Evaluation of H.265 Video Delivery Based on Dynamic Adaptive Streaming over HTTP (DASH) Using Mobile Devices

Hamid Azwar[1]

***Abstract*—MPEG Dynamic Adaptive Streaming over HTTP (MPEG-DASH) is a standard for HTTP streaming that allows interoperability between servers and clients from different vendors which is able to adjust the speed of video delivery to changing bandwidth. Today there is the latest H.265 video coding known as the High Efficiency Video Coding (HEVC). This research implemented the delivery of the H.265 video through the method of Dynamic Adaptive Streaming over HTTP (DASH) using mobile devices. The test was conducted by sending videos over 3G networks with GSM length video segment consists of 2 seconds, 5 seconds, 10 seconds and 20 seconds. The video quality provided in this study consisted of 5 video levels with different bit rates. From the test results, the submitted videos using a smaller video segment generated greater throughput and acquired more video quality with a higher bit rate than the video to a larger segment. Video delivery with segment of 2 seconds acquired video quality with the highest level of 83%, meanwhile the video delivery segment of 20 seconds obtained video quality with the highest level of 33%.**

Keywords: Video, Adaptive, H.265, Streaming, Segment

I. Introduction

One of the alternatives to streaming video is to use progressive download method via HTTP server. The HTTP client can search for the position of the video content stored on the HTTP server and request the content to be sent to the client. However, this method has the disadvantage of requiring sufficient bandwidth to download. If the available bandwidth is not enough, then the wait time known as the bufferring time will increase.

Many commercially adapted HTTP streaming methods have been developed with platforms, such as Apple HTTP Live Streaming (HLS) [1], Microsoft Smooth Streaming [2], and Adobe HTTP Dynamic Streaming (HDS) [3]. All these platforms use HTTP streaming as their content delivery method. Additionally, in 2012 MPEG collaborated with 3GPP released an adaptive video delivery via HTTP standard known as MPEG Dynamic Adaptive Streaming over HTTP (MPEG-DASH) [4].

Besides video streaming methods, video compression techniques are also improving. This is because standardization of video compression is required to facilitate the exchange of data in the form of digital video globally in the form of video streaming applications. A coding standardization is considered to be efficient when it supports good compression algorithms and implements efficient encoder and decoder designs. For multimedia communication, there are two main standards organizations, namely ITU-T and International Organization for Standardization (ISO). Over the past few decades, a number of video compression standards from ITU-T and ISO, such as MPEG-1, MPEG-2, MPEG-4, H.261, H.263, and H.264 have been developed to support many video delivery applications. On April 13, 2013, ITU has recommended a new video coding standard called H.265 known as High Efficiency Video Coding (HEVC) [5]. H.265/HEVC is joint research conducted by ITU-T Video Coding Experts Group (VCEG) organization and ISO/IEC Moving Picture Experts Group (MPEG) organization. This joint research is also known as Joint Collaborative Team on Video Coding (JCT-VC). Fig. 1 shows the development of video coding that has occurred in recent decades.

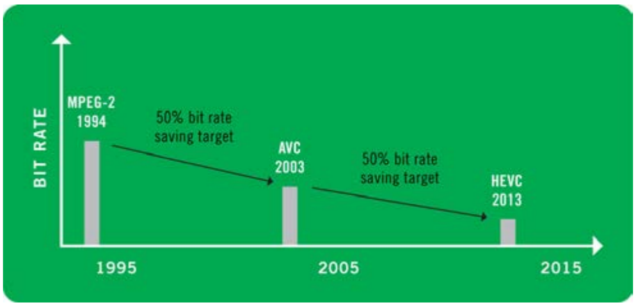


Fig. 1 The development of video coding [3].

Observations on the efficiency of HEVC video coding have been undertaken, which results show that with the same Peak Signal Noise to Ratio (PSNR) value, the HEVC Main Profile has a 35.4% bit rate reduction compared to H.264/AVC High Profile. In addition, based on subjective assessment, HEVC Main Profile has a bit rate reduction of 49.3% compared to H.264/AVC High Profile [6].

