IJICTR International Journal of Information & Communication Technology Research

ITRC

Volume 5- Number 4- Autumn 2013 (27-35)

A New Method for Evaluating Video Quality of Experience on Content-Aware Packet Loss Effect Analysis

Amir Hossein Mandegar
Department of Electrical, IT &Computer Engineering,
Qazvin Branch, Islamic Azad University
Qazvin, IRAN
amandegar@qiau.ac.ir

Behzad Akbari
Department of Electrical & Computer Engineering,
Tarbiat Modares University
Tehran, IRAN
b.akbari@modares.ac.ir

Received:February 6, 2013- Accepted:July 27, 2013

***Abstract*—According to the popularity of variant video streaming applications over the Internet, video quality assessment is essential for video solutions and network operators in order to evaluate video stream quality and measure the end-user Quality of Experience. Packet loss affects both the decoder performance and the user's perception of video quality due to several network issues. The main goal of this paper is to estimate the perceived video quality at the receiver's side by passive measurement in the network and also evaluate how errors propagate through temporal prediction. Our proposed method is based on the network level parameters by considering the video codec and frame information. Our experimental analysis shows the accuracy and efficiency of the proposed method.**

Video Quality Estimation; Passive Traffic Measurement; Packet Lost Evaluation;

I. INTRODUCTION

Nowadays, Video streaming applications are growing fast and significant part of the Internet traffic are dedicated to various video services. Therefore, it is essential for service providers to give undivided attention to the traffic management and quality assurance and find efficient approaches to improve their measurement techniques and reduce their processing overheads. Packet loss is a common phenomenon in IP networks. Since video streams are very sensitive to network packet loss due to the interdependency of video packets, it is important to measure and estimate the effect of packet loss on quality of video streams, for example to provision service quality of experience and users' satisfaction by the network operators. Moreover, the estimation is useful to measure the relationship between the network Quality of Service (QoS) and user Quality of Experience (QoE), which might be complex.

Video quality measurement is a classic subject in which numerous researches have been undertaken during the recent years to devise more advanced methods to satisfy the essential requirements. In general, the methods of video quality measurement are divided into two major category; Subjective and Objective. The former category relies on the human perception and it based on the real time statistical information gathered from end-users' feedbacks. Although objective methods produce accurate results, they are usually time consuming, costly and require huge human resources for the data collection and estimation. Due to these problems the latter category was introduced to propose some methods based on statistical and mathematical models to process the data and infer the results. Although each method`s category has own peculiarity, the objective methods are more popular and useful than subjective ones. For example, ITU-T p.800 was introduced as a quality assessment standard for subjective methods in 1996[12] and well-known methods such as PSNR, SSIM and PEVQ are introduced as objective methods. In Addition, Objective methods are divided into three sub categories: Full-Reference, Reduced-Reference and

No-Reference - which have different accessibility to the original video. Full-Reference methods consist of methods with access to the original video and are usually suitable for laboratory experiments. In contrast, No-Reference methods are independent of original video and are the most applicable methods for real networks. Reduced-Reference methods are obviously between these two groups and they have partially access to the original video information. In other word, at the processing point some information about the original video are accessible to Reduced-Reference`s methods.

To measure the packet lost effect, the authors in [1] have proposed a model to calculate the degradation and error propagation effect on video quality while taking temporal prediction into account. They work on typical Digital Video Broadcasting (DVB) IPTV traffic in the network which is available at the transport level without any needs to decode the streams. The rewrapper used for preprocessing, simplifies the network processing and its reorders, and also encapsulates TS video to RTP packets. This study proposed the packet loss effect prediction (PLEP) model based on RTP deep packet inspection, which measures the fraction of every affected frames by artifacts resulted from packet losses. They first calculate the degradation value then estimate the error propagation. The work in [3] provides an overview of the most relevant challenges to perform Quality of Experience (QoE) assessment in IP networks, and highlights the particular considerations needed during comparison with the alternative mechanisms already deployed. In this paper, the authors covered different available approaches for the QoE evaluation of the delivered multimedia. In addition, they described the most useful techniques and metrics as well as the challenges in designing of an IPTV QoE assessment platform. The authors in [2] have studied the packet loss effect and visibility according to the generalized linear model which uses some video and bit-stream features. They model the probability of the visibility which is the type of the logistic regression and an extension of the classical linear model. Reibmanand et al. in [4] study the effect of packet loss by using a model to estimate the effect in terms of Mean Square Error. They specifically addressed MPEG-2 videos and find a quite significant correlation between MSE and relatively a high packet loss value.

In this paper, we estimate packet loss effect on video streams by passive measurement which is categorized as a subjective method. We consider the packets' contents based on the frames types and temporal estimation in each Group of Pictures (GOP). Since different frames types in a compressed video (i.e. I, P and B) have different impacts on the quality of the perceived video at the receivers, we propose a method to measure error propagation and estimate frame type destruction effect that will be used for packet loss impact analysis. Although the proposed method considers more parameters, it has a low processing overhead (it is completely based on the network level information) while providing suitable accuracy. Moreover, it is completely passive and, therefore, does not induce any extra traffic into the network and also is a fully No-Reference method. Hence, it is suitable to be used at any place inside the network and thus it can be a valuable method for network providers and also for video service providers to evaluate their quality assurance level.

