

© 2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted of this work in other works.

I. Sedano, G. Prieto, K. Brunnström, M. Kihl and J. Montalban, "Application of full-reference video quality metrics in IPTV," 2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Cagliari, 2017, pp. 1-4.

doi: 10.1109/BMSB.2017.7986191

URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7986191&isnumber=7986123>

Application of full-reference video quality metrics in IPTV

Iñigo Sedano

Instrumentation & Smart Systems
Tecnalia Research & Innovation
Derio, Spain
inigo.sedano@tecnalia.com

Gorka Prieto,

Department of Communications Engineering (Faculty of
Engineering of Bilbao)
UPV/EHU
Bilbao, Spain
gorka.prieto@ehu.eus

Kjell Brunnström

Visual Media Quality
RISE Acreo AB and Mid Sweden University
Kista, Sweden and Sundsvall, Sweden
kjell.brunnstrom@ri.se

Maria Kihl

Department of Electrical and Information Technology
Lund University
Sweden
maria.kihl@eit.lth.se

Jon Montalban

Department of Communications Engineering (Faculty of
Engineering of Bilbao)
UPV/EHU
Bilbao, Spain
jon.montalban@ehu.eus

Abstract— Executing an accurate full-reference metric such as VQM can take minutes in an average computer for just one user. Therefore, it can be unfeasible to analyze all the videos received by users in an IPTV network for example consisting of 10.000 users using a single computer running the VQM metric. One solution can be to use a lightweight no-reference metrics in addition to the full-reference metric mentioned. Lightweight no-reference metrics can be used for discarding potential situations to evaluate because they are accurate enough for that task, and then the full-reference metric VQM

can be used when more accuracy is needed. The work in this paper is focused on determining the maximum number of situations/users that can be analyzed simultaneously using the VQM metric in a computer with good performance. The full-reference metric is applied on the transmitter using a method specified in the recommendation ITU BT.1789. The best performance achieved was 112.8 seconds per process.

Keywords—IPTV & Internet TV, Performance evaluation, Objective evaluation techniques, QoE

I. INTRODUCTION

Video quality assessment has come a long way [1][2]. Analyzing quality of experience is interesting because it enables the optimization of resources in IPTV networks and the correct management of problems in the service delivery. Usually subjective video quality assessment is considered the most accurate method, but subjective assessments are also considered expensive and time consuming. Therefore, many objective video quality metrics have been proposed and developed. These methods are usually divided into three groups depending on how much reference information they are using in the processing. Full-Reference (FR) metric requires a high-quality video of the same size as video under test i.e. two input videos, the reference and the distorted. Reduced-Reference (RR) metrics also needs two input videos in the way as the FR metrics, but a reduced representation is calculated from each of them and only this is transmitted to the other side. No-Reference (NR) methods, as the name suggests, do not use any reference video and therefore just need one video as input. Full-Reference metrics can be more accurate than Reduced-reference or No-Reference metrics due to the higher amount of information available.

Executing an accurate full-reference metric such as VQM [3] can take minutes in an average computer for just one user. Therefore, it can be unfeasible to analyze all the videos received by users in an IPTV network for example consisting of 10.000 users using a single computer running the VQM metric. Lightweight no-reference metrics [4] can be used for discarding potential situations to evaluate because they are accurate enough for that task, and where more accuracy is required the full-reference metric VQM could be used. The main work in this paper is focused on determining the number of situations/users that can be analyzed simultaneously using the VQM metric in a computer with good performance. The full-reference metric is applied on the transmitter using a method specified in the recommendation ITU BT.1789 “A method to reconstruct received video using transmission error information for packet video transmission” [5].

II. SCENARIO

The scenario of the recommendation ITU BT.1789 that was considered is shown in Figure 1. The head-end is composed of an encoder and a transmitter and the receiver is composed of a decoder. Additionally, for the video quality assessment in the transmitter, new blocks are needed.