This paper applies video delivery using Dynamic Adaptive Streaming over HTTP (DASH) over a 3G network using a mobile devices as clients. The delivery of the video used H.265 video coding according to the length of the video segment at the time of the DASH implementation.

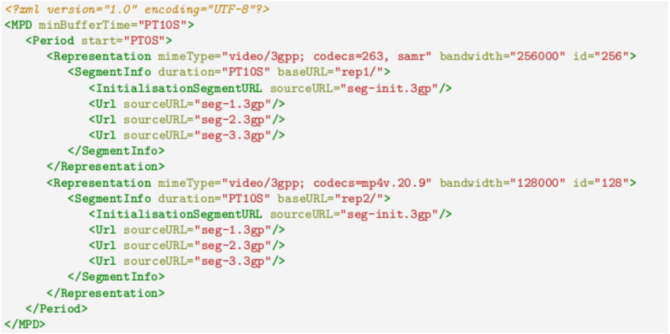
II. MPEG Dynamic Adaptive Streaming over HTTP (MPEG-DASH)

MPEG that gets broad participation and support from many industries develops MPEG-DASH technology in order to standardize adaptive video streaming over HTTP by making it simpler and more flexible enabling clients and servers from different vendors. MPEG-DASH is an international ISO standard that attempts to unify HTTP streaming for a single

[1] *Electrical Engineering Department, Politeknik Caltex Riau, Jl. Umbansari No. 1 Rumbai, Pekanbaru, INDONESIA (tlp: 0761-53939; e-mail: hamid@pcr.ac.id)*

standard [7]. It provides a universal format for content delivery using existing device and infrastructure content. This improves the quality of streaming media on the user side by interactively switching between different video encodings depending on the network conditions at that time.

The MPEG-DASH structure consists mainly of three file types: "Manifest" (.mpd) which is an XML file depicting a segment, "File Initialization" that contains the headers is needed to decode byte on a segment, and "Segment Files" contain Media being played. Media Presentation Description (MPD) provides metadata for the media requesting a segment through a Uniform Resource Locator (URL) in order to locate and perform a segment download. It also provides information on the number of representations and characteristics of each representation used for adaptation level requirements, such as minimum buffering time, bandwidth representation, and segment duration [8]. Examples of the form and structure of MPD are respectively shown in Fig. 2 and Fig. 3.



```
<?xml version="1.0" encoding="UTF-8"?>
<MPD minBufferTime="PT10S">
   <Period start="PT0S">
      <Representation mimeType="video/3gpp; codecs=263, samr" bandwidth="256000" id="256">
         <SegmentInfo duration="PT10S" baseURL="rep1/">
            <InitialisationSegmentURL sourceURL="seg-init.3gp"/>
            <Url sourceURL="seg-1.3gp"/>
            <Url sourceURL="seg-2.3gp"/>
            <Url sourceURL="seg-3.3gp"/>
         </SegmentInfo>
      </Representation>
      <Representation mimeType="video/3gpp; codecs=mp4v.20.9" bandwidth="128000" id="128">
         <SegmentInfo duration="PT10S" baseURL="rep2/">
            <InitialisationSegmentURL sourceURL="seg-init.3gp"/>
            <Url sourceURL="seg-1.3gp"/>
            <Url sourceURL="seg-2.3gp"/>
            <Url sourceURL="seg-3.3gp"/>
         </SegmentInfo>
      </Representation>
   </Period>
</MPD>
```

Fig. 2 Example of MPD [8].

