Towards a Video QoE Definition in Converged Networks

Mukundan Venkataraman, Shamik Sengupta, Mainak Chatterjee
Electrical Engineering and Computer Science
University of Central Florida
Orlando, FL 32826
{mukundan, shamik, mainak}@eecs.ucf.edu

Raja Neogi
C-Cor Inc.
15797 NW Andalusian Way
Portland, OR 97229
raja_neogi@ieee.org

Abstract

Most resource allocation decisions in converged multi-service networks are presently driven by QoS parameters and service level agreements. Network operators are realizing that such policy decisions ought to be based on Quality of Experience *(QoE), which is a better measure of subjective video perception. Video QoE has two components: (a) Zapping and (b) Video Quality or Video Mean Opinion Score (MOS). Zapping anomalies are well understood and easily characterized, but there is no consensus on what the definition of a Video-MOS should be. This is largely because there is little understanding of qualitative and quantitative* subjective *degradations caused by networking events like loss, delay, jitter, and error. In this work we conduct extensive experiments on a simulation testbed in a search for a good Video-MOS definition, with specific emphasis on quality degradations due to network transmission. We deploy "collector" nodes that can gather various statistics of a given flow. We also identify key parameters that such collector nodes must maintain such that an overlay of such nodes can identity impairment points. Lightweight Video-MOS definition evolving out this work can be used in collector overlay networks to solve problems like root-cause analysis, capacity-planning and various network optimization problems.*

1 Introduction

Network service providers are actively deploying triple and quadruple play networks that deliver voice, video and data services over converged infrastructure. There is already a need to deliver high quality real time streaming applications like Video on Demand (VOD) and IPTV. Quality-of-service (QoS) parameters like bandwidth, delay, and jitter are typically used to guarantee services. These metrics however fail to capture the subjectiveness associated with human perception and understanding. Network operators instead need to mutate policies based on *quality-of-experience* (QoE). It is not just sufficient to guarantee peak rate for voice or video service, but instead, being able to guarantee good mean opinion scores (MOS) will certainly bring in new subscribers.

Video QoE is influenced by both video MOS and zapping irregularities. Video MOS captures the subjective assessment of video quality, while poor zapping indicates signaling anomalies. Signaling anomalies are well understood, while video MOS lacks well accepted semantic definition and is a much harder problem to deal with. Before one can gather video quality degradations in the network, there is a need to agree on how to measure this.

Currently used video evaluation schemes are either housed on the end systems or in the network core. Examples of end system schemes are the peak signal to noise ratio (PSNR), MPQM [2] and VQM [3]. PSNR is a simple frame-to-frame calculation, VQM a subjective evaluation, and MPQM considers human eye perception and subjectivity. Differing in levels of complexity of operation and the need for specialized hardware, such end system implementations which characterize a stream at playout (after the network transmission is *complete*) can do little to isolate or correct any network induced impairment. In fact, their primary purpose is to evaluate the effectiveness of different transcoding schemes only. The media delivery index (MDI) [1], on the other hand, is housed inside the network and provides statistics on delay factor and media loss rate on a given stream. The scheme is too simple and there is an extremely loose correlation between networking events and a subjective perception of a playout at destination. Broadly speaking, these processes can be divided into two segments: processes in active network elements of the end-to-end journey (transcoding points), and intermediate events that potentially happen throughout the network. These two processes need to evolve independently, with each process complimenting the other to reach the common goal of high quality user perceived streaming content.

We focus on the networking part and seek to understand the effect of various network events on the final quality of content at the end user. Our aim is to identify properties of a video evaluation metric that has the following properties: (i) must be intuitive, easy to implement and deploy, and not require specialized hardware; (ii) should be housed in the network core (or *collector* nodes) where events occur; (iii) able to identify the source of the problem as it happens, is aware of cause and effects (i.e., a network event, and its *user perceived* effect on playout); and (iv) bridge the process of encoding and network transmission, such that there is more information passed between transcoding and networking. All these properties create a possibility for corrective actions when possible. Also, a harmony in the entire process could help the network core draw some inferences on the importance of a particular packet in transit.

