IMPLEMENTATION AND ANALYSIS OF USER ADAPTIVE MOBILE VIDEO STREAMING USING MPEG-DASH

by

ABHIJITH JAGANNATH

Presented to the Faculty of the Graduate School of
The University of Texas at Arlington in Partial Fulfillment
of the Requirements
for the Degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING
THE UNIVERSITY OF TEXAS AT ARLINGTON
August 2014

Acknowledgements

It would not have been possible to complete this thesis without the guidance and the help of several individuals who in one way or another contributed and extended their valuable assistance

First and foremost, my utmost gratitude to my thesis advisor, Dr. K. R. Rao, for his sincerity and encouragement. I really appreciate his helpful nature and care he takes for the students. I feel really proud to be Dr. Rao’s student and a member of Multimedia Processing Lab.

I would like to thank Dr. W Alan Davis and Dr. Jonathan Bredow for their valuable time and taking interest to review my thesis work.

I want to thank Dr. Yuriy Reznik for the continuous support of my studies and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of internship and writing of this thesis. I also want to thank the team members at Inter Digital, Dr. Rahul Vanam, Dr. Eduardo Asbun and Dr. Louis Kerofsky for being patient and guiding me whenever I asked for help.

This thesis would never have been completed without the encouragement and devotion of my family and friends. I express my sincere gratitude towards them.

July 7, 2014

Abstract

IMPLEMENTATION AND ANALYSIS OF USER ADAPTIVE MOBILE VIDEO STREAMING USING MPEG-DASH

Abhijith Jagannath, MS

The University of Texas at Arlington, 2014

Supervising Professor: K. R. Rao

Modern mobile devices are already matching and surpassing HDTV sets in terms of graphics capabilities. They often feature high density retina screens with 1280X720, 1920X1080 and even higher resolutions. But, the density of the information seen will depend on several factors. The visual perception of the video played on these devices mainly depends on the user, where and how the video is consumed, quality of the video perceived changes when in dark room to that of bright sun light, device held at an arm's distance to device kept away on a table. Streaming high bitrate video to these devices when in reality it is not perceived is an added burden to the network.

This thesis is an attempt to help the streaming clients to select the bit streams considering the perceptual factors. Reduction of bandwidth usage without compromising perceptual quality is achieved by calculating the sufficient resolution to play the video for particular viewing condition. These viewing conditions are determined using several sensors in the smart phones. A web based implementation of MPEG-DASH JavaScript player is used to implement the streaming client. Considerable bandwidth savings (10% – 40%) are observed when user adaptation is taken into account.

Table of Contents

List of Illustrations

List of Tables

Chapter 1

Introduction

1.1 Challenges in Mobile Video Streaming

Internet video streaming has experienced a dramatic growth and transformation from an early concept into a mainstream technology in the past two decades [1] [2] [3]. A recently issued MPEG-DASH standard [4] consolidates many advances achieved in the design of streaming media delivery systems, including full use of the existing HTTP infrastructure, bandwidth adaptation mechanisms, latest audio and video codecs, etc. Yet, some challenges in implementation and deployment of streaming systems still exist. In particular, they arise in the delivery of streaming video content to wirelessly connected mobile devices, such as smartphones and tablets.

On one hand, many mobile devices are already matching and surpassing HDTV sets in terms of graphics capabilities. They often feature high-density “retina” screens with 1280X720, 1920X1080, and even higher resolutions. They also come equipped with powerful processors, making it possible to receive, decode and play HD-resolution videos. On the other hand, network and battery/power resources in mobile devices remain limited. Wireless networks, including the latest 4G/LTE networks, are fundamentally constrained by capacities of their cells. Each cell’s capacity is shared between its users, and it can be saturated by as few as 5 - 10 users simultaneously watching high-quality videos [5]. High data rates used to transmit video also cause high power consumption by the receiving devices, draining their batteries rapidly. All these factors suggest that technologies for reducing bandwidth and power use in mobile video streaming are very much needed.

In this thesis, one of the approaches towards improving performance of mobile streaming systems is reviewed and implemented. It is based on understanding of user behavior and environmental factors affecting perception of visual information delivered to mobile screens.

1.2 Factors Affecting Perception of Visual Information

There are several factors that can affect a user's ability to discern the visual information rendered on a mobile screen.

In Figure 1-1, several important parameters of a viewing setup is shown. These include viewing distance, display size, viewing angle, and ambient light.

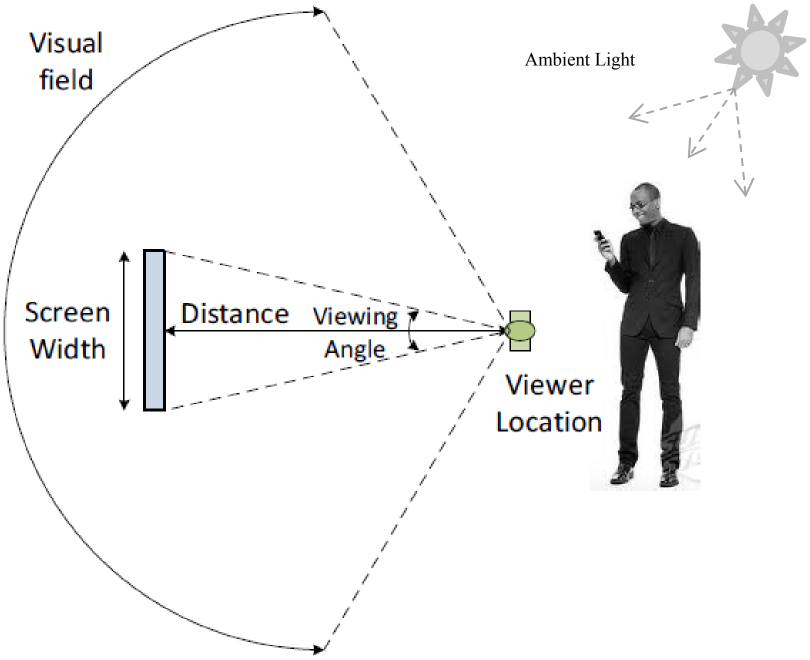


Figure 1-1 Characteristics of viewing setup. [6]

The variation of ambient illuminance across several possible types of environments is captured in Figure 1-2. This shows a very broad (5 orders of magnitude) range of this characteristic.

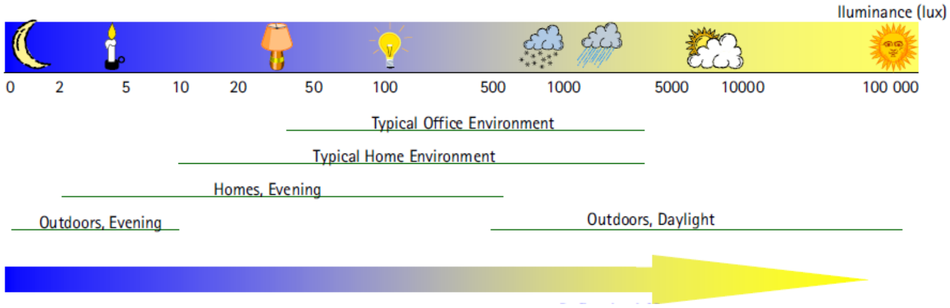


Figure 1-2 Ambient illuminance in different environments [7]

Figure 1-3 illustrates some effects of ambient light on the visibility of information presented on a mobile screen when indoors and in direct sunlight conditions.

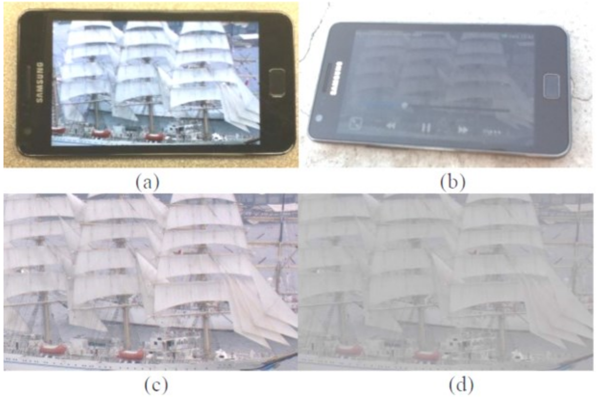


Figure 1-3 Mobile video viewed with different surrounding light level. (a) indoors, (b) direct sunlight, (c) contrast perceived as in (a), (d) contrast perceived as in (b) [8].

The examples in Figure 1-3 show that the impact of the viewing setup and the environmental factors can be very significant. A high-quality video becomes completely washed away under sunlight.

Changes in viewer pose, e.g. when he/she is placing a tablet or a phone on his/her lap or on a stand, may also significantly impact the amount of information that will be delivered. The further the viewer is from the screen, the more content in the video that falls beyond the resolution capability of human vision, making it invisible to the viewer.

1.3 User Adaptive Streaming

The main idea explored in this thesis is to make delivery of video to mobile screens adaptive – by measuring parameters of the viewing setup and other environmental factors and then using them to select video resolution and other encoding parameters – ensuring the most effective delivery of video to the user. This concept is called User Adaptive Mobile Video Streaming [6].

In this thesis, a prototype of a user adaptive video streaming system is implemented to study its performance. Key aspects of this work include:

- Development of models and formulas for computing sufficient resolution for delivery of video under certain viewing conditions,
- Development of models and algorithms for estimating ambient contrast of mobile displays,
- Development of algorithms for detecting user presence and measuring his distance from the screen,
- Implementation of MPEG-DASH framework to make adjustments related to different viewing conditions, and
- Experimental study of the effectiveness of developed system.

The results obtained in this work suggest that user adaptive streaming can offer an appreciable (in the range of 10-40%, based on different condition) reduction in bandwidth usage compared to traditional adaptive streaming to mobile devices. The Most significant savings are achieved for streaming of HD (1920X1080) content to small form-factor devices (phones and tablets).

1.4 Thesis Outline

This thesis is organized as follows. In Chapter 2, several aspects and limitations of human vision are reviewed. Chapter 3 introduces the concept of maximum visible frequency, which shows how it can be used to compute sufficient resolution for delivery of video to a given reproduction environment. In Chapter 4, the design principles used by adaptive video streaming and MPEG-DASH are reviewed. It also describes the architecture of MPEG-DASH client used in this work. The implementation of a user adaptive video streaming is discussed in Chapter 5. The results and analysis of implementation are discussed in Chapter 6. Chapter 7 offers conclusions.

Chapter 2

Limits of Human Vision

In this chapter, some relevant aspects of human visual limits will be reviewed. Particular attention is given to explanation of the concepts of spatial frequency and Contrast Sensitivity Function (CSF) of human vision.

2.1 Visual Acuity

Visual acuity (VA) is a quantitative measure of the ability to identify black symbols on a white background at a standardized distance as the size of the symbols is varied. It is the most common clinical measurement of visual function which represents the smallest size that can be reliably identified. A visual acuity of 20/20 is frequently described as meaning that a person can see detail from 20 feet away the same as a person with normal eyesight would see from 20 feet.

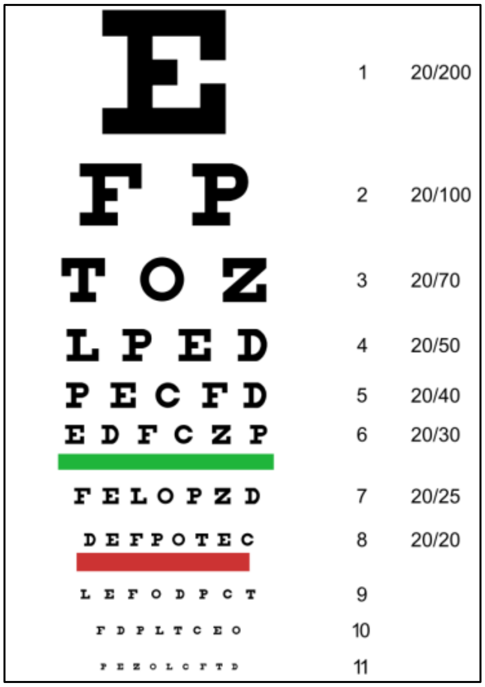


Figure 2-1 Snellen's chart [9]

Visual acuity often is referred to as “Snellen” acuity [10]. The chart and the letters are named for a 19th-century Dutch ophthalmologist Hermann Snellen (1834–1908) who created them as a test of visual acuity.

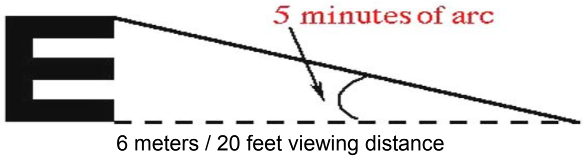


Figure 2-2 Visual acuity calculation [11]

Snellen letters are constructed so that the size of the critical detail (stroke width and gap width) subtends 1/5th of the overall height. To specify a person’s visual acuity in terms of Snellen notation, a determination is made of the smallest line of letters of the chart that he/she can correctly identify. Visual acuity (VA) in Snellen notation is given by the relation:

$$\mathrm{VA} = D'/D \tag{2.1}$$

where D’ is the standard viewing distance (usually 6 metres or 20 feet) and D is the distance at which each letter of this line subtends 5 minutes of arc.

2.2 Viewing Distance

The human visual system (HVS) uses two mechanisms to focus on objects: convergence and accommodation. Convergence denotes the eyes moving inward when focusing on nearby objects, and accommodation describes the focusing of objects of different distance by means of physically deforming the lens of the eye. The default distance at which objects appear sharp is called the resting point of accommodation (RPA). RPA is around 75 cm for younger people and increases in distance with age [12] .

The distance at which the eyes are set to converge when there is no object to converge on is called the resting point of vergence (RPV) [13]. RPV is 114 cm when looking straight ahead and drops to 89 cm when looking 30 degrees down [12].

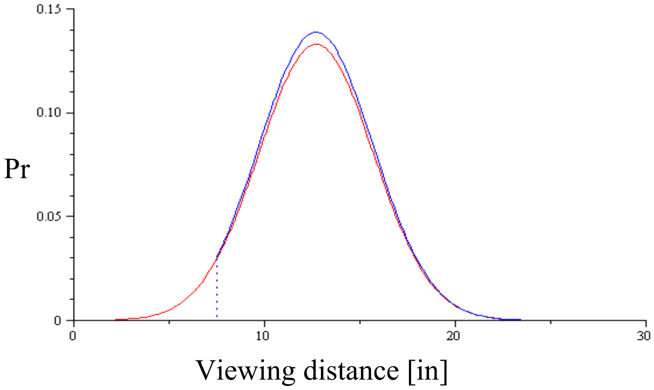


Figure 2-3 Probability distribution of smartphone reading distances [6]

Recently a study [14] has been conducted pertaining to the distances with which a person with normal 20/20 vision can be comfortable in reading text on smart phones. The result shows that viewing distances for a smartphone range from 7.5” to 23.6” with a mean distance 12.7” and standard deviation of 3”. The approximate shape of such a distribution, obtained by fitting Gaussian model, is shown in Figure 2-3.

2.3 Spatial Frequency

The spatial frequency is a measure of how often sinusoidal components of a structure repeat per unit of spatial distance. It is also often described as the frequency of change per angular unit, capturing the relative position of a viewer to the image as being projected to the screen (see Figure 2-4).