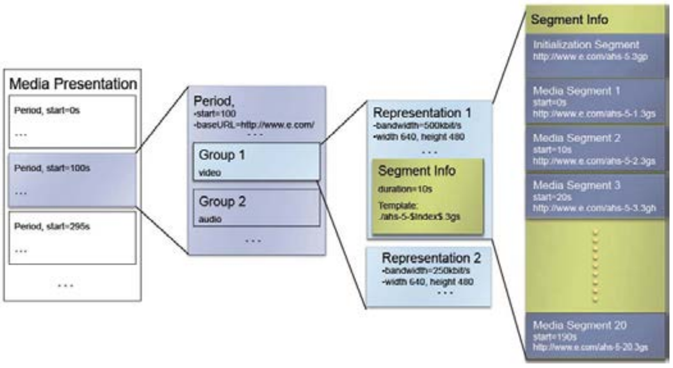


Fig. 3 Structure of MPD [7].

The MPEG-DASH defines the MPD format and video delivery format using ISO BMFF and MPEG2-TS. It defines the provision of content, such as the size and duration of segments, numbers, and bit rate representations. In addition, it also determines normative client behaviour, such as switching representations [7]. Fig. 4 shows the architecture that composes the MPEG-DASH.

On the left side of Fig. 4 is shown media presentations in the form of video segments that have been coded with varying quality and placed on the HTTP server. This media variation was sent to the clients using MPD. The exchange of data could be done using the HTTP protocol.

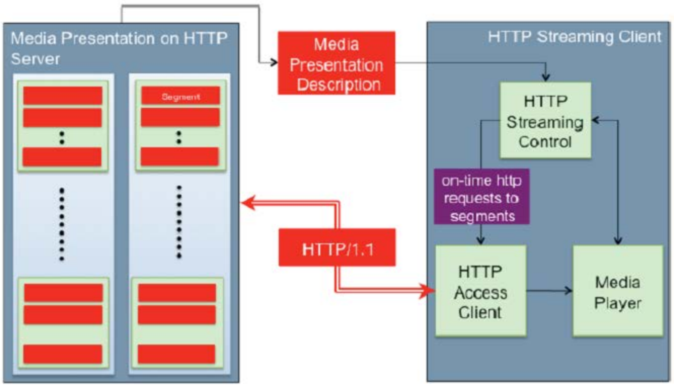

Media Presentation on HTTP Server
Segment
Media Presentation Description
HTTP Streaming Client
HTTP Streaming Control
on-time http requests to segments
HTTP/1.1
HTTP Access Client
Media Player

Fig. 4 Architecture of MPEG-DASH [7].

The MPEG-DASH process on the client when the video was delivered is shown in Fig. 5 [9].

The video server stores the MPD files and all the different levels of encoding from a video streaming. In Fig. 5, the video is encoded into three levels of quality ie A, B, and C. These clients can be streaming devices, such as desktops, laptops, smartphones, and other devices. The video player on the client is responsible for streaming video adaptively depending on the quality recommended by the quality adaptation algorithm on the client. Initially, the client starts requesting the MPD file from the server. Once accepted, the client parses the file and creates a list of segments accessed for each representation. Then, the client begins to ask for the lowest available encoding on the server. After that, quality client-side adaptation algorithms begin to update recommended quality based on several criteria, such as network throughput. Next, the client selects the suggested segment from the selected representation and then sends the request (HTTP GET request) on the server. For example, in this example the quality level selected from the third segment is quality B. When the client performs a video playback, the process of checking the appropriate quality level and demand of the video through the segment is continuous on every occurrence of the request. Then, the video player will reinstall the video fragment. At the end of this process, the video will be obtained with segments of varying quality levels.

III. H.265

HEVC is a video coding standard produced from a joint research project between the ITU-T VCEG organization and ISO/IEC MPEG organization, known as JCT-VC [10]. This HEVC is a continuation of H.264/AVC video coding (Advanced Video Coding). The H.264/AVC video coding has also been developed for Scalable Video Coding (SVC) technology [11]. The HEVC video coding has better video quality and compression level than H.264/AVC and is capable of supporting 8K UHD (Ultra High Definiton) video type with resolution reaching 8192x4320 [10].