We investigate for a good video-MOS definition, with specific emphasis on transmission aspects. Quality degradations due to transcoding are beyond the scope of this work. We perform simulations with realistic workload scenarios, taking parameters (namely loss, error, delay and jitter) to study the effects of different events on video quality.

The rest of the paper is organized as follows. In Section 2, we discuss the need for this kind of study. The simulation testbed is explained in Section 3. The experimental results are discussed in Section 4. Key observations and conclusions are presented in the last section.

2 Motivations and Background

Currently, there is little understanding on how to isolate the major *causes* of user perceived degradation. Various end system video quality assessment schemes exist, such as the Peak Signal to Noise Ratio (PSNR), MPQM [2] and VQM [3]. PSNR is a ratio of the maximum possible power of the signal to that of its disrupting noise. PSNR is a purely frame-to-frame calculation, and does not take into consideration metrics such as human perception or subjective evaluations. It is, however, a popular way of reporting video assessment. MPQM is a complex mechanism which requires extensive machinery to analyze human eye perception and its effects on a playout and a subjective evaluation. Similarly, VQM returns a subjective rating of the quality of a video on a scale of 0 to 5, with 5 denoting the worst quality. Note that all the above schemes are necessarily evaluated at the *end* host. In fact, these metrics were primarily developed to evaluate different transoding schemes. End systems implementations cannot isolate network induced impairments, and hence, cannot completely recover from such errors other than apply more intelligent patches, which has its upper bounds. We argue that the only place a network induced impairment can be corrected (provided correction is possible) is at the point where the error actually occurs. The Media Delivery Index (MDI) [1] gather statistics on various networking events on a given flow over a specified period of time, and can return statistics such as the delay factor (DF) and the media loss rate (MLR) and is denoted by a number such as DF:MLR. This statistic conveys little about what the impact of these events is on a streams playout at the destination based on a *subjective* perception.

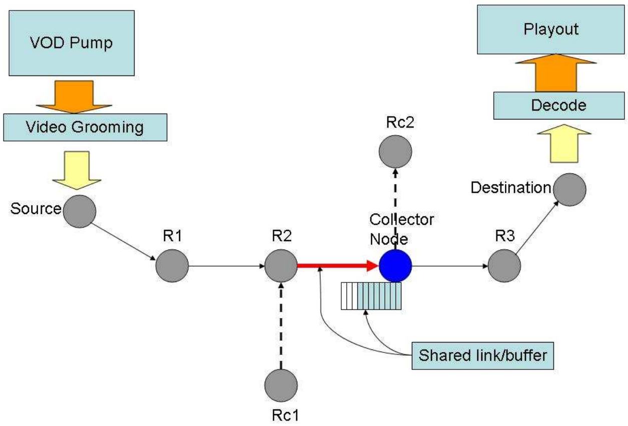


Figure 1. Simulation topology

Moreover, when content flows through network elements, there is no way to infer flow characteristics unless explicitly specified though in-band or out-of-band signaling. One way to achieve this is to introduce probes in the form of collector nodes. By correlating probe results, any kind of flow anomaly (e.g., video MOS) could be detected and isolated. Proper recovery mechanisms (e.g., link layer protocol boosting) could then be invoked to fix the problem.

3 Testbed Set-up

Our testbed consists of a source video, encoders, a simulated network with tunable parameters and a decoder. We use the MPEG-4 [8] specification to encode the source video stream and the network simulator (ns-2) [7] to simulate the network. We use packages such as Evalvid [5] and MSU Video Evaluation Toolkit [6] to better model the network to as close to reality as possible. Ns-2 has well tested simulation suites to model various transport protocols, buffering strategies, links and network dynamics.

We perform the following sequence of events. First, we take a raw video in a YUV format and perform MPEG-4 encoding to generate frames. These frames are then converted to IP packets with a fragmentation limit. At the destination, IP packets are reassembled to form the MPEG-4 frames which are then played out. At playout, we apply two different evaluation schemes: the PSNR and the VQM to understand the effects of various network events.

