QUALITY OF EXPERIENCE ESTIMATION FOR ADAPTIVE HTTP/TCP VIDEO STREAMING USING H.264/AVC

Kamal Deep Singh, Yassine Hadjadj-Aoul and Gerardo Rubino

INRIA, Campus Universitaire de Beaulieu, 35042 Rennes Cedex, France
Email: {kamal.singh,yassine.hadjadj-aoul, gerardo.rubino}@inria.fr

ABSTRACT

Video services are being adopted widely in both mobile and fixed networks. For their successful deployment, the content providers are increasingly becoming interested in evaluating the performance of such traffic from the final users' perspective, that is, their Quality of Experience (QoE). For this purpose, subjective quality assessment methods are costly and can not be used in real time. Therefore, automatic estimation of QoE is highly desired. In this paper, we propose a no-reference QoE monitoring module for adaptive HTTP streaming using TCP and the H.264 video codec. HTTP streaming using TCP is the popular choice of many web based and IPTV applications due to the intrinsic advantages of the protocol. Moreover, these applications do not suffer from video data loss due to the reliable nature of the transport layer. However, there can be playout interruptions and if adaptive bitrate video streaming is used then the quality of video can vary due to lossy compression. Our QoE estimation module, based on Random Neural Networks, models the impact of both factors. The results presented in this paper show that our model accurately captures the relation between them and QoE.

Index Terms— QoE, Video Quality, adaptive HTTP streaming, TCP video streaming, H.264, IPTV

1. INTRODUCTION

Distribution of digital video is undergoing dramatic transformation at present. Many developments in this area are changing the way contents are generated, distributed and consumed. In fact, reductions in the cost of digital video cameras coupled with the development of easy-to-use content sharing platforms (e.g. the YouTube phenomenon) have clearly stimulated media delivery over HTTP. The current success of streaming over HTTP is also due to the intrinsic advantages of the protocol, which has the ability, notably, to bypass firewalls while exploiting the standard HTTP entities in the media delivery.

With these considerations, HTTP-based streaming is experiencing a widespread adoption by Services providers and CDN operators for both mobile and fixed networks. In this context, content providers are increasingly becoming interested in evaluating the performance of such applications from the final users' perspective. Indeed, more importance is being attached to the quality as perceived by the final users, or Quality of Experience (QoE), as compared to Quality of Service. In fact, with HTTP streaming, it is possible to have a bad QoS, but a good QoE, since HTTP provides a reliable mean of transportation. This can happen for example when the QoS parameters are packet loss rate and packet delay. In that case, even if there will be some packet loss and delay reflecting bad QoS, the HTTP streaming, with TCP retransmissions, will still be able to provide good QoE, at least for up to some values of these parameters.

QoE is defined in [1] as "the overall acceptability of an application or service, as perceived subjectively by the end-user". Moreover, it is noted that QoE includes everything such as network, client, terminal, etc., and overall acceptability may be influenced by the context and the expectations of the user. As will be stated in next section, several methods to measure the QoE are available. However, none of the proposed methods address the problem of measuring QoE in the case of adaptive bitrate video using a reliable transport protocol, the situation in adaptive streaming over HTTP.

In light of the above observations, the main concern of the present work is to design a no-reference QoE monitoring module for HTTP/TCP video streaming using H.264/AVC video codec in the context of IPTV. The proposed approach uses a methodology called Pseudo-Subjective Quality Assessment (PSQA) [10,13], which is based on Random Neural Network (RNN) [9], to estimate the QoE in the above context. In this work, instead of packet loss patterns and latencies, we consider the Quantization Parameter (QP) used in video compression and the playout interruptions as metrics that directly impact QoE. Indeed, when using adaptive HTTP streaming the perceived quality depends directly on QP, which reflects changes in quality, and on the playout interruptions.

The remainder of this paper is organized as follows. Section 2 gives an overview of the background and discusses the state of the art. Section 3 focuses on monitoring QoE. Section 4 presents and discusses performance evaluation results. Finally, the paper concludes in Section 5 with a summary re-

The work described in this paper is partially supported by the national French project ANR VERSO ViPeer.

capping the main achievements of the proposed scheme.

