adaptive streaming in 2D and 3D scenarios as well as symmetric and asymmetric coding in stereoscopic video. Section III presents the architecture of the testbed system and its components. Section IV shows a survey of the main aspects related to 3D video objective a subjective quality assessment. Section V presents the figures, metrics and subjective evaluation results that allow the assessment of the multimedia streaming system. Finally, Section V presents the conclusions of the work.

2. RELATED WORK

As shown in Figure 1, the first stage in a QoE evaluation process in a DASH streaming system is related to the processes of encoding and the compression of the video sequences. If we consider that both the losses associated with the coding and compression processes, as well as errors and losses during transmission, can affect the quality perceived by the user, it is clear why it has become a research topic with numerous contributions in recent years. The various representation formats for 3D video can be classified in three groups according to the adopted strategy [10][11] frame-compatible methods [12], Multi-view Video Coding (MVC) [13][14] and View-plus-Depth (V+D) [15][16] coding schemes. To make it easier the introduction of stereoscopic services through the existing infrastructure and equipment, frame-compatible formats have been introduced. Frame-compatible video formats can be compressed with existing encoders, transmitted through existing channels, and decoded by existing receivers and players.

In a previous work carried out by the authors, a comparison in terms of QoE and the objective quality metrics [17] PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity) of the most popular video encoding standards H.264/AVC (Advanced Video Coding) [18] and H.265/HEVC (High Efficiency Video Coding) [19] with their respective extensions for multi-view formats using the MVC standard [20][21] was carried out. The results obtained agree with those obtained in other works such as [22] and [23] where performance comparison between encoders are carried out. In general, the results obtained in the different studies agree that the encoders based on HEVC have a better Rate Distortion relationship than the others based in H.264. Despite the high compression degree achieved with MVC and MVC-Full, its main limitation is that the secondary view decoding is conditioned to the correct receipt of the main view. On the contrary, with H.264 and H.265 Symmetric/Asymmetric coding, each frame can be predicted only from frames of the same view. This is an advantage in transport scenarios where a special treatment for each view can be guaranteed.

It is well known than the human visual system (HVS) can perceive high frequencies in 3D stereo video, even if that information is presented in only one of the views. Asymmetric coding aims to exploit the binocular suppression of the HVS getting more efficient video compression by representing one of the two views with a lower quality [24][25]. This is similar to what has been done for monocular color video, where chrominance channels are coded with fewer bits than the luminance, because HVS is less sensible to changes in color. In [26] the authors study the visual effect of asymmetric and symmetric encoding for immersive media conducting two different experiments using an Absolute Category Rating (ACR) [27] and boosted preference of experience (PoE) [28] protocol. They validate the value of Pair Comparison (PC) approach which main advantage is its high discriminatory power which is of great value when several test items are nearly equal in quality.

As mentioned above, the growing interest in improving the QoE of the users, making better use of network resources, has led to the adoption of HTTP Adaptive Streaming (HAS) technology among the most popular video content providers and applications. This has made the study of the performance of adaptive video streaming systems over the Internet and their impact on the user's QoE, research topics with numerous contributions in recent years [29][30].

Several publications focused on the theoretical or experimental study of the diverse players and adaptation algorithms can be found in the bibliography. In [31], the authors present a survey over the state-of-the-art of the bitrate adaptation algorithms for HAS. Bitrate adaptation schemes are classified based on the entity of the system where the logic is implemented: Server-based adaptation, Client-based adaptation, Network-assisted adaptation as well as hybrid adaptation, using information from any combination of the client, server(s) and network. In [32] and [33] the authors conducted experimental evaluations of different commercial and open sources adaptive HTTP streaming players/algorithms. As in most works of this type, the objective was focused on the rate-adaptation mechanisms and how they face the network bandwidth evolution.

So far, the experimental methodology has been based on the connection between two computers (Server and Client) and implies that the host that runs the video player also runs a packet sniffer and a network emulator such as DummyNet, NetEm or Netlimiter, among others.

Likewise, in addition to the need to understand the black box that the adaptation algorithm represents, the objective in many cases is its optimization [34] considering, for example, the segment size variation, the estimated path bandwidth and the current buffer occupancy, to accurately predict the time required to download the next segment, or its assessment based on the results of the users subjective evaluation [35].

However, there are few works oriented to the implementation of a common reference system for the realization/replication of performance tests of multimedia players in an easy and accessible way. For example, we find a proposal in [36] that although it targets an objective equivalent to ours, and also focuses on the study of multimedia players for web environments, the proposed solution still involves the use of three servers, one for the storage of content (Web Server), another for the emulation of network variations (Mininet) and, finally, a third server (Selenium Server) that will automate access to different players.

We remark that the quality evaluation of 3D video playing in the terminal of the final user is crucial. There has been a lot of work done in QoE evaluation for 2D video streaming and now we can find some works for 3D video streaming services. Generally speaking, we follow the typical aspects for evaluating the QoE: objective and subjective assessment. (i) Objective assessment is based on well-known metrics already applied to 2D video like: the popular Peak Signal to Noise Ratio (PSNR) [37], Structural Similarity (SSIM) [38], Visual Information Fidelity (VIF) [39] and the newest Video Multimethod Assessment Fusion (VMAF) video quality monitoring system, which is used to control the picture quality of all encoded videos streamed by Netflix [40]. PSNR represents the ratio in decibels between the maximum achievable power of a signal and the power of undesirable noise that affects the fidelity of its representation. Under certain conditions PSNR can be considered a logarithmic representation of Mean Squared Error (MSE). SSIM is also a full-reference metric designed to improve PSNR and it is based on frame-to-frame measuring of three components (luminance similarity, contrast similarity and structural similarity) and their combination in a single value. VIF is a full reference image quality assessment index based on natural scene statistics and the notion of image information extracted by the human visual system. And finally, VMAF predicts subjective video quality based on a reference and distorted video sequence. The implementation of the measure of all these parameters is simple, has low computational cost and measures can be reproduced. (ii) Regarding subjective assessment, we found that the methods used in the subjective assessment of 2D video can also be used for 3D video streaming. This evaluation is user centred, which is important because it can remark users related aspects. The most common methods (full-reference or no-reference standardized by ITU-T P915 [41]) are: Double-Stimulus Impairment Scale (DSIS), Double-Stimulus Continuous Quality-Scale (DSCQS), Simultaneous Double Stimulus for Continuous Evaluation (SDSCE), Absolute Category Rating (ACR), Hidden References (HRR), ACR5, ACR5-HRR, ACR11 and ACR11-HRR. A full-reference technique compares the measured video against the original, uncompressed video, and checks for differences. A reference-free technique analyzes the video under test and looks for artifacts without making a comparison. Both approaches exhibit positive attributes and drawbacks, and the appropriate choice depends on the desired application, cost constraints, and issues that can be caused by false readings. The disadvantages of subjective evaluation are founded on the fact that its implementation needs high computational cost and requires a controlled environment, specific equipment and suitable number of users.

In conclusion, we find that there are many different QoE models available [42] [43]. Each of them proposes its own set of metrics, which takes into account aspects such as stalls or the video start-up time, among others. One of the objectives of our work is to obtain and integrate in our framework as many metrics as possible in order to improve the estimation of the QoE, both in real time and for offline analysis. Thus, we rely on the implementation of the ITU-T Recommendation P.1203 [44], which was published in 2017 and it is described in [45]. It has become the first standardized model for QoE evaluation of adaptive audio and video streaming services, and it has been used in works such as [8], where QoE is evaluated on YouTube under different bandwidth restriction scenarios.

3. DEMO SYSTEM ARCHITECTURE

The proposed framework includes the following components:

- Pre-processing and coding videos in 3D format.
- HTTP-based web server that hosts pre-processed content (encoded and segmented video).
- Tool for emulation of network conditions (bandwidth and/or latency variations).
- Client with the DASH player.
- Tools for post-processing data related to transmission and extraction of metrics.
- QoE estimation.

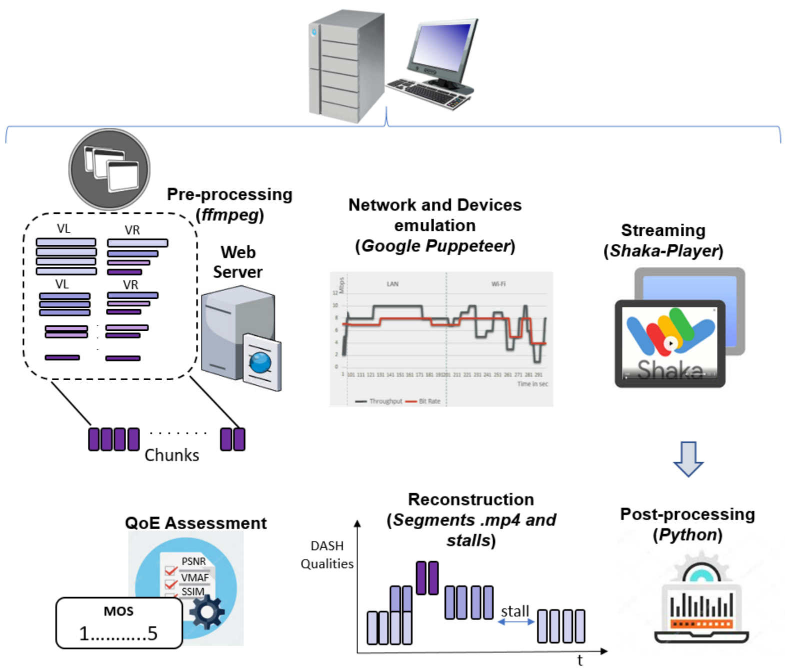

Figure 2. Proposed framework architecture.

Figure 2 illustrates the architecture of the proposed test framework. The blocks of the proposed system, whether pre-processing, server or client, has been implemented on a ninth generation Intel Core i7 PC with Ubuntu (version 18.04.2 LTS). The blocks are modular and can be executed on the same machine. Thus, a virtualization system or containers-based, such as Docker, could be used to deploy each of the modules on the same computer. The fact of implementing the entire testbed process in a single device allows the developed system to be versatile and easily exportable and replicable by the scientific community. Detailed descriptions of each component are provided in the following subsections.

