A DASH Based HEVC Multi-View Video Streaming System

Tianyu Su, Ashkan Sobhani, Abdulsalam Yassine, Shervin Shirmohammadi, Abbas Javadtalab

Distributed and Collaborative Virtual Environments Research Laboratory (DISCOVER), University of Ottawa, Ottawa, Canada

tsu083@uottawa.ca , asobh034@uottawa.ca,{ayassine | shervin | javadtalab}@discover.uottawa.ca

Abstract

Recent advancement in cameras and image processing technology has generated a paradigm shift from traditional 2D and 3D video to Multi-view Video (MVV) technology, while at the same time improving video quality and compression through standards such as High Efficiency video Coding (HEVC). In multi-view, cameras are placed in predetermined positions to capture the video from various views. Delivering such views with high quality over the Internet is a challenging prospect, as MVV traffic is several times larger than traditional video since it consists of multiple video sequences each captured from a different angle, requiring more bandwidth than single view video to transmit MVV. Also, the Internet is known to be prone to packet loss, delay, and bandwidth variation, which adversely affects MVV transmission. Another challenge is that end users' devices have different capabilities in terms of computing power, display, and access link capacity, requiring MVV to be adapted to each user's context. In this paper, we propose an HEVC Multi-View system using Dynamic Adaptive Streaming over HTTP (DASH) to overcome the above mentioned challenges. Our system uses an adaptive mechanism to adjust the video bitrate to the variations of bandwidth in best effort networks. We also propose a novel scalable way for the Multi-view video and Depth (MVD) content for 3D video in terms of the number of transmitted views. Our objective measurements show that our method of transmitting MVV content can maximize the perceptual quality of virtual views after the rendering and hence increase the user's quality of experience.

Keywords

High Efficiency Video Coding (HEVC), Dynamic Adaptive Streaming over HTTP (DASH), Multi-view plus Depth Coding(MVD),Multi-view Video Coding Video Scalability and Adaptation, Quality of Experience, Depth Image Based Rendering, Rate Distortion Optimization.

1. Introduction

Traditional 3D video representation is usually achieved with two cameras. Users can observe the 3D scene by wearing shutter goggles or polarized goggles [1]. Although, stereoscopic immersive 3D video is popular both in theaters and in home entertainment, the flexibility for the users is low [1, 6] due to the fixed conditions in which the stereo content is captured as can be shown in Figure 1(a). Recent advancement in camera and image processing technology has generated a paradigm shift from traditional 2D and 3D video to multi-view video technology. In multi-view video (MVV), cameras are placed in predetermined positions and angles to capture the video sequences. The video sequences of MVV are fixed [2]. MVV content is then compressed and transmitted in a suitable way so that the users' viewing device can easily access the relevant views to interpolate new views. Multi-view representation allows the users to freely change their viewpoints [2].

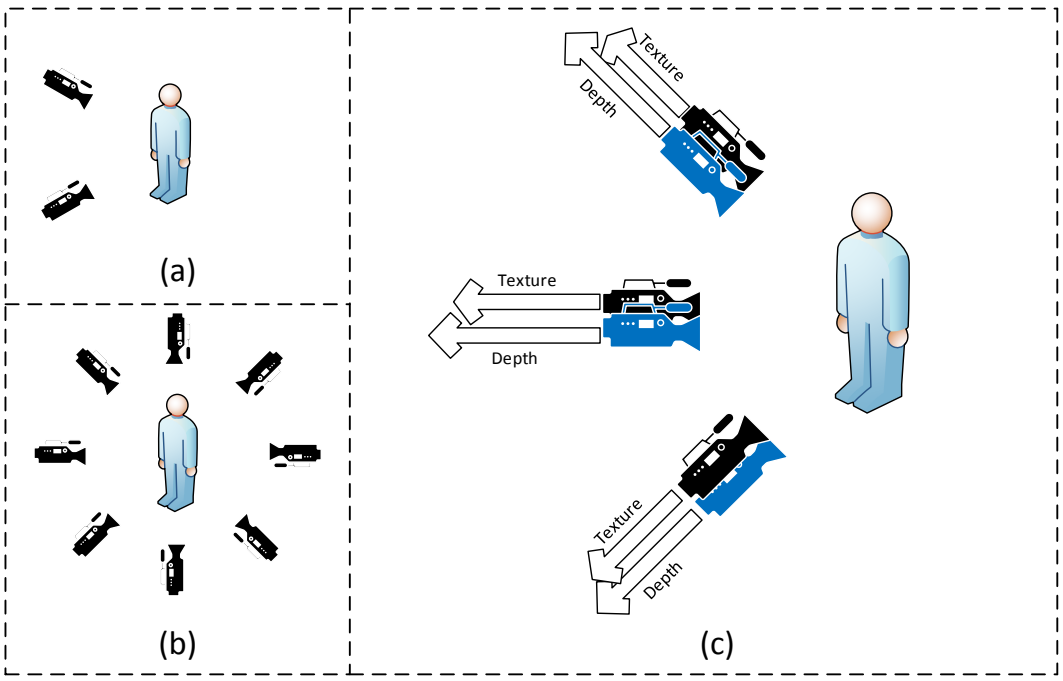

Figure1. Different camera capture ways

Currently, Multi-view Video Coding (MVC) for MVV or Free Viewpoint Video (FVV) is standardized as an extension of H.264/MPEG-4 Advanced Video Coding (AVC). This technology is used in several applications such as stereoscoptic 3D video (temporal interleaving or spatial interleaving), FVV, and auto-stereoscoptic 3D video [2]. Furthermore, the Free Viewpoint Television (FTV) is based on this concept, which allows the users to interactively control the

viewpoint and generate new views of a dynamic scene from any 3D position [3] as can be seen from Figure1(b). This means providing a realistic feeling of natural interaction to the users. A more recent Multi View plus Depth (MVD) format [4, 5] encodes a depth signal for each camera, as illustrated in Figure1(c). The main advantage of MVD is that it allows synthesizing virtual views at the client via the depth signals. These signals are sent along MVC by means of Depth Image Based Rendering (DIBR) that can render several virtual views based on the few real views and their associate depth map [4]. The DIBR method can convert transmitted multi-views to more virtual views for auto stereoscoptic 3D display in order to efficiently represent the 3D views to transmit. The popularity of this technology has gained attention from researchers and practitioners to develop innovative 3D techniques to meet new demands.

While MVV and MVD technologies are very promising, the development of appropriate mechanisms that support the delivery of these technologies to the end users over best effort networks is not progressing at the same speed [6]. There are two major challenges encumbering the delivery of MVV over present networks [6]: First, MVV traffic is several times larger than traditional video since it consists of multiple video sequences captured by multiple cameras, which means more bandwidth is required to transmit it. Since the Internet is prone to packet loss, delay, and bandwidth variation, MVV transmission is even more challenging in the presence of such network lag. Second, end users' devices have different capabilities in terms of computing power, display, and access link capacity, which requires adaptive mechanisms to adjust for the variations of video characteristics while traversing the network's paths.

The above challenges motivated us to propose a dynamic rate adaptation system and its associated rate-distortion model for multi-view 3D video transmission, which will address the issue of varying network bandwidth for Internet consumers of multi-view video. Our rate adaptation system is built on top of two state-of-the-art key technologies: High Efficiency Video coding (HEVC) [7, 8], and MPEG's Dynamic Adaptive Streaming over HTTP (DASH) [11, 12]. HEVC, introduced by the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group, provides about 50 percent bit rate reduction at the same video quality compared to H.264 [9]. Furthermore, HEVC 3D extension [10] is flexible in generating a bit stream format that is suitable for different setups: from traditional

2D to stereo 3D and Multi-view video. MPEG-DASH [11,12] divides the video contents into segments with equal length and stores them in a server. The copies of the video segments are encoded with different bit rates that represent different qualities and resolutions, as shown in Figure2.