The remaining of this paper has been organized as following: section II will describe our adopted assumptions. A new metric has been introduced in section III, which explains the main idea and formula that contains the mathematical and analytical method to achieve the result. The section IV provides the experimental results and their analysis. Finally, we conclude the paper in the last section.

II. Assumptions

The analysis described in this paper was performed on the common video streams. We extract required information from network level traffic without any need to decode the video stream. Although, the proposed scheme is related to the video compression and encoding theory, the proposed method and the result is fully applicable to any type of video streaming system, scenario and application. We focus on the stream's packet information which is available in the transport level [8].

It is assumed that the video stream uses UDP/RTP/RFC2250 [7] to transport the packets on the network. RTP is one of the most common protocols in transporting video streams. The RTP protocol separates video and audio in two different parallel streams and sends bidirectional feedbacks and statistics to both sides of a connection [6]. In this study, we use an RTP sequence number to find the lost packets in streams and then calculate the effect of the lost packets on the video quality with regard to the video GOP frames awareness. RFC2250 [7] defines RTP payload which carries important information of the video stream. This RTP payload makes it possible to extract the frame information such as the frame type and the last packet of the each frame. The RTP and mentioned payload information are shown in Fig. 1. The main fields of a RTP packet are as following in which the fields S, E and P are the most useful in our proposed method:

- MBZ: Unused.
- T: MPEG-2 specific header extension.
- TR: Temporal-Reference.
- AN: Active N bit for error resilience.
- N: New picture header.
- S: Sequence-header-present.
- B: Beginning-of-slice.
- E: End-of-slice (ES).
- P: Picture-Type.
- FBV: Full pel backward vector.
- BFC: backward f code.
- FFV: Full pel forward vector.
- FFC: Forward f code.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|V=2|P|X|  CC   |M|     PT      |       sequence number         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           timestamp                           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           synchronization source (SSRC) identifier            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    MBZ  |T|         TR        | |N|S|B|E|  P  | | BFC | | FFC |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                AN              FBV     FFV
```

Figure 1. Real Time Protocol and RFC2250 header elements [6, 7]

RTP sequence number(S) field is used to detect and count the number of the lost packets in a video stream. Gap detection method counts how many packets are lost in a video stream by calculating the sequence number difference and the gap space between them [5]. To calculate packet lost ratio based on GOP, gap detection counts the number of the lost packets from the start of each GOP up to the end. Usually the numbers of lost packets are measured in periods of time. However, in our proposed method, we use the GOP length to evaluate the packet loss effect by the end of each GOP. A GOP is a fixed set of frames coming one after another in a specific order inside a stream. This sequence of frames is repeated in a video stream session. The first frame of each GOP is an I-frame, which is intra-coded and against the error propagation along the stream. I-frame plays a reference frame`s role for each GOP in a compressed video stream. These information are critical for estimation and our proposed estimation based on the frame worthiness which is derived from the GOP structure and its capability in error propagation.

We also assumed random packet loss in the network, that is difficult to evaluate and estimate quality degradation in compare to burst lost.

III. Packet Loss Effect Measurement

In this section we introduce a metric called Packet Loss Effect in order to study the impact of network packet loss on the experienced video quality by the receivers. Packet loss effect is a network level and frame aware metric that estimates the destructive effect of lost packets in a video stream. Equation (1) suggests how Packet Loss Effect should be estimated in a video stream based on the structure of the frames and GOPs inside the stream which can be extracted from the packets` payload. The required parameters which must be calculated for each GOP are as following:

GOP_{pkt} : Total number of packets in the GOP

$Frame_{pkt}$: Number of packets of a "Frame" type

$Frame_{qty}$: Quantity of each "Frame" type

GOP structure of any video stream is extracted from the first correctly received GOP (without any packet lost) with regards to the I-frame type. GOP_{pkt} shows the total number of packets in one GOP. $Frame_{pkt}$ shows the number of packets for a specific frame type in the GOP and $Frame_{qty}$ shows the quantity of a specific frame type in the GOP. For example, Table I shows the structural information of Coastguard video stream. It has 23 packets in one GOP consisting of 1 I-frame, 4 P-frames and 10 B-frames. Each of these frames occupies 5, 8 and 10 packets in a GOP respectively. In addition the sample video stream has IBBPBBPBBPBBPBB GOP structure which is a typical structure for video compression for streaming over network (Fig. 2). This information is obtained from video stream and used to estimate the loss effect.

TABLE I. Coastguard video structural information

Frame type	Quantity	Packets
I	1	5
P	4	8
B	10	10
GOP packets = 23 packets		

Equation (1) is proposed for Packet Loss Effect estimation. This is a new approach which consists of the probability of packet loss and their perceptual effect on the video stream to obtain the quality degradation. In this equation, P(I) is the probability occurrence of I-frame, and obtained from dividing the number of I-frame's packets by the total number of the GOP packets. P(P) and P(B) are the same as the P(I) for frame types P and B. P(x) in Equation (2) is the general equation which shows occurrence probability of frame type x and must be calculated for all three types of frames. $Loss_{pkt}$ is the number of the lost packets measured in a GOP.

$$
\begin{aligned}
e(Loss) = {} & P(I) \times E(I) \times Loss_{pkt} \\
& P(P) \times E(P) \times Loss_{pkt} \\
& + P(B) \times E(B) \times Loss_{pkt}
\end{aligned}
\qquad (1)
$$

E(x) for a given frame type denotes the average number of frames in the GOP that affected by loss of frame type x (x= I, P and B). This value is calculated according to Equation (2).

$$
\begin{aligned}
& P(x) = \frac{frame(x)_{pkt}}{GOP_{pkt}} \\
& E(I) = I_{qty} + P_{qty} + B_{qty} \\
& E(B) = 1
\end{aligned}
\qquad (2)
$$

Since, frame type I is a reference frame for the GOP, its loss affects all frames in the GOP. In contrast, the loss of a frame type B has no any impact on other frame except itself. Frames types P have only impact on following frames in the frame dependency graph (Fig. 2(b)). As can be seen in equation (1) by multiplying frame occurrence probabilities P(x) by the corresponding effect value E(x) and the number of lost packets in the GOP, Packet Loss Effects of each frame type in a GOP is calculated. Aggregation of all the values for each frame type gives the Packet Loss Effect estimation, which is exactly the quality reduction magnitude.

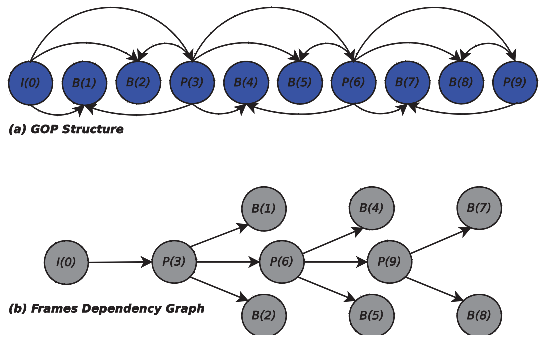

Figure 2. A typical GOP structure (a) and related decoding dependency graph (b)

According to the dependency graph of a GOP (Fig. 2(b)), I-frame is the most valuable frame type inside a GOP and it carries the full data of the picture, consequently any damage to I-frame affects all the other frames of the GOP. Therefore, all the frame types inside a GOP are strongly dependent on I-frames, while these frames are completely independent of the others. In Fig. 2(b) all frames of a given GOP directly or indirectly dependent on frame I. E(I) introduces the effect of this type of frame by the total number of I, P and B-frames due to the dependency graph. Then P-Frames are the most important type of video stream frames. Whenever a P-frame is lost, all the next P-frames, next and previous adjacent B-frames will be affected. Therefore, any damage to each P-frame makes a chain reaction to the next and some of the previous frames which is strictly dependent on P-frames' positions in a GOP. Like I-Frame, dependency graph in Fig. 2(b) shows each P-Frames` dependent frames.

On the other hand, Fig. 3 visualizes how P-frames affect video quality in one GOP. In this example, the first P-frame affects 11 frames, the next P-frame affects 8 frames and the last one affects 5 frames. As mentioned before and can be seen in Fig. 2(b) any packet loss damaging the first P-frame affects all the next P-frames and adjacent B-frames. To obtain P-frame type effect E(P) on a GOP, average of the effects of all individual P-frames inside the GOP is considered.

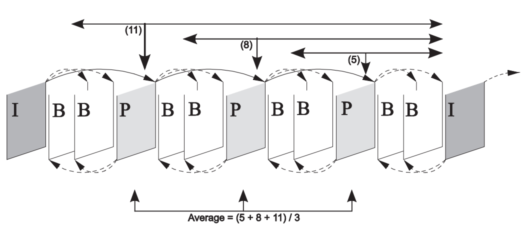

Figure 3. P-Frame Effect

The pseudo-code in Fig. 4 is used for calculating the effect of P-frames on a GOP by traversing the GOP structure from the last frame to the first frame and counting all individual P-frames and the adjacent B-frames.

Although B-frames have the least value inside a GOP in comparison to the other frame types, they are very useful to reduce the bit rate of a video stream. B-frames are the smallest frame type in a GOP, but they produce a high processing overhead and no other frames depend on them. Since the loss of B-frames has no impact on other frames, we set the effect value of B-frames to one (Equation (2)).