3.1 3D Video coding and DASH content

As a previous step to the generation of DASH contents, an encoding process must be carried out. Focused on the generation of asymmetric encodings, 3D distribution systems use frame-compatible 3D and full-resolution frame-compatible 3D coding formats, that allow each of the views to be independently encoded, unlike in multi view methods such as MVC or V+D, although the latter methods exhibit higher coding efficiency.

In this sense, the encoding process is carried out using the *libx264* library of the *ffmpeg* application and the *Quantization Parameter* (QP) as encoding constraint parameter. The system is open to other encoders and any other encoding option, including other encoding constraint parameters, CRF (Constant Rate Factor) or bitrate. The use of other encoders such as HEVC, VP9 or AV1, depends on their compatibility with the web browser, the DASH player and the actual implementation of the standard for subjective evaluation.

The bitrates of the representations that should be available on the server were chosen by analyzing the RD (Rate Distortion) curve of the encoded sequence with different values of QPs in a range from 18 to 42. A preliminary perceptual quality test is performed to select the quantization parameters that spanned a wide range of visual quality. The stereoscopic video sequences were built following different HRCs (Hypothetical Reference Circuits) [46]:

- Symmetrically coded stereoscopic sequences. Left and Right views are encoded with the same quantization parameter QP in the range from 18 to 42 with two length steps. In the following, this condition will be referred to as SYM.
- Left view is encoded with QP in the range from 18 to 42 with two length steps and for the Right view the quantization parameter varies in the range from 20 to 42. These conditions will be referred to as ASYM.

Figure 3 illustrates the VMAF value for all the ASYM and SYM transform-domain quantization coding representations. The labels represent de QP value of the Left view, and the first mark in each curve correspond to the SYM representation.

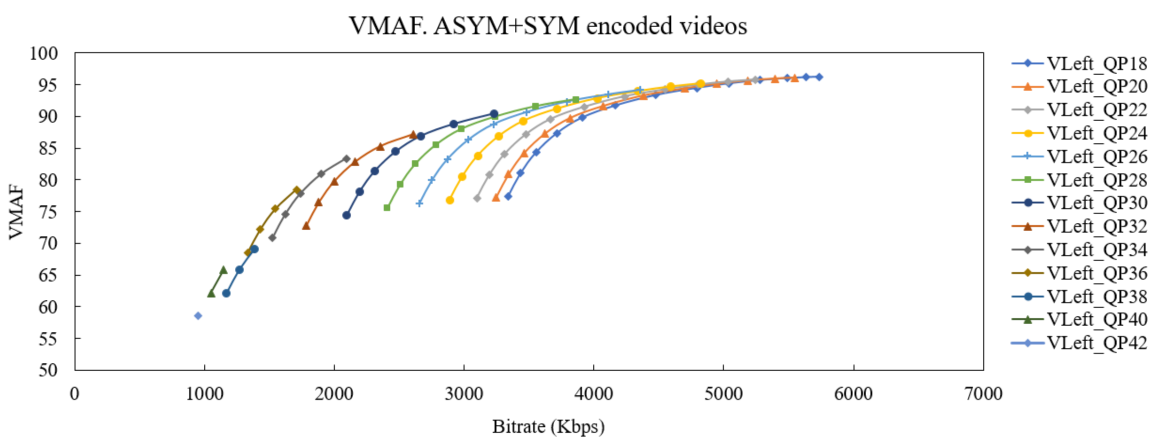

Figure 3. VMAF for all the ASYM and SYM transform-domain quantization encoded videos (representations).

Once the video sequences have been encoded, the DASH segments and the MPD are generated, containing all the information about the different video coding parameters and bandwidth. We are currently working with MPEG-DASH as the delivery format, although the option of using HTTP Live Streaming (HLS), which is the HTTP-based multimedia streaming protocol implemented by Apple, is also allowed.

3.2 Network conditions and devices emulation using Puppeteer

Adaptive video streaming can be conducted in a variety of environments, and a single change in context conditions can give a major impact on player behaviour and, most likely, the end-user viewing experience. In this first test scenario, emulation of network conditions will focus on bandwidth variation (periodic changes, staggered changes, etc.), which will allow us to evaluate the performance of the DASH player in each situation. Puppeteer [9] will be used as a tool for automation of end-to-end testing, including emulation of network bandwidth evolution. Puppeteer library has been recently released by Google. It offers an interface based on *node.js* that allows to execute and control Chrome (or Chromium) in headless mode through the *DevTools* protocol by running a script from the command line. Specifically, a CDP (Chrome Devtools Protocol) session is established with the web server where the Shaka Player implementation is hosted. By accessing the *Network.emulateNetworkConditions* resources, the necessary parameters for throttling activation are provided. The required network conditions to be defined are *downloadThroughput* (bytes/s), *uploadThroughput* (bytes/s) and *latency* (ms), which will be selected according to the desired context conditions [47].

Figure 4 shows the two network scenarios or bandwidth evolution considered within our evaluation. Taking into account that the frequency, type and location of the switching events between different video quality levels during a video streaming session may disturb the user's visual attention and therefore affect the user's QoE, the emulated scenarios represent persistent and non-persistent bandwidth fluctuations that correspond in some cases to the Network Presets available for throttling management in Chrome.

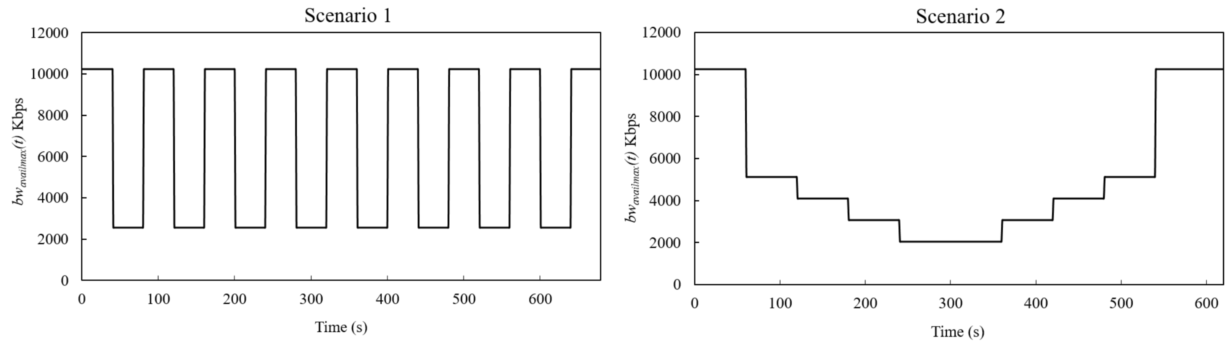

Figure 4. Network scenarios or bandwidth changes used for evaluation.

3.3 Client player DASH

From the various implementations of players compatible with the DASH standard currently available, we have selected the Shaka Player [48] for the development of this work. Shaka Player is an open source JavaScript library that allows the playback of multimedia content in both DASH and HLS formats in a standard browser, without requiring the use of plugins or Flash. The player must be hosted on a web server, the local computer in this case. Through the use of Puppeteer, a script has been developed in node.js that allows the following functions: access to the web where the Shaka Player is placed; select the video; activate the log in console (where the data will be collected for post-processing) and start the video playback by automatically clicking on the corresponding button.

All video players for modern streaming formats (e.g. HLS and MPEG-DASH) have a common feature set. Many of the features are subject to various tradeoffs between QoE and other parameters, which means that it is often possible to improve QoE by coming up with better heuristics. For some players (including ShakaPlayer), there is an easier way to improve some QoE metrics at the expense of others, since the heuristics can be tweaked using configuration options in the player. The two major important features in modern video players that have an impact on QoE are:

- Bitrate selection: to pick a suitable bitrate when there are multiple renditions in different qualities for a video stream. This feature is known by many names, e.g. adaptive bit rate (ABR) strategy, multi bit rate (MBR) strategy or automatic bitrate selection.
- Buffering strategy: for deciding the amount of media data to keep in the player's internal buffer, when to fetch media data, and how much media data is needed at startup before playback is initiated.

Shaka Player's buffering system has three parameters: *bufferingGoal*, *rebufferingGoal*, and *bufferBehind*. All are expressed in seconds. The *bufferingGoal* parameter is the amount of content we try to buffer, *rebufferingGoal* is the amount of content we have to have buffered before we can play, and *bufferBehind* is the amount of content we keep in buffer behind the playhead.

The modification of the Buffering Strategy is outside the objectives of this work that does not seek to make changes on the algorithm of adaptation of the player. However, tests have been carried out modifying the values associated with the *bufferingGoal*, in particular simulations were performed using *bufferingGoal* values of 30, 20 and 10 seconds. Using the value of 10 s, set by default in the player, the number of interruptions was multiplied in each of the scenarios under study, but in particular the situation was critical in scenarios with a high rate of bitrate variation. Finally, the value of *bufferingGoal* equal to 20 s was selected, which allows to have a fluid behaviour during the video streaming process since it maintains a good proportional relationship with the size of the segment used (5s).

Figure 5 shows a snapshot of what the user would see when running the system and disabling headless mode. A message is shown in the upper left corner of the screen, which indicates that an automated testing software is controlling Chrome. The right frame of the screen shows that access to DevTools is active, allowing access to throttling bandwidth and network statistics.

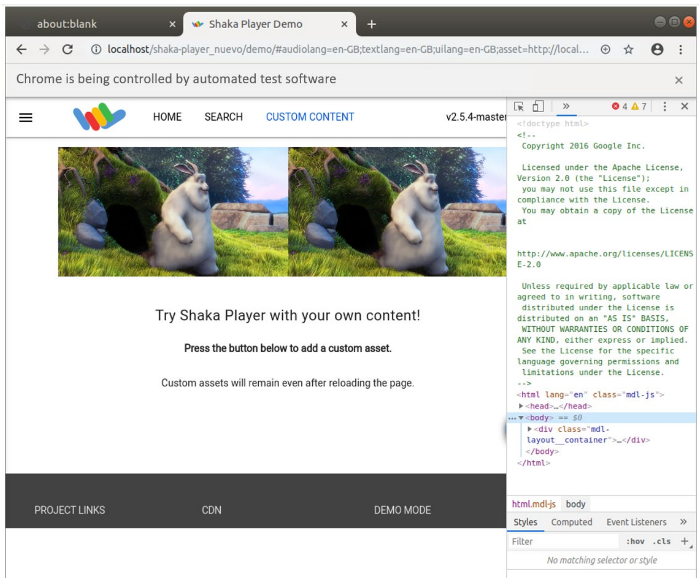

Figure 5. Chrome controlled by Puppeteer. Headless mode off.