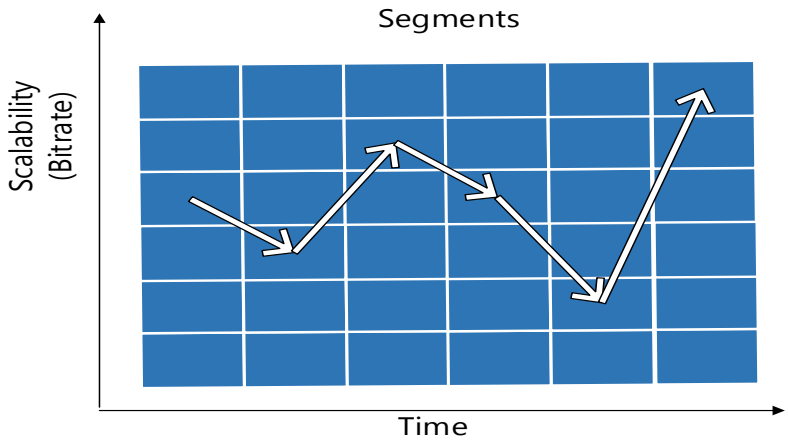

Figure2 Scalability of Video content in DASH

In MPEG-DASH, all segments that are stored in the server can be accessed via an XML-based Media Presentation Description (MPD) file. All streaming sessions are managed by the clients; i.e., a client chooses a bit rate by requesting a specific MPD based on the network condition and its decoding process. Our system uses an adaptive mechanism to adjust for the variations of bandwidth through a novel scalable method for MVD content in terms of the number of transmitted views. Our objective tests show that our method of transmitting multi-view video content can maximize the perceptual quality of virtual views and hence increase the user's quality of experience.

The rest of this paper is organized as follows: In the next section, we discuss the background and related work. In section 3, we present our proposed system. The simulation and the objective test results of the proposed system are provided in section 4. Finally, in section 5, we provide the conclusion and our future work.

2. Background and Related Work

In this section, we present an overview of related work relevant to the proposed study. In particular, we examine a sample of studies in 2D live streaming bit rate

adaptation, DASH and 3D streaming, and multi-view plus depth that we believe to be representative and specifically related to our work. We also describe in this section the main advantages and drawbacks of existing work and the needs for alternative mechanisms as proposed in this paper.

2.1 Bitrate Adaptation for 2D High Definition Video

In order to reach a given bit rate in conventional 2D video streaming, some parameters inside the encoder such as the Quantization Parameters (QP) are adjusted based on the rate distortion model for the lossy coding [14]. Also, changing the resolution of the video frame, and the frame rate can be useful for bitrate adaptation. In [22], a method for available bandwidth detection over best effort networks is presented, specifically for video streaming. This algorithm is used to ensure the video is adapted to available bandwidth in time to provide the highest quality at that bit rate level.

2.2 Bitrate Adaptation for HTTP live Streaming

HTTP plays an important role as a protocol of delivery for video streaming [15, 16]. HTTP is pervasive and it can pass through all firewalls and Network Address Translators (NATs). Video streaming deployments that are based on HTTP would not impose considerable costs in comparison to other protocols. This is one of the main reasons that most popular video hosting service providers such as Apple HTTP live steaming [17], Microsoft smooth streaming [18], Adobe HTTP dynamic streaming [19], YouTube, and Akamai [20], prefer to use HTTP compared to alternatives such as the Real-Time Transport Protocol (RTP). Traditionally, approaches that used HTTP required progressive download, which uses HTTP for playing out the online video contents [11]. But this did not support the main aspects of real streaming like adaptively changing the resolution and quality with respect to network conditions. Viewers must select the most suitable representation (bit rate) before playing out the video; otherwise they may experience interruption and freezes if the network condition is changed during the play time [11]. To overcoming the limitations of progressive download, adaptive streaming [17-20] is has been proposed to resolve the drawbacks while trying to retain the simplicity of progressive download. Such proposals were behind the creation of Dynamic Adaptive Streaming over HTTP (DASH) standard, which

was introduced by the Motion Picture Experts Group (MPEG) and 3rd Generation Partnership Project (3GPP) with the goal of integrating all individual efforts in adaptive streaming [15]. Like other adaptive streaming methods, each video in DASH is encoded and compressed into a variety of video bitrates corresponding to different resolutions and qualities. It is worth noting that DASH is encoder-agnostic so that HEVC can be easily used along with DASH. These compressed versions are called different representations of the video. After that, all representations are fragmented to several segments usually with constant duration in order of a few seconds. These segments are then stored in common web servers in a company with a generated XML-based file called Media Presentation Description (MPD), which is sent to the client by the server to determine the available representations and corresponding URLs. DASH is a pull-based method [15] that allows the client to start playing the video by asking for the MPD file using HTTP GET requests. After becoming aware of the available videos by parsing the received MPD file, the client sends requests for fetching and downloading the appropriate segments based on its knowledge about the conditions of the network, like incoming bandwidth, and the status of incoming buffer [21].

2.3 Dynamic Adaptive Streaming Over HTTP for Free Viewpoint Video Streaming and Stereo 3D Streaming

A DASH based stereoscopic 3D video method is proposed in [23], which encodes the stereo views in a scalable way on the server to adaptively stream the 3D stereo content using DASH. In [24], a DASH-based free viewpoint video streaming system is proposed. An adaptation mechanism is used to maximize the virtual views based on the rate distortion model that is rendered from the texture reference views and its associate depth map with the tracking of the user's head at the client side. The authors' methodology is to choose the best quality of synthesized views between two reference views in one fixed baseline distance. However, in spite of its the final goal to provide the best virtual view quality to users, it ignores the user's experience about depth; in other words, whether different users feel comfortable with the rendered views. Furthermore, the head tracking process may introduce more delay in the system, which negatively affects DASH streaming.

In [25], the design of a DASH based stereo 3D video system is introduced. While the approach is interesting, stereo 3D only provides two views to the viewers and different viewers have different preferences over the depth that makes them feel comfortable.

2.4 High Efficiency Video Coding (HEVC)

High Efficiency Video Coding (HEVC) [26], introduced by the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group, is a hybrid video codec design. Compared to H.264/AVC [27] and MPEG-2 [28], HEVC has the following improvements:

- Hierarchical Quad Tree Structure: introduces a quad-tree coding structure within a picture including Coding Tree Unit (CTU), the smaller Coding Unit (CU) that is split by CTU, and further Prediction Units (PU) from CU that is used for intra and inter prediction inside the CU. The Transform Unit is furthered by PU, which defines for the transform (e.g. DCT) and Quantization [26].
- The larger size of "Macroblock", the Coding Tree Unit consists of a luma Coding Tree Block (CTB) and Chroma CTBs, which is analog to Macroblock in H.264/AVC [7]. The size of CTB is selected by the encoder and can be larger (64×64) than traditional Macro block (16×16) introduced in previous standards (e.g.H.264 and MPEG2). The larger size of CTU is good for better compression performance for High Definition or 4K video.
- Parallelization Design: In order to accelerate the speed of the codec to tackle the issue of improvement of the computation complexity, parallelization design for HEVC introduces Tiles, which are several rectangular parts inside each frame.
- Support for 3D extension: The new HEVC standard not only tackles video compression for 2D High Definition Video and 4K Video, but also support for the views plus depth format of 3D multi-view video [5].

2.5 Objective and Subjective Approach for the Multi-view Plus Depth content

In [29] and [30], the objective methodology and approach for the multi-view video were proposed to build suitable objective quality assessment metrics for different scalable modalities in multiview 3D video.