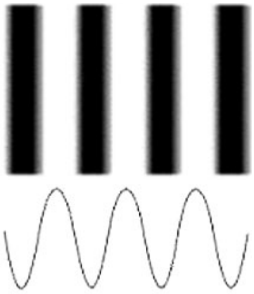


Figure 2-4 Sinusoidal grating

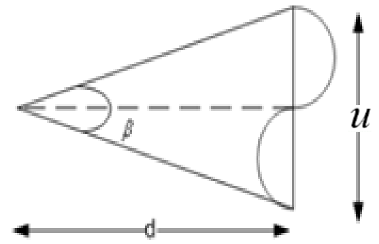


Figure 2-5 Computation of spatial frequency

As shown in Figure 2-5, spatial frequency,‘u’, of a sinusoidal grating (see Figure 2-4) with a cycle length of n pixels can be computed (in cycles per degree) as

$$u = \frac{1}{\beta}, \qquad \beta = \frac{\pi}{360}\arctan\left(\frac{n}{2d\rho}\right) \tag{2.2}$$

where ρ is the display pixel density (in pixels per inch), d is the distance between viewer and the screen (in inches) and β is the angular span of one cycle of grating (in degrees).

2.3.1 Spatial Frequency Limits Implied by Visual Acuity

The concept of visual acuity or 20/20 vision can also be understood as a limit in spatial frequency space. To illustrate this, consider Snellen’s E grating conversion presented in Figure 2-6.

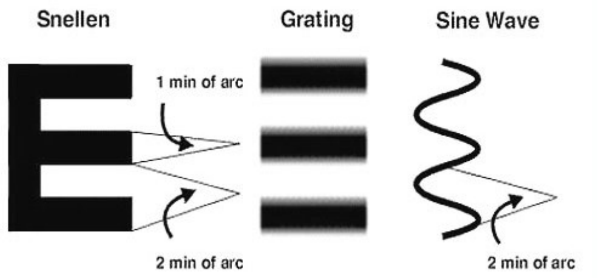


Figure 2-6 Finding highest spatial frequency limit for the HVS [11]

It can be observed that for the 20/20 Snellen’s letter E, there are 2 minutes of arc in one cycle. It is known that 60 minutes makes one degree and hence in a 20/20 letter, there are 30 cycles per degree (cpd). This means that 20/20 acuity implies ability to resolve frequencies of at least 30 cpd.

2.4 Contrast

Contrast is a fundamental characteristic of displays or other visual sources capturing the dynamic range of luminance that they can reproduce. There are several alternative definitions of contrast used in the literature. The important ones for this work will be the Michelson contrast, the Contrast Sensitivity, and the Contrast Ratio.

2.4.1 Michelson Contrast (C):

The Michelson definition of contrast is used very commonly in vision research. Michelson contrast C is defined as [15]:

$$\mathrm{C} = \frac{L_{max} - L_{min}}{L_{max} + L_{min}} \tag{2.3}$$

where L_{max} and L_{min} are luminances of darkest and brightest colors in an image or video projected to a screen. It follows from definition, Michelson contrast C ranges from 0 to 1.

2.4.2 Contrast Sensitivity (S):

Contrast sensitivity S is most commonly defined as an inverse of the Michelson contrast:

$$\mathrm{S} = \frac{1}{C} = \frac{L_{max} + L_{min}}{L_{max} - L_{min}} \tag{2.4}$$

The range of contrast sensitivities is from 1 to infinity. I.e. contrast sensitivity cannot be lower than 1.

2.4.3 Display contrast ratio (CR):

Contrast ratio is the ratio between the luminances of the brightest (typically white) L_{white} and the darkest (typically black) L_{black} colors that a display device can reproduce:

$$\mathrm{CR} = \frac{L_{white}}{L_{black}} \tag{2.5}$$

This Contrast ratio (CR) is commonly used by the display industry to characterize contrasts of TVs and monitors being produced. Such manufacturer-reported contrast ratios are typically measured in a dark room, and they can be very high (contrast ratios of 1000:1 or even 100000:1 are very common for modern displays). However, in the presence of ambient light contrast ratios can be several orders of magnitude lower.

2.4.4 Effect of Ambient Light on Display Contrast Ratio

Ambient light is a background (typically highly diffused) illumination present in most reproduction environments. It is typically measured as the luminous flux incident per unit area (lux).

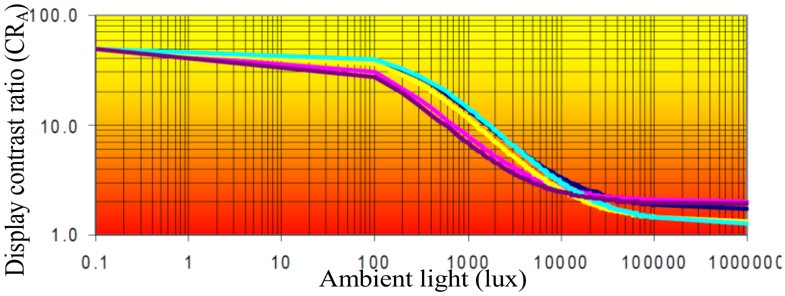


Figure 2-7 Contrast ratio of displays as fucntion of illuminance [7]

The reflection of ambient light by displays results in lowering their contrast.Figure 2-7 shows the contrast of different displays as a function of ambient illuminance.

The impact of ambient light on contrast can be easily quantified. Given the incident ambient illuminance, I [lux] , the reflected luminance from the display, L_{ref}, can be computed as [16]:

$$L_{ref} = \left(\frac{I}{\pi}\right)K \qquad (2.6)$$

where K is the reflection coefficient of a display. For typical consumer-grade displays, this coefficient may vary in the range from 4% to 10%.

Given this amount of reflected light, the effective contrast ratio of the display becomes:

$$CR_A = \frac{L_{white} + L_{ref}}{L_{black} + L_{ref}} \qquad (2.7)$$

This quantity, CR_A , is often called the ambient contrast.

2.5 Contrast Sensitivity Function

Visual acuity is measured using high contrast letters (black symbols on white background). The contrast Sensitivity Function (CSF) is a more complete characteristic of human vision, obtained by considering images of different contrasts.

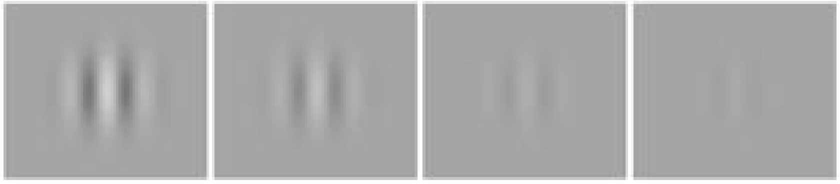


Figure 2-8 Gabor patches with progressively reduced contrast

Some example images, (so-called Gabor patches) as used in CSF measurements, are shown in Figure 2-8. Such patches are viewed from a distance limiting their angular span to a certain angle χ (usually between 2 and 12^0). The maximum and minimum luminances, L_{max} and L_{min}, of such patches are also controlled so as to achieve different levels of their contrast. During each test, the contrast of the patch is progressively reduced until the point when a viewer can no longer detect it. This test is repeated for patches with different frequencies. It is also performed involving a fairly large (20 viewers +) panel of viewers.

The Michaels contrast level C_T at which 50% of viewers say that they can see oscillations and the other 50% of viewers cannot – is called the contrast visibility threshold C_T. The inverse of it

$$S = \frac{1}{C_T} \tag{2.8}$$

is called the sensitivity threshold for Gabor patches with a certain spatial frequency u.

The Contrast Sensitivity Function is the collection of sensitivity thresholds measured across different spatial frequencies. An approximate shape of the CSF curve is illustrated in Figure 2-9.

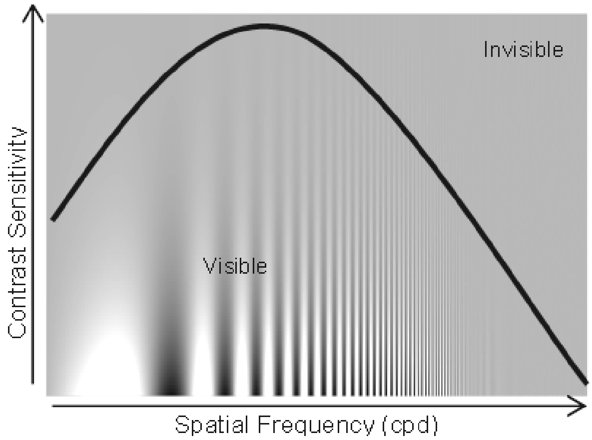


Figure 2-9 Illustration of shape of contrast sensitivity [17]

The two important points on the CSF curve are: maximum point – corresponding to about 3-5 [cpd], and a point at which it approaches contrast sensitivity of 1. This farthest right point coincides with the visual acuity limit.

2.5.1 CSF Models

Many models [18] [19] [20] [21] of CSF have been proposed in the literature. The recent models account for factors like object luminance, filed size, oblique effect, background luminance etc. Figure 2-8 shows the variations in CSF with changes in illumination and field size.

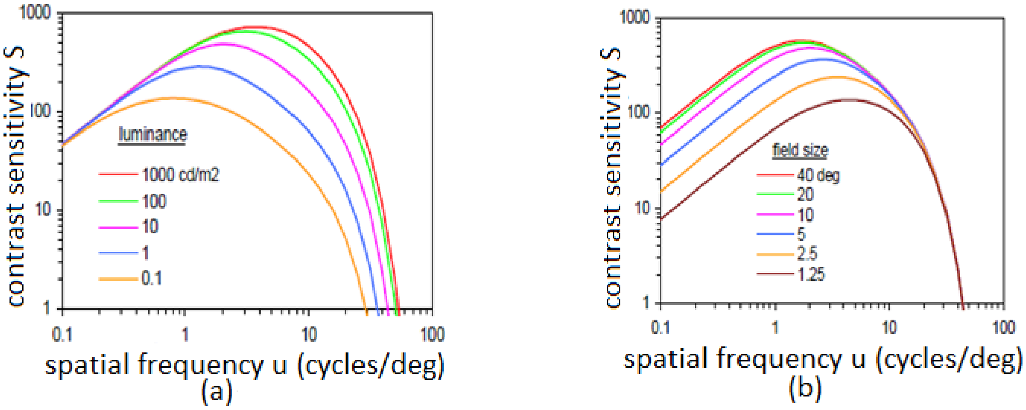


Figure 2-10 CSF Dependence (a) on illuminance, (b) on field size [20]

The formula for contrast sensitivity function $\mathrm{S(u)}$ derived by Barten [20] will be used as a CSF model in this thesis. It will be referred to as Barten-04 model. This formula is:

$$\mathrm{CSF(u)} = \mathrm{S(u)} = \frac{A\,e^{-D\,u^2}}{\sqrt{(B+u^2)\left(C+\frac{1}{1-e^{-0.02u^2}}\right)}} \tag{2.9}$$

where u is the spatial frequency in cpd, and A,B,C,D, are terms depending on the average luminance of the patch L and its angular size χ. They are defined as follows:

$$\mathrm{A} = \frac{5200E}{\sqrt{0.64}}, B = \frac{1}{0.64}\left(1+\frac{144}{X_o^2}\right),$$

$$C = \frac{63}{L_o^{0.83}},\; D = 0.0016\left(1+\frac{100}{L_o}\right), \tag{2.10}$$

$$E = exp\left(-\frac{ln^2\left(\frac{L_S}{L_O}\left(1+\frac{144}{X_O^2}\right)^{0.25}\right)-ln^2\left(\left(1+\frac{144}{X_O^2}\right)^{0.25}\right)}{2\,ln^2(32)}\right)$$

where L is the luminance in cd/m^2 and X_o^2 is the angular patch area in square degrees.

2.6 Summary

Several phenomena and facts from human vision have been reviewed in this chapter. This chapter also explained several related characteristics of displays and reproduction environments. A set of formulae have been provided to enable quantitative operations with these characteristics and phenomena. Some of these concepts and formula are used in the next chapter to derive the maximum visible frequency and the sufficient resolution.

Chapter 3

Maximum Visible Frequency and Sufficient Resolution

In this chapter, the concepts of maximum visible frequency and sufficient resolution will be introduced. Both are derived using limits imposed by the contrast sensitivity function (CSF) and ambient contrast characteristic of the display and reproduction environment.

3.1 Display Nyquist Frequency

The Display Nyquist frequency is the highest spatial frequency a display can reproduce with the limit implied by the display pixel density and the viewing distance in a particular viewing set up. It can be calculated from the expression (2.2), by setting the wavelength, n, to be 2 pixels, and computing the resulting spatial frequency:

$$u_{D,Nyq} = \left[\frac{\pi}{360} arctan\left(\frac{1}{d\rho}\right)\right]^{-1} \qquad (3.1)$$

Here d is the distance between viewer and the screen, ρ is the display pixel density and $u_{D,Nyq}$ is the display nyquist frequency measured in cpd.

3.2 Maximum Contrast

Consider an environment with a certain amount of ambient light present. Then, the peak L_{max}^{D} and base L_{min}^{D} luminances of a display can be expressed as:

$$L_{max}^{D} = L_{white} + L_{ref}\ , \qquad L_{min}^{D} = L_{black} + L_{ref} \qquad (3.2)$$

where L_{ref} is the reflected luminance and L_{white}, L_{black} are the absolute peak and dark luminance. Hence, the Michelson contrast (2.3) of the display becomes:

$$C_D = \frac{L_{max}^{D} - L_{min}^{D}}{L_{max}^{D} + L_{min}^{D}} = \frac{\mathrm{CR_A} - 1}{\mathrm{CR_A} + 1}$$

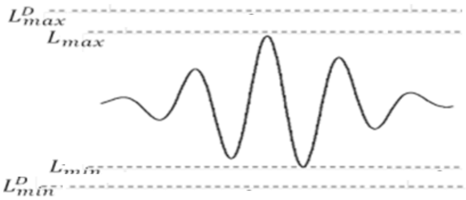


Figure 3-1 Image vs display luminance limits.

If it is now assumed that an image is rendered to such a screen. Then for any oscillation rendered on this device, it follows that [22]

$$L^D_{min} \leq L_{min} \leq L_{max} \leq L^D_{max} \tag{3.3}$$

By plugging these inequalities in an equation of Michelson contrast (2.3), it can be shown that the Michelson contrast of an image rendered to a screen has a natural upper bound:

$$\mathrm{C} = \frac{L_{max} - L_{min}}{L_{max} + L_{min}} \leq \frac{L^D_{max} - L^D_{min}}{L^D_{max} + L^D_{min}} = C_D \tag{3.4}$$

where C_D is the ambient contrast of a display.

3.3 Minimum Contrast Sensitivity

The reciprocal of inequality (3.4) implies that contrast sensitivity, *S,* for oscillations within an image rendered to the screen will satisfy

$$S \geq S_{min} \tag{3.5}$$

where

$$S_{min} = \frac{1}{C_D} = \frac{\mathrm{CR_A} + 1}{\mathrm{CR_A} - 1} \tag{3.6}$$

is a contrast sensitivity corresponding to the ambient contrast of the display.