3.4 Post-processing and reconstruction

The possibility of having access through Puppeteer to all the resources or developer tools oriented to the evaluation of the performance of the web services allows, once the video playback has started, the capture of metrics and records related to network statistics, buffer level, number of stalls, duration of stalls, playback time and transmitted versions, among others. This information is generated using methods that allow us to define a callback in which we can specify the elements of the page to extract programmatically. This data is collected through the console output or the generation of a JSON file that registers the interaction between the client and the server or by extracting the statistics and variables of the player through JavaScript methods. These files are then processed to retrieve the data required for the reconstruction of the video that will be used in the objective and subjective evaluation stages. Figure 6 shows an example of a reconstructed video (Q_1 corresponds to the highest available quality representation and Q_n to the lowest available quality representation). In free-of-stalls scenarios, the reconstruction is carried out by concatenating the downloaded segments according with their playback timestamps. However, in the scenarios where there are stalls, it is also necessary to have the start time and the duration of the stalls.

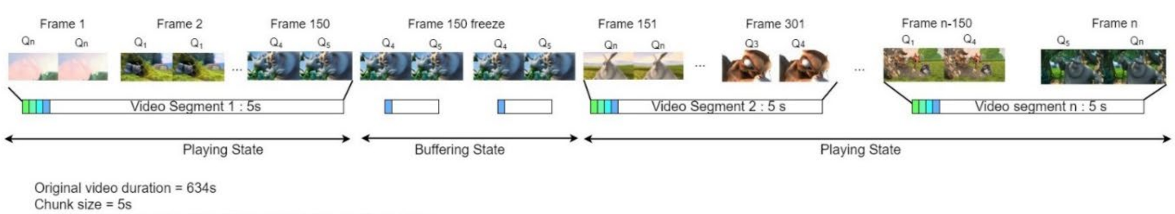

Figure 6. Received video reconstruction process.

With the data extracted from the network statistics and the variables obtained from the player we have the following objective metrics that will be used in the performance evaluation of the streaming service: *Playback Start Time*, *Number of Stalls*, *Duration of Stalls*, *Switching Events* and *Average Bitrate*.

3.5 Quality Assessment

While the ability to toggle between video representations significantly reduces the risk of stalls, the quality variations that this entails can reduce the global perceived quality of the video and can be annoying to the user. The users of video streaming services usually have a different set of expectations as compared to other types of web services. For this reason, knowing the QoE perceived by the user is a key factor to assess the performance of adaptive streaming algorithms. If there is a noticeable delay after a video link is clicked, or if there is an interruption during the playback, or any perceivable drop in the visual quality of the video, the user's perceived QoE can be affected.

In traditional video broadcasting systems, the quality metrics are based on a comparison of the output video with respect to the original video, so we find that the quality metrics have been classified into three different categories according to their dependence on the reference [49]. At this stage, as mentioned above, the evaluation of the quality of the received video will be performed using objective metrics: PSNR, SSIM, VIF and VMAF. Likewise, the subjective tests will be carried out to evaluate how the quality perceived by the user is affected by aspects such as the switching frequency, the variation range and the type of variation (ascending or descending).

In addition, at this stage we use a Python implementation of the ITU-T Recommendation P.1203. The overall architecture of the model is shown in Figure 7. The ITU-T P.1203 model consists actually of three individual modules: for video quality estimation (P.1203.1, Pv), for audio quality estimation (P.1203.2, Pa) and for audiovisual integration (P.1203.3, Pq). Input I.01 denotes the bitstream, from which specific input information for audio (I.11), video (I.13) and initial loading delay and stalling (I.14) are derived.

The Pv model (P.1203.1) can be run in four operation modes according to the available input information, from mode 0 to mode 3. These modes are distinguished according to the amount of available input information I.13, ranging from metadata (codec, resolution, target bitrate, frame rate, coding and display resolution and segment duration) in mode 0, to bitstream level in mode 3. A detailed description of the P.1203.1 model algorithm can be found in [50].

P.1203 offers different output information (Figure 7) that can be used for service diagnostics and provides a score on a scale of Mean Opinion Score (MOS) from 1 to 5. Moreover, P.1203 takes into account information of stalls for quality estimation and can be used for videos encoded in H.264 format with a resolution of up to 1080p.

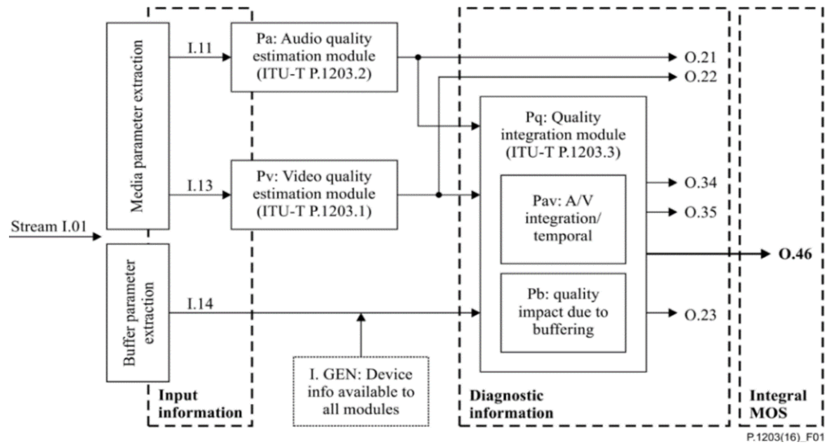

Figure 7. Model ITU-T P.1203 architecture overview.

4. DASH STREAMING ASSESSMENT AND RESULTS

The Big Buck Bunny [51] video sequence with 1080p resolution and 30 fps has been used for the experiments. This sequence is commonly used in this type of testbeds [52], since its time duration (635 s) is suitable in this type of experiments and allows to reduce biases in the analysis. The video of each view (Left view and Right view) has been encoded using ffmpeg libx264, preset "medium", a closed GOP (Group of Pictures) structure and variable QP. Following the procedure commented in section 3.1, out of the total of 91 available encoded sequences, 19 HRCs were selected to be segmented and hosted on the server. The length of the segments (Sd) was 5 s. Once the segments are generated, they are placed together with the MPD within the Web Server so that they can be retrieved by the DASH player. Figure 8 shows the VMAF curve of the selected encoded sequences. The represented VMAF corresponds to the stereoscopic video stream and has been calculated by averaging the individual VMAF values of the Left and Right views. Table 1 details the information about the conformation of each stereoscopic video stream.

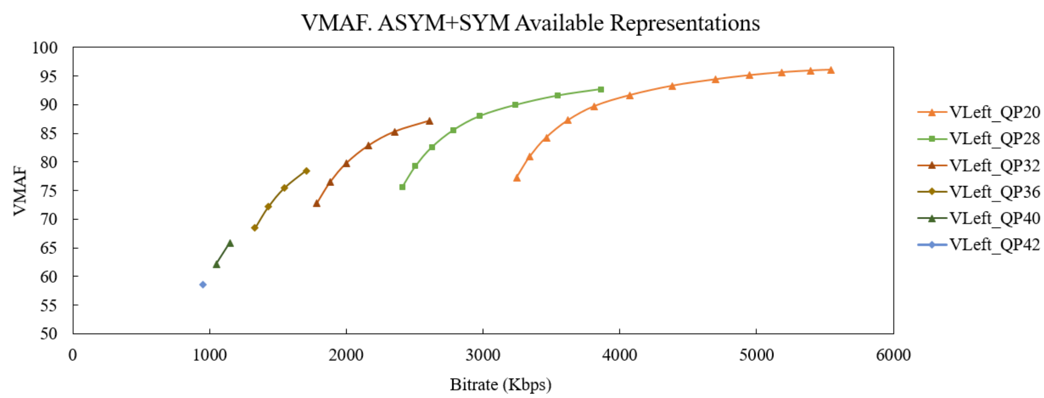

Figure 8. VMAF. ASYM+ SYM selected encoded videos (available representations).

Table 1. Available representations

		V0	V1	Average Bitrate
SYM	HRC1	QP20	QP20	5544.12
ASYM	HRC2	QP20	QP22	5394.88
ASYM	HRC3	QP20	QP24	5183.74
ASYM	HRC4	QP20	QP26	4948.15
ASYM	HRC5	QP20	QP28	4698.32
ASYM	HRC6	QP20	QP30	4383.89
SYM	HRC7	QP28	QP28	3860.92

ASYM	HRC8	QP28	QP30	3546.49
ASYM	HRC9	QP28	QP32	3235.4
ASYM	HRC10	QP28	QP34	2977.57
SYM	HRC11	QP32	QP32	2610.81
ASYM	HRC12	QP32	QP34	2352.98
ASYM	HRC13	QP32	QP36	2160.4
SYM	HRC14	QP36	QP36	1708.85
ASYM	HRC15	QP36	QP38	1548.13
ASYM	HRC16	QP36	QP40	1429.78
SYM	HRC17	QP40	QP40	1148.55
ASYM	HRC18	QP40	QP42	1051.41
SYM	HRC19	QP42	QP42	954.07

The tests are carried out using the framework for automatic video streaming developed in this proposal. The proposed framework is based on the use of Puppeteer, which was discussed in section 3.2. It is an API based on node.js, developed by Google, and is aimed at automating functional tests in web environments. Using Puppeteer, we can automatically access the web server and the video content, while emulating the network conditions corresponding to each of the proposed scenarios. At the end of the streaming simulation process, the system also allows us to perform the reconstruction of the video played on the client and to perform the analysis of the streaming performance having access to values related to parameters such as playback timestamp per segment, initial buffering delay, download segment size and time, buffer state per second, requested and downloaded representations, playback states (buffering, paused or playing), and bandwidth switching levels, frequency and location. In the following sections, the analysis of the main aspects evaluated in this type of studies is presented.

4.1 DASH streaming performance evaluation

Table 2 summarizes the notation and definition of the parameters used for the evaluation. As Figure 9 and Figure 10 show, Scenario 1 and Scenario 2 will be considered to show the results. Scenario 1 (Figure 9) emulates periodic short-term variations of the available bandwidth. In this context, short-term changes are fluctuations produced at intervals of 40 s. On the other hand, Scenario 2 (Figure 10) produces long-term variations of the available bandwidth. This scenario exhibits two parts: from t=0 s to t=300 s, bitrate decreases from 7 Mbps to 2 Mbps each minute, and from t=300 s to t=635 s bitrate increases from 2 Mbps to 7 Mbps each minute. The latency has been set to 5 ms in all cases.