We seek to understand the correlations between network events like losses, jitter, delay to metrics like PSNR and subjective quality perception. The topology used for simulations is shown in Fig. 1. Video grooming is used to inject frames into the network at a *constant* rate instead of overwhelming the network with a naive UDP blasting

Simulation Parameters	Value
Resolution	352x288 pixels
Frame rate	30 frames/sec
Video Color Mode	Y, U, V $(4-2-0$ scheme)
GOP Length	30 Frames
Unshared Links	1Mbps
Link Delay	1ms
Buffer length	50
Queue behavior	Droptail
Sequence length	10 secs of playout
Packet Size	1052 bytes
Max. Fragmented Size	1024 bytes

Table 1. Simulation parameters

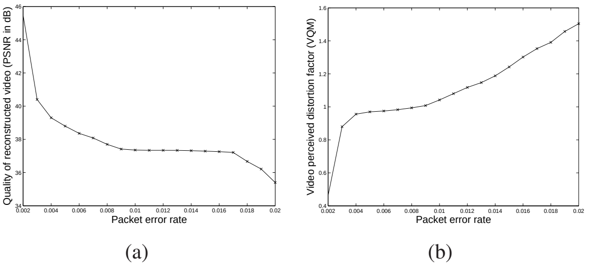
strategy. The path taken by the MPEG-4 stream is as follows: the source node (node "Source"), the intermediate IP hops (with a designated collector node (node "Collector") that collects various event statistics), and a destination node (node "Destination"). We also set up a competing traffic originating at "Rc1" and terminating at "Rc2". The rate and duration of this competing traffic, in combination with a choice of various other simulation parameters (shared buffer/queue length, link speeds etc.) are tuned to create various events in the network like delay, jitter, and loss. Other simulation parameters are shown in Table I.

4 Simulation Results

Simulations were performed to study the impact of individual network events in isolation and combinations. We evaluate video quality in terms of PSNR and VQM. For the rest of the paper, PSNR is synonymous to "Y-YUV PSNR".

The link between node "R2" and "Collector node" (CN) is shared by the video source and the competing flow, along with the transmission buffer of the CN (Fig. 1). We switch the competing traffic on/off at different rates and durations to introduce various network events: (i) delay: when both flows share the router buffers; (ii) jitter: when traffic pulses of varying rates are injected from the competing traffic; (iv) error: Network induced noise, which corrupts the packet contents; and (iii) loss: when the router buffer overflows.

Effect of Packet Errors: Error is a network induced noise, such that the information content is corrupt. Since successful decoding depends a lot on reference frames being intact, errors tend to *propagate* and spill into successive frames. We conducted simulations at various packet error rates (PERs). The link qualities were chosen so as not to create packet drops, and each simulation for a given packet error rate was run for at least 30 times. We report the average PSNR and VQM in Figures 2(a) and 2(b).



(a) (b)

Figure 2. (a) PSNR vs. PER; (b) VQM vs. PER.

PSNR reacts sharply with a noticeable drop initially (Fig. 2(a)) with increasing PER when error starts to propagate into many successive frames. At high PERs, PSNR is almost constant. This happens due to a combination of events: (i) at low error rates, a few corrupt reference frame packets effect reconstruction of several other dependent frames; and (ii) at higher error rates, reference frame packets *and* the dependent frame packets are simultaneously corrupt, making PSNR dip relatively less. Note, however, that in both cases, we found a similar number of reference and dependent frames corrupt either because of error spilling or network errors.

VQM reports are not analogous, however. The plot for VQM (Fig. 2(b)) shows a twofold initial increase in scores with the introduction of PER (which creates error spilling). At higher PERs, much unlike PSNR, VQM continues to rise. This is reflective of *growing* user dissatisfaction with increasing error rates. It is interesting to note that we (human subjects) were also able to distinguish between error spilling (at low error rates) and network errors (at high errors, where many frames are naturally corrupted).