2. RELATED WORK AND BACKGROUND

2.1. Background

Several factors can influence QoE for video applications depending on their type and the underlying network. For example, packet losses in the case of applications using an unreliable transport protocol and packet delay, in the case of real time applications, can significantly impact QoE. Moreover, the video content itself may have some impact

In the context of HTTP streaming, considered in this paper, a reliable transport protocol such as TCP is assumed and thus video data will not be lost. However, there may be playout interruptions caused by either bandwidth fluctuations or long delays due to retransmissions after some packet losses. Such playout interruptions are annoying for the users and should be taken into account for QoE estimation.

Quantization is another important factor that is relevant especially in the case of *adaptive* HTTP streaming. In fact, to adapt to a diminution in the available bandwidth, the adaptive HTTP streaming client will ask for low bitrate video chunks with more compression that may degrade QoE.

2.2. Related Work

Objective quality measurement tools, such as PSNR, may not correlate well with human perceived video quality and may not be able to assess the impact of video playout interruptions. On the other hand the subjective quality assessment methods [2] are costly and time consuming. Thus, automatic QoE monitoring is highly desired. Some existing approaches [4–7] take the full original video as a reference (can be very costly), or at least as a partial reference for QoE estimation.

About no-reference models: an "opinion model" is described in ITU G.1070 [11], mainly for planning purposes and is not very accurate [3]. The work in [3] uses a methodology called Pseudo-Subjective Quality Assessment (PSQA) [10], which is based on Random Neural Network (RNN), a family of Artificial Neural Networks based on queueing models [9], to estimate the QoE of H.264 video streaming over DVB-H networks experiencing packet losses. However, the above module is not valid for HTTP streaming where lost packets are retransmitted and playout interruptions may occur.

A QoE model considering video playout interruptions is proposed in [8]. However, that approach does not take into account the impact of video bitrate change on the perceived quality, which is the case in adaptive video streaming.

In this paper, we use new subjective tests and studies to propose a model considering video playout interruptions as well as an encoding parameter called Quantization Parameter (QP) that impacts the bitrate as well as the video quality. We consider the number of playout interruptions, the average interruptions delay as well as the maximal interruptions delay, which are direct consequences of network conditions. The above choices are explained in the following section.

3. QOE ESTIMATION

3.1. Pseudo-Subjective Quality Assessment (PSQA)

A general technology called Pseudo-Subjective Quality Assessment (PSQA) has been proposed in [10]. PSQA is based on a specific type of queuing network used as a learning tool called Random Neural Network [9]. For every different context, such as when the video codec or the parameters affecting the QoE change, a new PSQA based module is to be designed after analysing the associated parameters and after conducting new subjective tests. The idea is to have several distorted samples evaluated subjectively by a panel of human observers. Then the results of this evaluation are used to train a RNN in order to capture the relation between the parameters causing distortion and the perceived quality.

3.2. Adaptive HTTP Streaming

The keen interest towards multimedia streaming over the Internet, which was clearly encouraged by the development of easy-to-use content sharing platforms (e.g. YouTube), is making HTTP/TCP streaming the leading technology in the media delivery sectors [14] for both mobile and fixed networks. As opposed to the former protocols, HTTP enables a reliable and an adaptive streaming process. These properties are directly inherited from those of the TCP protocol. It also allows to seamlessly bypass firewalls and adapt the streaming quality to the bandwidth, which makes the technology particularly interesting for a wide deployment. For adaptive bitrate streaming, the media file is fragmented into small segments or chunks of same duration (e.g. a few seconds) [15].

In order to allow adaptive streaming, each chunk is decoded independently enabling seamless switching from one quality to another when network conditions change. Thus, once the playout of a chunk is finished, the video player can start playing the next chunk with a different quality.

3.3. QoE estimation for Adaptive HTTP Streaming

In order to use PSQA for Adaptive HTTP Streaming, first the relevant parameters need to be identified and their effect on the perceived quality needs to be studied. Then the selected parameters have to be simulated, which will result in distorted video sequences. These distorted video sequences will be used to train the PSQA tool with the help of a panel of human observers. The trained PSQA will, then, be used in real time to estimate the video quality. In the following text, we describe the parameters that are considered for QoE estimation in the context of HTTP streaming over TCP/IP networks.