Table 2. Simulation Parameters

Notation	Unit	Definition
T	s	Total session time
S_d	s	Segment duration
S	segments	Total number of segments in session. $S= T/S_d$
R	level	Total number of representations offered on the server
r_i	kbps	Representation levels $1<i<R$. r_R represents the highest quality level
$bw_{available}(t)$	kbps	Available bandwidth emulated
$bw_{availablemax}(t)$	kbps	Maximum available bandwidth
$\tau(t)$	kbps	Throughput, measured bitrate on client's side
$b(t)$	kbps	Bitrate selected on client's side

T_{buf}	s	Total buffering time, total time the playback is interrupted due to buffer underrun.
ε_{buf}		Buffer efficiency. $\varepsilon_{buf} = {}^{T_{buf}}/_{T}$

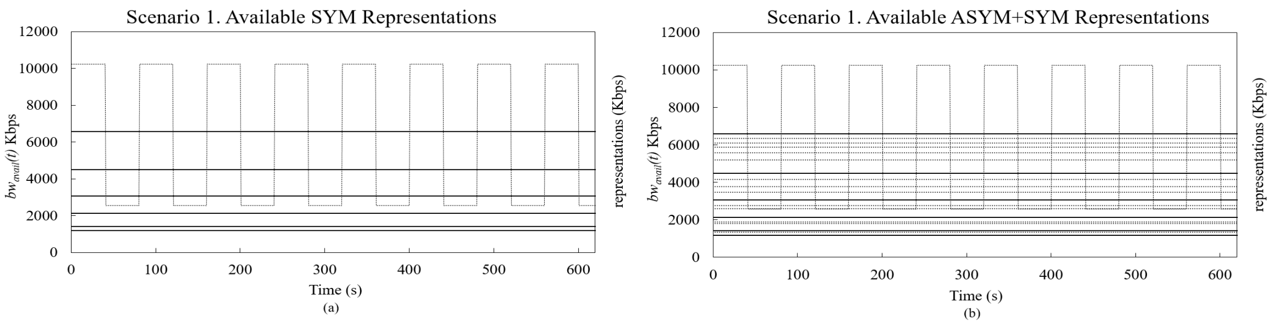

Figure 9. Bandwidth variations and available representations for Scenario 1. (a) SYM Representations. (b) ASYM+SYM Representations

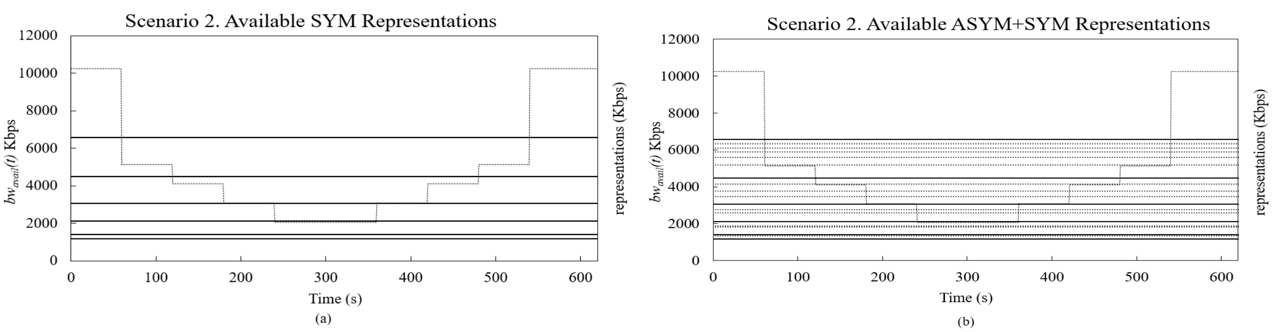

Figure 10. Bandwidth variations and available representations for Scenario 2. (a) SYM Representations. (b) ASYM+SYM Representations.

Figure 11 and Figure 14 show the available bandwidth *($bw_{available}(t)$)*, the requested bitrate *b(t)* as a function of the bandwidth variations and throughput function *τ(t)* or bitrate measured on the client side for Scenario 1 and Scenario 2 respectively. *τ(t)* is obtained by calculating the effective throughput of each segment, which is the size of the segment divided into the segment download time. Although it cannot be appreciated due to the figures scale, in all the cases the adaptation algorithm downloads the first two segments using the lowest quality level and, then, switches to the highest available quality level within the first 12 seconds, which is usually the case of most adaptation algorithms. The results obtained are consistent with what is expected from the adaptation algorithm of the Shaka Player, which is based on 2 EWMA (*Exponentially Weighted Moving Average*), according to the literature. In the end, it managed to adapt quickly to bitrate drops but gradually increase the quality when the bandwidth goes up, as it can be seen in Figure 11 and 14. As can be seen in both scenarios, the inclusion of the asymmetric representations (ASYM+SYM) allows to optimize the use of the available bandwidth, since the qualities downloaded using the ASYM+SYM representations allow us to experiment better overall video quality. For instance, in Figure 14 Scenario 2, it can be observed at t= 200 s that for an available bandwidth of 3000 kbps, the chosen representation using SYM corresponds to 2118 Kbps bitrate versus 2750 Kbps bitrate when both symmetric and non-symmetric representations are chosen.

Figure 12 and Figure 15 show the time between requests and the download time of a segment for the two scenarios under study, respectively. It can be appreciated how the *Segment Download Time* behavior is affected by the state of the buffer and the available bandwidth, since its value increases significantly when the buffer's occupancy decreases or an underrun of the buffer is taking place.

There are no interruptions simulated in the experiments. However, the behavior of the system under severe decrements of the buffer occupancy is shown. Thus, it can be observed in the Scenario 1 that the segment download time reaches values around 1 s when the available bandwidth is 10 Mbps. It can also reach up to 14 s when the available bandwidth decreases suddenly to 2.5 Mbps.

The gain of the ASYM+SYM encoded representations over the SYM encoded representations is more evident in Scenario 2. In this case, due to the fact that the player has a greater number of available representations using ASYM+SYM, a smoother adaptation process can be carried out, selecting higher quality representations in each of the required quality change transitions as a consequence of bandwidth variations. For both scenarios, Figure 13 and Figure 16 show respectively the behaviour of the buffer as a function of the available bandwidth. Using a *bufferingGoal* of 20 seconds, playback interruptions can be minimized or avoided.

Adaptation algorithms use buffer occupancy and throughput as parameters to choose the next downloading segment. It can be shown that SYM+ASYM representations allow to reproduce better quality segments keeping a buffer occupancy level similar to SYM representations in most scenarios.

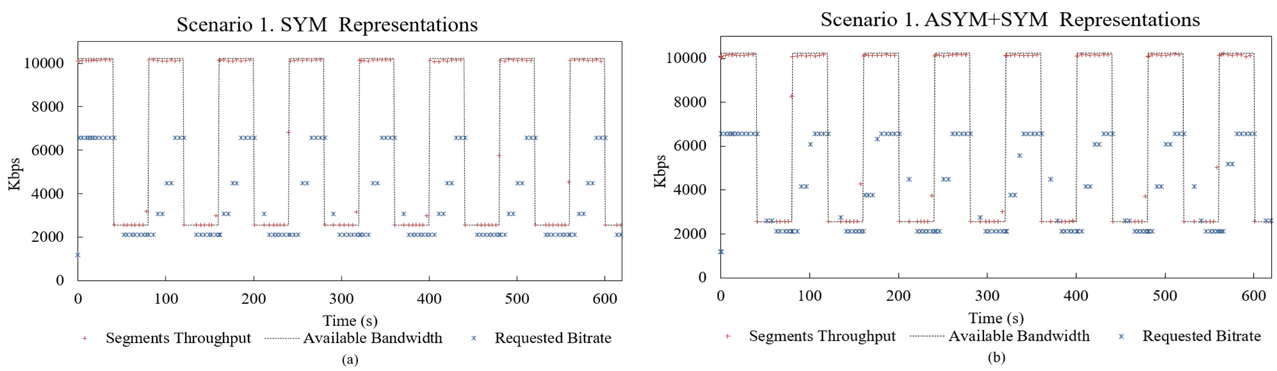

Figure 11. Scenario 1. Segments throughput, available bandwidth and requested bitrate. (a) SYM. (b) ASYM+SYM.

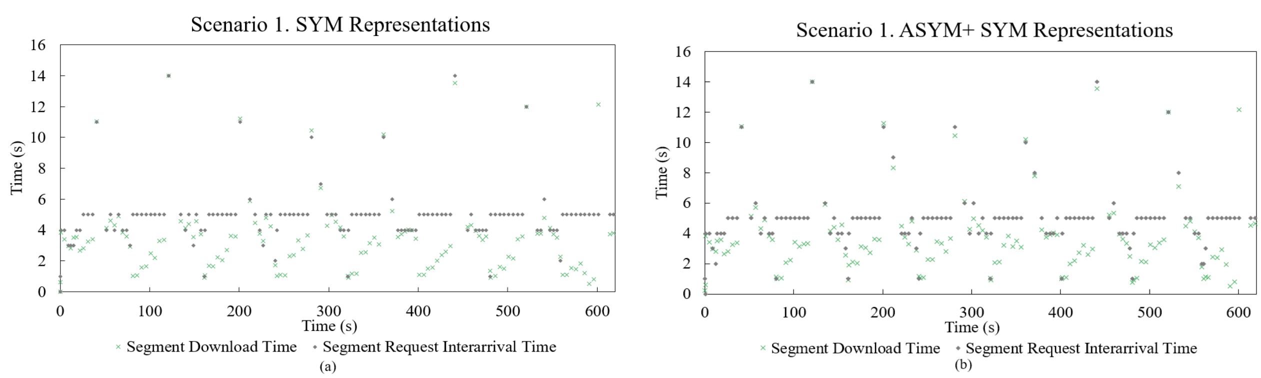

Figure 12. Scenario 1. Segment download and requested interarrival time of video segments under restricted available bandwidth conditions. (a) SYM. (b) ASYM+SYM.