```
Begin
sum = 0;
count = 1;
Bcount = false;
for i= sizeof(GOP) To 1
    if frame[i-1] != B
     sum += count;
     count++;
endfor
E(P) = sum / P-Quantity;
End
```

Figure 4. P-frame effect algorithm pseudo-code

Although e(Loss) in Equation (1) reflects a good attempt to show the effect of packet loss in a GOP, Equation (3) illustrates how we need to incorporate other information extracted from the GOP before it can be applicable to video stream quality measurement. To do so, we first need to normalize the value of e(Loss) and then make it non-linear. The normalization can be achieved by dividing e(Loss) by the number of packets inside the GOP. The square root of the division makes it non-linear and results in a curve. Therefore, according to Equation (3), whenever the number of packets lost in a GOP increases, its destructive effect on video quality will accelerate.

$$LossEffect = \sqrt{\frac{e(Loss)}{GOP_{pkt}}} \tag{3}$$

Equation (3) is the final packet loss effect formula, and it will be evaluated in the next section for different situations and GOP structures. In comparison to the simple packet loss measurement without any GOP structure awareness, the proposed method is more accurate and will shows a more accurate reflection of video quality perceived by user. In addition, the awareness of the frame types and GOP structure makes it possible to measure and evaluate the effect of the identical packets lost in different video stream structures and understand the actual effect of each packet on the video quality.

IV. Experiment Results

In order to evaluate the proposed method to analyze the effect of packet loss on video quality, we first apply our method to a set of three video streams with the same GOP structure but different GOP lengths. Then we apply the method to another set of six video streams with the same GOP length but different structures and frame orders. Ultimately, we use three short sample videos such as Akiyo, Coastguard and Football from a dataset and evaluate them with Qt framework implemented software in the emulation environment based on the CORE[1] network emulator. Qt is a popular C++ multi-platform development framework with huge number of features and capabilities for rapid application development [10]. CORE is a suitable GNU/Linux based tool for network emulation which uses the para-virtualization and network name services and is capable of emulating hosts, routers and servers [9]. The evaluator software runs in the predesigned emulation scenario (Fig. 6) and captures all packets and only picks up those relevant to the process and video quality estimation.

Our proposed scheme has been implemented with Qt 4.7.4 framework, which is a multi-platform framework with brilliant capabilities for developers [10]. We use Qt containers, C++ Polymorphism technique and pcap library to develop the evaluator software. Fig. 5 shows the Evaluator software architecture which first captures packets from the network then distributes packets to relevant classes for processing, each classes are responsible for specific protocol analysis and produce results respectively. For instance, RTSP and RTP are the classes which are responsible for the proposed method and produce quality estimation data. Moreover, the Evaluator software is completely open source and source code is accessible for developers[2].

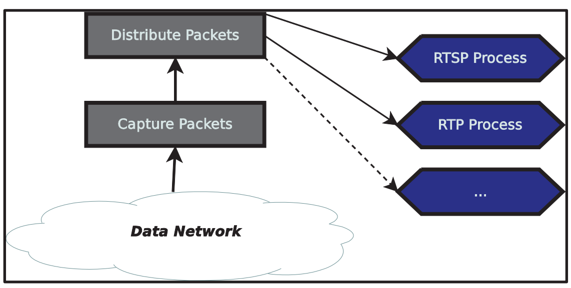

Figure 5. Diagram of Evaluator software