It should be noted that other parameters, not described below, like resolution and frame rate, are either constant in

our study or, like losses, delay and jitter that cause packets to miss their deadline, are converted into playout interruptions.

3.3.1. Playout Interruptions

TCP/IP networks are characterised by frequent packet losses and fluctuating bandwidth over time. In contrast to the streaming over UDP, HTTP streaming implies an automatic recovery of lost packets by TCP, that may eliminate video distortions. However, the retransmissions and bandwidth fluctuations may cause playout interruptions which should be considered for QoE estimation. In this paper, by playout interruptions we mean pauses in the playout without any skip of video data. This case is more relevant for Video On Demand applications where video can be paused when some immediate chunks are not yet downloaded.

We model the playout interruptions using three parameters measured over an interval containing a fixed duration of video data (16 seconds in this paper, but interval can be longer due to interruptions): the total number of playout interruptions N, the average D_{avg} and the maximal D_{max} values of interruption delays. Note that, for longer videos, a different QoE score will be provided after every 16 seconds of video.

It should be noted that in order to train our QoE module, different values of N, D_{avg} and D_{max} must be generated *uniformly* to avoid missing regions in the input parameter space of the training database. However, the existing relation between D_{avg} and D_{max} makes it difficult to uniformly generate such values. Thus, we decided to model the playout interruptions as a function of N, D_{avg} and D_{max} and to use the following cumulative distribution function $F(x)$ of the delay:

$$F(x) = \begin{cases} \dfrac{\alpha x}{D_{avg}} & \text{if } x \leq D_{avg}, \\ (1-\alpha)\dfrac{x - D_{avg}}{D_{max} - D_{avg}} + \alpha & \text{if } x \in [D_{avg},\, D_{max}], \\ 1 & \text{otherwise.} \end{cases} \tag{1}$$

As the desired average value of the distribution function is D_{avg}, we must have $D_{avg} = \int_0^{+\infty} \big[1 - F(x)\big]\, \mathrm{d}x$ and combining it with (1), we get $\alpha = 1 - \dfrac{D_{avg}}{D_{max}}$. In order to generate the D_{avg} values, we need to invert function F.

This gives $F^{-1}(u) =$

$$\begin{cases} \dfrac{D_{avg} D_{max}}{D_{max} - D_{avg}} & \text{if } 0 \leq u \leq \alpha, \\ \dfrac{(u-1)D_{max}^2 - (u-2)D_{max}D_{avg}}{D_{avg}} & \text{if } \alpha \leq u \leq 1. \end{cases} \tag{2}$$

To sample different independent values of the delay, we simply generate i.i.d. pseudo-random numbers uniformly distributed on $[0,\, 1]$, say $U_1, U_2, \ldots$ and return the values $F^{-1}(U_1), F^{-1}(U_2), \ldots$. They are then used to simulate playout interruptions in the videos by inserting pauses with durations equal to these values.

3.3.2. Quantization Parameter (QP)

In the HTTP streaming context, another parameter significantly impacting QoE is the quantization parameter that controlles the amount of video compression. H.264 codec uses QP (from 0 to 51) to quantize the transform coefficients obtained while encoding the video. The trade-off is that a higher value means more loss of information and lower quality, but a lower bitrate; and vice versa. In H.264, QP can vary in different frames as well as in different macro blocks (MB) in each frame. QP is adapted over time to attain the target video bitrate for the encoded chunks. For QoE estimation, we consider the average of QP values, QP_{avg}, over all MBs in all video frames present over the measurement time window. Also note that from now onwards we use QP and QP_{avg} interchangeably in the paper, unless otherwise specified.