3.4 Highest Visible Frequency

The Barten-04 formula for contrast sensitivity function is now retrieved from equation (2.8), and by introducing the following alteration, an approximation that fully retains the shape of the upper (u>4 cpd) branch can be obtained (see Figure 3-2):

$$\mathrm{S(u)} = \frac{A\,e^{-D\,u^2}}{\sqrt{(B+u^2)\left(C+\frac{1}{1-e^{-0.02u^2}}\right)}} \rightarrow \ \mathrm{S_1(u)} = \frac{A\,e^{-D\,u^2}}{\sqrt{(B+u^2)(C+1)}} \qquad (3.7)$$

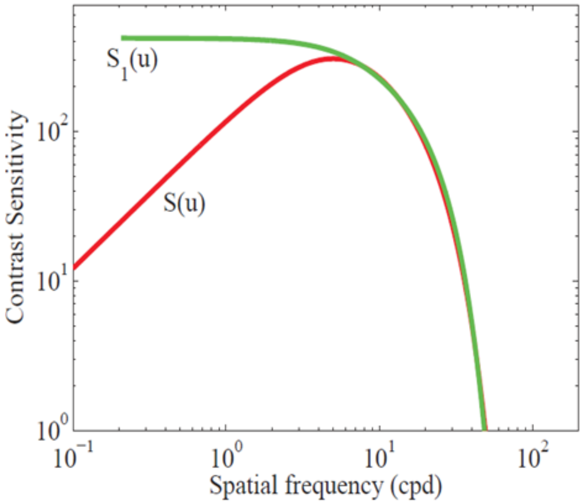


Figure 3-2 CSF approximation [23]

This simplified function is sufficient for modeling low-pass characteristic of CSF, and it can be analytically inverted [23]. The inverse function becomes:

$$u = S_1^{-1}(s) = \sqrt{\frac{\mathrm{Lambert}W\left(\frac{2DA^2e^2DB}{(C+1)s^2}\right)}{2.D}} \qquad (3.8)$$

where $\mathrm{Lambert}W(z)$ is a solution of equation:

$$\mathrm{Lambert}W(z).\,e^{\mathrm{Lambert}W(z)} = z \qquad (3.9)$$

known as the Lambert W function [24].

The plot of this inverse function is presented in Figure 3-3. It is noted that this provides the connection between threshold contrast sensitivity s and spatial frequency u. Hence, if it is known that there is a limit s_{min}, such that for any oscillation within an image, its contrast sensitivity satisfies:

$$S \geq S_{min}$$

then, by plugging S_{min} in the inverse CSF formula, the frequency, u_{max}, is obtained by providing an upper bound for frequencies that are visible (detectable with probability ½).

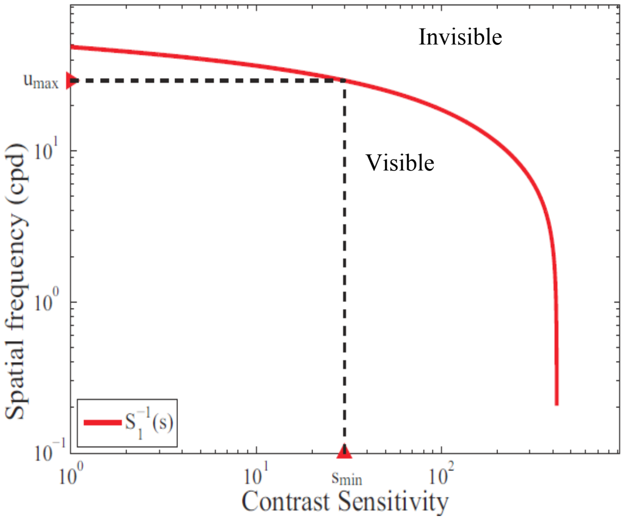


Figure 3-3 Inverted CSF Model [23]

Therefore, u_{max} is called as a maximum visible spatial frequency. The exact formula for u_{max} derived from Barten-04 CSF model is:

$$u_{max} = S_1^{-1}(S_{min}) = \sqrt{\frac{\mathrm{Lambert}W\left(\frac{2\,DA^2e^2DB}{(C+1)\,s_{min}{}^2}\right)}{2\,D}} \tag{3.10}$$

where S_{min} is the lower bound on contrast sensitivity implied by ambient contrast of the display (3.6), and where terms A,B,C,D are given by (2.10).

3.5 Sufficient Resolution

If u_{max} is known, it can be used to derive resolution of video / display sufficient for reproducing oscillations with frequencies $u \leq u_{max}$. For example, using (2.2) an expression for pixel density can be obtained:

$$\rho = \frac{n}{2\,d\tan\left(\frac{\pi}{360\,\mathrm{u}_{max}}\right)} \tag{3.11}$$

By further setting n=2, so that u_{max} corresponds to the Nyquist frequency.

$$\rho \geq \rho_{min} = \frac{1}{d \ \tan\left(\frac{\pi}{360 \ \mathrm{u}_{max}}\right)} \tag{3.12}$$

where ρ_{min} can be understood as minimum sufficient density.

The expression (3.12) can be further simplified. It can be noted that for realistic (20-40 cpd) values of u_{max} the argument of $\tan(x)$ function will be small. Therefore, series expansion can be used

$$\tan(x) = x + \frac{1}{3}x^3 + \frac{2}{15}x^5 + \cdots \quad |x| < 1 \tag{3.13}$$

and pick its first term:

$$\tan(x) \approx x \tag{3.14}$$

With this simplification, the minimum density becomes:

$$\rho_{min} \approx \frac{360}{\pi} \frac{\mathrm{u}_{max}}{d} \tag{3.15}$$

By considering now a device has physical display dimensions M X N (width x height) [inches^2], we can now turn ρ_{min} in pixel counts:

$$\begin{aligned} W_{suff} &= M \ \rho_{min}, \\ H_{suff} &= N \ \rho_{min} \end{aligned} \tag{3.16}$$

where $W_{suff} \times H_{suff}$ is the sufficient resolution of the display.

It can also be computed using display's true pixel resolution $W_{disp} \times H_{disp}$ and pixel density ρ:

$$\begin{aligned} W_{suff} &= W_{disp} \frac{\rho_{min}}{\rho}, \\ H_{suff} &= H_{disp} \frac{\rho_{min}}{\rho} \end{aligned} \tag{3.17}$$

3.6 Summary

The chapter introduced the concept of the maximum visible frequency. The limitations posed by a display and the surrounding environment in reproducing this maximum visible frequency, leads to the calculation of resolution sufficient to display content. These calculations will be used in the 5th chapter for implementing the user adaptive video player. In the next chapter, the adaptive video streaming technology standardized as MPEG-DASH will be discussed.

Chapter 4

Adaptive Video Streaming and MPEG-DASH

In this chapter, the reference implementation of recently issued MPEG-DASH [4] standard is reviewed along with the concepts of adaptive video streaming.

4.1 Adaptive Bitrate Streaming

Adaptive bitrate streaming is a technique used in streaming multimedia over computer networks. While in the past most video streaming technologies used streaming protocols such as real-time transport protocol (RTP) with real time streaming protocol (RTSP). Today's adaptive streaming technologies are almost exclusively based on HTTP [25] and designed to work efficiently over large distributed HTTP networks such as the Internet.

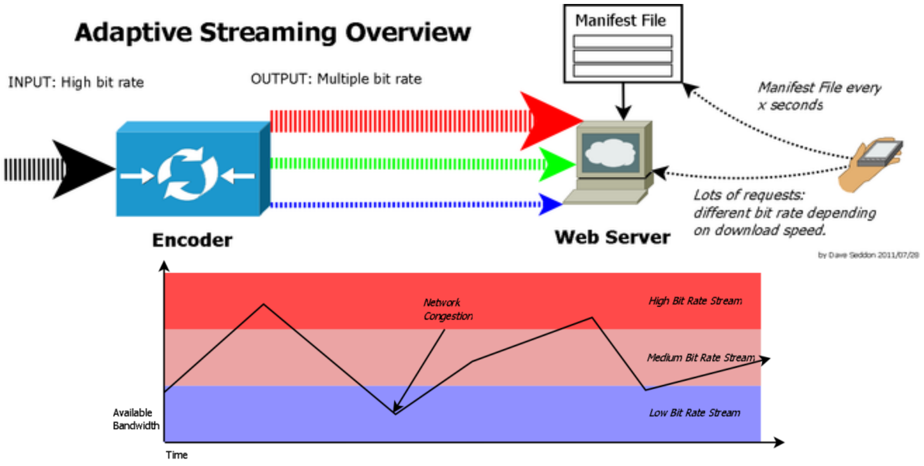


Figure 4-1 Overview of adaptive bitrate streaming [27]

In adaptive streaming, the quality of video stream is adjusted by detecting the user's bandwidth and CPU capacity in real time. This requires the use of an encoder which can encode a single source video at multiple bit rates. The player client switches between streaming different encodings depending on the available resources.

4.2 MPEG-DASH Overview

Dynamic Adaptive Streaming over HTTP (DASH) [3], also known as MPEG-DASH [4], is an adaptive bitrate streaming technique that enables high quality streaming of media content over the Internet delivered from conventional HTTP web servers. MPEG-DASH works by breaking the content into a sequence of small HTTP-based file segments, each segment containing a short interval of playback time of a content that is potentially many hours in duration, such as a movie or the live broadcast of a sports event. The content is made available at a variety of different bit rates, i.e., alternative segments encoded at different bit rates covering aligned short intervals of play back time are made available.

As the content is played back by an MPEG-DASH client, based on current network conditions, it automatically selects the next segment to download. The client selects the segment with the highest bit rate possible that can be downloaded in time for play back without causing stalls or re-buffering events in the playback. Thus, an MPEG-DASH client can seamlessly adapt to changing network conditions, and provide high quality play back without stalls or re-buffering events.

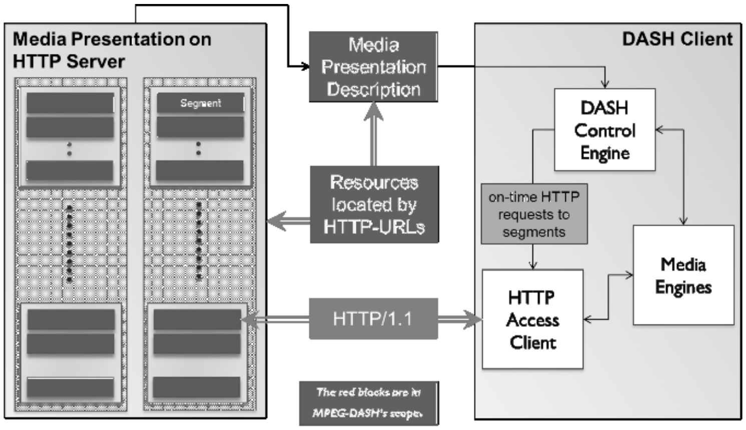


Figure 4-2 Overview of MPEG-DASH [26]

Figure 5-1 illustrates a simple streaming scenario between an HTTP server and a DASH client. In this figure, the multimedia content is captured and stored on an HTTP server and is delivered using HTTP. The content exists on the server in two parts: Media Presentation Description (MPD), which describes a manifest of the available content, its various alternatives, their uniform resource locator (URL) addresses, and other characteristics; and segments, which contain the actual multimedia bit streams in the form of chunks, in single or multiple files.

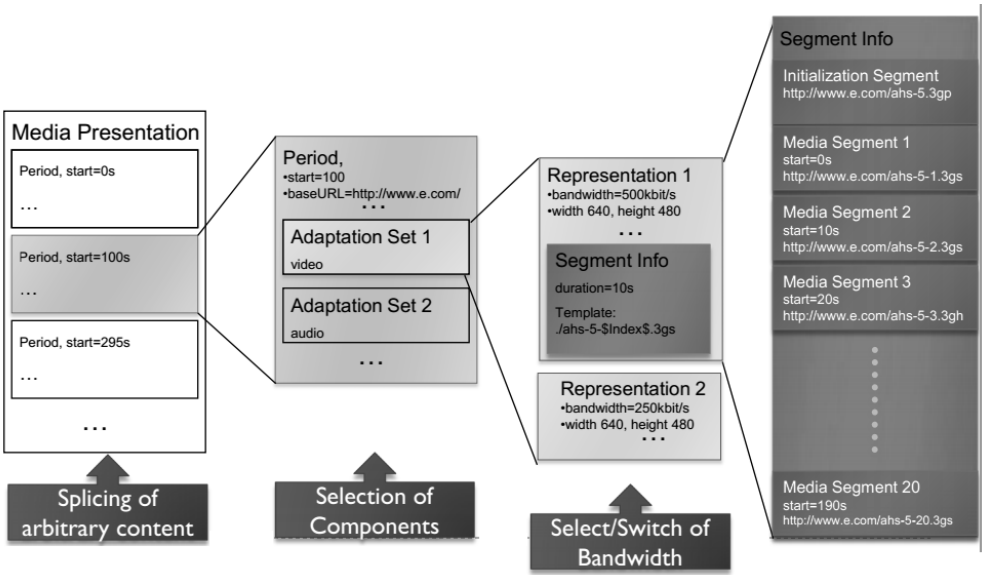
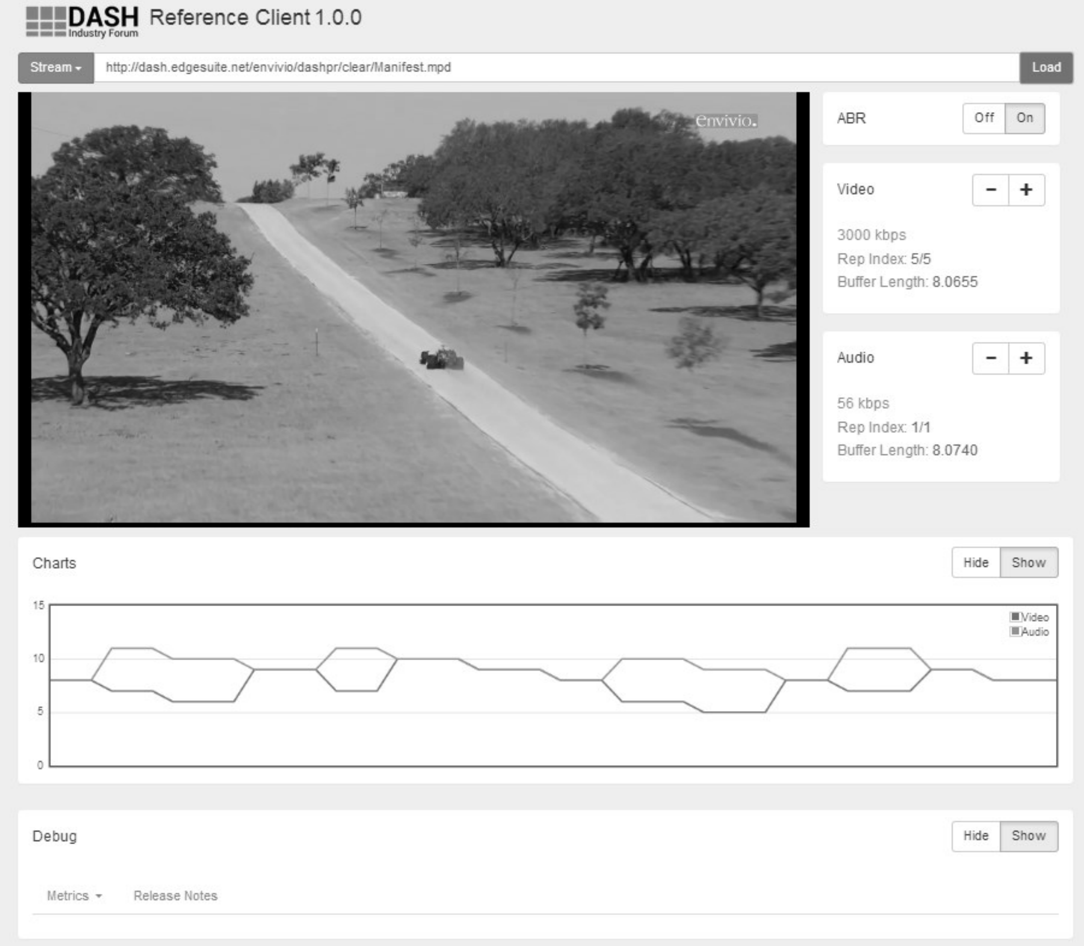


Figure 4-3 Media Presentation Description (MPD) [27]

To play the content, the DASH client first obtains the MPD. The MPD can be delivered using HTTP, email, thumb drive, broadcast, or other transports. By parsing the MPD, the DASH client learns about the program timing, media-content availability, media types, resolutions, minimum and maximum bandwidths, and the existence of various encoded alternatives of multimedia components, accessibility features and required digital rights management (DRM), media-component locations on the network, and other content characteristics. Using this information, the DASH client selects the appropriate encoded alternative and starts streaming the content by fetching the segments using HTTP GET requests.

