

Cognitive No-Reference Video Quality Assessment for Mobile Streaming Services

Maria Torres Vega*, Emanuele Giordano[†], Decebal Constantin Mocanu*,
Dian Tjondronegoro[‡], Antonio Liotta*

*Eindhoven University of Technology
Email: m.torres.vega, d.c.mocanu, a.liotta@tue.nl

[†]University of Padua
Email: emanuelegiordano99@gmail.com

[‡]Queensland University of Technology
Email: djan@qut.edu.au

Abstract—The evaluation of mobile streaming services, particularly in terms of delivered Quality of Experience (QoE), entails the use of automated methods (which excludes subjective QoE) that can be executed in real-time (i.e. without delaying the streaming process). This calls for lightweight algorithms that provide accurate results under considerable constraints. Starting from a low complexity no-reference objective algorithm for still images, in this work we contribute a new version that not only works for videos but, is general enough to adjust to a diverse range of video types while not significantly increasing the computational complexity. To achieve the necessary level of flexibility and computational efficiency, our method relies merely on information available at the client side and is equipped with a lightweight Artificial Neural Network which makes the algorithm independent from type of network or video. Its resource efficiency and generality make our method fit to be used in mobile streaming services. To prove the viability of our approach, we show a high level of correlation with the well-known full-reference method SSIM.

Keywords—No-Reference Quality of Experience, Neural Networks, Mobile Streaming Services, Network Quality Assessment.

I. INTRODUCTION

Mobile networks are exponentially growing in complexity as the number of interconnected devices does not cease to increase [1]. In this situation, assessing accurately and in real-time the network performance becomes fundamental [2] [3]. Traditionally, Quality-of-Service (QoS) based evaluations have been used for this purpose. However, factors like jitter, latency, packet loss or bitrate, while presenting statistics about the network behavior, cannot accurately assess how the unpredictable network impairments may affect the perception of the final beneficiary of these services, i.e. the user's Quality of Experience (QoE) [4] [5].

QoE is defined as the degree of delight or annoyance of the user of an application or service [6]. Due to its subjective essence, the legitimate judges of visual quality are the humans, whose opinion can be obtained through subjective analyses [7] [8]. In practice, presented stimuli (for

example impaired video sequences) are rated by subjects under controlled conditions [9]. These ratings express the subjective QoE (sQoE) described typically by the Mean Opinion Score (MOS). However, due to the time-consuming nature and bias of subjective experiments, in the last years, great effort has been placed onto developing objective quality metrics which could provide with a valid alternative, i.e. objective QoE (oQoE) [10] [11]. These can be broadly classified, depending on the amount of reference information required, in three categories: Full-Reference (FR), Reduced-Reference (RR) and No-Reference (NR). FR metrics perform a comparative assessment, and so access to the original (unimpaired) material in its entirety is necessary. Examples of this group are the image and video quality assessment methods Peak Signal to Noise Ratio (PSNR), based on the frame's Mean Square Error (MSE) value, and Structural Similarity (SSIM) [12], which combines comparisons in terms of luminance, contrast and structure for its quality assessment. RR metrics evaluate the material using a subset of features [13] [14]. Due to their need of partial or the totality of the original data both FR and RR have been proven not suitable for channel restricted environments, such as mobile networks [15]. Furthermore, their assessment complexity and time-consumption make them difficult to use for online network and service monitoring. To fill this gap, NR metrics appeared, due to the fact that they do not rely on comparisons but on absolute measurements of external factors and received data to model the oQoE. Thus, not requiring access to the original material and providing a faster analysis.

In this work, we present a novel NR-Video Quality Assessment method fit for automating oQoE analysis and that can work on lightweight mobile clients. We start our work from the method presented in [16], focusing on the problem of a generic calculation of the necessary weights of the original objective algorithm. To this end, we put the video bitstream through an Artificial Neural Network. We test our approach in a real-time, end-to-end video streaming scenario whereby a realistic network is emulated. We benchmark our method against the FR state-of-the-art metric SSIM due to its high correlation with the Human Visual System (HVS) [12]. We should stress, though, that SSIM would not be usable to carry out a real-time (online) evaluation of mobile streaming services (which is our main goal) since it would require the client to have access to both the original (unimpaired video) and the impaired stream under

Pre-printed version.