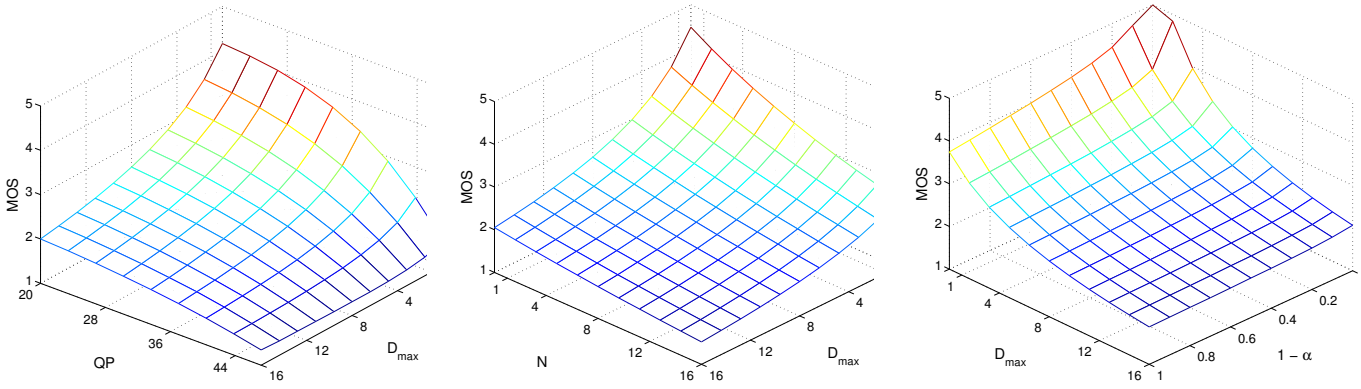
In the case of adaptive HTTP streaming, we want to cover the context of several chunks having different bitrates. Thus, we encoded video chunks with different bitrates that in turn resulted in different values of QP.

4. RESULTS

In order to generate the QoE Estimation module, 4 different video sequences of 16 seconds each were considered: Aspen, ControlledBurn, RushFieldsCuts and RedKayak from VQEG. The resolution was 720p, fps = 30, GOP size = 60 frames. The high profile of H.264 was used. The encoder used was x264 [12]. The value of QP was varied from 22 to 48.

Some videos having similar quality were discarded using random sampling for limiting the final size of the video database to avoid too long subjective testing sessions. 15 users evaluated the videos using a MOS scale of 1, very bad, to 5, excellent, using Single Stimulus Impairment Scale (SS) testing methodology [2]. Similar to the screening procedure in [10], the subjective test scores of the users, who did not give consistent scores, were removed (1 in our case). The MOS scores were used to train the RNN. Out of total 118 videos, 100 videos were used for training and 18 videos for validation. The trained RNN is validated only if the root mean square error (RMSE) for both training and validation data is less than the target. The target is equal to the average RMSE of users in the subjective panel that in turn is 0.61.

The trained RNN had 3 layers. The first layer has 4 neurons, for the input parameters, the hidden layer has 5 neurons (this number was varied and 5 provided the best RMSE) and the last layer has 1 neuron for output. To understand RNN, some very compressed details (please see [10] for more info.): the output of the network is a fraction where the numerator is the sum in h of the "state" ϱ_h of hidden neuron h, weighted by $W^+(h, o)$, the positive weight from h to the single output neuron o; the denominator is the sum of the rate of neuron o and the sum of the ϱ_h weighted by the $W^-(h, o)$. State ϱ_h is in turn equal to a similar fraction using layers 0 and 1.



(a) $1-\alpha = 0.9$ and $N = 2$ (b) $QP = 23$, $1-\alpha = 0.9$ (c) $QP = 23$, $N = 4$

Fig. 1. MOS scores with respect to different parameters.

Table 1 provides the weights of the trained RNN. The 3 layers are indexed using l and the neurons in each layer using k. The weights connecting the neurons in different layers are denoted W^+ and W^-. The 4 input parameters explained in section 3.3 were normalised as follows: $1-(QP-20)/(54-20)$, $1-\alpha$, $D_{max}/16$ and $N/16$.

The results are shown in Figure 1 which shows the estimated QoE with respect to different pairs of parameters, the remaining ones being fixed. Figure 1(a) shows that users are more sensitive to video playout interruptions as compared to an increase in QP for lower values of QP. When QP increases or the quality degrades due to increased quantization, initially the QoE scores fall very slowly. Only after reaching a high value of QP, the QoE scores start to decrease faster. Whereas, QoE drops faster with increasing D_{max}, initially, but after a higher value of D_{max} the decrement of QoE becomes slow. This is because after a high value of D_{max}, around 6 to 8 seconds, the dropped QoE becomes saturated and the users are less sensitive to further increments in D_{max}.