Effect of Packet Loss: IP fragmentation breaks individual frames to packets: an I-Frame was usually broken into 16 packets with a fragmentation of 1024 bytes in our simulations. A lost packet would contain a *part* of a frame. Further, from the MPEG-4 header it is easy to observe that the I-Frame losses are more important than other frame losses, since the other frames are constructed with reference to an I-frame. We studied the effect of packet losses, with varying levels of packet *errors* simultaneously present.

Our results, however, show that not only are the I-Frames important, the *first* such frame in a given temporal sequence is the most important. This is because for a given sequence with temporal redundancy, successive I-Frames may be reconstructed or error patching techniques applied to generate missing frames. With the loss of the first reference frame, the effect on playout is degrading: the stream freezes at one frame for an entire GOP playout time or presents a complete "white out" if nothing could be done (especially the case when the *very* start of a video has the frame lost). We ran an experiment where the competing traffic was so chosen to cause just enough congestion for the network to lose a given GOP sequence packets for a 10 GOP stream. This is

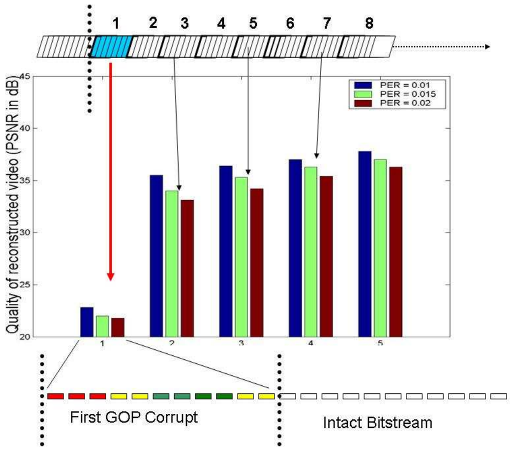


Figure 3. Observed PSNR when the *first* frame occurs in various GOP intervals.

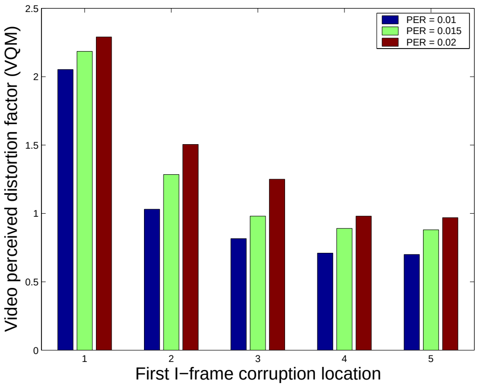
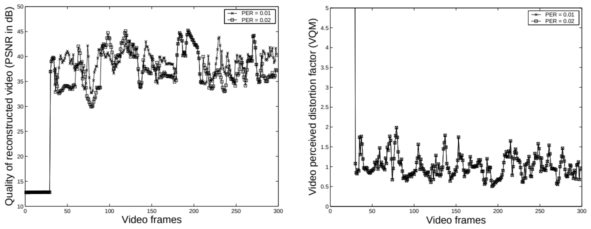


Figure 4. Various VQM values with a first frame loss in different intervals.

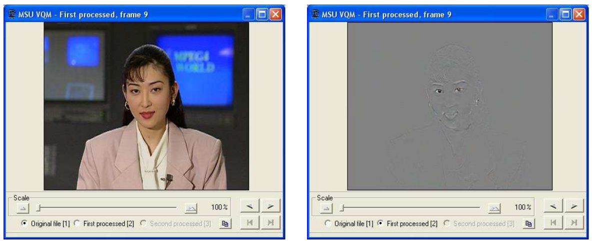
illustrated for the case of PSNR in Fig. 3, where the loss of the *first* GOP in a temporal sequence has a severe impact on the PSNR value (registering around 23 dB), whereas PSNR is around 35-37 dB when the first frame loss happens in any other GOP interval. Also shown in the figure are the effects of different error rates in combination with a reference frame loss, to provide a better understanding of PERs in combination with losses. The differences in PSNR values for different PERs given that losses occurred at a particular GOP sequence are marginal, indicating that the effect of errors is far less than the effect of a loss.