Fig. 6 illustrates the designed scenario which is used in the CORE emulator. Node n1 is a VOD(Video on Demand) server and node n6 is a customer who requests a video stream from the video source. Node n3 and n7 are used to generate continual background traffic while random packet loss occurs at the physical link between n4 and n5 which link produces 1, 2, 5 and 10 percent random packet error in different experiments. The Evaluator software listens to the network, captures the packets and processes the payload at router n5 and estimates video quality while packets are flowing. The RTP packets of each video stream session are processed and the results save into a log file for a further evaluation. Moreover, at the same time the video stream receives at node n6 and content saves into a file for further validation and comparison.

[1] Common Open Research Emulator (http://cs.itd.nrl.navy.mil/work/core/)

[2] https://github.com/amandegar/evaluator

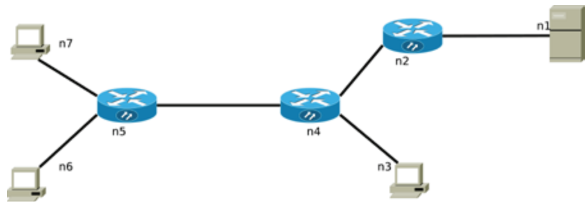

Figure 6. CORE Emulator Scenario

To establish the video on demand server, VLC is used as a VOD server and is configured to respond to the requests for the three mentioned videos. On the other side, MPlayer sends request to establish a new video stream session and receives video packets which some affect by random packet loss. As mentioned, MPlayer saves the received video into a file for further validation which will be described later.

Fig. 7 shows three graphs of the packet loss effect on the quality of the three video streams. The graph`s horizontal axis is the number of packets lost in the GOP and the vertical axis illustrates the result of loss effect. Each line shows the effect of packet loss by considering a different GOP length. The non-linear curve shapes of the graphs are due to the square root in Equation (3). As it was explained before, these three video streams have the same GOP structure but different number of packets inside a GOP due to the different frame size. In fact, these three video streams have a similar GOP structure containing 1 I-frame, 6 P-frames, and 10 B-frames. However, one of the video streams has 18 packets inside its GOP, the other 23 packets and the last one 35 packets inside their corresponding GOP. More packets inside a GOP means Lost Effect will grow more slowly.

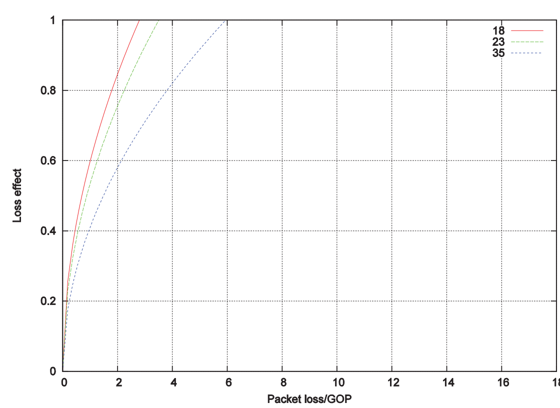

Figure 7. Video loss effect impact (Packet loss per GOP)

In addition, Fig. 8 illustrates a different presentation of the same data has shown in Fig. 7. The figure presents the loss effect based on packet loss ratio. As it can be seen, all samples almost follow each other and this behavior shows the appropriate estimation equation of packet loss impact on different lengths of GOP. Video quality measurement needs an equation which can estimate the packet loss impact without considering the GOP length and the number of packets in each GOP. Consequently, Fig. 8 presents the unique behavior of the proposed equation at the different GOP sizes.

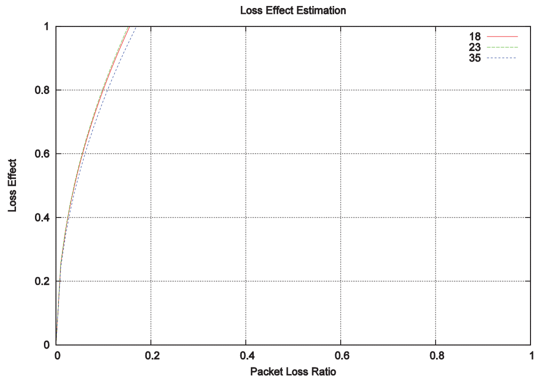

Figure 8. Video loss effect impact (Packet Loss Ratio)

Furthermore, six GOP samples with 15 packets in each GOP are used for more evaluation. Table 2 shows the six sample GOP structures and their total number of P-frames. Although, GOP samples 2, 3 and 4, 5 have the same number of P-frames, they have different frames order inside their GOP. Hence, the frames orders have an impact on the estimation because each frame type and their combination have different value and effect on the video stream.

TABLE II. SAMPLE GOP

Name	GOP	P-Quantity
Sample 1	IBBPBBPBBPBBPBBI	4
Sample 2	IBBBBPBBBBPBBBBI	2
Sample 3	IBBBBBPBBBBBPBBI	2
Sample 4	IPPPPBPPPPBPPPPI	12
Sample 5	IPPPPPPBPPPPPPBI	12
Sample 6	IPPPPPPPPPPPPPPI	14