Increasing QP decreases the video bitrate. Thus, when the bandwidth decreases in the network, the QP can be increased significantly to adapt the streaming bitrate to the available bandwidth, rather than risking even a single playout interruption. This depends on the obtained QoE function, whose weights are given in Table 1, as well as the trade-off between increasing QP or risking a playout interruption. With respect to previous observations, our QoE model can be integrated with the controller of adaptive HTTP streaming. It will help the controller to make decisions that optimize QoE for the given values of QP and network parameters.

Figure 1(b) shows the QoE with respect to D_{max} and N. It can be seen that for lower values of D_{max} or N the QoE is very sensitive to both. However, for higher values, QoE decreases very slowly because after a certain value of D_{max} or N the quality is already bad enough and users are not sensitive to further increments. Also note that the worst value of predicted MOS in this figure is 1.5, but for such high values of D_{max} and N the MOS should be 1.0, that is the minimum MOS possible. This prediction error of 0.5 is because RNN shows saturation near bad quality and because while training we wanted to be accurate when quality is good instead of being accurate when quality is already bad.

Table 1. Weights for RNN function.

From l,k	To l,k	W^+	W^-	From l,k	To l,k	W^+	W^-
0:0	1:0	0.13656	0.00946	0:2	1:3	0.04169	0.06672
0:0	1:1	0.04683	0.02828	0:2	1:4	0.13656	0.01814
0:0	1:2	0.05276	12.3531	0:3	1:0	0.05961	0.20870
0:0	1:3	0.25218	0.02317	0:3	1:1	1.90036	0.05845
0:0	1:4	1.13732	1.70886	0:3	1:2	0.07383	1.83576
0:1	1:0	0.00599	0.32773	0:3	1:3	0.29561	12.2138
0:1	1:1	0.02582	67.1723	0:3	1:4	0.00972	1.15571
0:1	1:2	8.03364	1.12285	1:0	2:0	0.11194	2.47160
0:1	1:3	0.01362	1.39952	1:1	2:0	0.01686	0.00329
0:1	1:4	0.00821	0.03468	1:2	2:0	0.00024	0.04832
0:2	1:0	0.30222	0.18335	1:3	2:0	0.01047	12.4031
0:2	1:1	0.02602	0.54512	1:4	2:0	0.11556	0.00370
0:2	1:2	1.92420	1.25226				

Figure 1(c) shows QoE behaviour with respect to $1-\alpha = D_{avg}/D_{max}$. The real significance of $1-\alpha$ is that it characterizes the spread of pause durations. When the value of $1-\alpha$ is low then the interruptions are spread out between 0 and D_{max}, but for high values of $1-\alpha$ the durations of all the interruptions are closer to D_{max}. Thus the QoE scores would be low for higher values of $1-\alpha$. Figure 1(c) shows that the behaviour is as expected.

Figure 2 shows the good accuracy of the estimation using a scatter plot with estimated MOS vs. real MOS obtained from the subjective tests. The points corresponding to the training as well as the validation data are shown. The overall Root Mean Square Error (RMSE) of 0.37 for all data (with slightly higher and lower RMSE for training and validation sets, respectively) on the MOS scale going from 1.0 to 5.0. The RMSE is less than that of the human test panel (0.61) and thus is satisfactory.

We also compared our QoE estimation model with a

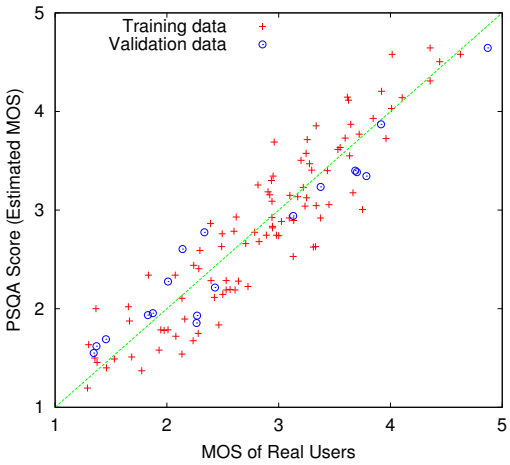


Fig. 2. Real vs Estimated QoE scores with our QoE module.

model proposed in [8], that estimates video quality based on freeze distortions. In Figure 3, we plot the values of mean square error (MSE) for the estimated MOS vs. different QPs. Similar MSE values are observed for low QPs. However, our model performs significantly better in QoE estimation by maintaining low values of MSE even when QP increases, whereas with the existing model the MOS estimation error, increases significantly. This is because, unlike our model, the freeze distortion based model does not take QP into account.