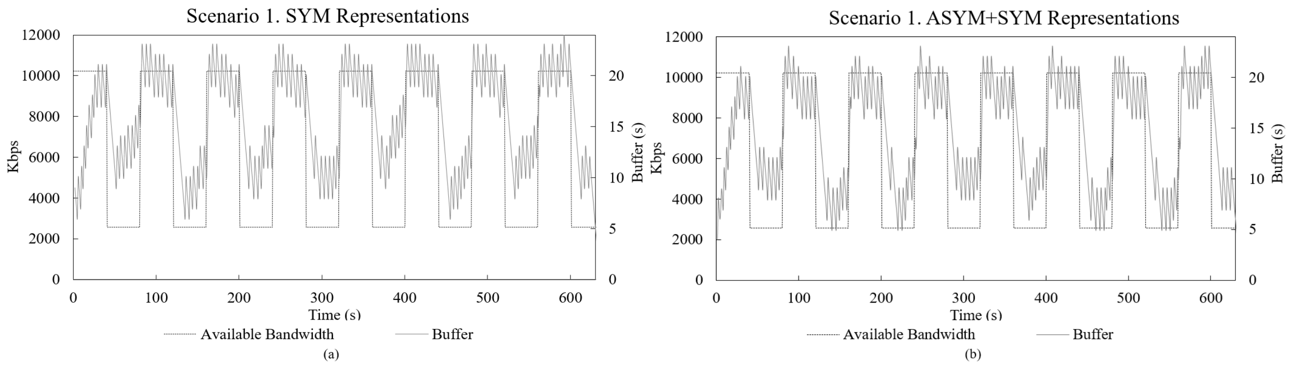

Figure 13. Scenario 1. Available bandwidth and buffer state. (a) SYM. (b) ASYM+SYM.

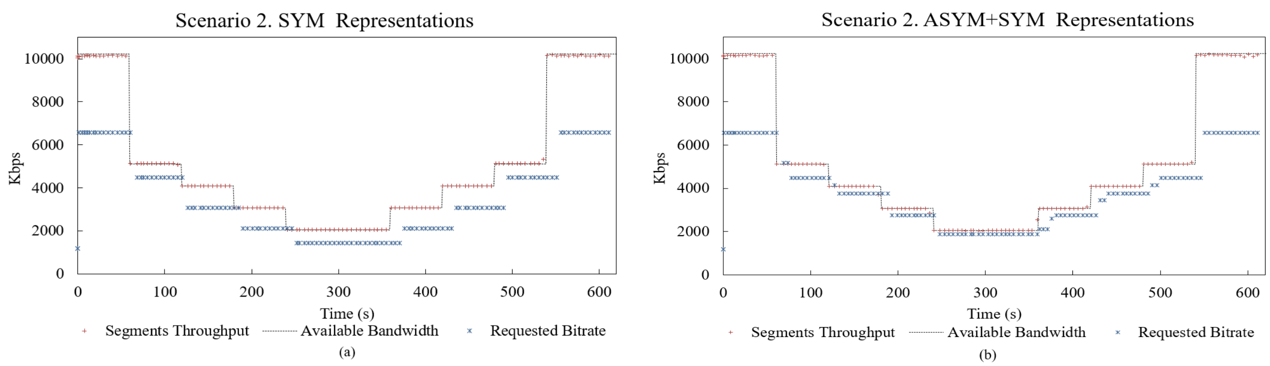

Figure 14. Scenario 2. Segments throughput, available bandwidth and requested bitrate. (a) SYM. (b) ASYM+SYM.

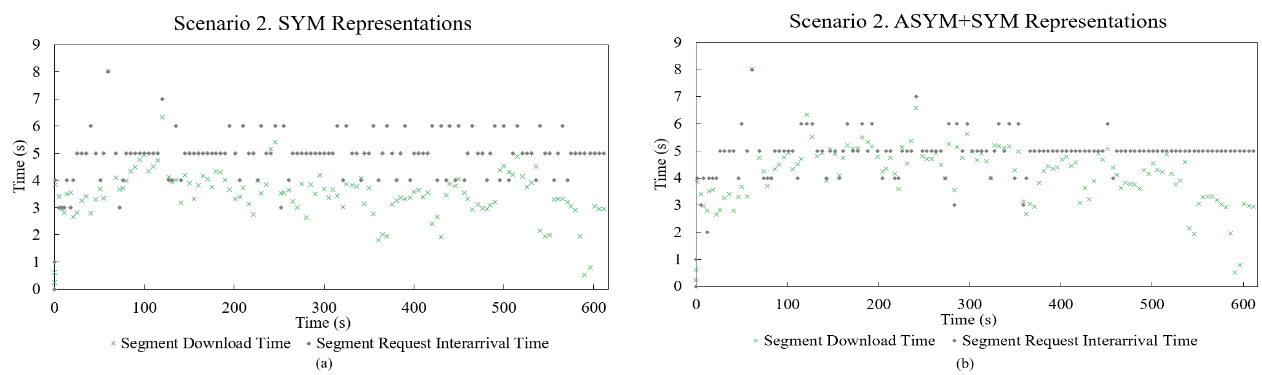

Figure 15. Scenario 2. Segment download and requested interarrival time of video segments under restricted available bandwidth conditions. (a) SYM. (b) ASYM+SYM.

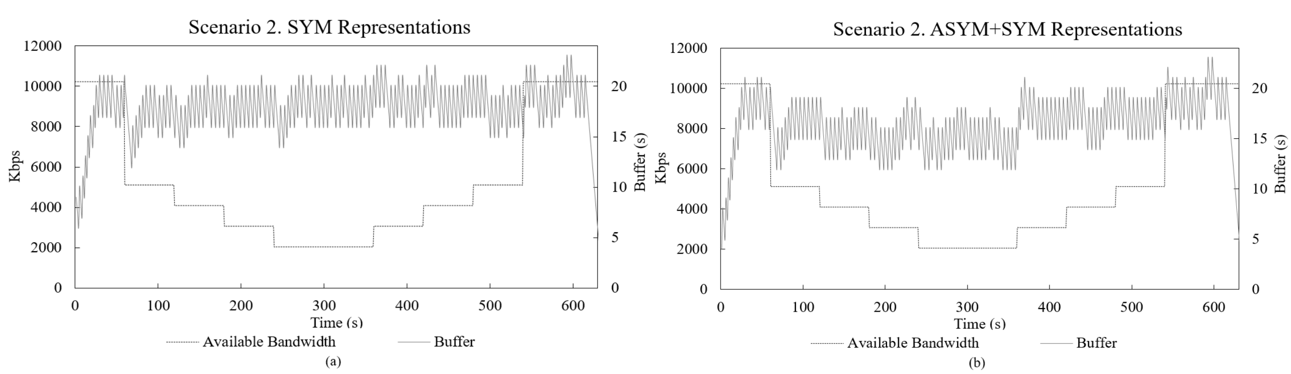

Figure 16. Scenario 2. Available bandwidth and buffer state. (a) SYM. (b) ASYM+SYM.

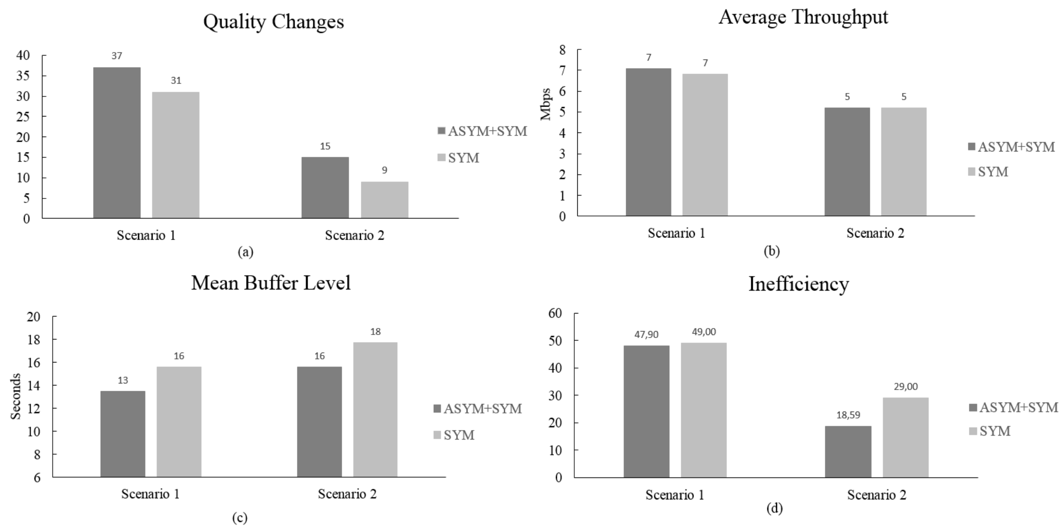

Figure 17. Performance results for Scenario 1 and Scenario 2. (a) Quality Changes, (b) Average Throughput, (c) Mean Buffer Level and (d) Inefficiency.

Figure 17 illustrates some important performance results that can be also retrieved from the system, such as number of quality changes, average throughput, mean buffer level and the inefficiency, but also others can be calculated from the resulting analysis, like number of stalls and the percentage of time in buffering state with respect to total playing time (these cases are not shown since they exhibit zero value). In particular, inefficiency is defined according to equation (1) [7] and it determines to what extent the algorithm properly utilizes the available network bandwidth.

$$\boldsymbol{Inefficiency} = \sum_t \frac{|\boldsymbol{b}(\boldsymbol{t}) - \boldsymbol{\tau}(\boldsymbol{t})|}{\boldsymbol{\tau}(\boldsymbol{t})} \qquad (1)$$

Regarding to the quality changes parameter, we can see that for both Scenario 1 and Scenario 2, more Quality Changes are observed in the context of the ASYM+SYM representations because there are more available and fine-grained representations. Although a higher number of representation switches is punished in most quality assessment algorithms, the use of a finer-grained representation list using asymmetric views allows the transitions between the different available bitrates to be smoother, thus improving the quality of user experience. We can also observe that the Average Throughput is slightly higher in the ASYM+SYM context than if we use only the SYM representations, but it is not a representative difference in this case.

On the other hand, the Mean Buffer Level and Inefficiency parameters show that the use of ASYM+SIM representations allows a better use of the bandwidth. This trend is particularly remarkable in Scenario 2 due to the bandwidth variation pattern (taking into account the frequency, duration and magnitude of the change in the available bandwidth).