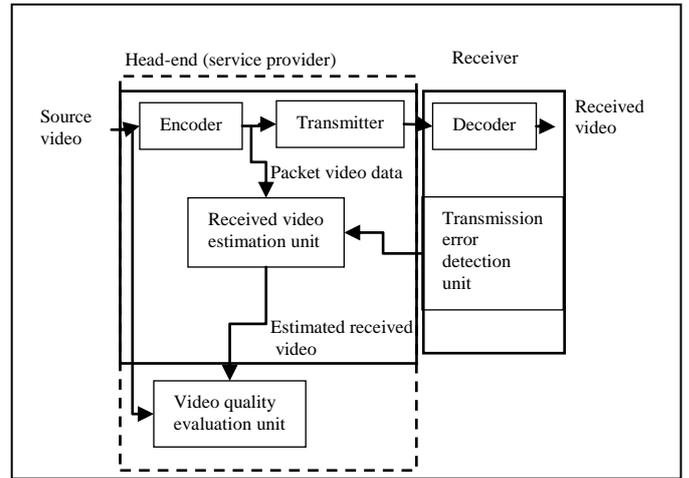


Figure 1 Scenario depicted in the recommendation ITU BT.1789

The transmission error detection unit in the receiver detects the numbers of the lost packets and sends that information to the received video estimation unit, which is part of the transmitter. With that information, the received video estimation unit can reconstruct the degraded video, which was shown to the user in the receiver. Finally, the video quality evaluation unit, which is a full-reference metric, compares the degraded and non-degraded video and produces a final assessment on the video quality that the user perceived.

This scenario is further detailed in Figure 2. The orange blocks in the figure represent what is present in a normal transmitter and receiver. The blue blocks in the figure are all related to the full-reference evaluation by VQM.

In [4], VQM was found to perform better than Peak Signal-to-Noise Ratio, Structural Similarity [6], Multi-scale SSIM [7], Visual Signal-to-Noise Ratio [8] and MOTion-based Video Integrity Evaluation [9]. Therefore, we have selected VQM for our evaluation. A version of VQM with support for 64 bits was used and the model parameter was set to general model. The operating system used in all cases was Windows 10.

The scripts used to create the different processes and start the simulator, decoder and metric were programmed in Python 3 language.

The simulator is a tool that takes as input an H.264 file and a pattern of zeros and ones representing which slices must be lost and generates as output another H.264 file that doesn't contain the slices to be lost [10]. The generation of the pattern of zeros and ones is done using another python program in which we specify the packet loss percentage.

The decoder used was ffmpeg that now support the presence of lost packets and conceals the errors as best as possible. In all the studied cases, ffmpeg generated a decoded video of the length expected (same length as the original video). This fact is important, because if the videos do not have the same length it would be problematic to apply the full-reference metric in a correct way.

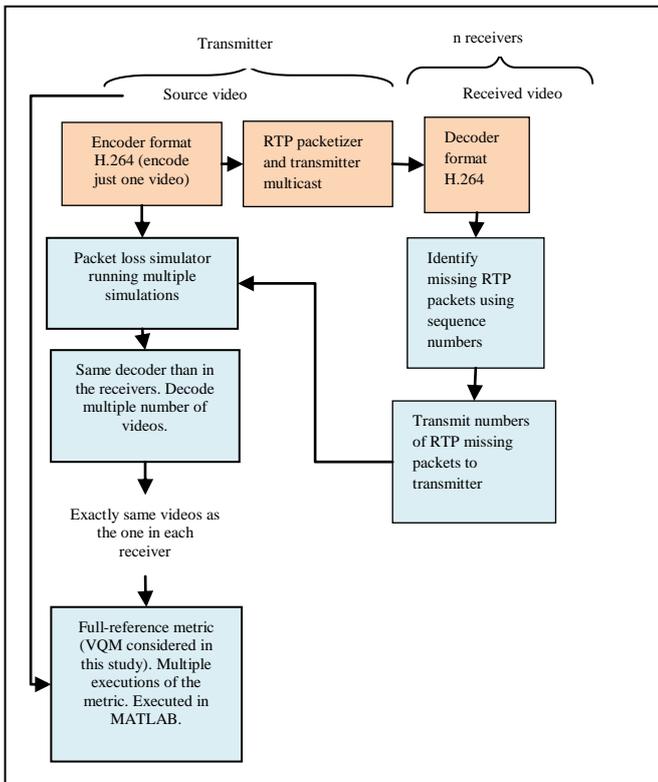


Figure 2 Scenario considered for practical evaluation.