Analogous is the case for VQM (Fig. 4), which is indicative of a medium to poor subjective score when the first GOP is lost (scores at around 2.5), whereas lower VQM scores for losses at other points indicate lesser user irritation. Note that the values of VQM seem to drop steadily with the first frame loss happening at higher GOP sequences, which indicate that user perceived degradation is lesser as more intelligent error patching is applied to reconstruct frames using previous (intact) frames.



(a) (b)

Figure 5. (a) PSNR fluctuations with a first I-Frame loss; (b) Analogous case for VQM.



(a) (b)

Figure 6. (a) Original playout of the first GOP at source; (b) Playout at destination.

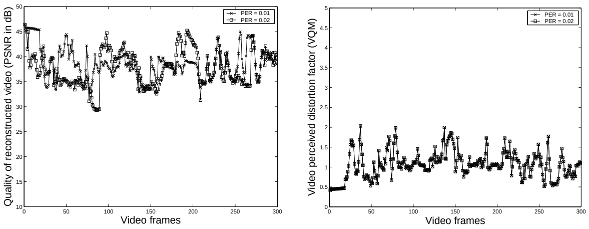
PSNR and VQM variations in a playout sequence: To better understand the effects of losses and error on a playout sequence, we plot PSNR and VQM values on a frame by frame basis at the destination. We consider two cases: loss at the first and third GOP sequences. In both cases, we ran the experiment at two different values of PERs (0.01 and 0.02).

Given that the first frame of a temporal sequence is lost, we study its effects on PSNR on a frame-by-frame basis (Fig. 5(a)). The loss of the first frame holds PSNR to a low 13dB for the entire playout of the first GOP[1]. From the arrival of the second GOP, PSNR stabilizes (or rather oscillates) at around 40dB. It is interesting to note the *amplitude* of oscillations in PSNR values at different PERs. Higher error rates seem to have a bit of an effect, since we consistently found the amplitude of oscillations to grow higher with increasing packet error rates.

Frame by frame PSNR for a loss of the third GOP sequence are shown in Fig. 7(a). Notice that there are no major degradations (to sub 15dB) in the PSNR value throughout the playout, which indicates a reasonably acceptable reconstruction based on the first two lossless GOP sequences. As in the previous case, PSNR fluctuations registered a slightly larger amplitude at higher packet error rates.

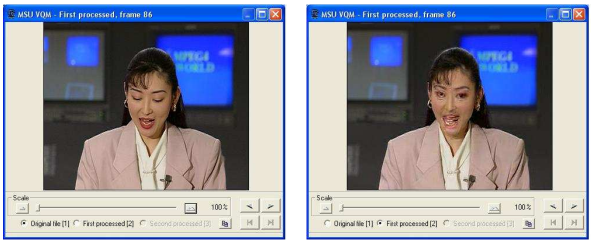
Subjective reactions to the playout as indicated by VQM are analogous. For the sequence we considered, the first GOP happened to be the *very* beginning of the sequence as well. Since it could be hardly reconstructed, we observed a complete "white-out" or frame freezing (as shown in Figs. 6(a) and 6(b)). VQM reflected a score of 5.0 (or the "worst"

[1] In the experiment, only a subset of packets carrying the first I-Frame were actually lost.



(a) (b)

Figure 7. (a) PSNR fluctuations with the *third* I-Frame packets in error; (b) The case for VQM.