Please cite as: M. Torres Vega, E. Giordano, D. C. Mocanu, D. Tjondronegoro, A. Liotta, *Cognitive No-Reference Video Quality Assessment for Mobile Streaming Services, QoMEX, 26-29 May 2015, Costa Navarino (Greece)*.

scrutiny. Our key contribution is to demonstrate the viability of our NR-method, which is shown to correlate well with SSIM and is suited to mobile streaming scenarios.

The remainder of this paper is organized as follows. Section II provides an overview of the state-of-the-art on NR oQoE metrics for video quality assessment. Section III, presents our proposed method. The experimental test-bed and experiments performed are shown in section IV and section V, respectively. Finally, section VI draws conclusions, highlighting key contributions and suggesting directions for future work.

II. NO-REFERENCE OQOE METHODS FOR VIDEO QUALITY ASSESSMENT

Reibman et al. [17] classified NR approaches as either stemming from statistics derived from pixel-based features, NR pixel (NR-P), or computed directly from the coded bitstream, NR bitstream (NR-B). In a more recent classification, Shahid et al. [7] added to this classification a third category in which approaches combining both pixel and bitstream assessments are included, i.e. Hybrid NR-P-B metrics.

NR-P methods have focused their attention on the employment of certain artifacts related to a specific type of degradation of the visual quality. Blur, noise, blockiness, ringing or temporal impairments have been quantified for measuring the end-user’s quality. Early metrics made use of one artifact to assess the quality. However, these techniques have proven not to provide satisfactory performance in the assessment of the overall quality in the presence of more than one artifacts. Therefore, methods combining contributions of different artifacts appeared. These have given better results than single-feature based ones, but their usual high computational complexity and time consumption make them unfit for real-time assessment, particularly when it comes to mobile streaming services, where the client is computationally constrained. Examples of these approaches are the multivariate data analysis with extra downgrading quality [18] or the maximum a posteriori lost macroblocks pattern estimation [19].

NR-B methods are relatively simpler to compute and quality can often be obtained in the absence of the full decoder. But, they tend to have a limited scope of application as they are usually designed for specific coding techniques, such as H.264/AVC [20]. Another issue is to find the correlation among the bit-layer parameters and quality, without increasing the complexity. Learning tools have proven to be a promising solution for this type of approaches as well as for assessing QoE [21]. Shahid et al. [22] proposed a model combining different bitstream-layer features using an Artificial Neural Network to estimate the quality. However, their method focused on a very particular case, not providing results with sufficient general validity.

The performance of NR-B metrics can be enhanced by adding inputs in the form of NR-P based quality assessment, these methods are called hybrid methods. Shanableh et al. [23] suggest a multi-pass prediction system based on stepwise regression using features included in the coding information of a Macro Block (MB). This method has shown good correlation to SSIM but its complexity makes it not quite fit for online analysis. Keimel et al. in [24] measured quality by linearly combining bitstream and pixel related features.

Despite substantially reducing complexity, their approach does not, however, correlate well with either PSNR or SSIM, two of the FR state-of-the-art objective metrics in current use.

In this research, we present a Hybrid NR-P-B video quality assessment method. Consisting of a frame-based linear combination of noise and blur quality measurements, our method includes an Artificial Neural Network driven by the video bitstream information to select the appropriate weights. The low complexity and time consumption of our approach makes it suitable for real-time network evaluation.

III. NR-VIDEO QUALITY ASSESSMENT METHOD

This section presents our proposed NR method for video quality assessment, including its technicalities, assumptions and strengths.