The most related work to ours is [31], it is a subjective test approach for streaming MVD content that examines the effect of the number of views on the quality of the synthesized views. The subjective study shows that by decreasing the distance of the baseline and the number of transmitted views, one can maintain a satisfactory subjective quality. The study uses the Constant Quantizer Parameter (CQP) method to encode the different perceptual qualities of the content by setting one specific QP. This method; however, is not part of the MPEG-DASH standard [15, 16]. Compared with the Constant Bit Rate (CBR) controller that is based on the R-Lambda Model [9], it cannot guarantee the best quality at one specific bit rate given the fluctuation of the bandwidth.

In [32]，the study seeks the best bit rate allocation ratio in terms of the rendered synthesized views using the depth map image rendering technique between the depth map and the texture views in the multi-view plus depth map compression. Experiments are based on both H.264/MVC and the HEVC 3D extension. The results show that even though the optimal ratio varies from different test sequences, the best ratio is between 30 to 60 ratio depth to texture in percentage regarding to the PSNR value. Such conclusion is used in our experiment to set the parameters in the DASH server.

3. The Proposed System

The architecture of our HEVC- multi-view video system using dynamic adaptive streaming over HTTP is shown in Figure 3 and Figure 4. The architecture shows the DASH server side and the client side. The main goal of the proposed architecture is to adaptively transmit the multi-view plus depth map content for 3D auto-stereoscopic video over the Internet. Both the server side and the client side of the architecture are described in detail in the following subsections.

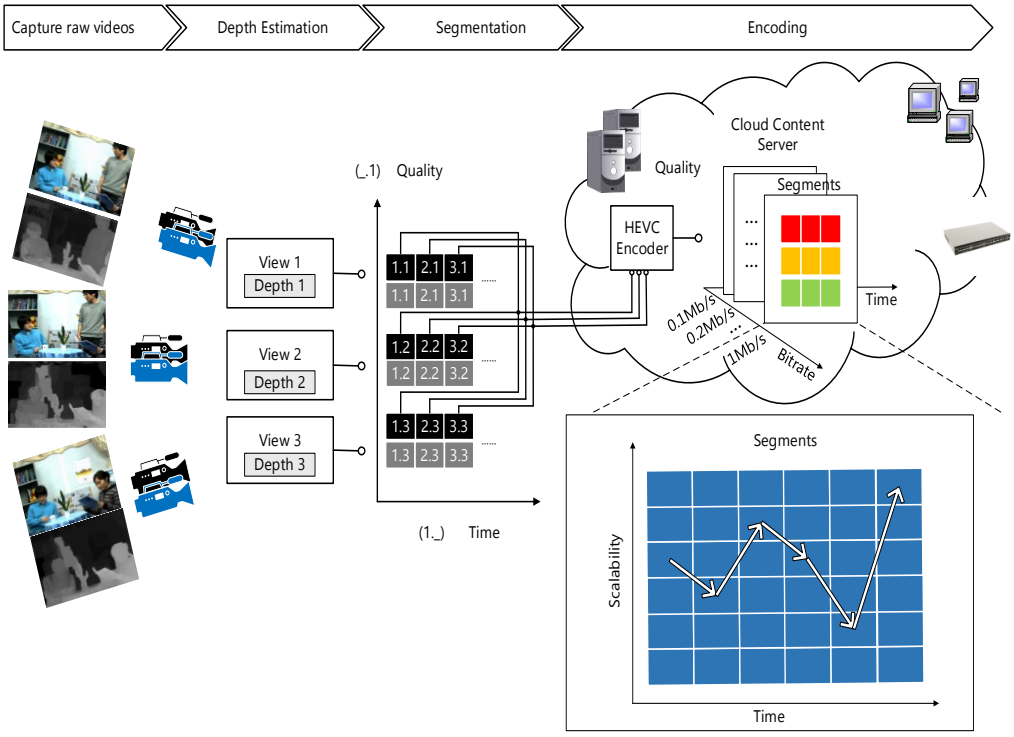

Figure 3 DASH based multi-view video transmission system on the Server Side

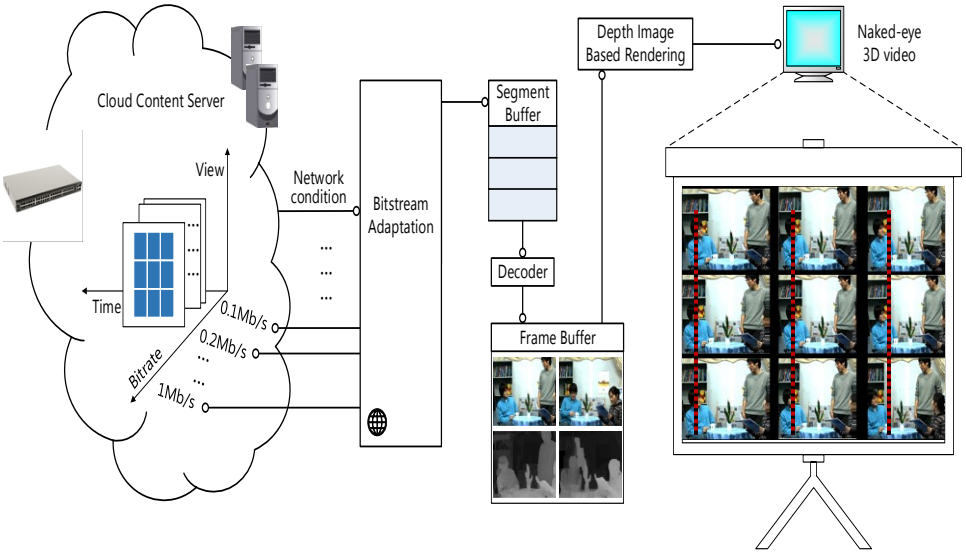

Figure 4 DASH based multi-view video transmission system on the Client Side

3.1 The DASH Server

The responsibility of the DASH server is to offer various versions of the video so that each client can adaptively select the video representation segment (i.e. the video bit rate) according to the network condition. There are several factors that

contribute to the server's selection strategy such as network congestion, the available bandwidth, the capacities of the buffer size on the client side, and the resolution of the client display. It must be noted that the choice of the video segment cannot be decided at the very beginning of the transmission period due to the stochastic nature of the bandwidth over best effort networks as well as the diversity of the video content. Therefore, the DASH server divides the whole video into temporal segments, for which each segment contains the video range from 2 to 10 seconds. Then it encodes each segment into different video bit rates and groups them into an adaptation set. The ultimate goal is to allow the DASH client to switch among different video bit rates according to the available bandwidth. In our DASH server, we use MVD to represent the multi-view video, in other words, a scene is captured from a series of cameras from various viewpoints, and the associated depth information using depth cameras such as Kinect or depth map creation methods as in [1]. Furthermore, we use the HEVC 3D extension encoder in our server encoding engine [10] because it provides better compression efficiency compared to H.264/AVC as shown in [9]. The Rate-Lamda model in HEVC [33] provides the highest compression quality of bit streams at the target bitrate. This model can allocate the bit at Group of Picture (GOP) level as well as at Picture Level and Large Coding Unit (LCU) level. We revise the HEVC 3D extension encoder to encode multiple versions of segments in specific target bitrates and then store them in the server. The raw video sequences are segmented with equal duration time. Then, these segments are fed to the encoding engine to produce different bit rates of the MVD video segments. These segments are then stored with their MPD file in the server, most likely in a cloud.