4.3 Reference Implementation

Figure 4-4 shows the video (a car scene) played on the reference DASH Player implemented in JavaScript and HTML by the DASH Industry forum [28]. The MPD can be keyed in to the space available or selected from the pre-configured “Stream” dropdown. When the load button is pressed, the player parses the MPD and starts playing the video by displaying relevant information in the area provided.



DASH Industry Forum Reference Client 1.0.0

http://dash.edgesuite.net/envivio/dashpr/clear/Manifest.mpd

ABR: On

Video
3000 kbps
Rep Index: 5/5
Buffer Length: 8.0655

Audio
56 kbps
Rep Index: 1/1
Buffer Length: 8.0740

Figure 4-4 Reference DASH-JS Player [29]

The adaptive bit rate (ABR) on/off switch will enable/disable the adaptive selection of the streams. The Video / Audio information box shows the bitrate of selected stream, the selected video representation from MPD and the filled buffer length in seconds for video to be played. When ABR is off, the + / - buttons can be used to manually select the adaptation set to be played by the player. There are also other information tools like network adaptation charts and debug information made available to analyze and improve the player.

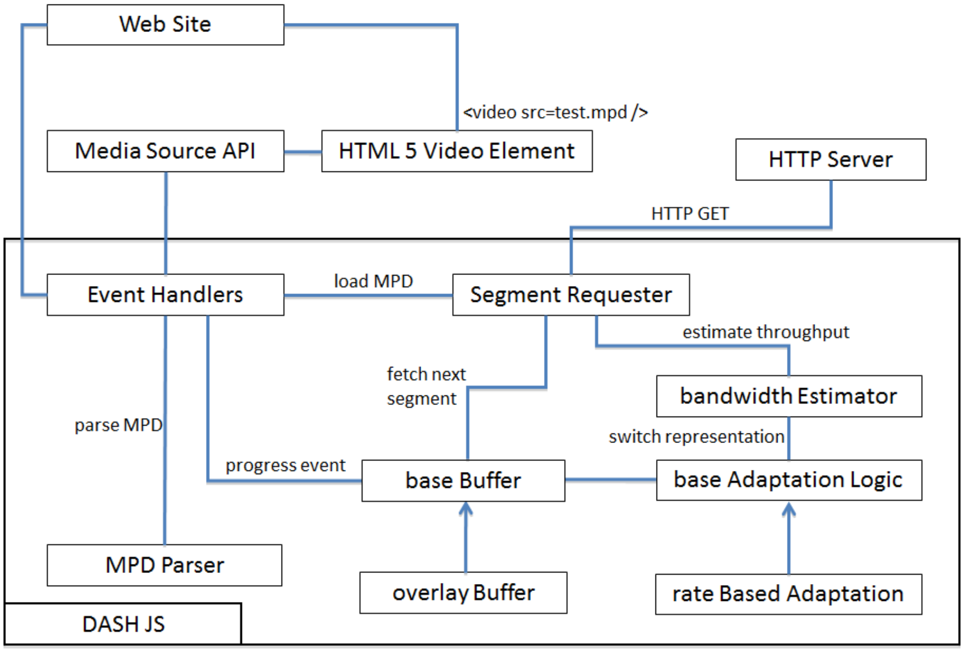


Figure 4-5 Reference DASH Player block diagram [30]

Figure 4-5 shows the parts of the JavaScript dash player. It uses several HTML and latest JavaScript enhancements to implement the adaptive streaming.

- *Media source APIs*: allow JavaScript to generate media streams for playback.
- *HTML5 video element*: added to HTML to playback video/animation content.
- *Event Handlers*: captures and processes the events like load video, parse MPD, etc.

- *MPD Parser*: checks and parsers the given MPD for adaptation set and URL.
- *Adaptation* Logic: estimates the next segment to be requested by calculating the buffer available and bandwidth estimate given by the bandwidth estimator.
- *Segment requester:* requests the selected stream by adaptation logic by posting appropriate HTTP GETs to the server.

4.4 Summary

In this chapter, the working principles of adaptive bitrate streaming is described with special attention to the MPEG-DASH. It is noted that using this method, any custom logic to select the video stream can be implemented with fewer modifications to the reference player described. In the next chapter, addition of user adaptive logic to the reference DASH implementation will be described,

Chapter 5

Implementation of User Adaptive Video using MPEG-DASH

In this chapter, the implementation method of user adaptive video using MPEG-DASH is described. It also describes the procedure for obtaining all the necessary data to compute the sufficient resolution of a device.

5.1 Streaming Client Model

In order to implement the user adaptive streaming, certain parameters of viewing conditions are necessary. These parameters can be obtained by the use of built in sensors of a mobile device, such as a front-facing camera and accelerometer to detect the presence of the user, his proximity, and viewing angle. The ambient illuminance reported form illuminance sensor and the information about brightness from the device settings can be used to estimate the effective contrast ratio of the screen.

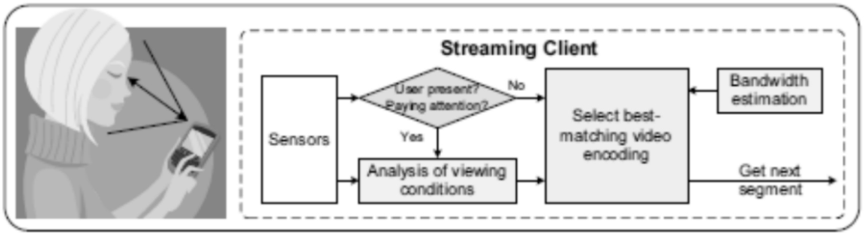


Figure 5-1 Streaming client with user adaptation [6]

The streaming client will obtain the characteristics of encoded video such as spatial resolution, bitrate, etc… from the MPD it parses. The user adaptive client can also estimate the sufficient resolution of the video to be played by using the viewing environment estimates given by the sensor data.

5.2 Determination of User Distance

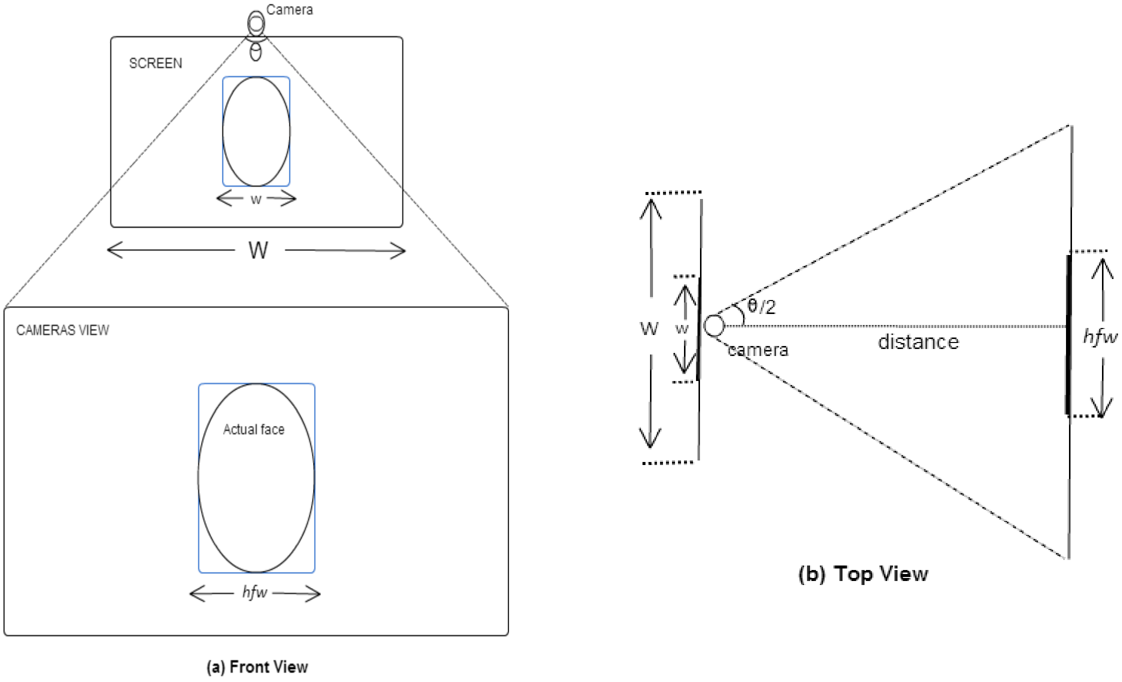
As discussed in Chapter 2, distance plays an important role for a viewer to discern the information presented on screen. To measure the distance of the screen from the viewer, inbuilt front facing camera is used. Sensors like accelerometer can be used to determine the device state which in turn can be used to estimate the distance.

5.2.1 Estimation of Distance Using Front Facing Camera:

Figure 5-2 shows a typical viewing setup. The front camera in the device is used to capture the users face. The distance of the user from the device can be calculated using Figure 5-2 (b) as:

$$d = \frac{W \cdot hfw}{2 \cdot w \cdot \tan(\theta/2)} \tag{5.1}$$

where d is the distance, W is the width of the screen in pixels, w is the width of the captured face in pixels, hfw is the actual human face width (5.3in for adults) [31] in inches and θ is the horizontal viewing angle of the camera.



(a) Front View

(b) Top View

Figure 5-2 Distance calculation using front facing camera

5.2.2 Estimation of Distance Using Accelerometer

Using the device camera along with video content playback will impact the performance and battery consumption of any mobile device and it would also result in privacy concerns of the user. According to a research [32], user positions while consuming multimedia content is predictable. Hence, by determining the user positions without the use of a camera will give performance benefits and take away the privacy concerns from the user.

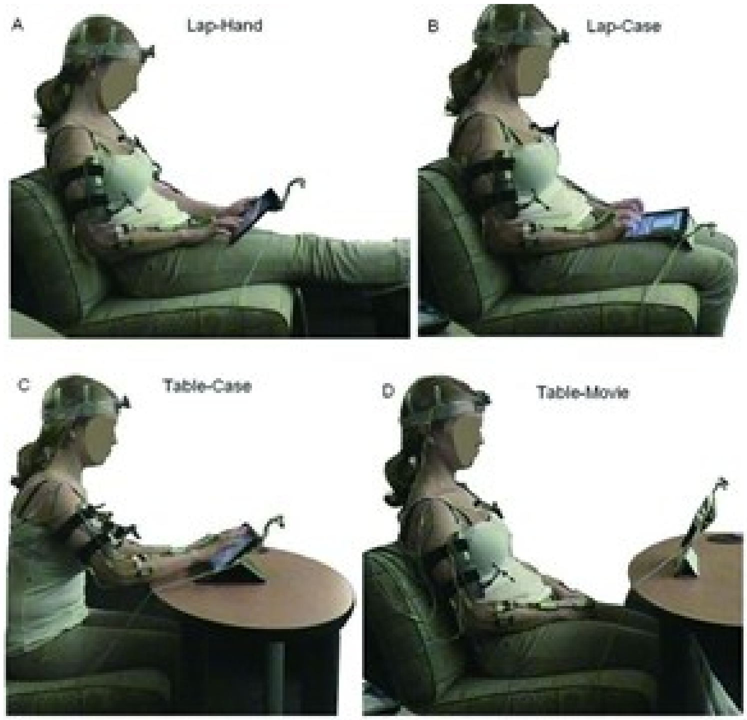


Figure 5-3 Tablet viewing positions [32]

Figure 5-3 shows different configurations in which a tablet can be placed according to the viewer’s preferences. These limited combinations of positions can be used to pre determine the distance of the screen from the viewer.

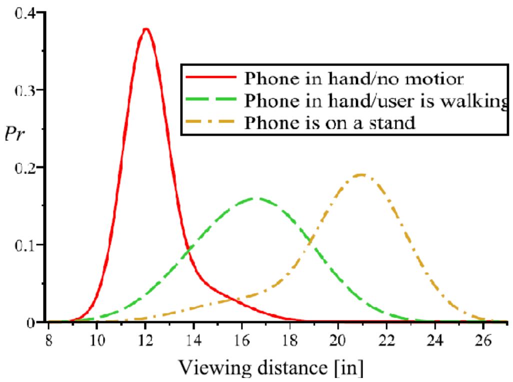


Figure 5-4 Probability distribution of device distances [6]

Figure 5-4 shows the distribution of viewer distances when device is used in different configurations. To estimate this device state, acceleration sensor (accelerometer) present in the device can be used.

The accelerometer outputs the acceleration applied to the device by measuring the forces applied to the sensor. The measured acceleration is always influenced by the force of the earth's gravity.

$$a_d = -g - \Sigma\,(F/m) \tag{5.2}$$

where a_d is the acceleration applied to the device, g the force of gravity, F the force acting on the device, and m the mass of the device. The sign Σ represents the sum of the x-, y- and z axis.