A. Overview

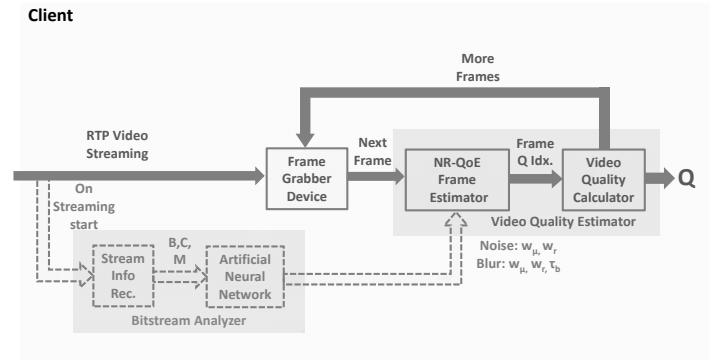


Fig. 1: NR-Video Quality Assessment Method Block Diagram

Figure 1 shows the block diagram of the proposed method. During a session, an RTP video streaming server starts transmitting a video to the client. On receiving the video header information, it is processed by the Bitstream Analyzer (BA) and the data required by the Video Quality Estimator (VQE) is obtained. This data will be used while the video is being streamed. Thus, this section is only entered once on session start. As frames from the streaming arrive, they are firstly grabbed by the Frame Grabber Device (FGD), which allows the possibility to change the rate of analyzed frames per video stream. This feature has been developed to ensure smooth and online quality assessment and it is to be set according to the client’s processor speed. Once the next frame to analyze is extracted, it enters the Video Quality Estimator (VQE). There, its quality is assessed and added to the overall video quality.

B. Video Quality Estimator

The video quality estimator (VQE) is carried out in two blocks, NR-Quality Frame Estimator (NFE) and Video Quality Calculator (VQC). The first estimates the frame quality based on pixel features. The later takes the frame quality derived by the first, includes it in the video average and requests the next frame to the FGD.

Hue, edge, noise, contrast and blur get affected when the image quality is degraded. However, it is typically assumed

that the most important factors for degradation are noise and blur. Based on this premise, Choi et al. designed a NR metric for image quality assessment in [16]. In the development of our method, we have followed up on Choi's work (which is limited to still images), adapting their algorithm to video streams and, also, making it independent from video type (thanks to the cognitive method described below). Equation 1 shows the quality metric as a linear combination of the contributions of blur, mean (μ_b) and ratio (r_b), and noise, mean (μ_n) and ratio (r_n), weighted by different constant values.

$$Q = 1 - (w_{\mu_b} * \mu_b + w_{r_b} * r_b + w_{\mu_n} * \mu_n + w_{r_n} * r_n) \quad (1)$$

The blur and noise components are derived using the procedures described in [16]. In order to obtain the blur contributions, the pixel value differences and means for the whole frame are firstly calculated. These values are used to derive the frame's edge pixels, which provide the decision on blurriness. Calculations are performed both in the pixel horizontal and vertical line. Second, the Inverse Blurriness Index is derived from the maximum between the vertical and horizontal blurriness and used for comparison with the blur threshold (Th_b) to decide the presence of blur in the pixel. Finally, the μ_b is derived as the sum of Inverse Blurriness Indexes divided by the number of blurry pixels in the frame and, the r_b , as the portion of edge pixels which have been considered blurry. Regarding the noise, as edge detection can be affected by it, the frames are pre-processed for noise filtering prior to edge detection. This is done by means of an averaging filter. Then, the edge pixels are detected on the filtered frame. Edge detection is performed following the same procedure as for blur. Once both horizontal and vertical edges have been spotted, pixel noise is obtained. The μ_n is then derived as the sum of all the noise values divided by the number of noisy pixels found and the r_n as portion of noisy pixels found in the whole frame.