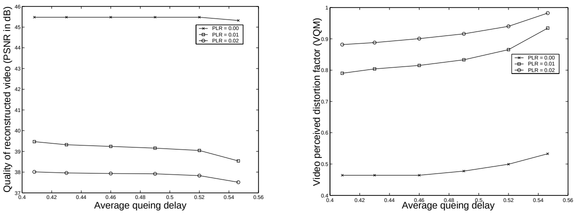
(a) (b)

Figure 8. (a) Playout of third GOP at source; (b) Playout at destination.

rating), indicating strongest user dissatisfaction when this happens. This clearly establishes the need to preserve the first I-Frame of a temporal sequence, especially if it happens to be the very first frame of the playout itself. Frame by frame VQM for the first GOP loss with different PERs are as shown in Fig. 5(b). Notice the value of 5.0 for the entire first GOP playout, which subsequently starts to stabilize (or oscillate) with the arrival of the second GOP sequence. VQM plot for a playout with the third GOP lost is shown in Fig. 7(b). The ratings in this case never rise above 2.0, which reflect that users were never overly dissatisfied with the playout. The screen shots of the playout at around the third GOP are shown in Figs. 8(a) and 8(b). Though the picture was reconstructed, the information content is not necessarily the same (while the person in the picture has eyes lowered in the original playout, she has her eyes fixed at the viewer in the reconstructed playout).

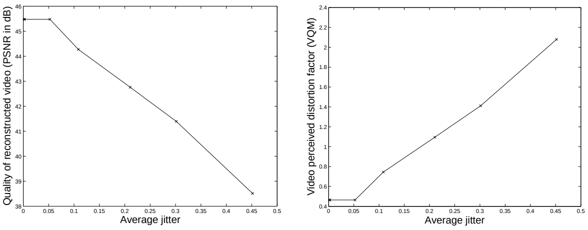
It is interesting, however, to note the *fluctuations* in VQM ratings on a frame by frame basis for different packet error rates. We observe *no* difference in the VQM values for two different PERs. In fact, the plots overlap. The amplitude of oscillations is the same, which means that users would not really distinguish between lower or higher error rates (provided it is less than a threshold). But the fact that there are errors present (which manifests itself as small rectangular blocks of corrupt content at various places in the screen) irritates the user to the same degree which accounts for the oscillations observed.

Effect of delay on playout: If the data stream has very tight delay bounds or are time critical (like interactive messages), time to deliver becomes most important and criteria.



(a) (b)

Figure 9. (a) PSNR with uniformly increasing average delay; (b) The case for VQM.



(a) (b)

Figure 10. (a) Effect of jitter on PSNR; (b) Jitter and VQM.

Apart from such data, queuing delays have little effect on quality if they are uniformly present throughout the playout. We seek to understand the effects of uniformly increasing average queuing delay on PSNR (Fig. 9(a)). Overall, PSNR remains at constant due to pure delay alone, decreasing marginally at high delay, especially when combined with packet losses in the network (queue overflow at collector node). The effects of loss are noteworthy: there is noticeable reduction of PSNR readings with the very introduction of loss. Notice that the difference between PSNR readings with loss rates of 0.01 and 0.02 are much smaller than that of 0.01 and no losses.

The observations from VQM are similar (Fig. 9(b)). When there are no losses, the uniform delay makes little difference to the subject. However, with the introduction of losses in combination with delay, VQM values indicate a stronger user dissatisfaction when there are little losses compared to no loss, while there is little change in opinion between a loss rate of 0.01 and 0.02. Note, however, that VQM calculations are not calculated with the metric of *expected* frame at a particular time of playout. Hence, even if the video starts after a while from its generation event at the source, VQM is high provided there are little losses.

Effect of Jitter: Jitter, the variance of the per-packet delay, is caused due to non-uniform sharing of the links and has direct implications on the receiver buffer occupancy. High jitter can lead to both buffer overflow and underflow– sometimes the buffer is overwhelmed with packets packet and sometimes, the lack of it. Both extremes lead to distortions, since information is at a loss either way. We study the effects of different jitter rates on the playout sequence.

We simulated jitter by introducing a competing VBR traffic. This makes the buffer occupancy among these two flows non-uniform: the video sequence sometimes sees many packets ahead, and sometimes very little competition. Also, to create high jitter, we assumed large buffer sizes at the intermediate nodes such that packets are not (tail) dropped. We plot PSNR for different values of jitter as shown in Fig. 10(a). PSNR reacts strongly to values of jitter exceeding 0.05 seconds, dropping rapidly. However, VQM suggests that human subjects can tolerate jitter levels of around 0.06 as evident from Fig. 10(b).