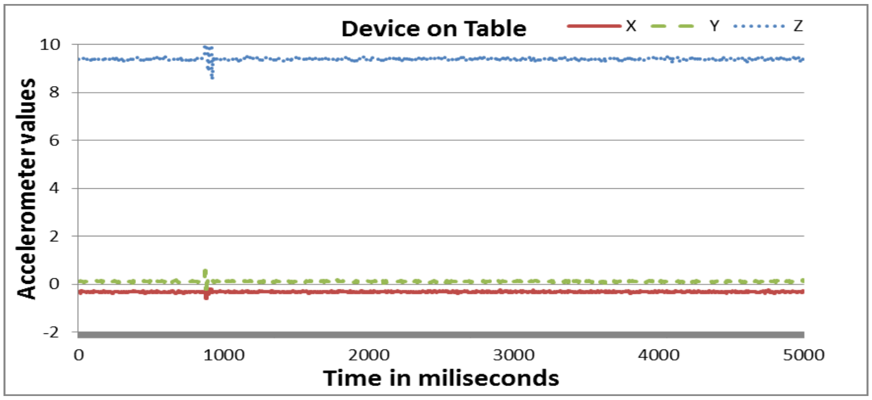


Figure 5-5 Accelerometer outputs for device on stand

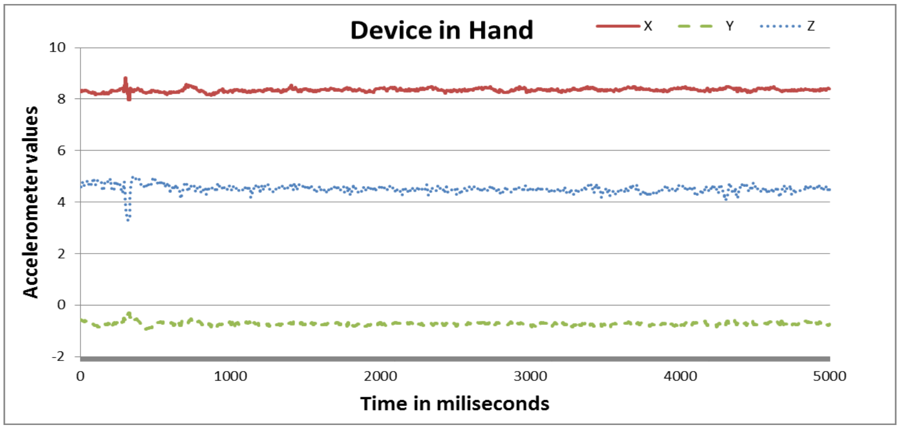


Figure 5-6 Accelerometer outputs for device in hand

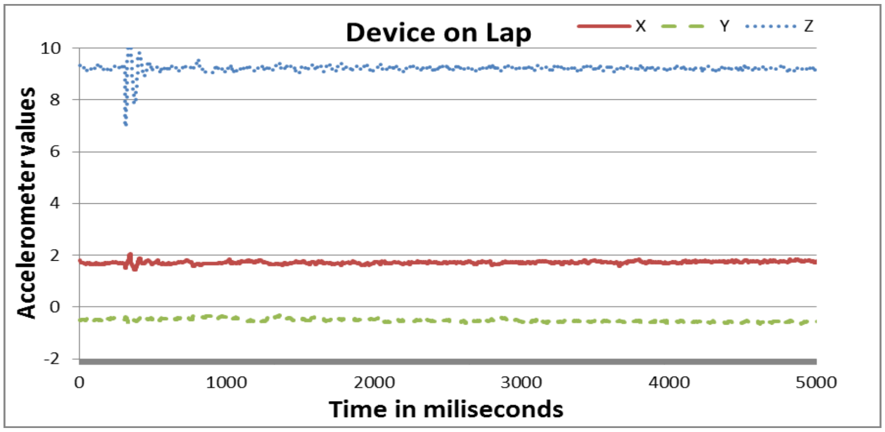


Figure 5-7 Accelerometer outputs for device on lap

Figure 5-4 through Figure 5-7 shows the accelerometer data for different activities. Using this accelerometer data, the state of the device like “In hand”, “on Stand” or “on lap” is determined. Once the state is known, pre-determined distances for the particular state can be mapped and used in the adaptation logic.

5.3 Obtaining Effective Display contrast

The Effective display contrast of the device depends on two factors. The brightness level set for the screen and the surrounding ambient light. Smart phones host an ambient light sensor to detect the amount of light incident on the phone and report it through application interfaces (API). Manufactures also expose the API for settings to get the brightness level set for the screen.

To obtain the display contrast ratio, the luminance limits are measured in the lab set up as shown in the Figure 5-8. Specifically a Konica Minolta CS 200 luminance meter is used to measure luminances at black and white points on the display displaying a reference image under several different ambient illuminance settings. KinoFLo lights and diffuser are used to produce ambient light up to 10000 lux.

Figure 5-8 Measurement of display contrast ratio

The following are the constants for a given device obtained from measurements

- Reflection coefficient K given by manufacturer.
- Luminance of screen with white background and maximum brightness (Lw_{max})
- Luminance of screen with white background and minimum brightness (Lw_{min})
- Luminance of screen with black background and maximum brightness (Lb_{max})
- Luminance of screen with black background and minimum brightness (Lb_{min})
- Brightness saturation level (B_sat) at which luminance with white background reaches maximum (see Figure 5-9)
- Best fit gamma (γ) calculated from power law model shown in the Figure 5-9

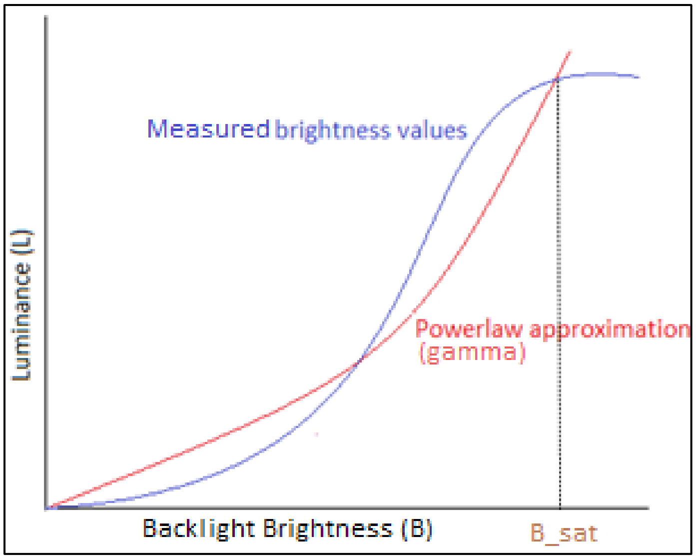


Figure 5-9 Brightness vs luminance

The procedure to get the effective contrast follows:

1. Obtain ambient illuminance (I) from the device sensor and calculate the reflected luminance

$$L_{ref} = \left(\frac{I}{\pi}\right)K$$

2. Obtain the actual brightness (B) from the device API and calculate Q.

$$Q(B) = \left(\frac{B}{B\_sat}\right)^{\gamma}$$

3. Calculate the intensity of the white (L_{white}) and the black (L_{black}) luminances

$$L_{white}(B) = Lw_{min} + (Lw_{max} - Lw_{min}) * Q(B),$$

$$L_{black}(B) = Lb_{min} + (Lb_{max} - Lb_{min}) * Q(B)$$

4. Now the maximum and the minimum display luminance can be calculated using

$$L_{max}^{D} = L_{white} + L_{ref}\,, \qquad L_{min}^{D} = L_{black} + L_{ref}$$

5. Finally, the effective Michelson contrast is calculated as

$$C_D = \frac{L_{max}^{D} - L_{min}^{D}}{L_{max}^{D} + L_{min}^{D}}$$

5.4 Display Pixel Density and Sufficient Resolution

Display pixel density and actual screen resolution for a display are constants and typically measured by the manufacturer. Also, there are API's exposed in most of the platforms to obtain the resolution and the display pixel density for the application. With all these necessary parameters obtained from the sensors and measurement data, the sufficient resolution for the display at a given distance, contrast and pixel density is calculated using (3.17).

5.5 User Adaptive DASH Implementation

MPEG-DASH gives the freedom to implement and test custom adaptation logic to select one of the available video streams from an MPD. This feature permits adding a separate user adaptation block to the reference DASH implementation which calculates the sufficient resolution among the resolutions provided by MPD.

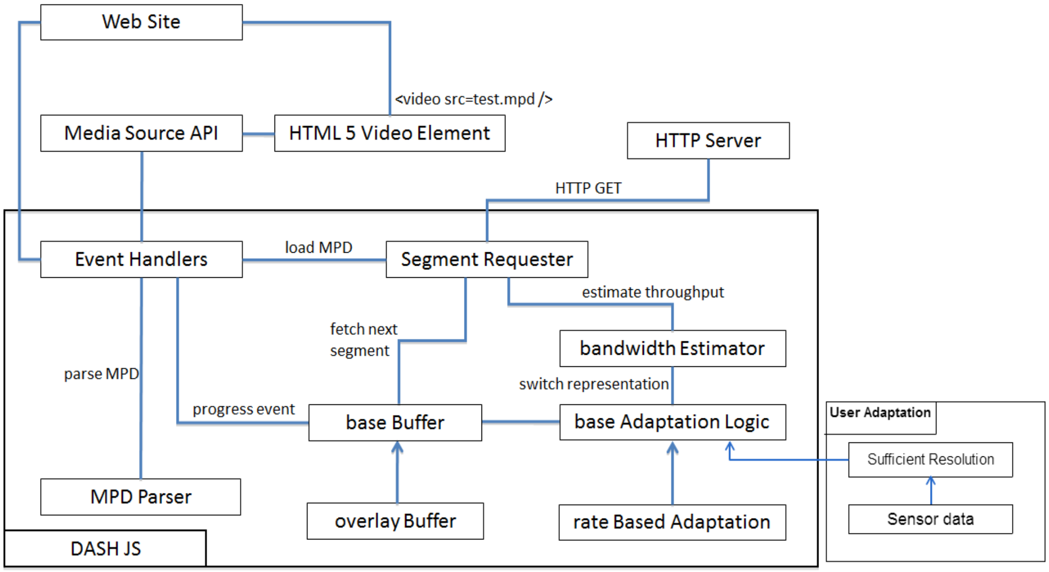


Figure 5-10 User adaptive Implementation

Figure 5-10 shows the addition of viewing environment measurement in selecting the bitrate and resolution. The required parameters like distance, ambient light, contrast and screen properties are obtained by sensors present in the device. Using this data, sufficient resolution is calculated. Once the sufficient resolution is obtained, the base adaptation logic is modified in such a way that, stream which is higher but closer to sufficient resolution is selected to play back.

2433 kbps
Resolution: 1056 x 592
Rep Index: 7/10
Buffer Length: 6.8701

64 kbps
Rep Index: 1/1
Buffer Length: 7.0328

Device State : On Stand
Without UAV: Rep Index: 8/10 Resolution: 1280 x 720 Bandwidth: 3287 kbps
With UAV : Rep Index: 7/10 Resolution: 1056 x 592 Bandwidth: 2433 kbps
Distance: 35 PPI:208 Sufficient Resolution:1054 x 593
Rate Savings: 25.98 %

Figure 5-11 Modified Player with user adaptation

Figure 5-11 shows an example video (a rock in the scene) played by the modified reference DASH player with user adaptive video streaming. A new information area called “User Adaptation” is created to display the user adaptive selections used for the current playback. They are:

- *Device State*: shows the accelerometer estimated state of the device
- *Wihtout UAV:* shows the representation that would have been selected without the user adaptation logic.

- *With UAV*: shows the current selected representation using the user adaptation.
- *Rate Savings:* will show the savings achieved by using user adaptation.

5.6 Implementation example

Consider that the bandwidth is available to play a 1980X1080 HD video on a smartphone with display density 330 ppi, screen resolution 1980X1080 which is placed at a distance of 15” from user and with an ambient light 200 lux.

Without considering the viewing conditions, the highest resolution 1080p with resolution 1920X1080 will be selected by the DASH client. Considering the viewing conditions, the calculation of sufficient resolution is described below:

The typical value of u_{max} for an ambient illuminance of 200 lux would be 35 cpd.

$$\rho_{min} = \frac{360}{\pi} \cdot \frac{35}{15} = 267\ ppi$$

$$W_{suff} = \frac{267}{330} \cdot 1080 = 875\ , \qquad H_{suff} = \frac{267}{330} \cdot 1920 = 1555$$

Hence, if the viewing conditions are considered, with the available streams from the MPD, the resolution selected will be “video 2: 880 X 1564, and 4.4 Mbps” instead of “video 1: 1920 X 1080, and 6 Mbps” saving almost 26.66% of bandwidth.

Chapter 6 outlines the results and analysis of this implementation with more examples.

Chapter 6

Results and Analysis

In this chapter, the device and media used to test the implementation is described. Bitrate savings reported by the device due to user adaptation are also discussed.

6.1 Device Used

Microsoft Surface Pro with the latest windows 8.1 [33] supported the java script player [29] and the sensor data API's. Hence this device was selected to implement the player.

6.2 Media Used

The DASH reference player uses media from different formats and resolutions. Amongst the reference media from DASH, the MPDs which contain streams of resolutions 1920X1080 or higher are selected to compare the bandwidth savings. All the MPDs used along with a screen shot of media are listed in Appendix A.

Tables 6-1 through 6-4 list the representations, resolutions available and the corresponding bitrates of the media used.

Table 6-1 Representations avaialbe: Sequence A

Representations available	Resolutions (pixels)	Corresponding bitrates (bits /sec)
1	1280 x 720	3000000
2	1024 x 576	2000000
3	704 x 396	1000000
4	480 x 270	600000
5	320 x 180	349952

Table 6-2 Representations avaialbe: Sequence B

Representations available	Resolutions (pixels)	Corresponding bitrates (bits /sec)
1	1920 X 1080	6000000
2	1568 X 880	4441000
3	1280 X 720	3287000
4	1056 X 592	2433000
5	848 X 480	1801000
6	688 X 384	1333000
7	576 X 320	986000
8	448 X 256	730000
9	368 X 208	540000
10	320 X 176	400000

Table 6-3 Representations avaialbe: Sequence C

Representations available	Resolutions (pixels)	Corresponding bitrates (bits /sec)
1	1920 X 1080	770663
2	1280 X 720	514793
3	640 X 360	194834
4	320 X 180	50842

Table 6-4 Representations avaialbe: Sequence D

Representations available	Resolutions (pixels)	Corresponding bitrates (bits /sec)
1	1920 X 1080	5933486
2	1280x720	3360441
3	960 X 540	2222352
4	640 X 360	985321
5	320 X 180	391544

6.3 Viewing Configurations

All the results are based on real time sensor data from the device. There was no ambient light sensor data available from JavaScript API's for this device, hence a constant of 200 lux is used for the experiment.

Accelerometer data is used to estimate the user device state and mapped to a corresponding pre calculated distance. The mapping of state to the distance used is tabulated in Table 6-5 with “In hand”, “On Stand” and “On Lap” configurations.

Table 6-5 Device state mapped to distance

Device State	Corresponding distance
In Hand	12”
On Lap	22”
On Stand	32”
On Table facing up	60”

6.4 Bandwidth Savings when Highest Bitrate Available

Figures 6-1 through 6-4 give the bandwidth savings when the user adaptive implementation is used and the network is supportive to allow the client to play the maximum bitrate available.