4.2 Objective Video Quality Evaluation

This section is focused on the study of the main parameters for objective video quality assessment (VMAF, PSNR, SSIM, VIF), comparing the use of symmetric views (SYM Representations) with the use of both asymmetric and symmetric views (ASYM+SYM Representations) for both studied scenarios (Scenario 1 and Scenario 2).

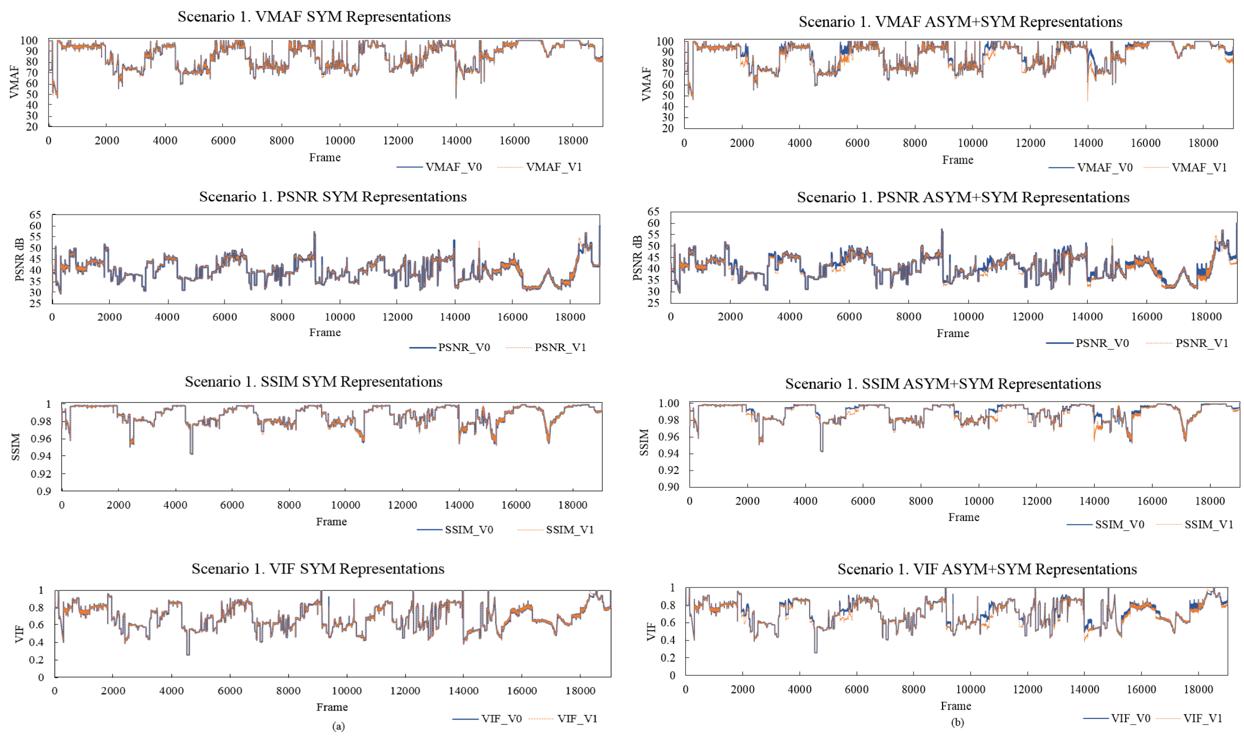

Figure 18. Objective metrics results for Scenario 1. (a) SYM Representations. (b) ASYM+SYM Representations.

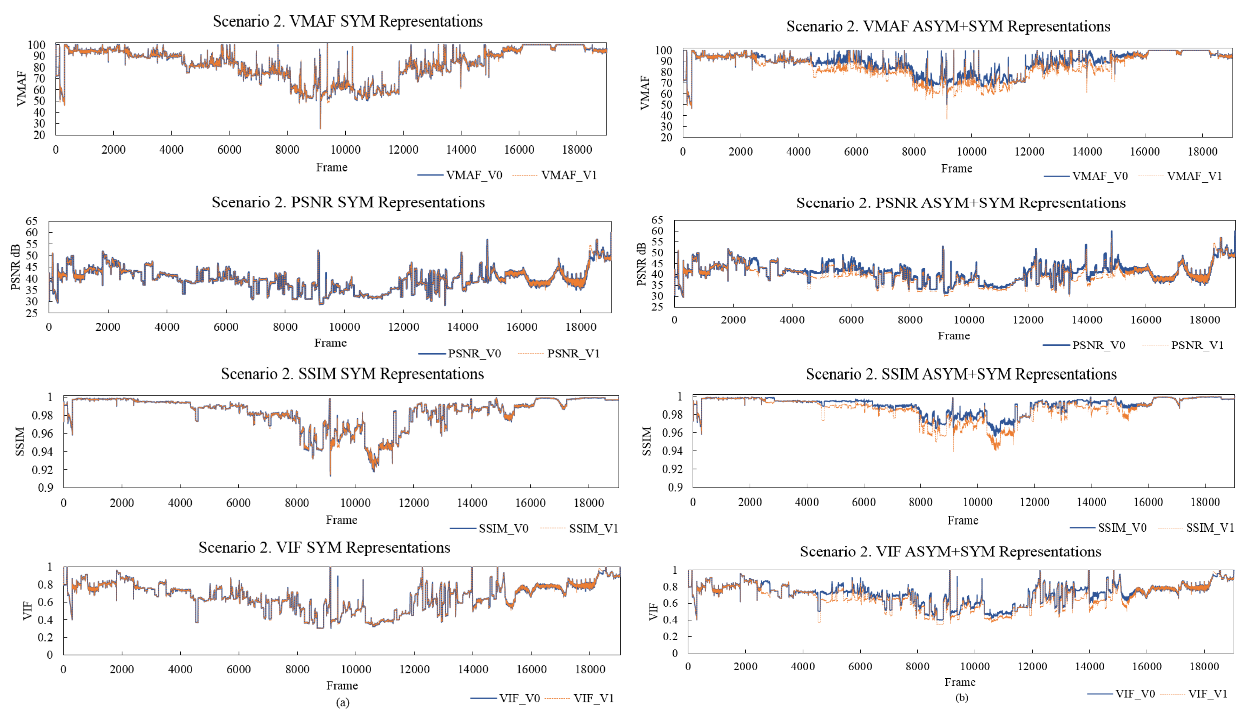

Figure 19. Objective metrics results for Scenario 2. (a) SYM Representations. (b) ASYM+SYM Representations.

As it can be observed in Figures 18 and 19, the behavior of the VMAF and SSIM curves have a clear correlation with the pattern of bandwidth variation in the two evaluated scenarios. Likewise, although the values obtained for the rest of the objective metrics (PSNR, VIF) seem to have similar behaviors in both cases, SYM and ASYM+SYM context, if we analyze the stereoscopic video played, we find that the average bitrate is higher when using ASYM+SYM representations compared with the use of SYM representations only. Analyzing the figures corresponding to the VMAF obtained for Scenario 1 and Scenario 2 using only the SYM representations with respect to the VMAF obtained using ASYM+SYM representations, it can be seen that a better result is obtained for the second condition for both Scenario 1 and Scenario 2. From a quantitative point of view, for Scenario 1 an average VMAF value of 85.73 is achieved when using only SYM encodings versus 87.13 when using ASYM+SYM encodings. The same behavior, but even more evident, is depicted in Figure 19 for Scenario 2, where we have an average VMAF value for 3D video of 82.47 when using only SYM encodings, compared to 86.61 when using ASYM+SYM encodings.

4.3 Subjective Assessment

Considering the drawbacks of performing subjective tests, like the number of users required and the time to perform the tests, we find that the requirement of a specific equipment and controlled viewing conditions during the test for 3D video context makes unfeasible the use of methodologies such as crowd sourcing, which would have been a highly cost-efficient, fast and flexible way of conducting user experiments. For this reason, in addition to the evaluation of QoE through subjective test, in this work we focus on how to predict the quality of a stereoscopic 3D video from objective and automatic subjective assessment of the 2D single-view videos.

The subjective assessment of each 2D single-view video was carried out with the Python implementation of the ITU-T P.1203 recommendation as discussed in section 3.5. The results of the simulations are shown in Table 3 and Table 4, and in Figures 20 and 21.

Table 3 shows the results obtained for Scenario 1 and Scenario 2 in both contexts SYM and ASYM+SYM representations. Firstly, output O.23 is shown, which is a perceptual buffering indication, as a single score expressed on a quality scale (from 1-5) for a given session. In this case O.23 has the maximum value, considering that there were no stalls during the transmission in any of the analyzed scenarios. Furthermore, O.35 and O.46 are also presented. O.35 gives the time-integrated version of O.34 (see definition below) and it represents the final audiovisual coding quality score, also given using a MOS quality scale (1-5). Finally, O.46 denotes the overall quality score that takes into account the initial buffering delay. From the O.35 output, it can be seen that ASYM+SYM encodings improves video quality results compared with the use of only SYM representations, although it is scenario dependent. That is, in Scenario 1, the improvement is around 1% whereas in Scenario 2 the improvement is over 4%.

Table 3. ITU-T P.1203 MOS results ASYM+SYM

	Output	Value V0	Value V1	Average V0V1
Scenario 1	O.23	5.00	5.00	5.00
	O.35	4.68	4.63	4.66
	O.46	4.63	4,60	4.62
Scenario 2	O.23	5.00	5.00	5.00
	O.35	4.81	4.69	4.75
	O.46	4.74	4.64	4.69

Table 4. ITU-T P.1203 MOS results SYM

	Output	Value V0	Value V1	Average V0V1
Scenario 1	O.23	5.00	5.00	5.00
	O.35	4.60	4.60	4.60
	O.46	4.58	4,58	4.58
Scenario 2	O.23	5.00	5.00	5.00
	O.35	4.56	4.56	4.56
	O.46	4.52	4.52	4.52

On the other hand, Figures 20 and 21 represent the behavior of O.34, which is the audiovisual quality per output sampling interval. Only the video is being considered, since the audio has been removed from the original sequence and is out of scope of this study. It can be seen in the figures how the MOS parameter decreases, which mean lower

QoE, when the player downloads lower quality versions to counteract the decrease of bandwidth and thus avoid interruptions.