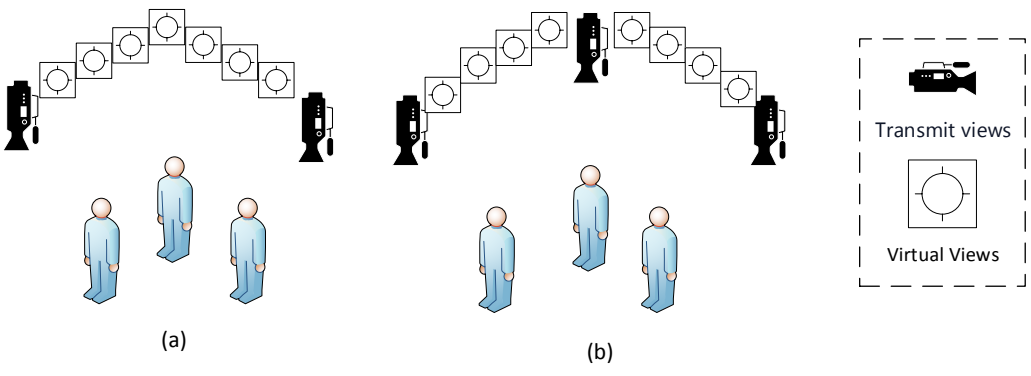

Figure 5 View Scalability Scenarios:

(a) 2View plus Depth with larger baseline. (b) 3View plus Depth with narrow baseline

Currently, there are several modes of scalability for 2D video including quality, temporal and spatial to produce the variant video bit rate based on priorities of perceptual quality [15]. For multi-view video, we use the number of transmitted views as a new scalability, as proposed in [34]. Our idea is to consider view scalability along with the other mentioned scalabilities to adapt to the available bandwidth. By increasing the number of the transmitted views (or decreasing the distance between the cameras) as can be shown in Figure 5b, we will be able to produce virtual views with higher quality than the one produced using larger distance between the cameras as shown in Figure 5b. By so doing, the client can adapt its requests for video segments with different number of views, lower or higher number of views, according to the available bandwidth. In turn, this has significant effect on the perceived quality of experience as we will show in our results.

3.2 The Adaptation Client

The adaptation client manages all transmission sessions when the client selects the video bit stream, which is scalable for each temporal segment such that the user can perceive the maximum quality of the 3D content given the available bandwidth. The client in DASH starts playing a certain video by asking for the MPD file using HTTP GET requests. After parsing the MPD file as can be seen on Figure 6, the client knows the representation of the content in the server, the decision about which version of segment to download is decided by the client. In DASH-based systems, the client controls the streaming session and manages the adjustment of video bit rates in reaction to network conditions, incoming bandwidth, and status of playback buffer. The adaptation algorithm on the client switches between different versions of the temporal segments.

In Figure 4, we show the segment buffer, which provides a safe margin in case of sudden decrease in the network bandwidth [39], but also can be useful for predicting the available bandwidth along with the throughput of the downloaded segment. The following subsection will show the algorithm of bitrate selection in the adaptation client.

```
</BaseURL>
<Period duration="PT1M">
<AdaptationSet segmentAlignment="true" group="1">
<Representation id="1" mimeType="video/hevc" codecs="HEVC3D" width="1024" height="768" frameRate="
30" numberofViews=3 sar="1:1" startWithSAP="1" bandwidth="1224800">
<SegmentList timescale="1000" duration="10001">
<Initialization sourceURL="
video/newspaper_10sec/300kbps/newspaper_720p_300kbps_3view_10sec_segmentinit.hevc"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1200kbps_3views_10sec_segment1.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1200kbps_3views_10sec_segment2.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1200kbps_3views_10sec_segment3.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1200kbps_3views_10sec_segment4.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1200kbps_3views_10sec_segment5.hevc
"/>
....
....
....
</SegmentList>
</Representation>
<Representation id="4" mimeType="video/hevc" codecs="HEVC3D" width="1024" height="768" frameRate="
30" numberofViews=2 sar="1:1" startWithSAP="1" bandwidth="1896448">
<SegmentList timescale="1000" duration="10001">
<Initialization sourceURL="
video/newspaper_10sec/300kbps/newspaper_720p_300kbps_2views_10sec_segmentinit.hevc"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1800kbps_2views_10sec_segment1.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1800kbps_2views_10sec_segment2.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_720p_1800kbps_2views_10sec_segment3.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_710p_1800kbps_2views_10sec_segment4.hevc
"/>
<SegmentURL media="video/newspaper_10sec/300kbps/newspaper_710p_1800kbps_2views_10sec_segment5.hevc
```

Figure6 A template of Multimedia Presentation Description (MPD) for the newspaper sequence

3.2.1 Bitstream Selection

The client is responsible for deciding which bitstream to use for different available bandwidths. In our system, the selection of the bitstream is based on current network conditions that can maximize the perceptual quality of the average rendered views. In section 4.2, we will introduce our objective measurement test results, which show the effect of different number of transmitted views at the same total bitrate level on the final perceptual quality of rendered views and transmitted views. We use these result to further set a policy to switch the bitrate selection from less or more number of views. Naturally, the selected total bitrate of the generated bitstream should be lower than the available bandwidth, and we further assume that each of the views shares the total bitrate equally, therefore, the total bitrate used in MVD can be expressed as:

$$\text{TotalBitrate} = Bitrate_{view} \times i \qquad (1)$$

where $\text{Bitrate}_{\text{view}}$ represents the bitrate per view, TotalBitrate represents the total bitrate of the bitstream, and *i* represents the number of transmitted views.

Although the adaptation logic in conventional DASH players selects the most appropriate video bitrate among the available representations, it only looks for the largest total video bitrate, which is less than the available network bandwidth. It is worth mentioning that our method complements conventional methods. We first use a conventional method to select the largest representations which are less than available network bandwidth. However, in MVD, it is possible to have the same video stream in terms of bitrate with different number of views; hence, we can use equation (2) to select the more suited video segment according to the corresponding computed SSIM.

$$\text{TotalBitrate} = \begin{cases} Bitrate_{view} \times i & \text{if } AvSSIM(i) > AvSSIM(i-1) \\ Bitrate_{view} \times (i-1) & \text{if } AvSSIM(i-1) > AvSSIM(i) \end{cases} \quad (2)$$

$AvgSSIM(i)$ represents the average of the SSIM value of all rendered views compared with the raw views without quality loss of compression. We show that in some specific total bitrates or available bandwidths, the user can ask the server to decrease or increase the number of transmitted views.

3.2.2 Available bandwidth prediction

We implemented a smoothed throughput based available bandwidth prediction method, which predicts the available bandwidth by moving average of the observed throughputs. This algorithm determines the optimal quality level considering the moving average of the throughput of downloading segments measurement Th_{inst} , the estimated throughput can be represented as:

$$Th_{est}(\mathrm{t}+1) = \begin{cases} (1-\alpha) \times Th_{est}(\mathrm{t}) + \alpha \times Th_{inst}(\mathrm{t}), & \text{if } \mathrm{t} > 0 \\ Th_{inst}(\mathrm{t}) & \text{if } \mathrm{t} = 0 \end{cases} \quad (3)\ [43]$$

Where $Th_{inst}(\mathrm{t})$ represents the instant throughput measurement, t represents the order of segment sequence downloaded, Th_{est} is the estimated throughput or available network bandwidth., and α is a weighting value. Our algorithm of bitstream selection is as follows:

Algorithm 1: Moving average throughput smoothed bitstream selection Algorithm

Input: Instant throughput Th_{inst}, playlist t, counter, level of video Representation S_n (1 representates the lowest quality level), number of transmitted views in each video representation $S_n(i)$, Bitrate of representation for Nth video representations $Totalbitrate_n$

Begin

if counter>0

Download from the minimum video bitrate:

$$S_1$$

Update the estimated throughput:

$$Th_{est}(\mathrm{t}+1) \leftarrow Th_{inst}(\mathrm{t})$$

Counter--

else

Calculate the available bandwidth based on the Lookup table**:**

$$Th_{est}(\mathrm{t}+1) \leftarrow (1-\alpha) \times Th_{est}(\mathrm{t}) + \alpha \times Th_{inst}(\mathrm{t})$$

Find the suitable representations in server for $Th_{est}(\mathrm{t}+1)$:

$$Totalbitrate_{n-1} \leq Th_{est}(\mathrm{t}+1) \leq Totalbitrate_n$$

$$Download\ S_n$$

While

Number of candidates for Nth representations

Number ($Totalbitrate_n$)> 1

do

Decide number of transmitted views based on (2)

if $AvSSIM(i) > AvSSIM(i-1)$

$$Download\ S_n(i)$$

else

$$Download\ S_n(i-1)$$

end if

Counter --

end if

As can be seen from algorithm (1), the prediction bandwidth has the following characteristics,

- The first time, it starts with the lowest quality of segments.
- The Moving Average of the throughput from the last downloaded segments is used for estimating the available network bandwidth.
- Multiple step switching; i.e., the selected quality level can be adjusted up and down based on the moving average of the estimated throughput. In

some specific total video bitrates provided by the server, the client can decide the different number of transmitted views to download based on the computed SSIM values. The switching operations are in the *while* loop in algorithm (1).