IV. System Planning

This section describes the design and implementation of DASH-based H.265 video delivery using mobile devices. The block diagram used in this system is shown in Fig. 6.

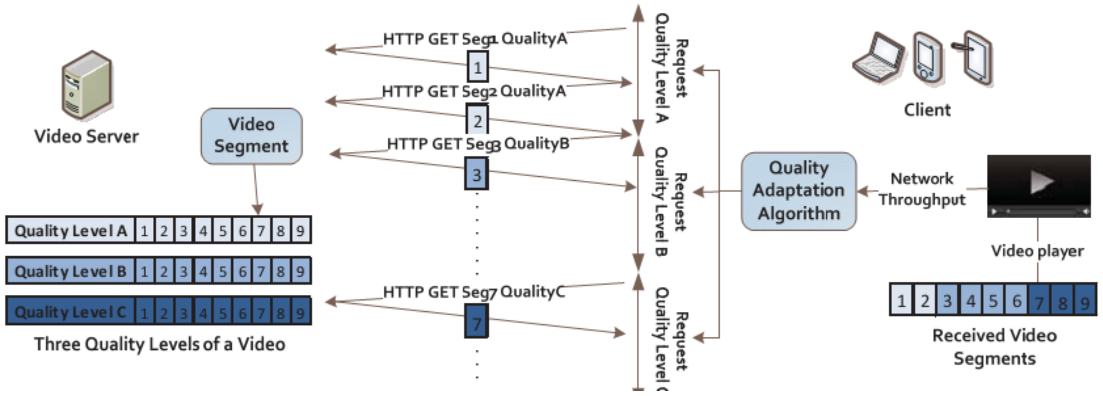


Fig. 5 Process of MPEG-DASH on clients [9].

A. Video Source

The video source used on this system was an uncompressed YUV video. The YUV video specifications are presented in Table I.

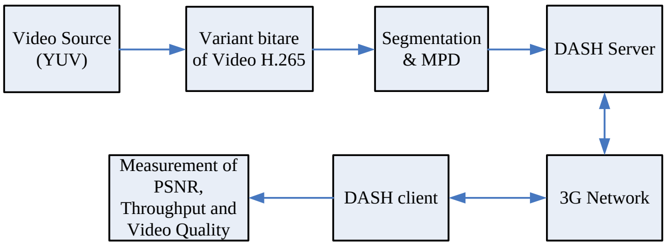

Video Source (YUV)
Variant bitare of Video H.265
Segmentation & MPD
DASH Server
Measurement of PSNR, Throughput and Video Quality
DASH client
3G Network

Fig. 6 Diagram of block system.

TABLE I
SPECIFICATIONS OF VIDEO SOURCE

Video Name	Resolution (Pixel)	Number of Frame	Description
Big Buck Bunny	1,280 x 720	5,760	Animation

TABLE II
SPECIFICATIONS OF VIDEO SOURCE

No.	Average bit rate (Kbps)	Resolution (Pixel)	Frame rate (fps)
1	100	1,280 x 720	24
2	500		
3	1,000		
4	2,000		
5	4,000		

The video specifications shown in Table I have a total of 14,315 frames [12]. However, on this system only used 5,760 frames.

B. Encoding Video H.265

H.265 video encoding can be done using DivX265 [13]. In this case, DivX265 was run on Windows operating system. At the moment, DivX265 can support video encoding process derived from lossless video (YUV). Video encoding can be done based on the bit rate and video resolution desired by the user. Therefore, to support video delivery via the DASH, then video encoding was done by generating a number of different video bit rates. Table II shows the video encoding specifications.

C. Network Topology

This section shows the network topology used as shown in Fig. 7.