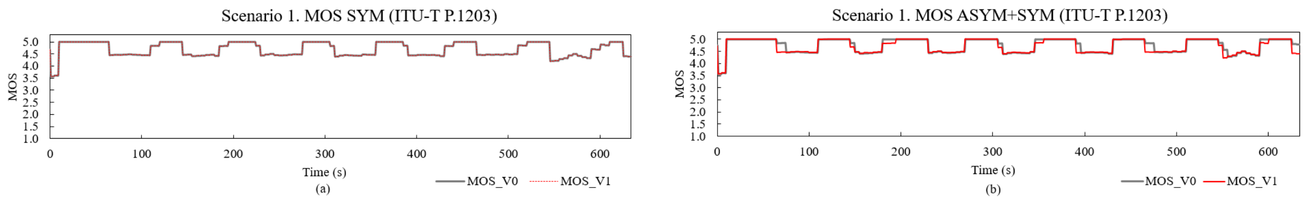

Figure 20. ITU-T P.1203 O.34 output for Scenario 1. (a) MOS per second using SYM representations. (b) MOS per second using ASYM+ SYM representations.

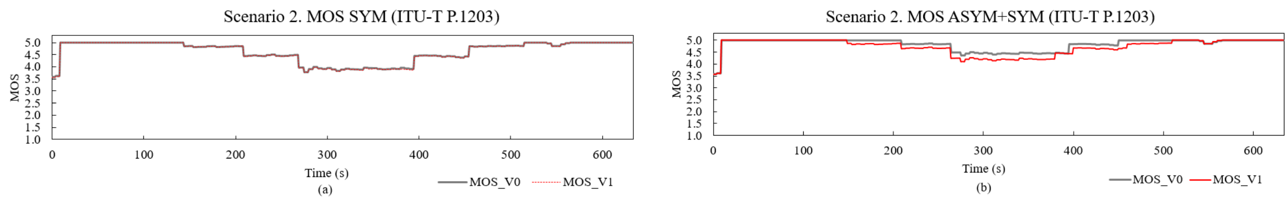

Figure 21. ITU-T P.1203 O.34 output for Scenario 2. (a) MOS per second using SYM representations. (b) MOS per second using ASYM+SYM representations.

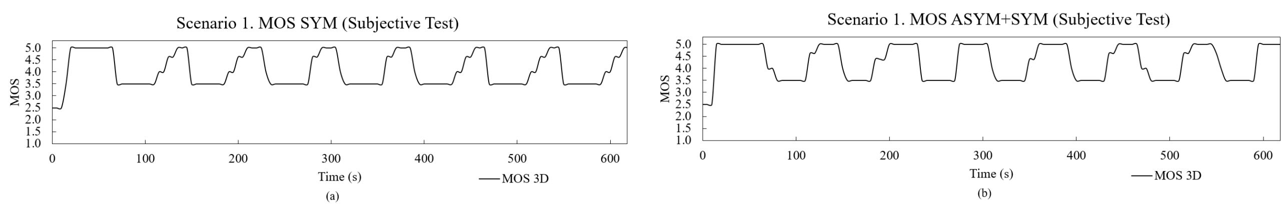

Figure 22. MOS Subjective Test with users Scenario 1. (a) MOS per second using SYM representations. (b) MOS per second using ASYM+SYM representations.

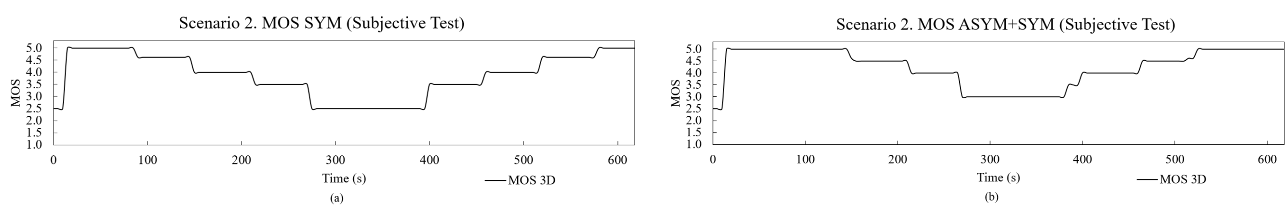

Figure 23. MOS Subjective Test with users Scenario 2. MOS per second using SYM representations. (b) MOS per second using ASYM+SYM representations.

Existing studies (e.g., [25] and [26]) suggest that averaging the quality of the 2D left and right views well predicts the quality of symmetrically distorted stereoscopic videos but generate substantial prediction bias when applied to asymmetrically distorted stereoscopic videos. According to that, a subjective assessment was carried out based on ITU-T P915 recommendation [41] using *Double-Stimulus* (DS) Quality Rating method with explicit reference using a 5-point MOS quality scale as shown in Table 5, in which video sequences were evaluated taking into account a reference version.

Table 5. MOS quality scale

Value	Quality
5	Excellent
4	Good
3	Fair
2	Poor
1	Bad

In subjective quality tests, 10 seconds of each of the 19 stereo video sequences (SYM and ASYM) available on the DASH streaming server were evaluated. During the assessment session, sequences where displayed in a random order, and before each test sequence, a video corresponding to the original video without any degradation was displayed. Each test sequence was individually evaluated just after being presented to the assessor, who had 5 s to evaluate the 3D video QoE in terms of the visual quality giving a score based on the MOS quality scale (Table 5). The assessors viewed each video sequence using NVIDIA 3D vision system and 17-inch LCD Monitor with 120 refresh rate. Figure 22 and Figure 23 show the MOS values obtained in the subjective tests for both Scenario 1 and Scenario 2, respectively. With these outcomes and in consistency with the objective assessment results, from the MOS values for the 2D sequence (Left view, Right view) provided by the implementation of the ITU-T P.1203 recommendation, a very good prediction of the MOS of the symmetric stereoscopic sequence is obtained. However, in the case of asymmetric representations, the prediction of the MOS values from the separate views is not straightforward and further study is required.

5. CONCLUSIONS

The evaluation of 3D QoE in DASH systems includes many aspects that, in general, do not make the replication of the experiments viable and complicates the possibility of comparing solutions or improvements. In order to contribute to the development of a versatile testbed that allows automating, scaling, replicating and simplifying the evaluation process, a system has been developed. The developed system is comprised from the description of the encoding process to the subjective evaluation using the ITU-T P.1203 recommendation. The provided implementation of the ITU-T P.1203 recommendation is a good first approximation for the prediction of the quality of symmetrically distorted stereoscopic videos. However, it generates substantial prediction bias when applied to asymmetrically distorted stereoscopic videos, as shown by comparing the results with those obtained through subjective evaluation by users.

For the process of network emulation and test automation, the Google Puppeteer tool and the options offered by Chrome DevTools have been used and proved helpful to manage the web client and player. In addition, the required programs have been developed in order to extract the necessary information for the reconstruction of the visualized video in the client and the adaptation of data to be properly processed with the ITU-T P.1203 recommendation software. With the use of Docker technology as part of future work, it is intended to include all the implemented tools in containers. This will ease the replication of the system by developers and researchers. The system will evolve and expand as new encoders (such as AV1, etc.) are incorporated both in the encoding process and in the reproduction and evaluation section according to the standard, as well as new players are developed, or different adaptation algorithms are proposed. However, all these new proposals do not imply changes in all modules since the adaptation can be done independently when necessary, following the evolution of the streaming technology.

ACKNOWLEDGEMENTS

This work has been partially supported by Spanish Ministry of Science, Innovation and Universities and by European Union through grant RTI2018-098085-BC41 (MCUI/AEI/FEDER), SSENCE project (Telefonica Chair UPV) , PROMETEO/2019/109 (Generalitat Valenciana) and UPV Line I+D+i "Technologies for distribution and processing of multimedia information and QoE".

REFERENCES

[1] Cisco, "Cisco Visual Networking Index : Forecast and Trends, 2017–2022 White Paper." 2019.

[2] ISO/IEC, "ISO/IEC 23009-1 Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats Technologies, 2019," 2019.

[3] T. Su, A. Javadtalab, A. Yassine, and S. Shirmohammadi, "A DASH-based 3D multi-view video rate control system A DASH-Based 3D Multi-view Video Rate Control System," 2014.

[4] D. K. Krishnappa, D. Bhat, and M. Zink, "DASHing YouTube: An analysis of using DASH in YouTube video service," in *38th Annual IEEE Conference on Local Computer Networks*, 2013, pp. 407–415.

[5] J. Martin, Y. Fu, N. Wourms, and T. Shaw, "Characterizing Netflix bandwidth consumption," *2013 IEEE 10th Consum. Commun. Netw. Conf. CCNC 2013*, pp. 230–235, 2013.

[6] T. Hoßfeld, M. Seufert, C. Sieber, T. Zinner, and P. Tran-Gia, "Identifying QoE optimal adaptation of HTTP adaptive streaming based on subjective

studies," *Comput. Networks*, vol. 81, pp. 320–332, Apr. 2015.

[7] N. Barman and M. G. Martini, "QoE Modeling for HTTP Adaptive Video Streaming-A Survey and Open Challenges," *IEEE Access*, vol. 7, pp. 30831–30859, 2019.

[8] W. Robitza, D. G. Kittur, A. M. Dethof, S. Goring, B. Feiten, and A. Raake, "Measuring YouTube QoE with ITU-T P.1203 under Constrained Bandwidth Conditions," in *2018 10th International Conference on Quality of Multimedia Experience, QoMEX 2018*, 2018, no. June.

[9] "Puppeteer." [Online]. Available: https://github.com/GoogleChrome/puppeteer. [Accessed: 01-Jul-2019].

[10] G. B. Akar, A. M. Tekalp, C. Fehn, and M. R. Civanlar, "Transport Methods in 3DTV—A Survey," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1622–1630, Nov. 2007.

[11] P. Merkle, K. Müller, and T. Wiegand, "3D video: acquisition, coding, and display," *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 946–950, May 2010.

[12] C. G. Gürler, B. Gorkemli, G. Saygili, and a. M. Tekalp, "Flexible transport of 3-d video over networks," *Proc. IEEE*, vol. 99, no. 4, pp. 694–707, 2011.