The main advantages of this algorithm are: First, it efficiently utilizes the available bandwidth and it is sensitive to the changes in estimated available bandwidth. Second, it uses a new scalable way for the MVD content for 3D video in terms of the number of transmitted views. This will further maximize the perceptual quality of virtual views after the rendering and hence increase the user's quality of experience.

3.2.3 Reconstruction based on the MVD format

The reconstruction of the MVD representation for potential 3D content is acquired after the decoding process, as can be seen from Figure 5. The rendering software introduced in [10] and [35] have already been proven to be better than MPEG VSRS in terms of both SSIM and PSNR for rendering the synthesised views from the MVD video. After the rendering process, the MVD video representation can produce multiple virtual views for auto-stereoscopic 3D display. The reason that we render the virtual views in the client side is to avoid the transmission of a large number of virtual views, which might not be optimal in the case of best effort networks. The number of virtual views plus multi-views ranges from 9 to 27 [36, 37], which will linearly increase the bandwidth burden.
In the next section, we provide the simulation results.

4. Simulation

4.1 Simulation Setup

In this section, we provide the simulation setup of our proposed. In order to test and evaluate our proposed system, we use Kendo [38] and Newspaper [39], which are recommend by MPEG [40]. The properties of the test sequences are listed in Table 1. We set the Group of Picture length and the Intra Period to 8 and 24 respectively for all test sequences, and the segment duration to 10 seconds for

both the Newspaper and Kendo test sequences. Since longer sequences are not possible with the above mentioned sequences, we had to repeat each of the test sequences ten times so that each test sequence can be considered as a segment.

Table1. Properties of the test sequences

Test Sequences Name	Frame Rate per second	Resolution Width*Height	Views	Length of the sequence	Distance of baseline
Newspaper	30	1024*768	2,4,6	300	5cm
Kendo	30	1024*768	1,3,5	300	5cm

As mentioned before, we use the HEVC 3D extension Encoder HM 11.0 [10]. The Sample Adaptive Offset (SAO) is enabled. In order to show how our policy of downloading segments works, we prepared two types of streams. The first type consists of 2 views plus their corresponding depths (2 V+D) while the second type includes 3 views plus corresponding depths (3 V+D). For preparing the first type of stream, 2V+D, we used views number 2 and 6, in Newspaper, and views number 1 and 5 in Kendo. For the second type of stream, 3V+D, we used the rest of the views in the aforementioned test sequences plus their related depths to emulate the streaming of higher number of views.

The segments in the 2V+D streams are encoded as follows: 300kbps, 500kbps, 800kbps, 1000kbps, 1200kbps, 1500kbps, and 2000kbps, using Constant Bit Rate (CBR) per view. So the total bit rates of different 2V+D streams are 1200kbps, 2000kbps, 3200kbps, 4000kbps, 4800kbps, and 6000kbps respectively. As pointed out in [4], the view and its depth have equal bit rate. For example, we have 2 views each at 300kbps and 2 depths at 300kbps to produce a stream at 1200kbps bit rate. In a similar way, we used the same encoded video bit rates for 3V+D such that the total 3V+D streams ranges from 1800kbps to 9000kbps. The segments belonging to all representations of 2V+D streams and 3V+D streams, as well as the MPD file are stored in the Internet Information Service (IIS) HTTP server[41], as described in Figure 6. We used the DummyNet tool [42] at the client side and we set the initial bandwidth to be 2.0 Mbps, which is then

increased by 1.0 Mbps after every 2 segments. The software introduced in [32], was chosen for rendering the virtual viewpoints based on the MVD content received by the client. The BlendMode and HoleFillingMode parameters are enabled. It is worth noting that for both of 2V+D and 3V+D experiments the total number of views including virtual views and the transmitted views are the same. For 2V+D, we rendered 7 virtual viewpoints between transmitted views, while for 3V+D, 6 virtual viewpoints were rendered so that there are 3 viewpoints between two transmitted views as can be shown in Figure 5.

4.2 Objective Quality Measurement

In this section, by using the well-known objective metrics Signal to Noise Ratio (PSNR) and the Structural Similarity (SSIM), we show how the different number of transmitted views (decreasing or increasing the distance of cameras) could affect the quality of virtual views and eventually the quality of user experience. The result of this experiment will be used to decide which policy for transmitting the MVD content for auto-stereoscopic 3D display would be better in terms of QoE under different circumstances.

We use a power curve fitting the f(bitrate) curve, which is computed in Equation (4) in order to linearly predict the perceptual quality in terms of PSNR and SSIM.

$$\mathrm{f(bitrate)} = \mathrm{a}\ bitrate^{b} + \mathrm{b} \qquad (4)$$

The inverse function $f^{-1}(\mathrm{Quality}) = \sqrt[b]{\frac{Quality-c}{a}}$ (5)

Based on the inverse function, the possible qualities of different bitrates can be predicted linearly.