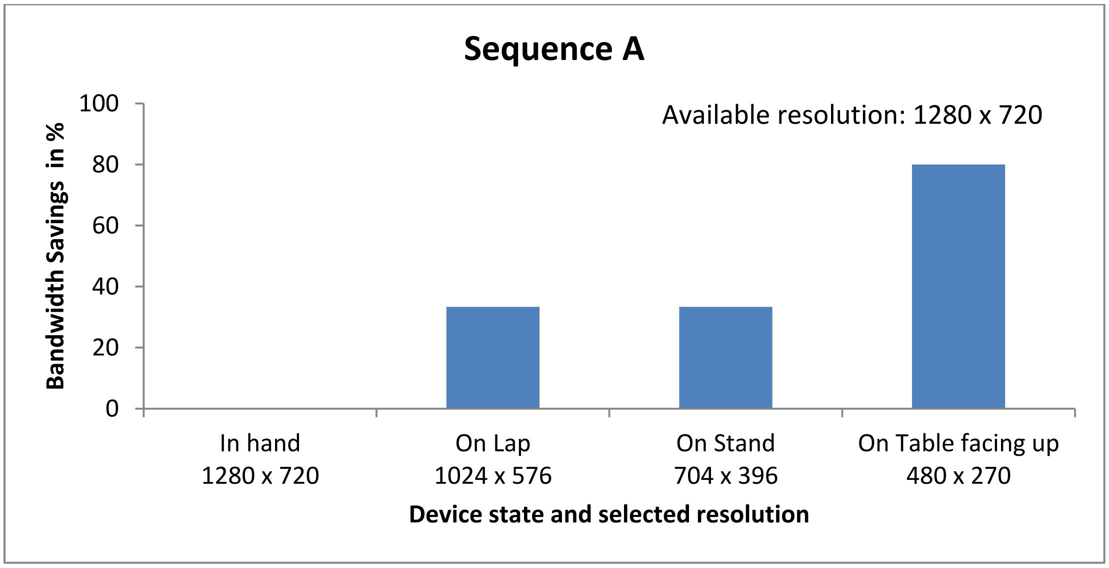


Figure 6-1 Savings for sequence A (All Bitrates available)

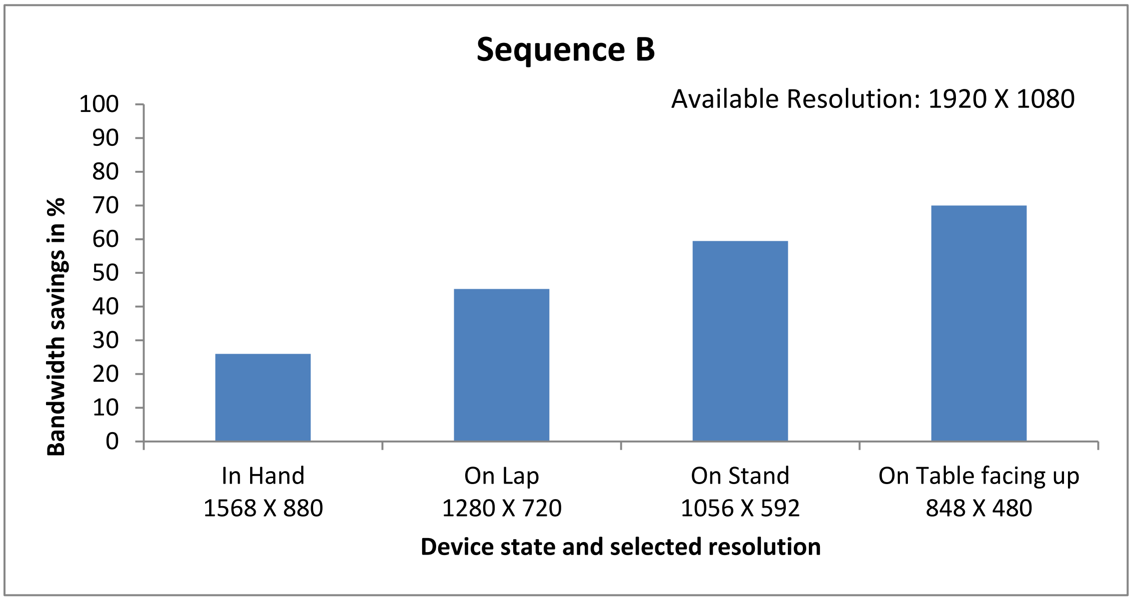


Figure 6-2 Savings for sequence B (All Bitrates available)

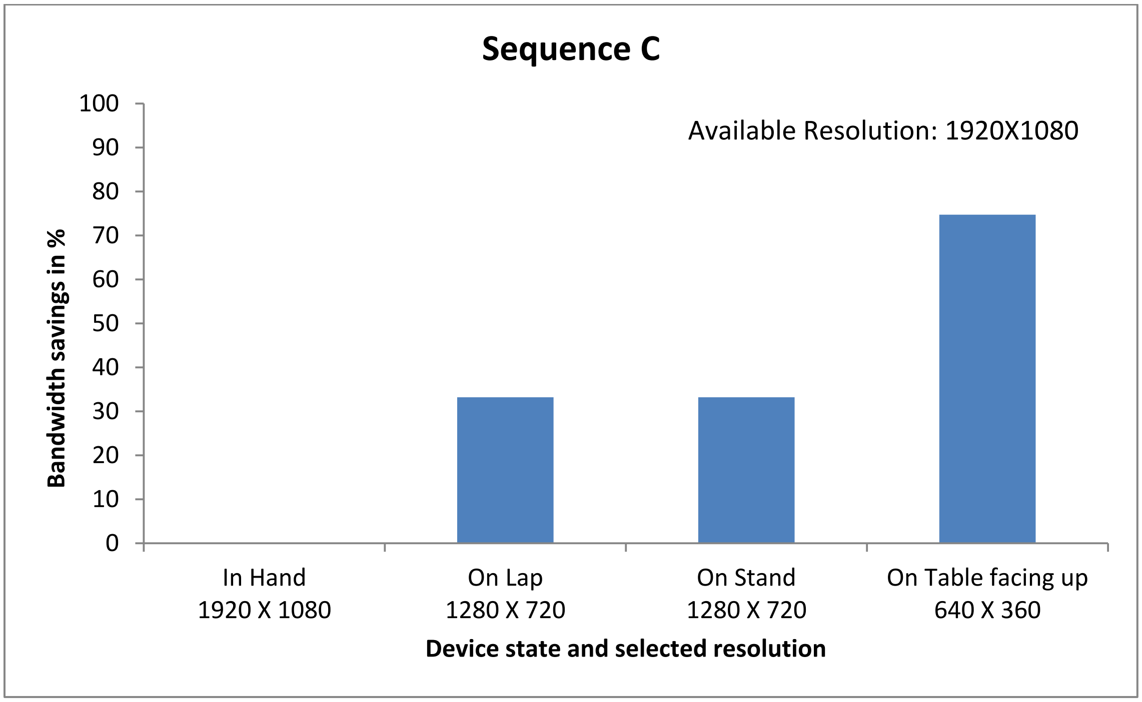


Figure 6-3 Savings for sequence C (All Bitrates available)

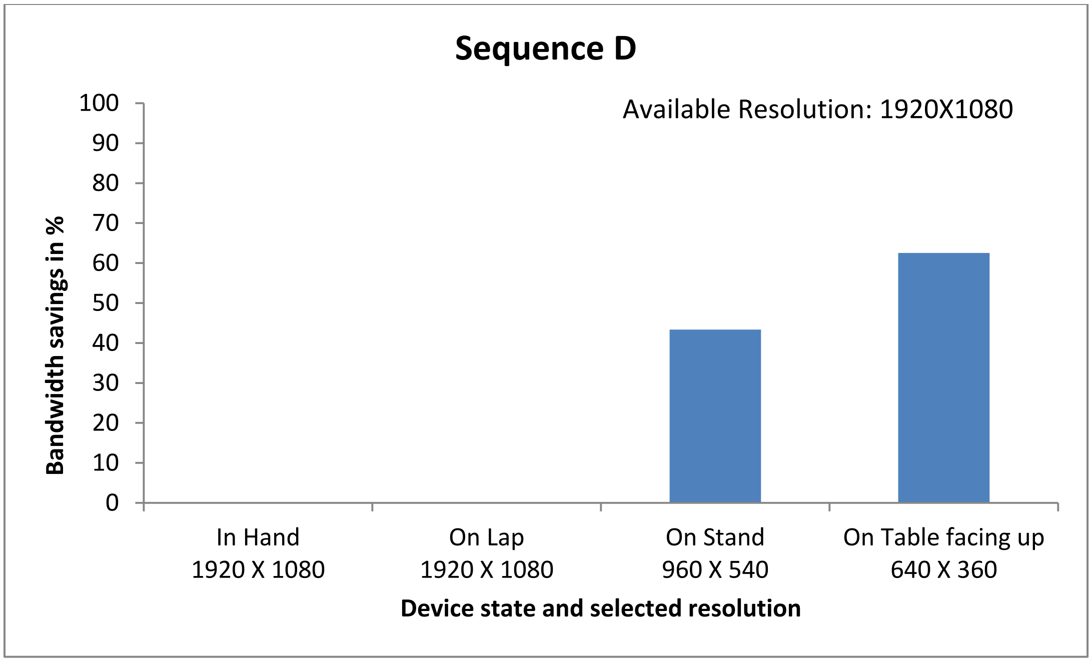


Figure 6-4 Savings for sequence D (All Bitrates available)

6.5 Bandwidth Savings when Resolution is Restricted

Figures 6-5 through 6-8 give the bandwidth savings when the available resolution is limited by the network.

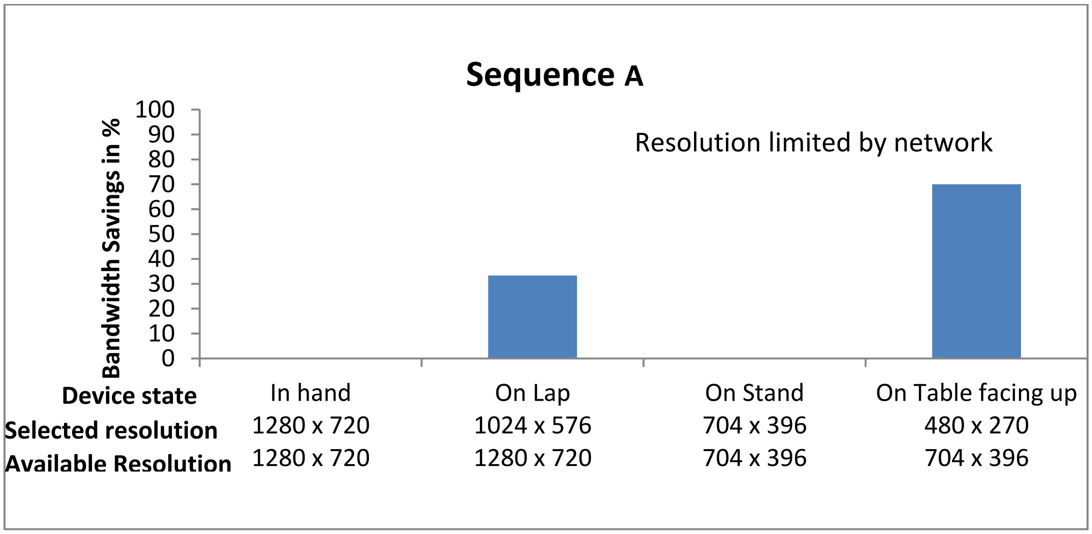


Figure 6-5 Savings for sequence A (Limited by network)

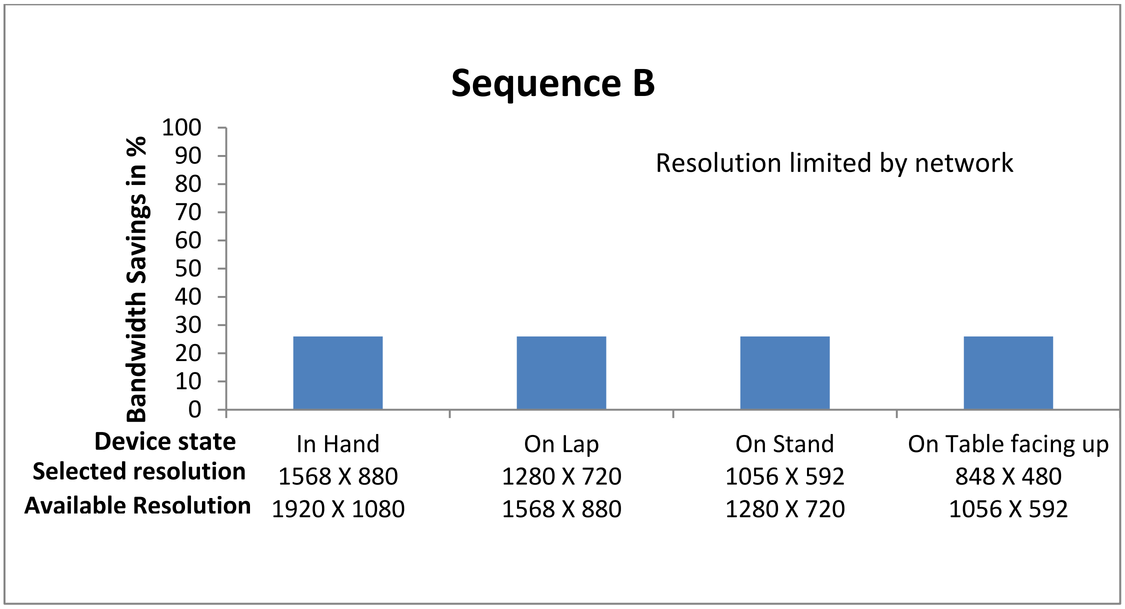


Figure 6-6 Savings for sequence B (Limited by network)

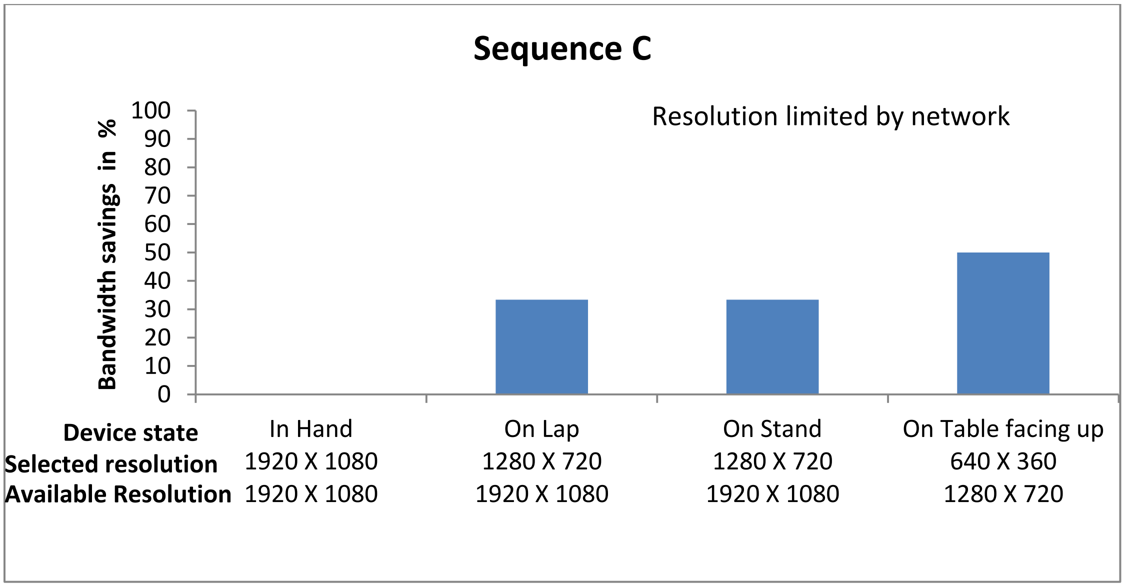


Figure 6-7 Savings for sequence C (Limited by network)