[13] A. Vetro, T. Wiegand, G. G. J. Sullivan, B. A. Vetro, F. Ieee, T. Wiegand, and G. G. J. Sullivan, "Overview of the Stereo and Multiview Video Coding Extensions of the H.264/MPEG-4 AVC Standard," *Proc. IEEE*, vol. 99, no. 4, pp. 626–642, Apr. 2011.

[14] Y. Chen, Y.-K. Wang, K. Ugur, M. M. Hannuksela, J. Lainema, and M. Gabbouj, "The Emerging MVC Standard for 3D Video Services," *EURASIP J. Adv. Signal Process.*, vol. 2009, no. 1, p. 786015, Jan. 2009.

[15] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Multi-View Video Plus Depth Representation and Coding," in *2007 IEEE International Conference on Image Processing*, 2007, vol. 1, p. I-201-I-204.

[16] M. M. Hannuksela, D. Rusanovskyy, W. Su, L. Chen, R. Li, P. Aflaki, D. Lan, M. Joachimiak, H. Li, and M. Gabbouj, "Multiview-video-plus-depth Coding based on the Advanced Video Coding Standard," *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3449–58, 2013.

[17] S. Chikkerur, V. Sundaram, M. Reisslein, and L. J. Karam, "Objective Video Quality Assessment Methods: A Classification, Review, and Performance Comparison," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 165–182, Jun. 2011.

[18] T. Wiegand and G. J. Sullivan, "The H.264/AVC Video Coding Standard," no. August 1999, pp. 148–153, 2007.

[19] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[20] O. Neṁić, S. Rimac-Drlje, and M. Vranjes, "Multiview Video Coding Extension of the H.264/AVC Standard," *ELMAR, 2010 PROCEEDINGS*. pp. 73–76, 2010.

[21] G. Tech, Y. Chen, K. Muller, J.-R. Ohm, A. Vetro, and Y.-K. Wang, "Overview of the Multiview and 3D Extensions of High Efficiency Video Coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 35–49, 2016.

[22] D. Grois, D. Marpe, A. Mulayoff, B. Itzhaky, and O. Hadar, "Performance comparison of H.265/MPEG-HEVC, VP9, and H.264/MPEG-AVC encoders," in *2013 Picture Coding Symposium (PCS)*, 2013, pp. 394–397.

[23] D. Grois, D. Marpe, T. Nguyen, and O. Hadar, "Comparative Assessment of H . 265 / MPEG-HEVC , VP9 , and H . 264 / MPEG-AVC Encoders for Low-Delay Video Applications," *SPIE Opt. Eng. + Appl.*, vol. 9217, no. June, p. 92170Q, Sep. 2013.

[24] G. Saygili, C. G. Gurler, and A. M. Tekalp, "Quality assessment of asymmetric stereo video coding," in *2010 IEEE International Conference on Image Processing*, 2010, pp. 4009–4012.

[25] J. Wang, S. Wang, and Z. Wang, "Asymmetrically Compressed Stereoscopic 3D Videos: Quality Assessment and Rate-Distortion Performance Evaluation," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1330–1343, 2017.

[26] F. Battisti, M. Carli, P. Le Callet, and P. Paudyal, "Toward the assessment of quality of experience for asymmetric encoding in immersive media," *IEEE Trans. Broadcast.*, vol. 64, no. 2, pp. 392–406, 2018.

[27] ITU-T, "ITU-T.P910. Subjective video quality assessment methods for multimedia applications," 2008.

[28] J. Li, M. Barkowsky, and P. Le Callet, "Subjective assessment methodology for preference of experience in 3DTV," *2013 IEEE 11th IVMSP Work. 3D Image/Video Technol. Appl. IVMSP 2013 - Proc.*, no. 1, pp. 1–4, 2013.

[29] H. K. Yarnagula, S. Luhadia, S. Datta, and V. Tamarapalli, "Quality of Experience Assessment of Rate Adaptation Algorithms in DASH : An Experimental Study," pp. 1–8, 2016.

[30] Jingteng Xue, Dong-Qing Zhang, Heather Yu, and Chang Wen Chen, "Assessing quality of experience for adaptive HTTP video streaming," in *2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, 2014, pp. 1–6.

[31] A. Bentaleb, B. Taani, A. C. Begen, C. Timmerer, and R. Zimmermann, "A survey on bitrate adaptation schemes for streaming media over HTTP," *IEEE Commun. Surv. Tutorials*, vol. 21, no. 1, pp. 562–585, Jan. 2019.

[32] S. Akhshabi, A. C. Begen, and C. Dovrolis, "An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP," in *Proceedings of the second annual ACM conference on Multimedia systems - MMSys '11*, 2011, p. 157.

[33] C. Timmerer, M. Maiero, and B. Rainer, "Which Adaptation Logic ? An Objective and Subjective Performance Evaluation of HTTP-based Adaptive Media Streaming Systems."

[34] P. Juluri, V. Tamarapalli, and D. Medhi, "SARA: Segment aware rate adaptation algorithm for dynamic adaptive streaming over HTTP," in *2015 IEEE International Conference on Communication Workshop, ICCW 2015*, 2015.

[35] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hobfeld, and P. Tran-Gia, "A Survey on Quality of Experience of HTTP Adaptive Streaming," *IEEE Commun. Surv. Tutorials*, vol. 17, no. 1, pp. 469–492, Jan. 2015.

[36] A. Zabrovskiy, E. Kuzmin, E. Petrov, C. Timmerer, and C. Mueller, "AdViSE: Adaptive Video Streaming Evaluation Framework for the Automated Testing of Media Players," *Proc. 8th ACM Multimed. Syst. Conf. - MMSys'17*, pp. 217–220, 2017.

[37] Q. Chen, Yanjiao ; Wu, Kaishun ; Zhang, “From QoS to QoE: A Tutorial on Video Quality Assessment,” *IEEE Commun. Surv. Tutorials*, vol. 17(2), pp. 1126–1165, 2015.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image Quality Assessment: From Error Visibility to Structural Similarity,” *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[39] H. R. Sheikh and A. C. Bovik, “Image information and visual quality,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, vol. 3, no. 2, pp. 430–444, 2004.

[40] Z. L. Li and A. Aaron, “Toward A Practical Perceptual Video Quality Metric,” *Netflix TechBlog*. .

[41] ITU-T, “ITU-T. P915. Subjective assessment methods for 3D video quality,” 2016.

[42] T. Maki, M. Varela, and D. Ammar, “A Layered Model for Quality Estimation of HTTP Video from QoS Measurements,” *Proc. - 11th Int. Conf. Signal-Image Technol. Internet-Based Syst. SITIS 2015*, no. January 2016, pp. 591–598, 2016.

[43] R. K. P. Mok, E. W. W. Chan, and R. K. C. Chang, “Measuring the quality of experience of HTTP video streaming,” in *12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops*, 2011, pp. 485–492.

[44] ITU Telecommunication Standardization Sector, “ITU-T Rec P 1203 Models and tools for quality assessment of streamed media,” 2017.

[45] W. Robitza, S. Göring, A. Raake, D. Lindegren, G. Heikkilä, J. Gustafsson, P. List, B. Feiten, U. Wüstenhagen, M.-N. Garcia, K. Yamagishi, and S. Broom, “HTTP Adaptive Streaming QoE Estimation with ITU-T Rec. P. 1203: Open Databases and Software,” *Proc. 9th ACM Multimed. Syst. Conf.*, pp. 466–471, 2018.

[46] R. I.-T. J.247, “Objective perceptual multimedia video quality measurement in the presence of a full reference.,” *Recomm. ITU-T J.247*, 2008.

[47] P. G. Castillo, P. A. Vila, and J. C. G. Cebollada, “Automatic QoE evaluation of DASH streaming using ITU-T standard P.1203 and google puppeteer,” *PE-WASUN 2019 - Proc. 16th ACM Int. Symp. Perform. Eval. Wirel. Ad Hoc, Sensor, Ubiquitous Networks*, pp. 79–86, 2019.

[48] “Shaka Player Demo.” [Online]. Available: https://shaka-player-demo.appspot.com. [Accessed: 01-Jun-2019].

[49] P. Fiedler, M.; Hossfeld, T.; Tran-Gia, “A generic quantitative relationship between quality of experience and quality of service,” *Network, IEEE*, vol. 24, no. April, pp. 36–41, 2010.

[50] A. Raake, M. N. Garcia, W. Robitza, P. List, S. Göring, and B. Feiten, “A bitstream-based, scalable video-quality model for HTTP adaptive streaming: ITU-T P.1203.1,” *2017 9th Int. Conf. Qual. Multimed. Exp. QoMEX 2017*, vol. 12, pp. 0–5, 2017.

[51] “Big Buck Bunny.” [Online]. Available: https://peach.blender.org. [Accessed: 01-May-2019].

[52] S. Lederer, C. Müller, and C. Timmerer, “Dynamic Adaptive Streaming over HTTP Dataset,” pp. 89–94, 2012.

Paola Guzmán Castillo received her Electronic Engineering degree and the M.Sc. degree in Electronic Engineering from Industrial University of Santander, Bucaramanga, Colombia, in 2001 and 2005 respectively. From 2006 to 2010 she worked as professor at St. Thomas University and Industrial University of Santander, Bucaramanga, Colombia. In 2014 she received her M.S. in Telematics from the Universitat Politècnica de València (UPV), Spain. Currently, she is a researcher at the Multimedia Communications research group (COMM) of the Institute of Telecommunications and Multimedia Applications (iTEAM), UPV, where she is working towards the Ph.D. degree. Her current research interests include multimedia QoE and adaptive video streaming over IP networks.

Pau Arce, received his Telecommunications Engineering degree, M.S. in Telematics and the Ph.D. degree in Telecommunications from the Universitat Politècnica de València (UPV), Spain, in 2005, 2007 and 2014 respectively. Currently he works as a researcher at the Institute of Telecommunications and Multimedia Applications (iTEAM). His research interests include multimedia QoS, routing on wireless ad hoc networks and performance evaluation of computer systems.

Juan Carlos Guerri, received the M.S. and Ph.D. degree, both in telecommunication engineering, from the Universitat Politècnica de València (UPV), Valencia, Spain, in 1993 and 1997, respectively. He is a Professor in E.T.S. Telecommunications Engineering at the UPV, where he leads the Multimedia Communications research group (COMM) of the iTEAM Institute. He is currently involved in research and development projects for the application of multimedia to industry, medicine, education, and communications.