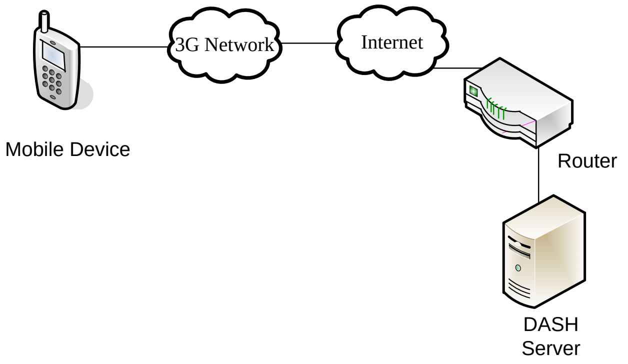


Fig. 7 Network topology.

In this research, it was used a client in the form of a mobile device to access video on HTTP Server. It played the video on demand via 3G network in Pekanbaru.

D. DASH Client

The client specifications in the form of the mobile device used are shown in Table III.

TABLE III
CLIENT SPECIFICATIONS

No.	Component	Specifications
1	Processor	Intel Octa-core 1.5 GHz
2	RAM	DDR2 2 GB
3	Sistem Operasi	Android versi 5.1.1
4	Playback	Osmo4

To support the DASH process on the client, on the mobile device must be installed Osmo4 to playout H.265 video [14]. After the installation process was done, then the mobile device could access the video melalaui HTTP server.

V. Results and Dicsussion

A. *Encoding Result of Video H.265*

The video source used was a video that had not been compressed with YUV format. The video was then encoded using H.265 video encoder. The H.265 video encoding results are presented in Table IV.

Table IV
Encoding Results of Video H.265

No.	Average bitrate (Kbps)	File Size (MB)	Video Level	Number of Frame	Frame rate (fps)	Resolution (pixel)
1	100	3.92	1	5,760	24	1,280x720
2	500	18.8	2			
3	1,000	36.19	3			
4	2,000	67.92	4			
5	4,000	12.79	5			

From Table IV, it can be seen that with the same number of frames, frames and resolutions. The video has a large file size when it has a high average bit rate. In addition, it can also be seen that the quality of the video is based on the value of PSNR as shown in Fig. 8.

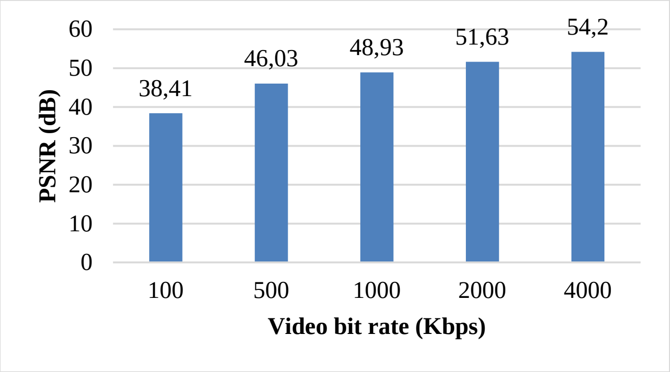


Fig. 8 PSNR of the encoding result of video H.265.

Comparable to the size of the file, the result of the H.265 video encoding as shown in Fig. 8 shows that the video that has a high average bit rate, it also has a high PSNR value so that the quality of the video is getting better. This is because the video with a high bit rate has a lot of information to represent. This is different from a video which has a low bit rate whose PSNR value is small, therefore it has a lower video quality. In general, the video produced on the H.265 encoding in this test has a relatively good PSNR value of above 31 dB [15].

B. *Result of Video H.265 Segmentation*

The video that had undergone the H.265 encoding process, then it ran a segmentation process to support video delivery via DASH. The video segmentation was done with segment length of 2 seconds, 5 seconds, 10 seconds, and 20 seconds. By using MP4Box, the segmentation could be done with the results as shown in Table V.

The video with segmentation of 2 seconds has a large number of segments. This is because each video with a certain bit rate was segmented with a segment length of 2 seconds, so each video produced 120 segments. Because the video bit rate varies to five levels, the total segment totals are 600 segments. This is in contrast to video segmentation with segment lengths of 20 seconds, resulting in fewer segment numbers, 12 segments and 60 segments are for five video levels.