Table 3 shows the results of applying our proposed method to the sets of six video streams in order to estimate the effect of packet loss on video quality. Each row indicates the corresponding GOP sample to which the method has been applied, the calculated P-Effect value on the GOP, the number of P-frames in the GOP and the value of e(Loss) for the GOP. It is clear from the table that the decrease in the number of P-Frames will results in an increase in Loss Effect. In addition, any error occurrence in GOP with a lower number of P-frames can make a bigger destructive effect on P-Frames due to the interdependency of frames. In case of P-Frame loss, more frames are affected. Consequently, this is the reason for higher value of P-Effect whenever there is less P-Frames in a GOP. On the other hand, a lower number of P-Frame reduces the probability of the lost occurrence. As the result, the value of the Loss Effect is against the number of P-frames. It can be observed in the results of table 3 in samples like 1 and 6 which higher number of the P-Frame produces lower value of P-Effect with a higher value of Loss Effect. The proposed method analyzes the effect of each frame type and its value inside its GOP and produces a different result by considering the frame order and neighboring. As mentioned before, the results depend on how many frames will be affected by damage which has visualized in Fig. 2(b) as a dependency graph.

TABLE III. SAMPLE'S GOP EVALUATION

GOP	P-Effect	P-Quantity	e(Loss)
Sample 1	9.5	4	2.53
Sample 2	11.5	2	1.53
Sample 3	11	2	1.47
Sample 4	7.6	12	6.08
Sample 5	8.08	12	6.46
Sample 6	7.5	14	7

V. RESULTS ANALYSIS

As mentioned before, three different short sample videos have been evaluated and their results have been saved into files. Results consist of the values of the estimated video stream quality and received video stream data at the end-node. To validate the results of the experiments, the well-known and Full-Reference objective method (SSIM[3][11]) is used and therefore, it requires both original and received videos. SSIM considers the structural similarities of pictures and improves the results of the traditional methods such as PSNR and MSE. It considers more parameters and also image structure which other methods like PSNR does not consider. The QPSNR software [13] is used for validating the results of the video quality estimation which is a multi-thread implementation of the PSNR and SSIM methods.

Fig. 9 illustrates the results of the proposed method and corresponding SSIM for all the three videos with 5 percent packet loss at the communication link. The main purpose of this comparison is to realize similarities between the results of both methods. As mentioned, we need to compare the results of the proposed method with a well-known method to proof its correctness.

We did the same experiment for 10 percent packet loss in the communication link and plotted the results in Fig. 10. Same as the results in Fig. 9, the evaluated measure by our proposed method shows very similar behavior to that of SSIM method.

[3] Structural Similarity Method

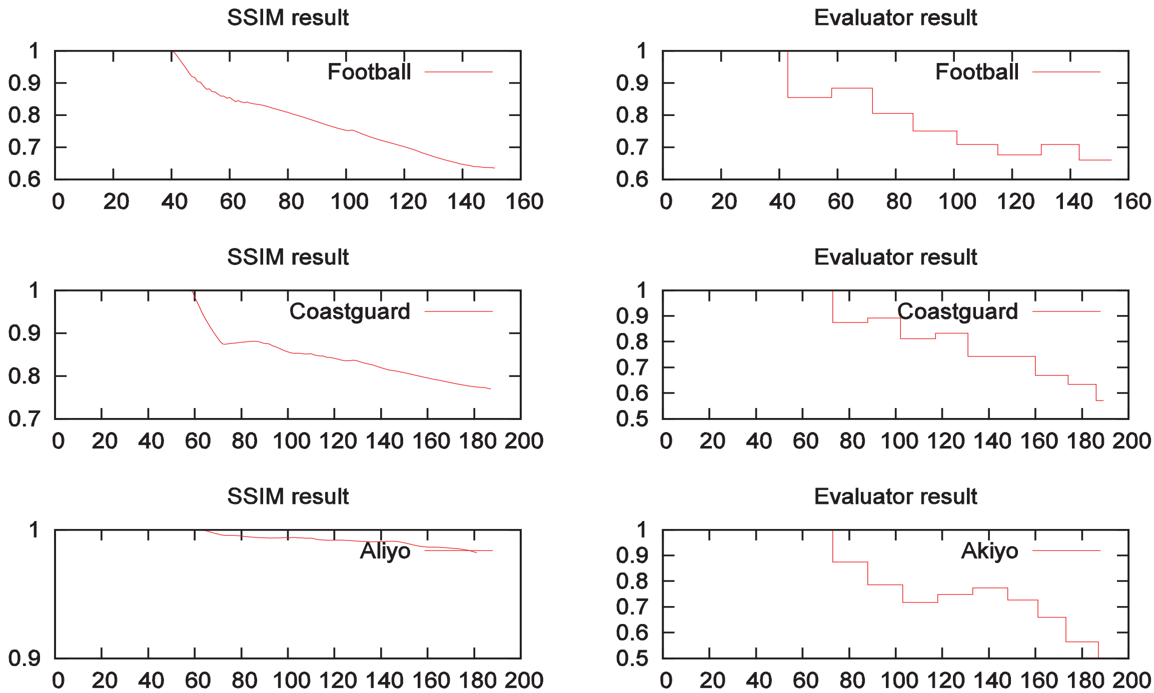

Figure 9. The proposed evaluator and SSIM (PER=5%)

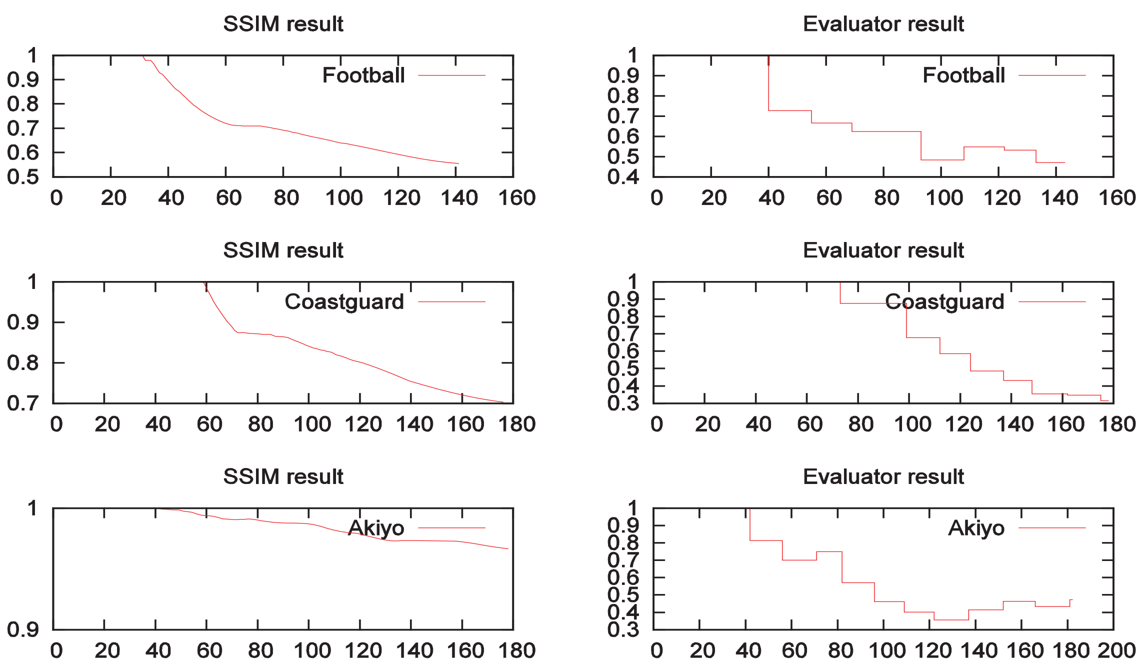

Figure 10. The proposed evaluator and SSIM (PER=10%)

In addition, Fig. 11 presents the results of different evaluation scenario. In this examination, Coastguard video (Medium dynamicity) quality is estimated for various bitrates. The goal is to study the packet loss effect on a video stream with different bitrates. Like Fig. 9 and 10, Fig. 11 illustrates similar behavior in video quality estimated by the proposed method and SSIM.

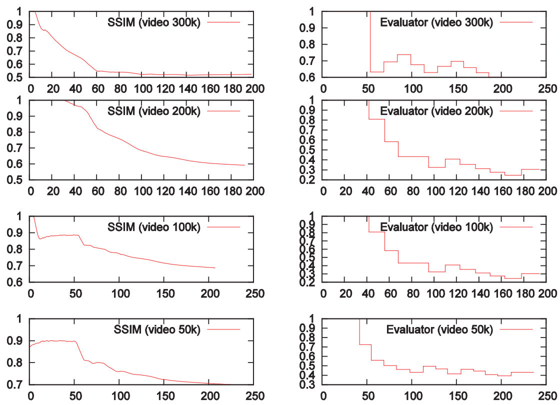

Figure 11. The proposed evaluator and SSIM results for different bitrate(Coastguard with PER=5%)

Since the analysis provides positive results, then a quantitative analysis is conducted to calculate the accuracy and mathematical validation [14]. Along the goal, Root Mean Square Estimation (RMSE) is used to measure the variance between the actual and predicted values. Similarly, the objective is to compare the experiment and SSIM results to demonstrate the similar behavior and accuracy with numerical.