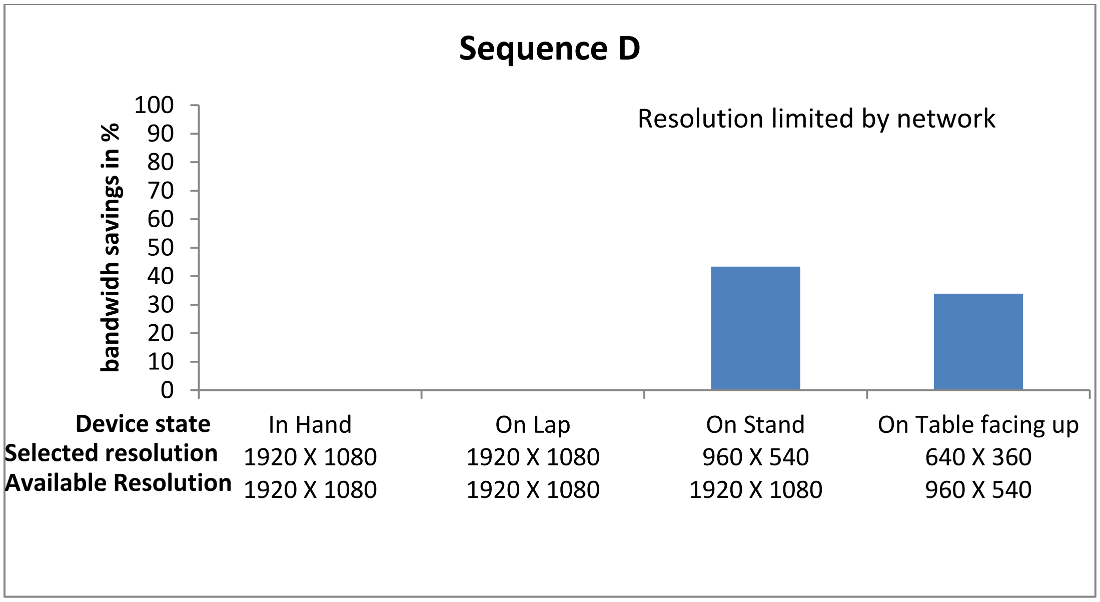


Figure 6-8 Savings for sequence D (Limited by network)

6.6 Summary:

Bandwidth savings is more evident in all the cases when the viewing distance is high (i.e The “On Table” case) when there is more bandwidth available to play the highest bitrate available. In the specific case of Sequence B, where there were more representations of bitrates available, user adaptation had more room to select the sufficient resolution. It is also noted that even if there is enough bandwidth available to play higher bitrates, selecting the lower and sufficient resolution helps in smooth playback of video with considerably less latency to buffer.

The test cases used are the examples from the DASH industry forum. They are selected in such a way that each one can offer different combinations of representations for viewing conditions. Chapter 7 relates to conclusions and possible future work.

Chapter 7

Conclusions and Future work

7.1 Conclusions

By incorporating the characteristics of viewing environment and by understanding the limits of HVS, the streaming of videos to mobile phones can be adaptive to viewing preferences of the user. This can be done through the MPEG-DASH standard which enables design of intelligent streaming systems adapting not only to bandwidth but also to factors affecting the user ability to see visual information. It is shown that such an adaptation can result in reduced bandwidth usage, increased battery life, and improved quality of user experience.

7.2 Possible Future Work

While this thesis has demonstrated the potential for bandwidth savings by the user adaptation, many opportunities for extending the scope of this thesis remain.

In MPEG-DASH, the server needs to have streams with many resolutions available for the client to select. This burden of having multiple streams of same content can be addressed using scalable video coding (SVC). It will be interesting to see how much of a burden can SVC can solve.

Human visual system has many other limits including but not limited to oblique effect, horizontal effect, contrast constancy, etc… Further research along these lines would add additional benefits in calculation of sufficient resolution.

As the sensors in the mobile device are constantly used to determine the viewing environment, they can also be used to collect data on user preferences of the content. User attentiveness while the content is played can be an area of research which will help the ad / film makers to provide appropriate content.

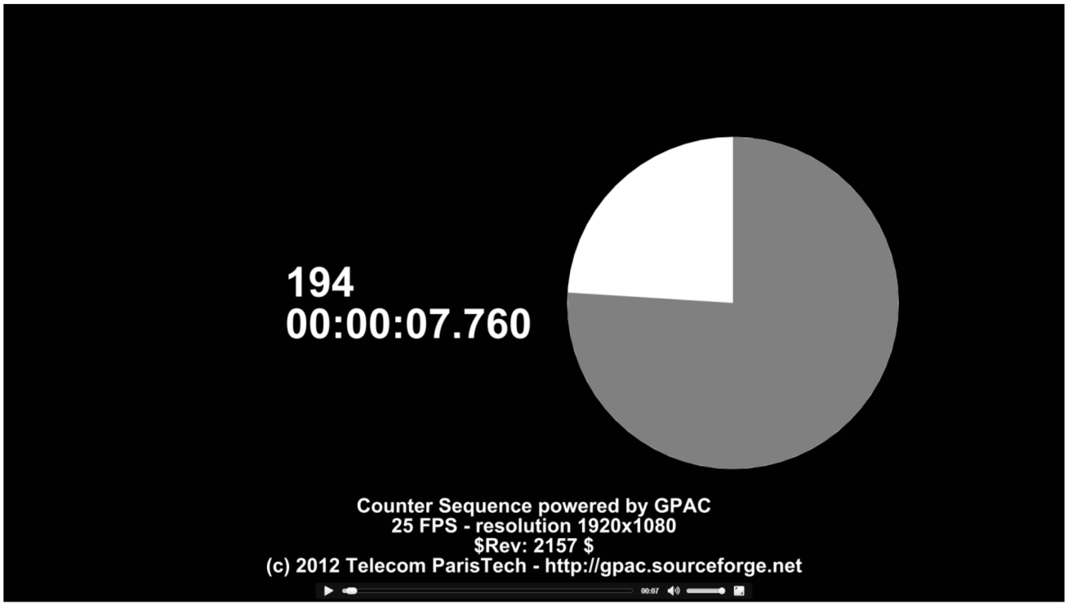
Appendix A

Test Media and MPDs [42 - 45]

Media Used.

Media 1 Frame from sequence A (1280X720)

Media 2 Frame from sequence B (1920X1080)



Media 3 Frame from sequence B (1920 X1080)

Media 4 Frame from sequence D (1920 X1080)

Corresponding MPDs

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:mpeg:DASH:schema:MPD:2011"
xsi:schemaLocation="urn:mpeg:DASH:schema:MPD:2011 DASH-MPD.xsd" type="static" mediaPresentationDuration="PT260.266S"
availabilityStartTime="2012-09-05T09:00:00Z" maxSegmentDuration="PT4.080S" minBufferTime="PT5.001S"
profiles="urn:mpeg:dash:profile:isoff-live:2011">
    <Period>
        <AdaptationSet mimeType="video/mp4" segmentAlignment="true" startWithSAP="1" maxWidth="1280" maxHeight="720" maxFrameRate="25"
        par="16:9">
            <SegmentTemplate presentationTimeOffset="0" timescale="90000" initialization="$RepresentationID$/Header.m4s"
            media="$RepresentationID$/$Number$.m4s" duration="360000" startNumber="0"/>
            <Representation id="video1" width="1280" height="720" frameRate="25" sar="1:1" scanType="progressive" bandwidth="3000000"
            codecs="avc1.4D4020"/>
            <Representation id="video2" width="1024" height="576" frameRate="25" sar="1:1" scanType="progressive" bandwidth="2000000"
            codecs="avc1.4D401F"/>
            <Representation id="video3" width="704" height="396" frameRate="25" sar="1:1" scanType="progressive" bandwidth="1000000"
            codecs="avc1.4D401E"/>
            <Representation id="video4" width="480" height="270" frameRate="25" sar="1:1" scanType="progressive" bandwidth="600000"
            codecs="avc1.4D4015"/>
            <Representation id="video5" width="320" height="180" frameRate="25" sar="1:1" scanType="progressive" bandwidth="349952"
            codecs="avc1.4D400D"/>
        </AdaptationSet>
        <AdaptationSet mimeType="audio/mp4" lang="en" segmentAlignment="true" startWithSAP="1">
            <SegmentTemplate presentationTimeOffset="0" timescale="48000" initialization="$RepresentationID$/Header.m4s"
            media="$RepresentationID$/$Number$.m4s" duration="192000" startNumber="0"/>
            <Representation id="audio" audioSamplingRate="48000" bandwidth="56000" codecs="mp4a.40.2">
                <AudioChannelConfiguration schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="2"/>
            </Representation>
        </AdaptationSet>
    </Period>
</MPD>
```

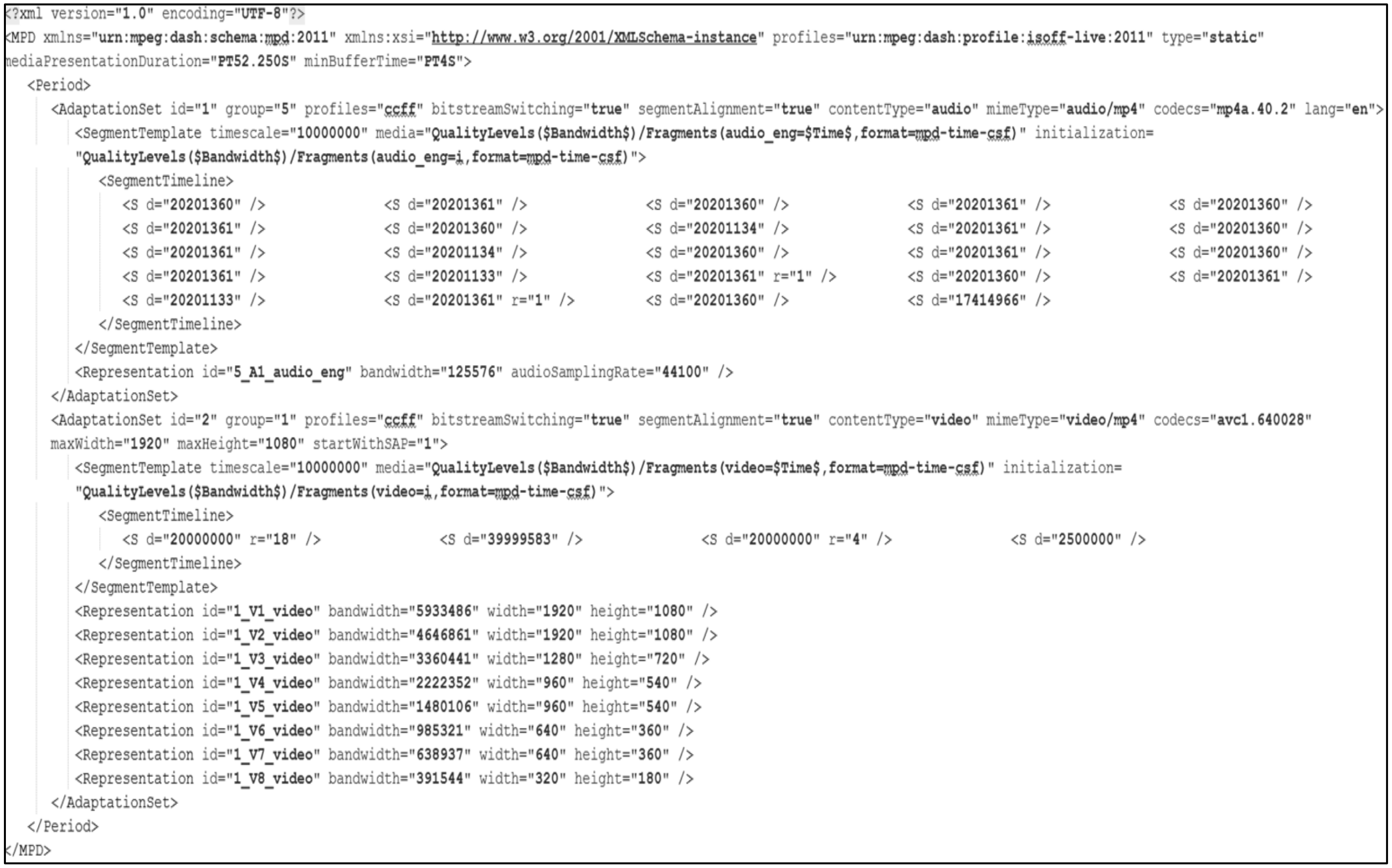
MPD 1 Representations from sequence A

```
<MPD  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="urn:mpeg:dash:schema:mpd:2011"
xsi:schemaLocation="urn:mpeg:DASH:schema:MPD:2011"  mediaPresentationDuration="PT126.439S"  minBufferTime="PT2.00S"    type="static"
profiles="urn:mpeg:dash:profile:isoff-main:2011">
   <Period start="PT0.00S" duration="PT126.439S">
     <AdaptationSet mimeType="audio/mp4" codecs="mp4a.0x40" lang="en" subsegmentAlignment="true">
       <SegmentTemplate media="$RepresentationID$/seg_$Number$.m4s" initialization="$RepresentationID$/seg_init.mp4" duration="2"
       startNumber="0"/>
       <Representation id="audio-64Kbps" bandwidth="64000"/>
     </AdaptationSet>
     <AdaptationSet bitstreamSwitching="true" mimeType="video/mp4" codecs="avc1.64001f" subsegmentAlignment="true">
       <SegmentTemplate media="$RepresentationID$/seg_$Number$.m4s" initialization="$RepresentationID$/seg_init.mp4" duration="2"
       startNumber="0"/>
       <Representation id="video-6000Kbps" bandwidth="6000000" width="1920" height="1080">
       </Representation>
       <Representation id="video-4441Kbps" bandwidth="4441000" width="1568" height="880">
       </Representation>
       <Representation id="video-3287Kbps" bandwidth="3287000" width="1280" height="720">
       </Representation>
       <Representation id="video-2433Kbps" bandwidth="2433000" width="1056" height="592">
       </Representation>
       <Representation id="video-1801Kbps" bandwidth="1801000" width="848" height="480">
       </Representation>
       <Representation id="video-1333Kbps" bandwidth="1333000" width="688" height="384">
       </Representation>
       <Representation id="video-986Kbps" bandwidth="986000" width="576" height="320">
       </Representation>
       <Representation id="video-730Kbps" bandwidth="730000" width="448" height="256">
       </Representation>
       <Representation id="video-540Kbps" bandwidth="540000" width="368" height="208">
       </Representation>
       <Representation id="video-400Kbps" bandwidth="400000" width="320" height="176">
       </Representation>
     </AdaptationSet>
   </Period>
 </MPD>
```

MPD 2 Representations from sequence B