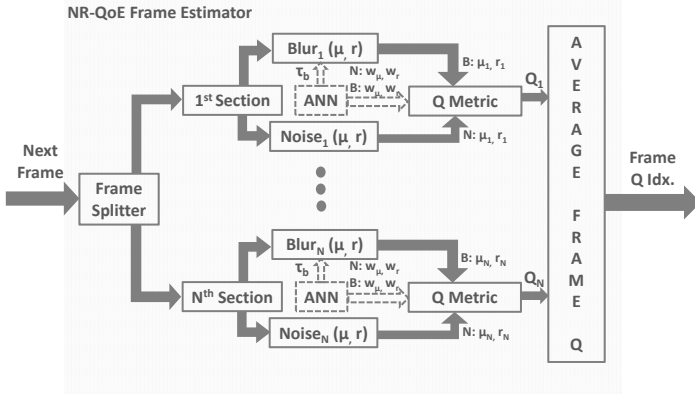


Fig. 2: NR-QoE Frame Estimator

Figure 2 shows how the NFE implements the previously explained metric. First of all, a frame is split in N blocks, which are analyzed in parallel threads in terms of blur and noise. The number of blocks N in which to split the frame has to be carefully selected in a way that it reduces the algorithm's running time while not significantly increasing its computational complexity. Thus, its value will depend on the type of processor and number of cores available in the

client device. Each section's blur and noise values are used to compute its quality, according to equation 1. Finally the frame quality index is obtained by averaging the results for all the frame blocks. This procedure improves the method's speed and makes it suitable for a real, online assessment.

C. Bitstream Analyzer

In order for the video quality to be accurately assessed, the constant weights w_{μ_b} , w_{r_b} , w_{μ_n} , w_{r_n} and blur threshold Th_b have to be adapted to the type of video stream, thus allowing to adjust to different combinations of spatial and temporal dynamics and obtaining a generic oQoE method. Inaccurate constant estimation can lead to wrong video quality index, and could, in the worst case, negatively bias the performance evaluation accuracy. The authors of [16] apply linear regression to derive the constants in the case of still images. However, videos, where encoding parameters have a significant non-linear influence on quality, require more complex learning techniques. The purpose of the BA is to set the weights and threshold to be used in the VQE, given the video stream encoding parameters. The analyzer is divided in two blocks (Figure 1): Stream Information Recorder (SIR) and Artificial Neural Network (ANN).

1) *Stream Information Recorder (SIR)*: The SIR is to obtain the information needed as input by the ANN from the encoding bitstream header. A video stream can be characterized by several parameters that will influence video types differently. Quality is affected by the number of bits transmitted per time interval, i.e. bitrate. Increases in bitrate lead to a direct increment in the perceived quality. Furthermore, parameters regarding the video scene composition have been demonstrated to affect quality to a large extent [25]. The scene can be characterized by its complexity and its level of motion, defined as the number of objects or elements and the amount of movement present in the video, respectively. Thus, we chose bitrate (B), level of motion (M) and scene complexity (C) as our ANN inputs. The SIR analyzes the incoming bitstream header to calculate these parameters. B can be directly recovered from the video codec. (C) and (M) are calculated according to equations 2 [25].

$$C = \frac{Bits_I}{2 * 10^6 * 0.91^{QP_I}} \quad M = \frac{Bits_P}{2 * 10^6 * 0.87^{QP_P}} \quad (2)$$

Where $Bits_I$, $Bits_P$ are bits of coded Intra (I) and Inter (P) frames, and QP_I , QP_P represent the average I-Frames and P-Frames quantization parameter. These values are also obtained directly from the encoding process and thus, do not increase the computational time or complexity of the method.

2) *Artificial Neural Network (ANN)*: The first time the client connects to the video server, it downloads the ANN model. This model has been trained and configured offline in the server. In this way, the server provides with a general model to all its clients and, as a consequence, the clients avoid the need to configure and train the network online. This would negatively influence and significantly increase the algorithm computational complexity and runtime. When the server's ANN is installed in the client, as part of it BA section, the client is ready to estimate the constant weights and threshold according to the specific video inputs B, M and C.