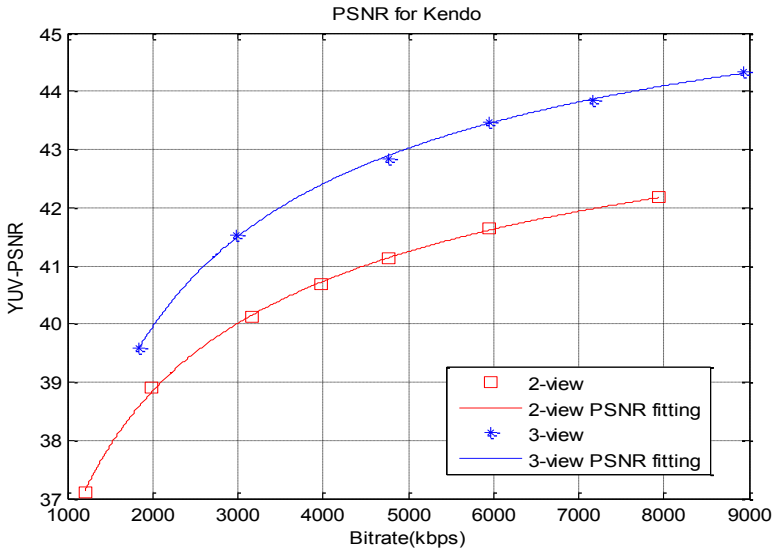

Figure7 (a) PSNR for Kendo

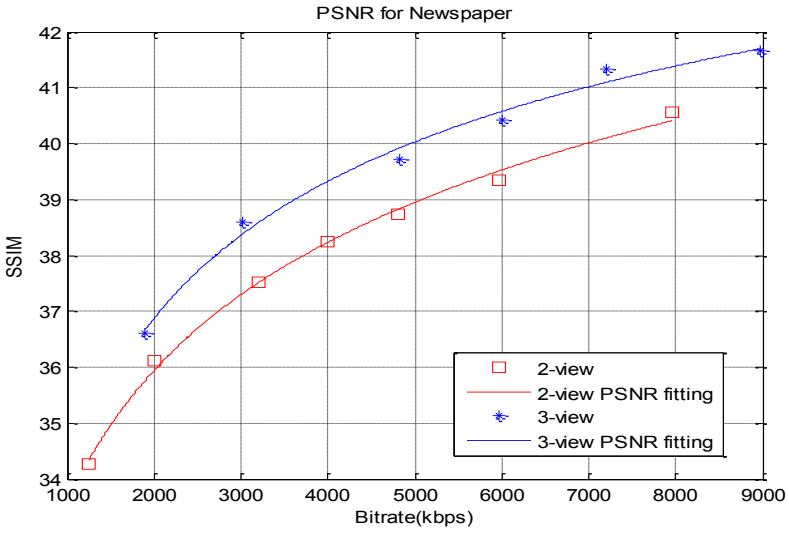

Figure 7(b) PSNR for Newspaper

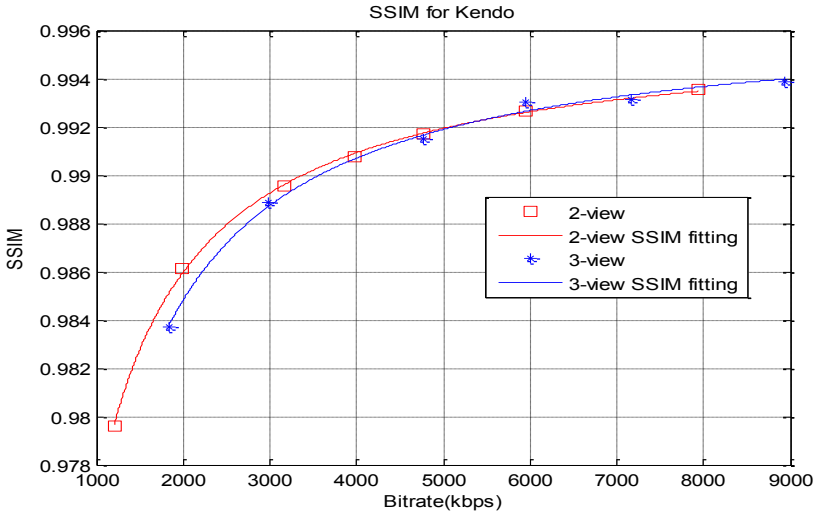

Figure8(a) SSIM for Kendo

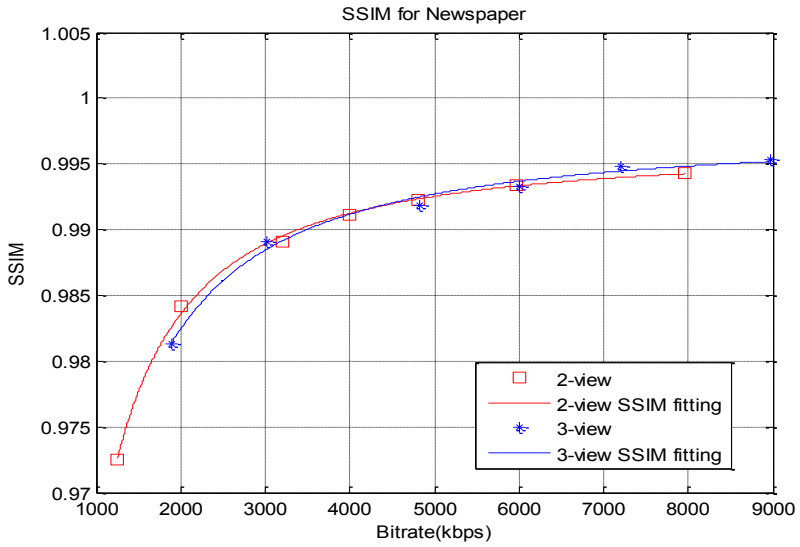

Figure8(b) SSIM for Newspaper

Figures 7(a) and 7(b) and Figures 8(a) and 8(b), show the result of the average PSNR and SSIM at different number of transmitted views(2V+D and 3V+D) for two test sequences.

When we use the PSNR as the Quality of Experience (QoS) metric for the average of virtual views, Figure 6(a) and 6(b), the same total video bitrate, the 3V+D format input always has a higher PSNR than the 2V+D at each specific total bitrate. This means that. for a fixed video bit rate, based on PSNR metric, 3V+D gives better quality. However, when we use the SSIM as a metric for evaluating the quality of virtual views of the different total bit rates, we can see that for lower total video bit rate, 2V+D has a higher SSIM than 3D+V, but as the total video bitrate increases, after passing about 5000kbps, 3V+D outperforms 2V+D as can be seen in Figure 7(a) and Figure 7(b). In other words, the effect of the transmitted number of views (distance of cameras' baseline) and the quality of each view on the rendered virtual views is depended on the total video bitrate. By using the SSIM metric, we can see that in some bit rate range, from 0 to 5000kbps, transmitting lower number of views with optimized quality is better. However, when the total video bit rate is higher than 5000kbps, it is better that we first increase the number of views and after that increase the quality of each view. Moreover, the variations among the rendered virtual views can be interpreted as a global scene movement to the amount of distance between two consecutive virtual viewpoints. Since SSIM takes into account the structural similarity and it has better correlation with human perception [13], it can predict the quality of

rendered virtual views better than PSNR and reveals a threshold of network bandwidth which allows us to accommodate higher number of views.

Based on the objective test results, we can define a policy for selecting the most appropriate MVD video segment with different number of views in terms of QoE, which can be used by the DASH client. In other words, increasing the number of views in reaction to the increasing network bandwidth does not mean higher quality. Hence, we can select the segments with lower number of views but with better quality until the available bandwidth is larger than a pre-defined threshold. For instance, in these test sequences, when the available bandwidth is lower than 5 Mbps, we select the segment with a total MVD bit rate to be lower than the available bandwidth. Otherwise, when the available bandwidth is higher than 5 Mbps, the priority of increasing the number of views would be higher than increasing the quality of each view. That is to say, when reaching 5 Mbps, we select the MVD segments with higher number of views from 2V+D to 3V+D. If the available bandwidth keeps increasing, we select the segments not only with higher number of views (3V+D), but also with higher quality for each view.

4.3 System Behavior

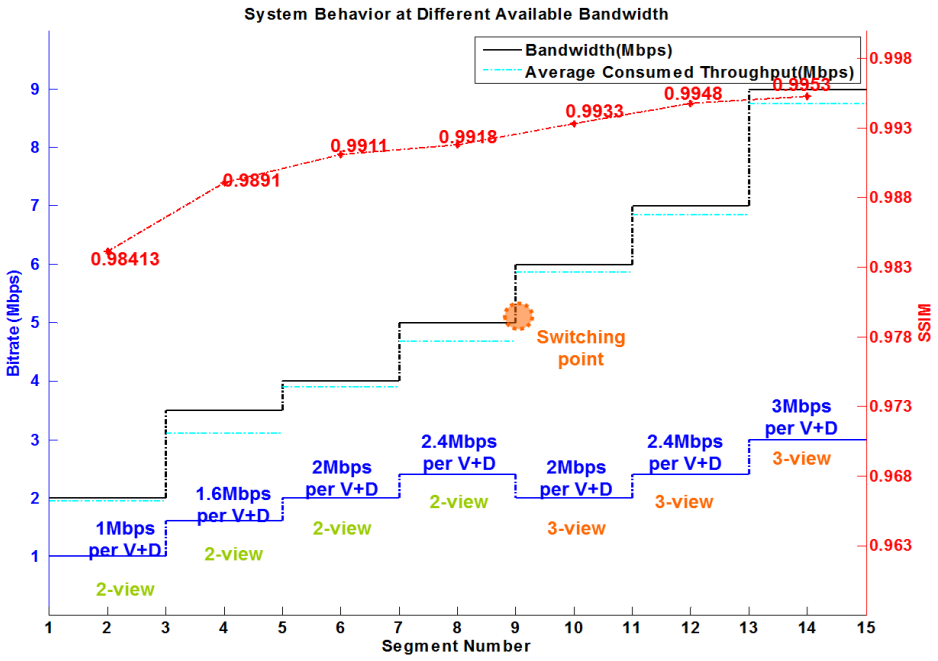

Figure9(a) System Behavior on different available bandwidth(Newspaper)