Table V
Results of Video H.265 Segmentation

No	Average bitrate (Kbps)	Segment Number			
		Segment 2 seconds	Segment 5 seconds	Segment 10 seconds	Segment 20 seconds
1	100	120	48	24	12
2	500				
3	1000				
4	2000				
5	4000				

C. *Result of Client's Video Sreaming*

After the MPD files were placed on the DASH server, the client could stream the video by accessing the content contained on the DASH server via HTTP using a mobile device. After filling the URL address to stream the video, a web page displayed for streaming as shown in Fig. 9.

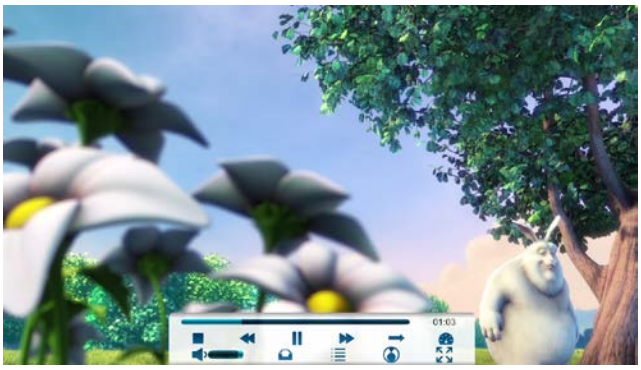


Fig. 9 PSNR of encoding result of video H.265.

The display shown in Fig. 9 was built with playback using Osmo4 that was capable of performing H.265 video playback and in the form of the DASH.

D. *Throughput Measurement Results*

In this section, the results of throughput measurement at the time of the video delivery using the length of the video segment vary, namely 2 seconds, 5 seconds, 10 seconds, and 20 seconds. From the measurement of throughput during video delivery with varying video segment lengths, it is found that the average throughputs of this measurement. The average throughputs of the video delivery are shown in Fig. 10.

Fig. 10 shows that video delivery of 2-second segment length has the largest average throughput value. Additionally, the smallest average throughput value occured at the time of video delivery with segment length of 20 seconds. This shows that a small value video segment could absorb large bandwidths to take video levels with large bit rate. Thus, the video which has a high bit rate, of course, has better video quality compared to a low bit rate video.

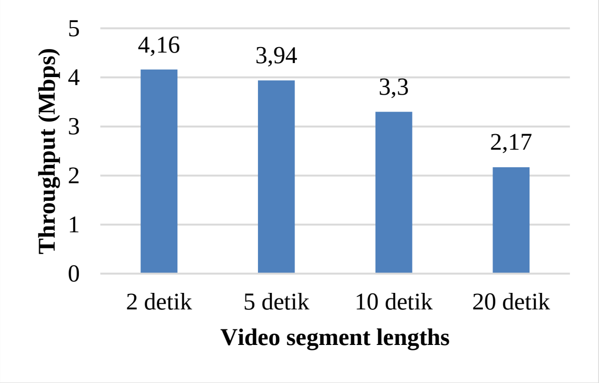


Fig. 10 Average throughputs of video delivery.

E. Quality of Video Received

In this section, it is observed the quality of the video delivery on users using 3G GSM network via mobile devices. Video delivery quality parameters are observed based on the percentage of video levels that a user receives. This study used five levels of video with varying bit rates, namely 100 Kbps, 500 Kbps, 1 Mbps, 2 Mbps, and 4 Mbps. Level 1 has a bit rate of 100 Kbps. Level 5 has a bit rate of 4 Mbps. The video delivery quality observation results based on different video segment lengths can be seen in Fig. 11.

