MODEL-BASED QOE PREDICTION TO ENABLE BETTER USER EXPERIENCE FOR VIDEO TELECONFERENCING

Liangping Ma[⋆] *Tianyi Xu*[†] *Gregory Sternberg*[‡] *Anantharaman Balasubramanian*[⋆] *Ariela Zeira*[⋆]

[⋆] InterDigital Communications, Inc., San Diego, CA 92121, USA
[†] Dept. of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716, USA
[‡]InterDigital Communications, Inc., King of Prussia, PA 19406, USA

ABSTRACT

The ultimate goal of network resource allocation for video teleconferencing is to optimize the Quality of Experience (QoE) of the video. We consider the IPPP video coding structure with macroblock intra refresh, which is widely used for video teleconferencing. Generally, the loss of a current frame causes error propagation to subsequent frames due to the video coding structure. Therefore, to optimize the QoE, a communication network needs to be able to accurately predict the consequence of each of its resource allocation decisions. We propose a QoE prediction scheme by considering QoE models that use the per-frame PSNR time series as the input, thus reducing the QoE prediction problem to a per-frame PSNR prediction problem. The QoE prediction scheme is jointly implemented by the video sender (or MCU) and the communication network. Simulation results show that the proposed per-frame PSNR prediction method is fairly accurate, with an average error well below 1dB.

Index Terms— QoE, video, prediction, scheduling, network.

1. INTRODUCTION

In real-time video applications such as video teleconferencing, the IPPP video coding structure is widely used, where the first frame is an intra-coded frame, and each P frame uses the frame immediately preceding it as the reference for motion compensated prediction. To meet the stringent delay requirement, the encoded video is typically delivered by the RTP/UDP protocol, which is lossy in nature. When a packet loss occurs, the associated video frame as well as the subsequent frames will be affected, which is called error propagation. Packet loss information can be fed back to the video sender or multipoint control unit (MCU) (which may perform transcoding) via protocols such as RTP Control Protocol (RTCP) to trigger the insertion of an intra-coded frame to stop error propagation. However, the feedback delay is at least about a round trip time (RTT). To alleviate error propagation, additionally, macroblock intra refresh – encoding some macroblocks of each video frame in the intra mode – is often used.

We realize that a video frame generally is mapped to one or multiple packets (or slices in the case of H.264/AVC) and thus a packet loss does not necessarily lead to the loss of a whole frame. In this paper, we focus on whole frame losses, and leave the more general packet losses for future work.

Although there is no difference in the video coding scheme for the P frames, the impact of a frame loss can be drastically different from frame to frame. As an example, Fig. 1 shows the average loss in PSNR for the subsequent frames if a P frame is dropped in the network for the Foreman-CIF sequence encoded in H.264/AVC with a quantization parameter (QP)= 30, where dropping frame 20 alone leads to a loss of 2.7dB, while dropping frame 22 alone leads to a loss of 5.9dB. This presents an opportunity for a communication network to intelligently drop certain video packets in the event of network congestion to optimize the video quality.

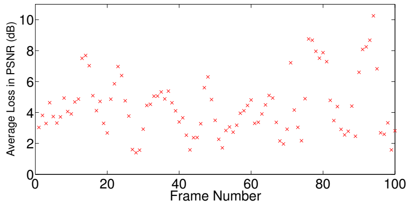


Fig. 1. The impact of a frame loss on the average PSNR of subsequent frames for the Foreman-CIF video sequence.

In this paper, we focus on video Quality of Experience (QoE), and propose a QoE prediction scheme that is low in delay, computational complexity and communication overhead to enable a network to allocate network resources so as to optimize the QoE. Specifically, with such a scheme, the network knows the resulting QoE for each resource allocation decision (e.g., dropping certain frames in the network) so that it can do optimal resource allocation by choosing the decision corresponding to the best QoE. Because of the prediction nature of the problem and the fact that the prediction has to be done within a communication network, we need a QoE model that is feasible to compute, which motivates us to consider

978-1-4799-0356-6/13/$31.00 ©2013 IEEE

those (e.g., [1][2]) that use the per-frame PSNR time series as the input. A per-frame PSNR time series is a sequence of PSNR values, each indexed by its corresponding frame number. The QoE prediction problem then reduces to one of predicting the per-frame PSNR. The proposed QoE prediction scheme is jointly implemented by the video sender (or MCU) and the communication network. Simulation results show that the proposed per-frame PSNR prediction method can achieve an average error much less than 1dB.