The main challenge in analyzing the data and comparing the two sets of the results is that the units of the quality assessment in SSIM and the estimation are different. To resolve this problem, the coefficient of variation is calculated for both, hence both category of data are normalized into the same domain to provide uniformity in the comparison and evaluation phase.

Equation 4 is a coefficient of variation formula which obtained from dividing the deviation value by the average value. Coefficient of variation provides appropriate values for calculating the margin of error and correlation value.

$$\mathrm{c.v.} = \frac{\sigma}{\mu} \tag{4}$$

On the other hand, RMSE is a popular method that is used to calculate the error between two sets of data and thus it is applied here to the evaluator and SSIM results to demonstrate accuracy. To calculate C.V. for each received frames, we consider the experiment's and SSIM results from the first GOP and determine the differences and errors with equation 5. This is the residual and typical error of model fitting [15].

$$\mathrm{RMSE}(x_1, x_2) = \sqrt{\frac{\sum_{i=1}^{n}(x_{1,i} - x_{2,i})^2}{n}} \tag{5}$$

As mentioned we used three videos Akiyo, Coastguard and Football which streamed in the emulated network and are evaluated by the proposed metrics. Table 4 is the outcomes of C.V.`s RMSE based on the results.

TABLE IV. TABLE IV. RMSE OF COEFFICIENT OF VARIATION (SSIM VS. EVALUATOR

PER	Akiyo	Coastguard	Football
1%	0.00139938	0.00275618	0.0526343
2%	0.0534977	0.0473584	0.0435976
5%	0.0126089	0.0639604	0.102745
10%	0.0816338	0.0540812	0.0627408

First column is the packet error rates in the communication link and the next columns present the RMSE`s resulted from streaming different video sequences. As we can see in this table, the results show more similarity between the results of the proposed method and SSIM.

Moreover, Table 5 presents the C.V.`s RMSE for Coastguard sequence with different bitrates and %5 packet error rate. . As the results in this table shows, the accuracy is of the proposed method is suitable for different bitrates.

TABLE V. COASTGUARD - PER=5%

Bitrate	C.V`s RMSE
50k	0.0615154
100k	0.0266851
200k	0.0648231
300k	0.0610942

In summary, the proposed method is a No-Reference method with low processing overhead and it has the same estimation behavior as SSIM. The experimental results show and validate the correctness of our proposed method. The proposed method has some preferences over the other methods. For example, it is capable of producing real time results and does not need the original video. Therefore, it is suitable for real time monitoring multiple sessions. Its results are completely based on the network level information and processing only the transport layer data without decoding video bit streams. Hence, this method can be very useful for quality assessment of video streams transported through a service provider network.

VI. CONCLUSION

In this paper, we studied the effect of packet loss on video stream quality and proposed a No-Reference and efficient method for video streaming quality assessment. This method has low processing overhead to estimate the packet loss effect by incorporating some new features that are extracted from video traffic in transport layer. It is more suitable for video service providers and also network operators to realize the effect of the network faults and congestion on the user perception. This approach is important for measurement of Quality of Experience, based on the network level measurement without any requirement to decode the video bit-stream.

Although our experiments shows the efficiency of the proposed method for estimation video quality by network level information, there are still some other issues remain to be considered, such as the complex pattern of loss and additional artifact for more accuracy. Our future work presents a combination of Loss Effect and another metric to increase the accuracy of video quality estimation. We are working on another metric to account the effects of pictures` motion. At the same packet lost condition on different sessions, various video motions have diverse effects on the end-user perception which is the goal of our future study.