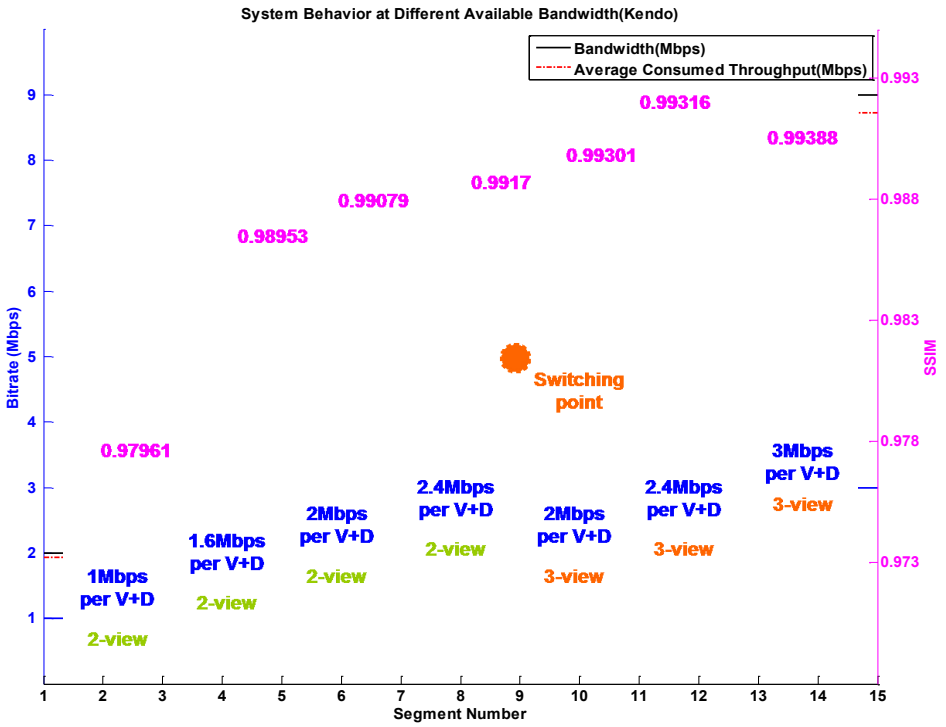

Figure9(b) System behavior on different available bandwidth(kendo)

In Figures 9(a) and 9(b), we demonstrate the transmission behaviour of our proposed system on the adaptive client. The results are shown on the two test sequences, Kendo and Newspaper. As mentioned early in this section, we initially limit the bandwidth to 2 Mbps, and after downloading each of the two segments, the available bandwidth is increased by 1 Mbps so the following available bandwidths are used 3.5Mbps, 4Mbps, 5Mbps, 6Mbps, 7Mbps and 9Mbps, at seconds 20, 40, 60, 80, 100, 120 respectively. It can be seen from both Figure 9(a) and Figure9 (b) that when the available bandwidth is below 5Mbps, the priority is to adjust the bit rate for each view and the depth to meet the perceived quality and not to increase the number of views. On the other hand, when the available bandwidth is above 5Mbps, the priority is to increase the number of views rather than the quality of each view. Our simulation results show that when the available bandwidth is above 5Mbps, the performance of 3V+D is better than 2V+D in terms of the average views' perceptual quality estimated by SSIM. Thus, we select the segments with more transmitted views instead of increasing the quality of each view. As can be seen from the Figure 9(a) and 9(b), when the available bandwidth at segment 9 and 10 (80 seconds) is increased from 5Mbps to 6 Mbps, we chose MVD segments which represent the 3V+D. However, the bitrate for

each view stays at the same level without any increase. In this way, we can select different bit streams from the server according to the variation of the available bandwidth to meet the maximum perceptual quality of the virtual views for the user.

5. Conclusions and Future work

In this paper, we proposed the architecture of our DASH based 3D multi-view video streaming system. Two of state-of-art techniques (HEVC and DASH) were used in our system. We described how to prepare the scalable server using the HEVC 3D extension encoder at the scalable server side. We also used a new scalability in terms of changing the number of views for adaptively streaming multi-view video for the auto stereoscopic 3D display. Based on the objective test results, we were able to devise a policy to adaptively selecting different versions of bit streams, which compressed by MVD format at different available bandwidth in order to present the best qualities for every view to the user.

In future work, we will be working to build a mathematical model to predict the available bandwidth using both the buffer size of the client and the throughput of the 3D multi-view video content. We also plan to test our results using a subjective test to see the effect of different bit rate allocation strategies on the users' perceived quality of experience in order to optimize our approach.

References

[1] P. Benzie, J. Watson, S. Member, P. Surman, I. Rakkolainen, K. Hopf, H. Urey, V. Sainov, and C. Von Kopylow, "A Survey of 3DTV Displays : Techniques and Technologies," IEEE Transaction Circuit Syst. Video Technol., vol. 17, no. 11, pp. 1647–1658, 2007

[2] A. Buchowicz, "Video coding and transmission standards for 3D television — a survey," Opto-Electronics Rev., vol. 21, no. 1, pp. 39–51, Dec. 2012.

[3] M. Tanimoto, "Free-Viewpoint Television," in Image and Geometry Processing for 3-D Cinematography, vol. 5, R. Ronfard and G. Taubin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 53–76.

[4] P. Kauff, N. Atzpadin, C. Fehn, M. Müller, O. Schreer, a. Smolic, and R. Tanger, "Depth map creation and image-based rendering for advanced 3DTV services providing interoperability and scalability," Signal Process. Image Commun., vol. 22, no. 2, pp. 217–234, Feb. 2007.

[5] K. Müller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D High-Efficiency Video Coding for

Multi-View Video and Depth Data," IEEE Trans. IMAGE Process., vol. 22, no. 9, pp. 3366–3378, 2013.

[6] C. Ozcinar, E. Ekmekcioglu, and A. Kondoz, "Dynamic adaptive 3D multi-view video streaming over the internet," in Proceedings of the 2013 ACM international workshop on Immersive media experiences - ImmersiveMe '13, 2013, pp. 51–56.

[7] G. J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 12, pp. 1649–1668, 2012

[8] K. Müller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D High-Efficiency Video Coding for Multi-View Video and Depth Data," IEEE Trans. IMAGE Process., vol. 22, no. 9, pp. 3366–3378, 2013

[9] J. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, S. Member, and T. Wiegand, "Comparison of the Coding Efficiency of Video Coding Standards — Including High Efficiency Video Coding (HEVC)," IEEE Trans. circuits Syst. video Technol., vol. 22, no. 12, pp. 1669–1684, 2012.

[10] H.-H.-I. Fruanhofer, "HEVC 3D extension Test Model(3DV HTM) version 11.0," 2013. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_3DVCSoftware/tags/HTM-11.0/.

[11] (MPEG) IJSW. Dynamic adaptive streaming over http. w11578, CD 23001-6, w11578, CD 23001-6. ISO/IEC JTC 1/SC 29/WG 11 (MPEG), Guangzhou, China, 2010.