We briefly review related work on channel distortion, which is the challenge for predicting the per-frame PSNR. Some aspects of the channel distortion model in the seminal work [3] are adopted in our work. An additive exponential model proposed in [4] is shown to have good performance. However, the determination of the model requires some information (the motion reference ratio) of the predicted video frames to be known a priori. This is possible only if the encoder generates all the video frames up to the predicted frame, introducing significant delay. For example, to predict the channel distortion 10 frames ahead, assuming a frame rate of 30 frames per second, the delay will be 333 ms. In [5], a model taking into account the cross-correlation among multiple frame losses is proposed for channel distortion. However, in the parameter estimation, the whole video sequence needs to be known in advance, making it infeasible for real time applications. Pixel-level channel distortion prediction models are proposed for optimizing the video encoder [6], which, although accurate, are an overkill for the problem we look at. Thus, in this paper, we consider the simpler frame-level distortion prediction.

The remainder of the paper is organized as follows. Section 2 describes the QoE prediction scheme, Section 3 gives the simulation results on the per-frame PSNR prediction, and Section 4 concludes the paper.

2. QOE PREDICTION

2.1. Choosing QoE Models

Subjective video quality testing is the ultimate method to measure the video quality perceived by the human visual system (HVS). However, subjective testing requires playing the video to a group of human subjects in stringent testing conditions [7] and collecting the ratings of the video quality, which is time consuming, expensive, and unable to provide realtime assessment results, not to mention predicting the video quality.

Alternatively, QoE models can be constructed by relating QoS metrics to video QoE [1][2][8][9]. The ITU recommendation G.1070 [9] considers the packet loss rate rather than the packet loss pattern in modeling the QoE, which is insufficient as indicated by our example in Section 1 where the pattern is a single frame loss. The model in [8] has the same problem. The ITU recommendation G.1070 also requires extensive offline subjective testing to construct a large number of QoE models, and extracting certain video features (e.g., degree of motion) [10] during prediction for desired accuracy, making it unsuitable for real-time applications.

The QoE model proposed in [1] uses the statics extracted from the per-frame peak signal-to-noise ratio (PSNR) time series, which are QoS metrics, as the model input. Some of the statistics are the minimum, maximum, standard deviation, the 90% and the 10% percentiles, and the difference in PSNR between two consecutive frames. Although the average PSNR of a video sequence is generally considered a flawed video quality metric, the model in [1] is shown to outperform Video Quality Metric (VQM) [11] and Structural Similarity (SSIM) [12] in terms of correlation to subjective testing results. With the choice of such QoE models, the QoE prediction problem reduces to one that predicts the per-frame PSNR time series.

2.2. The Proposed QoE Prediction Approach

Before discussing various approaches to QoE prediction, we note that the pattern of packet losses is important, because the video quality, or specifically statistics of the per-frame PSNR time series, depends on not only how many frame losses have occurred, but also where they have occurred in the video sequence.

There are three approaches to QoE prediction. In a *sender-only* approach, the per-frame PSNR time series for each frame loss pattern is obtained by simulation at the video sender. However, the number of frame loss patterns grows exponentially with the number of video frames. Even if the amount of computation is not an issue, the resulting per-frame PSNR time series need to be sent to the communication network, generating excessive communication overhead.

In a *network-only* approach, the network decodes the video and finds out the channel distortion for different packet loss patterns. However, the video quality depends on not only the channel distortion, but also the distortion from source coding. Due to lack of access to the original video, it is impossible for the network to know about the source distortion, making the QoE prediction inaccurate. Also, this approach becomes unscalable when the network serves a very large number of video teleconferencing sessions simultaneously. Finally, this approach may not be suitable when the video packets are encrypted.

We propose a *joint* approach that involves both the video sender (or MCU) and the network. The video sender obtains the channel distortion for single frame losses, and passes the results along with the source distortion to the network. The network knows the QoE model, and for each resource allocation decision, it calculates the total distortion for each frame (and hence the per-frame PSNR time series) by utilizing the linearity assumption for multiple frame losses [3][4]. This approach eliminates virtually all the communication overhead in the sender-only approach, takes into account source distortion absent in the network-only approach, and does not need to do video encoding/decoding in the network.