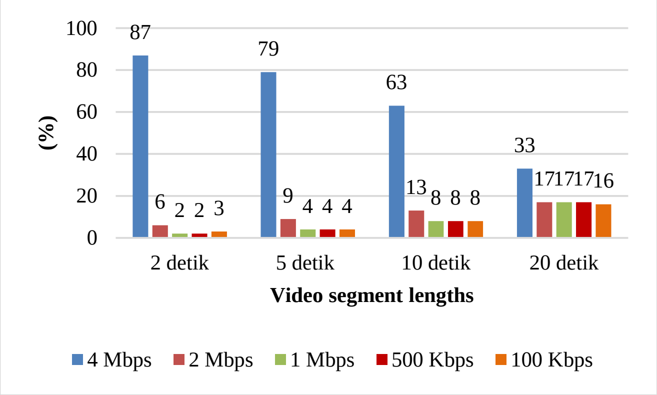


Fig. 11 Percentages of Client's quality video levels

Fig. 11 indicates that video level with 4 Mbps bit rate is delivered more when uses video with segment length of 2 seconds, with percentage of 88%, meanwhile the video delivery with a segment length of 20 seconds has a video level with 4 Mbps bit rate of 33%. This indicates that the length of the segment used during video delivery using the DASH greatly affects the quality of the video received by the user.

VI. Conclusion

Video delivery using the DASH method can be done using mobile devices. The segment length of the video delivery affects the throughput value and the percentage of the video-level quality received by the client. With a segment length of 2 seconds, the resulting throughput value is 4.16 Mbps, meanwhile the video with a segment length of 20 seconds has the throughput of 2.17 Mbps. The highest reception of the video quality (4 Mbps bit rate) yields 87% for the video segment of 2 seconds and 33% for the video segment length of 20 seconds.

References

[1] R. Pantos and W. May, "HTTP Live Streaming," *IETF draft*. pp. 1–33, 2012.

[2] A. Zambelli, "IIS Smooth Streaming Technical Overview," *Microsoft Corp.*, no. March, 2009.

[3] Adobe Systems Inc, "Adobe Flash Player," *[online]*, 2010. [Online]. Available: http://www.adobe.com/products/flashplayer/.

[4] MPEG, "Dynamic adaptive streaming over HTTP (DASH) -- Part 1: Media presentation description and segment formats," *ISO/IEC 23009-1:2014*, vol. 2013, pp. 1–61, 2015.

[5] "H.265 : High efficiency video coding." [Online]. Available: http://www.itu.int/rec/T-REC-H.265-201304-S. [Accessed: 30-Apr-2017].

[6] J. R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, 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.

[7] T. Stockhammer, "Dynamic Adaptive Streaming over HTTP: Standards and Design Principles," *Proc. Second Annu. ACM Conf. Multimed. Syst.*, vol. 2014, no. i, pp. 133–144, 2011.

[8] L. Rubio Romero, "A Dynamic Adaptive HTTP Streaming Video Service for Google Android," Master Thesis, Royal Institute of Technology (KTH), Stockholm, Sweden, Oct. 2011.

[9] V. J. H. Schulzrinne, S. Casner, R. Frederick, "RTP : A Transport Protocol for Real-Time Applications," *Internet Soc. RFC 3550*, pp. 1–89, 2003.

[10] 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, 2012.

[11] E. H. Putra, R. Hidayat, Widyawan, and and I. W. Mustika, "Cross-Layer Design of Wireless Multimedia Sensor Network Based on IEEE 802.11e EDCA and H.264/SVC," in *2015 International Conference on Science in Information Technology (ICSITech)*, 2015.

[12] "Big Buck Bunny." [Online]. Available: https://peach.blender.org/. [Accessed: 30-Apr-2017].

[13] "DivX HEVC Community Encoder | DivX Labs." [Online]. Available: http://labs.divx.com/divx265. [Accessed: 30-Apr-2017].

[14] "Osmo4 | GPAC." [Online]. Available: https://gpac.wp.imt.fr/player/. [Accessed: 30-Apr-2017].

[15] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal Processing*, vol. 90, no. 3. pp. 727–752, 2010.