```xml
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" type="static" minBufferTime="PT1.5S" mediaPresentationDuration="PT0H10M0.00S"
profiles="urn:mpeg:dash:profile:isoff-live:2011">
<ProgramInformation moreInformationURL="http://gpac.sourceforge.net">
<Title>mp4-live-mpd-AV-BS.mpd generated by GPAC</Title>
<Copyright>TelecomParisTech(c)2012</Copyright>
</ProgramInformation>
<Period start="PT0S" duration="PT0H10M0.00S">
<AdaptationSet segmentAlignment="true" bitstreamSwitching="true" maxWidth="1920" maxHeight="1080" maxFrameRate="25" par="16:9">
<ContentComponent id="1" contentType="video"/>
<SegmentTemplate timescale="1000" duration="10000" media="mp4-live-$RepresentationID$-$Number$.m4s" startNumber="1"
initialization="mp4-live-mpd-AV-BS_set1_init.mp4"/>
<Representation id="h264bl_low" mimeType="video/mp4" codecs="avc1.42c00d" width="320" height="180" frameRate="25" sar="1:1" startWithSAP="1"
bandwidth="50842"></Representation>
<Representation id="h264bl_mid" mimeType="video/mp4" codecs="avc1.42c01e" width="640" height="360" frameRate="25" sar="1:1" startWithSAP="1"
bandwidth="194834"></Representation>
<Representation id="h264bl_hd" mimeType="video/mp4" codecs="avc1.42c01f" width="1280" height="720" frameRate="25" sar="1:1" startWithSAP="1"
bandwidth="514793"></Representation>
<Representation id="h264bl_full" mimeType="video/mp4" codecs="avc1.42c028" width="1920" height="1080" frameRate="25" sar="1:1" startWithSAP="1"
bandwidth="770663"></Representation>
</AdaptationSet>
<AdaptationSet segmentAlignment="true" bitstreamSwitching="true" lang="und">
<AudioChannelConfiguration schemeIdUri="urn:mpeg:dash:23003:3:audio_channel_configuration:2011" value="1"/>
<ContentComponent id="1" contentType="audio"/>
<SegmentTemplate timescale="1000" duration="9520" media="mp4-live-$RepresentationID$-$Number$.m4s" startNumber="1"
initialization="mp4-live-mpd-AV-BS_set2_init.mp4"/>
<Representation id="aaclc_low" mimeType="audio/mp4" codecs="mp4a.40.2" audioSamplingRate="44100" startWithSAP="1" bandwidth="19042"></Representation>
<Representation id="aaclc_high" mimeType="audio/mp4" codecs="mp4a.40.2" audioSamplingRate="44100" startWithSAP="1" bandwidth="66341"></Representation>
</AdaptationSet>
</Period>
</MPD>
```

MPD 3 Representation from sequence C



```xml
<?xml version="1.0" encoding="UTF-8"?>
<MPD xmlns="urn:mpeg:dash:schema:mpd:2011" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" profiles="urn:mpeg:dash:profile:isoff-live:2011" type="static"
mediaPresentationDuration="PT52.250S" minBufferTime="PT4S">
  <Period>
    <AdaptationSet id="1" group="5" profiles="ccff" bitstreamSwitching="true" segmentAlignment="true" contentType="audio" mimeType="audio/mp4" codecs="mp4a.40.2" lang="en">
      <SegmentTemplate timescale="10000000" media="QualityLevels($Bandwidth$)/Fragments(audio_eng=$Time$,format=mpd-time-csf)" initialization=
      "QualityLevels($Bandwidth$)/Fragments(audio_eng=i,format=mpd-time-csf)">
        <SegmentTimeline>
          <S d="20201360" />           <S d="20201361" />           <S d="20201360" />           <S d="20201361" />           <S d="20201360" />
          <S d="20201361" />           <S d="20201360" />           <S d="20201134" />           <S d="20201361" />           <S d="20201360" />
          <S d="20201361" />           <S d="20201134" />           <S d="20201360" />           <S d="20201361" />           <S d="20201360" />
          <S d="20201361" />           <S d="20201133" />           <S d="20201361" r="1" />     <S d="20201360" />           <S d="20201361" />
          <S d="20201133" />           <S d="20201361" r="1" />     <S d="20201360" />           <S d="17414966" />
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="5_A1_audio_eng" bandwidth="125576" audioSamplingRate="44100" />
    </AdaptationSet>
    <AdaptationSet id="2" group="1" profiles="ccff" bitstreamSwitching="true" segmentAlignment="true" contentType="video" mimeType="video/mp4" codecs="avc1.640028"
    maxWidth="1920" maxHeight="1080" startWithSAP="1">
      <SegmentTemplate timescale="10000000" media="QualityLevels($Bandwidth$)/Fragments(video=$Time$,format=mpd-time-csf)" initialization=
      "QualityLevels($Bandwidth$)/Fragments(video=i,format=mpd-time-csf)">
        <SegmentTimeline>
          <S d="20000000" r="18" />          <S d="39999583" />          <S d="20000000" r="4" />          <S d="2500000" />
        </SegmentTimeline>
      </SegmentTemplate>
      <Representation id="1_V1_video" bandwidth="5933486" width="1920" height="1080" />
      <Representation id="1_V2_video" bandwidth="4646861" width="1920" height="1080" />
      <Representation id="1_V3_video" bandwidth="3360441" width="1280" height="720" />
      <Representation id="1_V4_video" bandwidth="2222352" width="960" height="540" />
      <Representation id="1_V5_video" bandwidth="1480106" width="960" height="540" />
      <Representation id="1_V6_video" bandwidth="985321" width="640" height="360" />
      <Representation id="1_V7_video" bandwidth="638937" width="640" height="360" />
      <Representation id="1_V8_video" bandwidth="391544" width="320" height="180" />
    </AdaptationSet>
  </Period>
</MPD>
```

MPD 4 Representations from sequence D

Acronyms

1. 3GPP: Third Generation Partnership Project
2. ABR: Adaptive Bit Rate
3. API: Application Interface
4. CDN: Content Delivery Network
5. CIE: Commission Internationale de l'Eclairage
6. CR: Contrast Ratio
7. CSF: Contrast Sensitivity Function
8. DASH-IF: DASH Industry Forum
9. DRM: Digital Rights Management
10. GPS: Global Positioning System
11. HDTV: High Definition Television
12. HTTP: Hypertext Transfer Protocol
13. HVS: Human Visual System
14. IP: Internet Protocol
15. ISO : International organization for standardization
16. JS: JavaScript
17. MPD: Media Presentation Description
18. MPEG: Moving Picture Experts Group
19. MPEG-DASH: Dynamic Adaptive Streaming over HTTP
20. MVC: Multiview Video Coding
21. NAT: Network Address Translation
22. RPA: Resting Point of Accommodation
23. RPV: Resting Point of Vergence.
24. RTP: Real-time Transport Protocol
25. RTSP: Real Time Streaming Protocol
26. SVC: Scalable Video Coding
27. TCP: Transmission Control protocol
28. UAV: User Adaptive Video
29. URL: Uniform Resource Locator

References

[1] D. Wu, Y. T. Hou, W. Zhu, Y.-Q. Zhang and J. M. Peha, "Streaming video over the Internet: approaches and directions," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 11, no. 3, pp. 282-300, Mar. 2001.

[2] G. J. Conklin, G. S. Greenbaum, K. O. Lillevold, A. F. Lippman and Y. A. Reznik, "Video coding for streaming media delivery on the Internet," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 11, no. 3, pp. 269-281, Mar. 2001.

[3] I. Sodagar, "The MPEG-DASH Standard for Multimedia Streaming Over the internet," *IEEE MultiMedia,* vol. 18, no. 4, pp. 62-67, Apr. 2011.

[4] ISO/IEC 23009-1, "Dynamic adaptive streaming over HTTP (DASH) - Part 1: Media presentation description and segment formats," *Information Technology,* 5 Jan. 2012.

[5] A. Talukdar, M. Cudak and A. Ghosh, "Streaming Video Capacities of LTE Air-Interface," *IEEE International Conference on Communications (ICC),* pp. 1-5, May 2010.

[6] Y. A. Reznik, "User-adaptive mobile video streaming using MPEG-DASH," *Proc. SPIE, Applications of Digital Image Processing XXXVI,* vol. 8856, Sept. 2013.

[7] J. Bergquist, "Resolution and contrast requirements on mobile displays for different applications in varying luminous environments," *2nd Int Symp Nanovision Sceince,* pp. 143-145, 2005.

[8] J. Xue and C. W. Chen, "A study on perception of mobile video with surrounding contextual influences," *Fourth International Workshop on Quality of Multimedia Experience,* vol. 5, no. 7, pp. 248-253, Jul. 2012.

[9] Wikipedia, "Snellen's Chart," [Online]. Available: http://en.wikipedia.org/wiki/Snellen_chart.

[10] H. Snellen, "Probebuchstaben zur Bestimmung der Sehschärfe (Sample letters to determine the visual acuity)", Berlin: Hermann Peters, 1873.

[11] K. Michael and L. Charles, "Visual Acuity," Web Vision, 5 June 2007. [Online]. Available: http://webvision.med.utah.edu/book/part-viii-gabac-receptors/visual-acuity/. [Accessed 29 June 2014].

[12] D. A. Owens and K. Wolf-Kelly, "Near work, visual fatigue, and variations of oculomotor tonus," *Investigative ophthalmology & visual science,* vol. 28, no. 4, pp. 743-749, 1987.

[13] J.Anshel, "Visual ergonomics in the workplace", CRC Press, 2002.

[14] Y. Bababekova, M. Rosenfield, J. E. Hue and R. R. Huang, "Font Size and Viewing Distance of Handheld Smart Phones," *Optometry and Vision Science,* vol. 88, no. 7, pp. 795-797, Jul. 2011.

[15] G. E. Legge, G. S. Rubin and A. Luebker, "Psychophysics of reading. V. The role of contrast in normal vision," *Vision Research,* vol. 27, no. 7, pp. 1165-1177, 1987.

[16] Recommendation ITU-R BT.500-13, "Methodology for the subjective assessment of the quality of television pictures," *International Telecommnication Union,* 2012.

[17] R. Vanam and Y. Reznik, "Improving the Efficiency of Video Coding by using Perceptual Preprocessing Filter," *IEEE Data Compression Conference,* 2013.

[18] J. A. Movshon and L. Kiorpes, "Analysis of the development of spatial contrast sensitivity in monkey and human infants.," *Journal of the Optical Society of America A,* vol. 5, no. 12, pp. 2166-2172, 1988.

[19] P. G. J. Barten, "Evaluation of subjective image quality with the square-root integral method," *Journal of the Optical Society of America A,* vol. 7, no. 10, pp. 2024-2031, Oct. 1990.

[20] P. G. J. Barten, "Formula for the contrast sensitivity of the human eye," *Proc. SPIE , Image Quality and System Performance,* vol. 5294, pp. 231-238, 2003.

[21] S. J. Daly, "Visible differences predictor: an algorithm for the assessment of image fidelity," *Proc of SPIE in Human Vision, Visual Processing, and Digital Display III,* vol. 1666, pp. 2-15, 1992.

[22] Y. A. Reznik and R. Vanam, "Improving coding and delivery of video by exploiting the oblique effect," *IEEE Global Conference on Signal and Information Processing,* Dec. 2013.

[23] L. J. Kerofsky, R. Vanam and Y. A. Reznik, "Improved Adaptive Video Delivery System Using a Preceptual Pre-processing Filter," *IEEE Global Conference on Signal and Information Processing,* 2014.

[24] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey and D. E. Knuth, "On the LambertW function," *Advances in Computational mathematics,* vol. 5, no. 1, pp. 329-359, 1996.

[25] A. Saamer, A. C. Begen and C. Dovrolis, "An Experimental Evaluation of Rate-Adaptation Algorithms in Adaptive Streaming over HTTP," *Proceedings of the second annual ACM conference on Multimedia systems,* pp. 157-168, 2011.

[26] "Overview of MPEG-DASH," DASH Industry Foundation, [Online]. Available: http://dashif.org/mpeg-dash/.

[27] T. Stockhammer and S. Iraj, "MPEG DASH: The Enabler Standard for Video Delivery

Over the Internet," *SMPTE Motion Imaging Journal,* vol. 121, no. 5, pp. 40-46, 2012.

[28] DASH-IF, "DASH Industry Forum," [Online]. Available: http://dashif.org/.

[29] DASH-IF, "Reference Client 1.0.0," DASH Industry Forum, [Online]. Available: http://dashif.org/reference/players/javascript/1.0.0/index.html.

[30] B. Rainer, S. Lederer, C. Muller and C. Timmerer, "A seamless Web integration of adaptive HTTP streaming," *IEEE Proceedings of the 20th European Signal Processing Conference (EUSIPCO),* pp. 1519-1523, Aug. 2012.

[31] M. Katsikitis, “The Human Face: Measurement and Meaning”, Springler, 2003.

[32] J. Young, M. Trudeau, D. Odell, K. Marinelli and J. Dennerlein, "Touch-screen tablet user configurations and case-supported tilt affect head and neck flexion angles," *Work: A Journal of Prevention, Assessment and Rehabilitation,* vol. 41, no. 1, pp. 81-91, 2012.

[33] Microsoft, "Surface pro," [Online]. Available: http://www.microsoft.com/surface/en-us/products/surface-pro-2.

[34] T. Montgomery, "Anatomy, Physiology & Pathology of the Human Eye," [Online]. Available: http://www.tedmontgomery.com/the_eye/acuity.html.

[35] J. Chipchase, C. Yanqing and Y. Jung, "Personal Television: A Qualitative Study of Mobile TV Users", Springer, 2007.

[36] K. R. Rao and H. Wu, "Digital Video Image Quality and Perceptual Coding", CRC Press, 2005.

[37] S. Dernbach, B. Das, N. C. Krishnan, B. L. Thomas and D. J. Cook, "Simple and Complex Activity Recognition through Smart Phones," *8th International Conference on Intelligent Environments (IE),* vol. 26, no. 29, pp. 214-221, 2012.

[38] D. Lamming, "Contrast Sensitivity," in *Vision and visual dysfunction*, London, Macmillan Press, 1991.

[39] V. Cerf and R. E. Kahn, "A Protocol for Packet Network Intercommunication," *IEEE Transactions on Communications,* vol. 22, no. 5, pp. 637-648, May 1974.

[40] Wikipedia, "Adaptive bitrate streaming," [Online]. Available: http://en.wikipedia.org/wiki/Adaptive_bitrate_streaming.

[41] K. R. Boff and L. E. Janet, "Engineering data compendium. Human perception and performance. User's guide," 1988.

[42] J. Mannos and D. J. Sakrison, "The effects of a visual fidelity criterion of the encoding of images," *IEEE Transactions on Information Theory,* vol. 20, no. 4, pp. 525-536, Jul. 1974.

Test media:

[43] ***A** : Envivo* : http://dash.edgesuite.net/envivio/dashpr/clear/Manifest.mpd

[44] ***B**: TimeScapes* : http://www.youtube.com/watch?v=EgKXcQ9PLuc

[45] ***C**: Counter* : http://www.digitalprimates.net/dash/streams/gpac/mp4-main-multi-mpd-AV-NBS.mpd

[46] ***D**: Sintel Trailer:* http://wams.edgesuite.net/media/SintelTrailer_MP4_from_WAME/sintel_trailer-1080p.ism/manifest(format=mpd-time-csf)

Biographical Information

Abhijith Jagannath was born in Bangalore, India in 1987.He received the Bachelor’s degree in Electronics and Communication Engineering from Visveswaraya technological University, India in 2008. He worked as Software Engineer at Adobe Systems PVT LTD, India.

He decided to pursue the Master’s degree from The University of Texas at Arlington in Fall 2012. He worked as a Graduate Research Assistant under Dr. Rao in the Multimedia Processing Lab from Fall 2012 to Summer 2013. He got an opportunity to work as an Intern at InterDigital Communications in San Diego, California from Summer 2013 till Spring 2014. He continues to work at InterDigital after graduation.