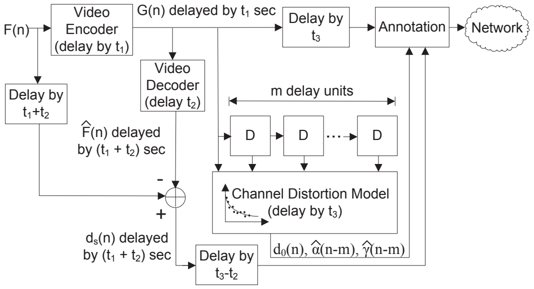

F(n)
Video Encoder (delay by t_1)
G(n) delayed by t_1 sec
Delay by t_3
Annotation
Network
Delay by t_1+t_2
Video Decoder (delay t_2)
m delay units
D
D
D
$\hat{F}$(n) delayed by (t_1 + t_2) sec
-
+
Channel Distortion Model (delay by t_3)
d_s(n) delayed by (t_1 + t_2) sec
Delay by t_3-t_2
d_0(n), $\hat{\alpha}$(n-m), $\hat{\gamma}$(n-m)

Fig. 2. System architecture of the video sender.

2.3. Per-frame PSNR Prediction

As mentioned before, frame-level channel distortion prediction is appropriate for the problem we focus on. We first look at the video sender side, whose architecture is shown in Fig. 2. Let the number of pixels in a frame be N. Let $F(n)$, a vector of length N, be the nth original frame, and $F(n,i)$ denote pixel i of $F(n)$. Let $\hat{F}(n)$ be the reconstructed frame without frame loss corresponding to $F(n)$, and $\hat{F}(n,i)$ be pixel i of $\hat{F}(n)$. Original video frame $F(n)$ is fed into the video encoder, which generates packet $G(n)$ after a delay of t_1 seconds. Packet $G(n)$ may represent multiple NAL units, which we call together a packet for convenience. Packet $G(n)$ is then fed into the video decoder to generate the reconstructed frame $\hat{F}(n)$, which takes t_2 seconds. Note that in a typical video encoder, this reconstruction is already in place. Let the distortion due to source coding for $F(n)$ be $d_s(n)$. Then,

$$d_s(n) = \sum_{i=1}^{N} (F(n,i) - \hat{F}(n,i))^2/N, \tag{1}$$

which is readily available at the video encoder.

As mentioned earlier, the construction of the channel distortion model in [4] requires some information (the motion reference ratio) of the predicted video frames to be known in advance, which results in significant delay. To address this problem, we propose using the current packet $G(n)$ and the previously generated packets $G(n-1), ..., G(n-m)$ to train a channel distortion model. In Fig. 2, D represents a delay of an inter-frame time interval. The training takes t_3 seconds. Note that $t_3 \geq t_2$, because the Channel Distortion Model needs to decode at least one frame. The values of the parameters for the model are then sent to the Annotation block for annotation. Also annotated is the source distortion $d_s(n)$. The annotated packet is then sent to the communication network.

We now look at the details of the channel distortion model. Prior results show that a linearity model performs well in practice [3][4]. For each frame loss, we define function $h(k,l)$ [4], which models how much distortion the loss of frame k causes to frame l for $l \geq k$

$$h(k,l) = d_0(k) \frac{e^{-\alpha(k)(l-k)}}{1 + \gamma(k)(l-k)} \tag{2}$$

where $d_0(k)$ is the channel distortion for frame k, resulting from the loss of frame k only and the error concealment, and $\alpha(k)$ and $\gamma(k)$ are parameters dependent on frame k. In this paper, we consider a simple error concealment scheme, namely, frame copy. Hence the distortion due to the loss of frame k (and only frame k) is

$$d_0(k) = \sum_{i=1}^{N} (\hat{F}(k,i) - \hat{F}(k-1,i))^2/N. \tag{3}$$

In (2), $\gamma(k)$ is called leakage, which describes the efficiency of loop filtering to remove the artifacts introduced by motion compensation and transformation [3]. The term $e^{-\alpha(k)(l-k)}$ captures the error propagation in the case of pseudo-random macroblock intra refresh. In [3], a linear function $(1 - (l-k)\beta)$, where β is the intra refresh rate, is proposed. We do not use this linear function, because the macroblock intra refresh scheme in [3] is cyclic, while the one used in our simulation software (JVT JM 16.2 [13]) is pseudo-random. The linear model states that the impact vanishes after $1/\beta$ frames (the intra refresh update interval for the cyclic scheme), which is not the case for the pseudo-random scheme as suggested by our simulation results. An exponential model such as the one in [4] is better. However, the model in [4] fails to capture the impact of loop filtering, while our model does capture it. The values of $\alpha(k)$ and $\gamma(k)$ can be obtained by methods such as least squares or least absolute value via fitting simulation data. In Fig. 2, the video sender drops packet $G(n-m)$ from the packet sequence $G(n), G(n-1), ..., G(n-m)$, performs video decoding, measures the channel distortions, and finds the value of $\alpha(n-m)$ (defined as $\hat{\alpha}(n-m)$) and the value of $\beta(n-m)$ (defined as $\hat{\gamma}(n-m)$) in (2) with the substitution $k = n-m$ that minimize the error between the measured distortions and the predicted distortions.