5 Observations and Discussions

Significant work on overlay networks such as [4] create the possibility for an overlay of collector nodes to initiate some viable action when possible. This could be done with a knowledge of *what* information a given packet of the flow carries, what its *relative* importance is, and most importantly, what effects would it have on the playout quality, both objectively and subjectively.

Should the network be content aware? It is important to note that relative *relevance* of a particular packet is highly dependent on the type of video stream it carries. For example, streams with very high temporal redundancy (a news program) are a lot different from streams with little or no redundancy (a football game).

If more information between transcoding and networking is to be passed, it is required to pass the *content information* of a particular stream onto the network. Gathering statistics is a lot more meaningful when it is combined with the content of the information flow. For example, if for a given stream, 50 packets are dropped towards the end, it is easy to predict that it would have little impact on a news cast. On the other hand, it would have a noticeable effect on a football game stream. This is simple to project from the experiments conducted.

PSNR vs. Subjective Evaluations: The end user is to be ultimately served, taking subjectivity as a factor in evaluation makes a lot of sense. There is a clear indication to diverge from metrics that have been long used to evaluate the effectiveness of transcoding alone.

During the course of our simulations, we found PSNR to report *false positives* in most cases, where even though there was enough degradation for VQM to report poor scores; PSNR continued to indicate high quality and vice versa. Though PSNR is a well established metric, we observe: (i) when subjective results show growth in displeasure with increasing error rates, PSNR readings remained constant; (ii) when subjective opinion would show no change between different error rates, PSNR's oscillation amplitude increases; (iii) when subjective scores indicate intolerance of jitter exceeding 0.05 secs of variations in receive time, PSNR remains a constant to 0.06 seconds and (iv) when subjective scores report increasing displeasure with rising delay, PSNR remains constant. This clearly establishes that subjective evaluations are an important dimension to be considered when assesing quality.

6 Conclusions

We set out to understand the correlations between networking events and subjective evaluation of streaming applications over IP networks. We have gone further than MDI in establishing correlations between networking events and subjective quality perception.

We find loss to be the most degrading networking event. We further find that not only are I-Frame packet losses important, the *first* such frame in a temporal GOP sequence is the most important.

Our results for PSNR and VQM readings clearly establish that subjective evaluations are an important dimension to video quality assesment. Also, end host quality evaluations make little sense. Monitoring the quality of a stream should be a continuous process since errors can be induced during transcoding and network transportation.

Collector nodes are a viable means of extracting impairments points in the network. A knowledge of metrics that a collector node might keep statistics about are: the content of a stream, the relative importance of information, network events on a packet and their subjective requirements.

References

[1] J. Welch and J. Clark, "*A Proposed Media Delivery Index (MDI)*", IETF RFC 4445. Apr 2006

[2] C. J. van den Branden Lambrecht and O. Verscheure, "Perceptual quality measure using a spatio temporal model of the human visual system", *Proc. IST/SPIE Conference Digital Video and Compression: Algorithms and Technologies 1996*, vol 2668, Feb 1996

[3] M. H. Pinson and S. Wolf, "A New Standardized Method for Objectively Measuring Video Quality", *IEEE Trans. on Broadcasting*, 50(3). Sept 2003.

[4] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, R. Morris, "Resilient Overlay Networks", *Proc. 18th ACM SOSP*, Oct 2001.

[5] J. Klaue, B. Rathke and A. Wolisz, "EvalVid - A Framework for Video Transmission and Quality Evaluation", *Proc. 13th International Conference on Modeling Techniques and Tools for Computer Performance Evaluation*, IL, Sept 2003.

[6] `http://graphics.cs.msu.su/` Video Evaluation Toolkit (Moscow State University, Moscow, Russia). Available for free download.

[7] `www.isi.edu/nsnam/ns`. The Network Simulator v2.30.

[8] R. Koenen, "Overview of the MPEG-4 Standard", *ISO IEC JTCI.SC29/WG11 M4030,* 2001.