[12] T. Stockhammer, "Dynamic Adaptive Streaming over HTTP – Standards and Design Principles," in Proceedings of the Second Annual ACM Conference on Multimedia Systems (MMSYS 2011), 2011, no. i, pp. 133–143.

[13] Z. Wang, A. C. Bovik, H. R. Sheikh, S. Member, E. P. Simoncelli, and S. Member, "Image Quality Assessment : From Error Visibility to Structural Similarity," IEEE Trans. circuits Syst. video Technol., vol. 13, no. 4, pp. 600–612, 2004.

[14] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, S. Member, and G. J. Sullivan, "Rate-Constrained Coder Control and Comparison of Video Coding Standards," IEEE Trans. circuits Syst. video Technol., vol. 13, no. 7, pp. 688–703, 2003.

[15] Begen, A., Akgul, T. and Baugher, M. 2011. Watching Video over the Web: Part 1: Streaming Protocols. J. IEEE Internet Comput. 15, 2 (Mar. 2011), 54–63.

[16] Kuschnig, R., Kofler, I. and Hellwagner, H. 2011. Evaluation of HTTP-based Request-Response Streams for Internet Video Streaming. In Proceedings of the second annual ACM conference on Multimedia systems. (San Jose, California, USA, February 23-25, 2011) MMSys `11. ACM, New York, NY, 245–256.

[17] Pantos, R. and May, W. 2010 HTTP Live Streaming. Internet Draft IETF Draft. IETF Tools. http://tools.ietf.org/html/draft-pantos-http-live-streaming-04

[18] Zambelli, A. 2009. IIS smooth streaming technical overview. Microsoft Corporation

[19] Hassoun, D. 2010. Dynamic streaming in flash media server 3.5. Adobe. http://www.adobe.com/devnet/adobe-media-server/articles/dynstream_advanced_pt1.html

[20] Akamai HD Network Demo. http://wwwns.akamai.com/hdnetwork/demo/flash/zeri/

[21] Lohmar, T.; Einarsson, T.; Frojdh, P.; Gabin, F. 2011. Kampmann, M.; Dynamic adaptive HTTP streaming of live content. In Proceedings of the 12th IEEE International Symposium on a

World of Wireless, Mobile and Multimedia Networks (Lucca, Italy, 20-24 June, 2011) WoWMoM '11. 1-8.

[22] A. Javadtalab, M. Semsarzadeh, A. Khanchi, S. Shirmohammandi, and A. Yassine, "Continuous One-Way Detection of Available Bandwidth Changes for Video Streaming Over Best-Effort Networks," IEEE Trans. Instrum. Meas., vol. 64, no. 1, pp. 190–203, 2015.

[23] K. T. Ba and A. M. Tekalp, "ADAPTIVE STEREOSCOPIC 3D VIDEO STREAMING," in Image Processing (ICIP), 2010 17th IEEE International Conference on, 2010, pp. 2409–2412.

[24] A. Hamza and M. Hefeeda, "A DASH-based Free Viewpoint Video Streaming System," in Proceedings of Network and Operating System Support on Digital Audio and Video Workshop, 2013, p. 55.

[25] K. Calagari, "Anahita : A System for 3D Video Streaming with Depth Customization Categories and Subject Descriptors," in Proceedings of the ACM International Conference on Multimedia, 2014, pp. 337–346.

[26] K. E. Psannis, M. Hadjinicolaou and A. Krikelis, "MPEG-2 Streaming Of Full Interactive Content", IEEE Transactions on Circuits and Systems for Video Technology, vol.16. no 2, pp. 280-285, 2006.

[27] T. Schierl, M. M. Hannuksela, Y-K. Wang, and S. Wenger, " System Layer Integration of High Efficiency Video Coding (HEVC)," IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, issue 12, pp. 1871-1884, 2012,

[28] S. Wenger, "H.264/AVC over IP," IEEE Transactions on Circuits and Systems, vol. 13, no. 7, July 2003.

[29]H. Roodaki, M.R. Hashemi, and S. Shirmohammadi, "A New Methodology to Derive Objective Quality Assessment Metrics for Scalable Multi-view 3D Video Coding", ACM Transactions on Multimedia Computing, Communications, and Applications, Vol. 8, No. 3S, Article 44, September 2012, 25 pages.DOI: 10.1145/2348816.2348823

[30] H. Roodaki, M.R. Hashemi, and S. Shirmohammadi, "Rate-Distortion Optimization for Scalable Multi-View Video Coding", Proc. IEEE International Conference on Multimedia and Expo, Chengdu, China, July 14-18 2014, 6 pages.DOI: 10.1109/ICME.2014.6890275

[31] B. Oztas, M. T. Pourazad, P. Nasiopoulos, I. Sodagar, and V. C. M. Leung, "A Rate Adaptation Approach for Streaming Multiview Plus Depth Content," in Computing, Networking and Communications (ICNC), 2014 International Conference on, 2013, no. Mvd, pp. 1006–1010.

[32] E. Bosc, F. Racapé, V. Jantet, P. Riou, M. Pressigout, and L. Morin, "A study of depth/texture bit-rate allocation in multi-view video plus depth compression," Ann. Telecommun. - Ann. Des Télécommunications, vol. 68, no. 11–12, pp. 615–625, Apr. 2013.

[33] B. Li, H. Li, L. Li, and Z. Jinlei, "Rate control by R-lambda model for HEVC," Jt. Collab. Team Video Coding(JCT-VC)of ITU-T SG 16 WP 3 ISO/IEC JTC 1/SC 29/WG 11, pp. 1–11, 2012.

[34]H. Roodaki, M.R. Hashemi, and S. Shirmohammadi, "New Scalable Modalities in Multi-view 3D Video", Proc. ACM Workshop on Mobile Video, Oslo, Norway, February 27 2013, pp. 25-30. DOI: 10.1145/2457413.2457420

[35] P. Ndjiki-nya, M. Köppel, D. Doshkov, H. Lakshman, P. Merkle, K. Muller, and T. Wiegand, “Depth Image-Based Rendering With Advanced Texture Synthesis for 3-D Video,” IEEE Trans. Multimed., vol. 13, no. 3, pp. 453–465, 2011.

[36] Dimenco, “Non-glass 3D displayer,” 2014. [Online]. Available: http://www.dimenco.eu/3d-displays/displays/65-inch-4k/.

[37] Alioscopy, “Alioscopy 3D HD 55" LV data sheet,” 2010. [Online]. Available: http://www.alioscopy.com/en/datasheet.php?model=Alioscopy 3D HD 47%22 LV.

[38] T. L. at N. University, “Kendo Test sequences.” [Online]. Available: http://www.tanimoto.nuee.nagoya-u.ac.jp/.

[39] Y.-S. Ho, E.-K. Lee, and L. Cheon, “Newspaper, Multiview Video Test Sequence and Camera Parameters,” in INTERNATIONAL ORGANISATION FOR STANDARDISATION ORGANISATION I ISO / IEC JTC1 / SC29 / WG11 CODING OF MOVING PICTURES AND AUDIO, 2008, pp. 1–6.

[40] “Call for Proposals on 3D Video Coding Technology.” ISO/IEC JTC1/SC29/WG11 MPEG2011/N12036, Geneva, Switzerland, 2011.

[41] Rizzo, L. 1997. Dummynet : a simple approach to the evaluation of network protocols. ACM SIGCOMM Computer Communication Review. 27, (1997), 31–41

[42] Internet Information Service, 2014. [Online]. Available: http://www.iis.net.

[43] T. C. Thang, H. T. Le, S. Member, A. T. Pham, S. Member, and Y. M. Ro, “An Evaluation of Bitrate Adaptation Methods for HTTP Live Streaming,” IEEE J. Sel. ATREAS Commun., vol. 32, no. 4, pp. 693–705, 2014