We next look at the network side. We assume that the network has packets $G(n), G(n-1), ..., G(n-L)$ available. Let $I(k)$ be the indicator function, being 1 if frame k is dropped and 0 otherwise. A packet loss pattern can be characterized by a sequence of $I(k)$'s. For convenience we denote a pattern by a vector $P := (I(n), I(n-1), ..., I(0))$. The channel distortion of frame $l \geq n-L$ resulting from P is then predicted as

$$\hat{d}_c(l,P) = \sum_{k=0}^{l} I(k)\hat{h}(k,l), \tag{4}$$

where the linearity assumption for multiple frame losses in

[3][4] is used, and

$$\hat{h}(k,l) = d_0(k)\frac{e^{-\hat{\alpha}(k-m)(l-k)}}{1+\hat{\gamma}(k-m)(l-k)}. \tag{5}$$

We realize that the model in (4) can be improved, for example, by considering the cross-correlation of frame losses [5]. However, as mentioned earlier, the model in [5] is not suitable for real time applications, and its complexity is very high. The simple model in (4) proves to be reasonably accurate [3][4].

In order to predict the per-frame PSNR for a particular packet loss pattern P, the network needs to know about the source distortion as well. The total distortion prediction can be represented as

$$\hat{d}(l,P) = \hat{d}_c(l,P) + \hat{d}_s(l), \tag{6}$$

where $\hat{d}_s(l) = d_s(l)$ for $n \geq l \geq n-L$, and $\hat{d}_s(l) = d_s(n)$ for $l > n$, and where we have applied the assumption that the channel distortion and the source distortion are independent, which is shown to be pretty accurate [14]. Note that the source distortion estimation $\hat{d}_s(l)$ for $n \geq l \geq n-L$ is precise and readily available at the video sender and is included in the annotation of the $L+1$ packets $G(n), ..., G(n-L)$.

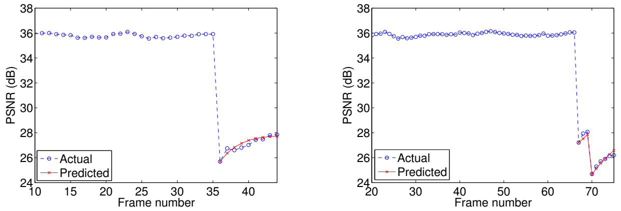
The PSNR prediction for frame $l \geq n-L$ with packet loss pattern P is then

$$\widehat{\text{PSNR}}(l,P) = 10\log_{10}(255^2/\hat{d}(l,P)). \tag{7}$$

The per-frame PSNR time series is then $\{\widehat{\text{PSNR}}(l,P)\}$, where l is the time index. The time series is a function of P. Thus, to generate the best time series, the network chooses the optimal P among those that are feasible under the resource constraint. Note that, part of P, i.e., $I(n-L-1), I(n-L-2), ..., I(0)$, is already determined, because a frame between 0 and $n-L-1$ has been either delivered or dropped. The variables subject to optimization are the remaining part of P, i.e., $I(n-L), ..., I(n)$. We define the prediction length λ as the number of frames to be predicted. That is, if the nth frame is to be dropped, then the predictor predicts frames n through $n+\lambda$. Note that, it is not necessary to predict many frames, since it takes the video encoder not more than one RTT to receive feedback about a frame loss.