The video had a length of 10 seconds and a resolution of 1920x1280.

III. RESOURCE CONSUMPTION

The VQM metric is the part that consumes most of the time when compared with the time needed by the decoder and the packet loss simulator. Therefore, in this study the focus is resources used by the VQM metric. It is of interest because the time needed by the VQM metric could be reduced if the bottleneck point is improved. The results are shown in Table I.

In the execution of the VQM metric in multiple processes, the bottleneck is in the CPU.

IV. TIME CONSUMPTION

This evaluation consists on analyzing the performance of running multiple processes, each of the processes composed of three parts that are executed consecutively:

- One packet loss simulation that introduces the same loss pattern that was detected in the receiver. The packet loss simulator [10] takes as input a file with a pattern of zeros and ones that indicate which slices are lost and which slices are kept in the encoded file.

- Decoding of the generated file by the simulation.

- Computation of the full-reference metric VQM on the decoded degraded video together with the original video as reference.

TABLE I.

RESOURCES VQM METRIC - DISK SSD - i7 - 8 cores - MEMORY RAM 32 GB (Packet loss percentage 1%)					
NUMBER OF SIMULTANEOUS PROCESSES	MEMORY (%)	CPU (%)	DISK TRANSFER (%)		
1	6	23	2		
2	8	46	3		
5	13	93	9		
10	21	100	8	Saturated CPU	
20	33	100			
50	66	100	50		
100	90	100	30		

The decoder used was ffmpeg [11]. The video had a length of 10 seconds, a resolution of 1920x1280 and a medium degree of movement.

We run multiple processes because we are emulating the scenario where the transmitter of the IPTV network must assess simultaneously the quality of n receivers.

The results are shown in Table II.

For example, the case 10+60 would mean 10 simultaneous maximum processes and 60 processes started when other processes finish.

As it is shown in Figure 3, the ideal is to have always 10 simultaneous processes being executed.

We show the case of 10 simultaneous processes in Figure 4 using a tool to display results that we developed.

In Figure 4, each bar in the graph is one process started, and it is located exactly in the moment in which it was started. In the final part of each bar, the process finishes.

Most of the time is consumed by VQM (in green) as opposed to the time needed for the execution of the simulator of packet losses (in red) and the time needed for the execution of the decoder (in blue).

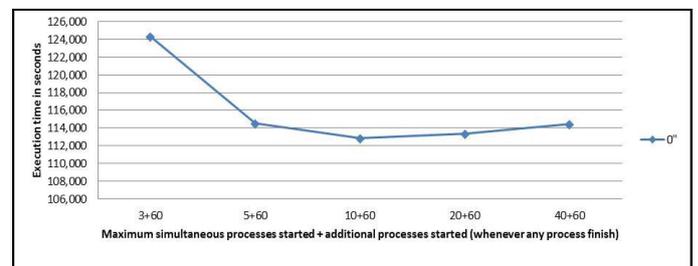


Figure 3 TIMINGS ITU-R BT1789 PER PROCESS IN SECONDS - Disk SSD - i7 - RAM MEMORY 32 GB - P+60 (the time between start of processes is 0 seconds)