5. CONCLUSIONS

In this study, we have addressed the problem of estimating the QoE of video streaming in TCP/IP networks. As a solution, we designed an automatic no-reference QoE estimation module for HTTP video streaming using TCP and H.264 video codec and the trained RNN function is provided in this paper. The proposed approach is different from the existing ones as it addresses the problem of measuring QoE in the combined case of adaptive bitrate video and the use of a reliable transport protocol. This is the case of the adaptive streaming over HTTP. Extensive simulations showed that our model accurately measures the QoE and performs better than an existing QoE model when the value of QP is varied. Finally, for future work, it should be stressed out that the QoE feedback can be used to take some corrective measures, in case the quality drops, to bring back QoE to satisfactory level.

6. REFERENCES

[1] ITU-T SG12, "Definition of Quality of Experience", COM12 - LS 62 - E, TD 109rev2 (PLEN/12), Geneva, Jan. 2007.

[2] ITU-R Recommendation BT.500-11, "Methodology for the subjective assessment of the quality of television pictures", June 2002.

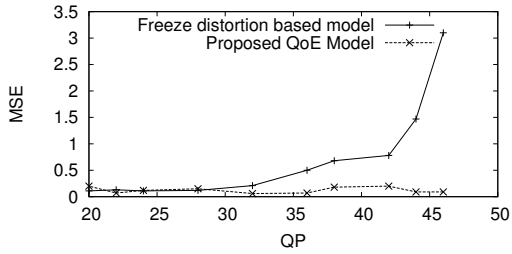


Fig. 3. Comparison of proposed QoE model in this paper with the freeze distortion based model that does not consider QP.

[3] K. Singh and G. Rubino, "No-Reference Quality of Experience Monitoring in DVB-H Networks", WTS 2010, Tampa, USA, April 2010.

[4] S. Wolf and M. Pinson "Video Quality Measurement Techniques", NTIA Report 02-392, June, 2002.

[5] C. J. van den B. Lambrecht. "Perceptual Models and Architectures for Video Coding Applications," PhD thesis, EPFL, Lausanne, Swiss, 1996.

[6] Z. Wang et al. "Video quality assessment using structural distortion measurement." Signal Processing: Image Communication, IEEE. Special issue on "Objective video quality metric", 19(2):121-132, Feb. 2004.

[7] Y. Wang. "Survey of objective video quality measurements." Technical Report WPICS-TR-06-02, EBU, Feb. 2006.

[8] Watanabe, K., Okamoto, J. and Kurita, T., "Objective video quality assessment method for evaluating effects of freeze distortion in arbitrary video scenes." SPIE Electronic Imaging, vol. 6494, 2006.

[9] E. Gelenbe, "Random neural networks with negative and positive signals and product form solution," Neural Comput., vol. 1, pp. 502-511, 1989.

[10] S. Mohamed and G. Rubino, "A Study of Real–time Packet Video Quality Using Random Neural Networks," *IEEE Trans. On Circuits and Systems for Video Tech.*, vol. 12, no. 12, pp. 1071–1083, Dec. 2002.

[11] ITU-T Telecommunication G.1070, "Opinion model for video-telephony applications". Apr. 2007.

[12] x264, "http://www.videolan.org/developers/x264.html".

[13] G. Rubino, "Quantifying the Quality of Audio and Video Transmissions over the Internet: The PSQA Approach," in *Design and Operations of Communication Networks: A Review of Wired and Wireless Modelling and Management Challenges*, edited by J. Barria, Imperial College Press, 2005.

[14] N. Zong, "Survey and Gap Analysis for HTTP Streaming Standards and Implementations", Internet-Draft: draft-zong-httpstreaming-gap-analysis-01, October 2010.

[15] Q. Wu, "Problem statement for HTTP streaming", Internet-Draft:draft-wu-http-streaming-optimization-ps-01, Sep. 2010.