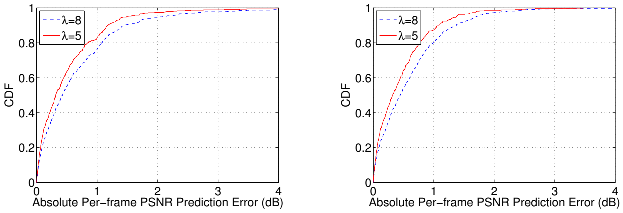
3. SIMULATION RESULTS

We evaluate the performance of the proposed per-frame PSNR prediction method via simulation. We consider both single frame losses and multiple frame losses. The Foreman CIF video sequence is used. For $m = 10$, $L = 5$, and $\lambda = 8$, Fig. 3(a) shows the prediction for frames $l \geq 36$ if frame 36 is dropped, and Fig. 3(b) for frames $l \geq 67$ if frames 67 and 70 are dropped. More simulation results are shown in Fig. 4. We plot the cumulative distribution function (CDF) of the *absolute per-frame PSNR prediction error*, i.e., the absolute value of the difference between the actual per-frame PSNR and the predicted value, both in dB. We consider prediction length of 8 (blue dashed lines) and of 5 (red solid lines). Fig. 4(a) is for single frame losses. Fig. 4(b) is for multiple frame losses, and in particular we consider two frame losses with a gap of 2 frames in between. We also calculate the mean value of the absolute prediction error. For single frame losses, it is 0.66dB and 0.51dB for prediction lengths 8 and 5, respectively. For multiple frame losses, it is 0.60dB and 0.46dB for prediction lengths 8 and 5, respectively.



(a) (b)

Fig. 3. The per-frame PSNR prediction for (a) a single frame loss at frame 36; (b) two frame losses at frames 67 and 70.



(a) (b)

Fig. 4. The CDF of per-frame PSNR prediction error for (a) single frame losses; (b) two frame losses with a gap of 2 frames in between.

4. CONCLUSION

We propose a QoE prediction scheme that allows a communication network to optimize resource allocation for video teleconferencing. By using QoE models that take the per-frame PSNR time series as the input, the QoE prediction problem reduces to a per-frame PSNR prediction problem. Simulation results show that the proposed per-frame PSNR prediction method achieves an average prediction error well below 1dB.

Acknowledgement The authors would like to thank Dr. Zhifeng Chen, Dr. Rahul Vanam, and Dr. Yuriy Reznik of InterDigital for their help and insightful discussion.

5. REFERENCES

[1] C. Keimel, T. Oelbaum, and K. Diepold, “Improving the prediction accuracy of video quality metrics,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Dallas, Texas, March 2010, pp. 2442–2445.

[2] C. Keimel, M. Rothbucher, H. Shen, and K. Diepold, “Video is a cube: Multidimensional analysis and video quality metrics,” *IEEE Signal Processing Magzine*, pp. 41–49, Nov. 2011.

[3] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, “Analysis of video transmission over lossy channels,” *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 6, pp. 1012–1032, June 2000.

[4] U. Dani, Z. He, and H. Xiong, “Transmission distortion modeling for wireless video communication,” in *IEEE Global Telecommunications Conference (GLOBECOM)*, Dec 2005.

[5] Y. J. Liang, J. G. Apostolopoulos, and B. Girod, “Analysis of packet loss for compressed video: Effect of burst losses and correlation between error frames,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 861–874, July 2008.

[6] Z. Chen and D. Wu, “Prediction of transmission distortion for wireless video communication: Algorithm and application,” *Journal of Visual Communication and Image Representation*, vol. 21, no. 8, pp. 948–964, Nov. 2010.

[7] *Subjective Video Quality Assessment Methods for Multimedia Applications, ITU-T Recommendation-P.910*, Sep. 1999.

[8] M. Venkataraman and M. Chatterjee, “Inferring video QoE in real time,” *IEEE Network*, pp. 4–13, Jan./Feb. 2011.

[9] *Opinion Model for Video-Telephony Applications, ITU-T Recommendation G.1070*, 2007.

[10] Jose Joskowicz and J. Carlos Lpez Ardao, “Enhancements to the opinion model for video-telephony applications,” in *Proceedings of the 5th International Latin American Networking Conference (LANC)*, 2009, pp. 87–94.

[11] M. Pinson and S. Wolf, “A new standardized method for objectively measuring video quality,,” *IEEE Trans. Broadcasting*, vol. 50, no. 3, pp. 312322, Sep. 2004.

[12] Z. Wang, L. Lu, and A. C. Bovik, “video quality assessment based on structural distortion measurement,” *Signal Processing: Image Communication*, vol. 19, no. 2, pp. 121–132, Feb 2004.

[13] ITU, “H.264/AVC reference software,” Online, Oct 2012, iphome.hhi.de/suehring/tml/download/.

[14] Zhihai He, Jianfei Cai, and Chang Wen Chen, “Joint source channel rate-distortion analysis for adaptive mode selection and rate control in wireless video coding,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 12, no. 6, pp. 511–523, June 2002.