ACKNOWLEDGEMENT

The authors would like to thank Mehdi Katanbaf and Precious Jasper De Leon for their contribution to improve the linguisticquality ofthis paper.

REFERENCES

[1] Pablo Perez, Jesus Macias, Jaime J. Ruiz, and Narciso Garcia, "Effect of Packet Loss in Video Quality of Experience," Bell Labs Technical Journal, issue 16(1), pp. 91-104, 2011.

[2] S. Kanumuri, S. G. Subramanian, P. C. Cosman, and A. R. Reibman, "Predicting H.264 Packet Loss Visibility Using a Generalized Linear Model," Proc. IEEE Internat. Conf. on Image Process. (ICIP â˜06), pp. 2245â"2248, 2006.

[3] R. Serral-Graci et. av., "An Overview of Quality of Experience Measurement Challenges for Video Applications in IP Networks," LNCS, 6074, pp. 252-263, 2010.

[4] A. R. Reibman, V. Vaishampayan, Y. Sermadevi, "Quality Monitoring of Video Over a Packet Network," IEEE Trans. Multimedia, vol. 6, no. 2, pp. 327-334, Apr. 2004.

[5] P. Calyam, E. Ekici, C. Lee, M. Haffner, N, Howes, "A GAP-Model based Framework for Online VOIP QoE Measurement," Journal of communications and networks, Vol. 9, Issue 4, pp. 446-56, 2007.

[6] RFC 3550, "RTP: A Transport Protocol for Real-Time Applications," [Online] Available: http://www.ietf.org/rfc/rfc3550.txt , 2003.

[7] RFC 2250, "RTP Payload Format for MPEG1/MPEG2 Video," [Online] Available: http://www.ietf.org/rfc/rfc2250.txt , 1998.

[8] F. Fitzek, P. Seeling, and M. Reisslein,"Video Streaming in Wireless Internet," CRC Press, 2004.

[9] J. Ahrenholz, C. Danilov, T. Henderson, and J.H. Kim, CORE: A real-time network emulator, Proceedings of IEEE MILCOM Conference, 2008. CORE emulation.

[10] QT(4.7.4) , "Cross Platform Application and UI Framework", http://qt.nokia.com/, retrieved November 2011.

[11] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli,"Image quality assessment: From error visibility to structural similarity", IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, April 2004.

[12] ITU-R Recommendations P.800, "Methods for subjective determination of transmission quality", International Telecommunication Union, Geneva, Switzerland, 1996.

[13] "qpsnr, http://qpsnr.youlink.org/, retrieved December 2011.

[14] A. Hore, D. Ziou,"Image Quality Metrics: PSNR vs. SSIM", 20th International Conference on Pattern Recognition (ICPR), 2010.

[15] M. Khamkong, "Goodness of Fit Test for Poisson Distribution based on Higher Moment", The 11th Statistics and Applied Statistics Conference, 2010.

Amir Hossein Mandegar received his B.Sc. and M.Sc. degrees in Computer Engineering and IT Engineering from Qazvin Islamic Azad University, Qazvin, IRAN, in 2005 and 2012, respectively. He was amember of IAU Young Research Club, Mechatronics Research Laboratory (MRL), Technical and Organizing committee of Iran Open Robocop competitions. He is interested in data science and analytics, traffic measurement and performance evaluation, network QoS and QoE, big data computing, parallel and distributed processing, as well as internet of things and UAV.

Behzad Akbari received his B.Sc., M.Sc. and Ph.D. degrees in Computer Engineering from Sharif University of Technology, Tehran, Iran, in 1999, 2002 and 2007, respectively. He joined department of Electrical & Computer Engineering at Tarbiat Modares University, Tehran, Iran as an assistant professor in 2007. His main research interests are Internet QoS, media streaming over the Internet, peer-to-peer networks, peer-to-peer video streaming, wireless video communications, data center networking, software defined networks (SDN) and network performance modeling & evaluation as well.