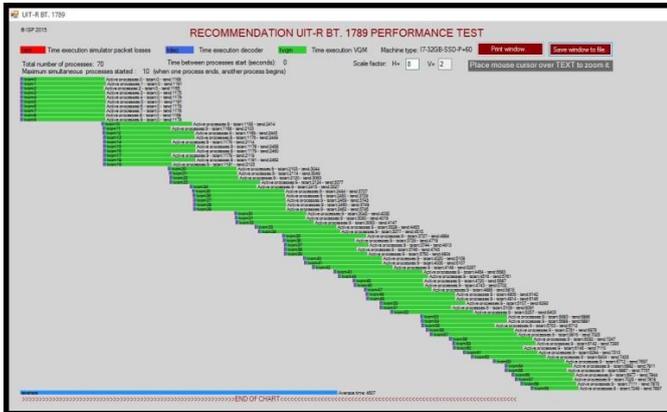


Figure 4 Best case ITU-R BT1789 - Disk SSD – i7 - RAM MEMORY 32 P+60, P=10

We can see that using this approach we arrive to a rather stable situation, where the number of simultaneous processes is stable, having therefore a stable situation regarding the consumption of CPU, RAM memory and disk transfer capacity. We select this case as the one that should be used in a real deployment.

V. CONCLUSION

The best performance achieved was 112.8 seconds per process. This performance is not enough for monitoring a network of 10.000 users using a single computer running the VQM metric. The solution is to use for example the packet loss rate information or any other no-reference or reduced-reference metric and only analyze the interesting situations for the IPTV provider (where there is doubt and interest) with the full-reference metric.

As future investigation, it would be interesting to explore how many users could be covered using other type of solutions, such as cloud computing, the use of GPUs or porting the code to other language different than MATLAB.

TABLE II.

TIMINGS ITU-R BT1789 IN SECONDS - Disk SSD - i7 - 8 cores - RAM MEMORY 32 GB - P+60		
<i>Total number of processes: Maximum simultaneous processes started + additional processes started (whenever any process finish)</i>	<i>Total time (seconds)</i>	<i>Average per process (seconds)</i>
3+60	7832,54	124,32
5+60	7445,07	114,54
10+60	7898,18	112,83
20+60	9071,80	113,39
40+60	11442,41	114,42
Average per process	115,58	

ACKNOWLEDGMENT

Iñigo Sedano would like to thanks Tecnalia Research & Innovation for their support and deceased Javier Sedano for his help running simulations. Maria Kihl is a member of the Lund Center for Control of Complex Engineering Systems (LCCC) and the Wallenberg Autonomous Systems and Software Program (WASP).

REFERENCES

- [1] S. Winkler and P. Mohandas, "The evolution of video quality measurement: From PSNR to hybrid metrics," *IEEE Trans. Broadcast.*, vol. 54, no. 3, pp. 660–668, 2008.
- [2] N. Aggarwal, "A Review on Video Quality Assessment," *Recent Adv. Eng. Comput. Sci.*, pp. 6–8, 2014.
- [3] "Video Quality Metric (VQM) software." [Online]. Available: <http://www.its.bldrdoc.gov/vqm/>. [Accessed: 01-Jan-2016].
- [4] I. Sedano, K. Brunnström, M. Kihl, and A. Aurelius, "Full-reference video quality metric assisted the development of no-reference bitstream video quality metrics for real-time network monitoring," *EURASIP J. Image Video Process.*, vol. 2014, no. 1, p. 4, 2014.
- [5] ITU-T, "Itu Bt.1789," 2007.
- [6] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.
- [7] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," *IEEE Asilomar Conf. Signals, Syst. Comput.*, vol. 2, pp. 9–13, 2003.
- [8] D. M. Chandler and S. S. Hemami, "VSNR: A wavelet-based visual signal-to-noise ratio for natural images," *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2284–2298, 2007.
- [9] K. Seshadrinathan and A. C. Bovik, "Motion tuned spatio-temporal quality assessment of natural videos," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 335–350, 2010.
- [10] Y.-K. W. Y. Guo, H. Li, "SVC/AVC loss simulator." [Online]. Available: http://wftp3.itu.int/av-arch/jvt-site/2005_10_Nice/. [Accessed: 01-Jan-2017].
- [11] "FFmpeg." [Online]. Available: <https://ffmpeg.org/>. [Accessed: 01-Jan-2017].