The reason for favoring ANNs against other machine learning techniques relies on the fact that these are universal function approximators, their ease to map input to output data, and their resistance to noise [26]. Moreover, they can work with unknown, noisy or incomplete input data, after being trained with just a reduced data set. An ANN consists of a group of basic interconnected processing units or neurons. Each neuron pair is linked by a weight, initialized randomly, and learned during the training process. Further information on the server’s ANN specifications and configuration as well as the training set can be found in the next section (section IV-B).

IV. EXPERIMENTAL TEST-BED

In this section we present the experimental test-bed developed to evaluate our algorithm’s performance.

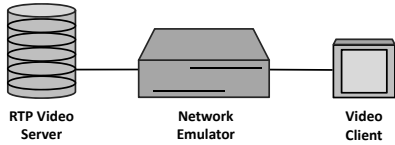


Fig. 3: Experimental test-bed

A. Test set-up

Our emulated network consists of an RTP video server and client, connected to input and output of a network emulator (PacketStorm Hurricane II) respectively, as it can be seen in figure 3. The video server provides a wide range of video types, to be streamed to the client on demand using the RTP/UDP protocol. The network emulator can generate impairments (such as delays, jitters or packet losses) in real time and according to a range of mobile network models. The client device is a laptop running Ubuntu with an intel core i7 2630QM cpu and 8Gb of RAM memory. It runs our NR-Video Quality Assessment method as well as SSIM (for benchmarking purposes). While SSIM is a well-accepted oQoE performance benchmark, our approach is functionally more applicable to realistic streaming systems whereby the original unimpaired video (necessary to measure SSIM) is unavailable to the network or client side of the streaming process.

When the client connects to the server for the first time, it receives the server’s ANN. Once it is installed, the client is ready for assessing quality.

During a streaming session, the client first triggers a video transmission from the server. While the streaming is taking place, our method analyses the frames received in real time (i.e. without delaying the streaming process in any way). Due to the client’s processor characteristics, we set the NFE splitter value N to 4, i.e. processing the frames in 4 parallel threads. With this configuration, the algorithm takes a maximum runtime of 0.300 seconds per frame. Thus, in order to fulfill the real-time condition capabilities, we set the FGD to grab 1 out of every 8 frames, which comes down to 3 frames per second, out of the 24 fps of the video under scrutiny. Were the client a lightweighted device such as a smart phone or tablet, the FGD would be set to fulfill the real-time condition capabilities accordingly. Once the streaming finishes, the client is allowed access to the original material and the FR assessment is performed as well, so that it can be used as benchmark.

B. ANN specifications, configuration and training-set

The server employs and sends to the client an ANN of the type Multi-Layer Perceptron (MLP) [26]. It consists of multiple layer-nodes in a directed graph, each layer being fully connected to the next one. We chose MLPs because its layered feed-forward topology allows the network to have a simple interpretation as a form of input-output model. Furthermore, it provides enough flexibility to model functions of almost arbitrary complexity.

On designing an ANN, it is fundamental to carefully select a suitable training set. This set consists of input signals assigned to their corresponding target, i.e. the desired output. Its representativity for the data distribution will determine the algorithm performance, causing inaccurate results if it is not well selected. For our case the server’s ANN was trained with a set of 10x10 videos obtained from 10 different video types selected from the Live Video Database [27]. Each one lasts 10 seconds, streams at 24 fps and is encoded in MPEG-4 AVC/H.264. The video types were considered at 10 different bitrates, ranging from 64 kbps to 2048 kbps. The characteristics of these 100 videos can be seen in Figure 4, in terms of M and C . Here we can see 10 points per video type (bs1, mc1, pa1, pr1, rb1, rh1, sf1, sh1, st1, tr1) whereby the various combinations of C and M values correspond to bitrates comprised between 64 kbps to 2048 kbps. In this way, we used a dataset consisting of a broad range of representative video types (dynamics). This selection gives a wider learning spectrum to the ANN and increases the strength of its generalization capabilities when it is exploited with different type of videos. For each video belonging to the training set, the weights and the thresholds from Equation 1, were set by performing linear regression to fit our metric’s quality index with the corresponding MOS value provided by the Live Video Database.

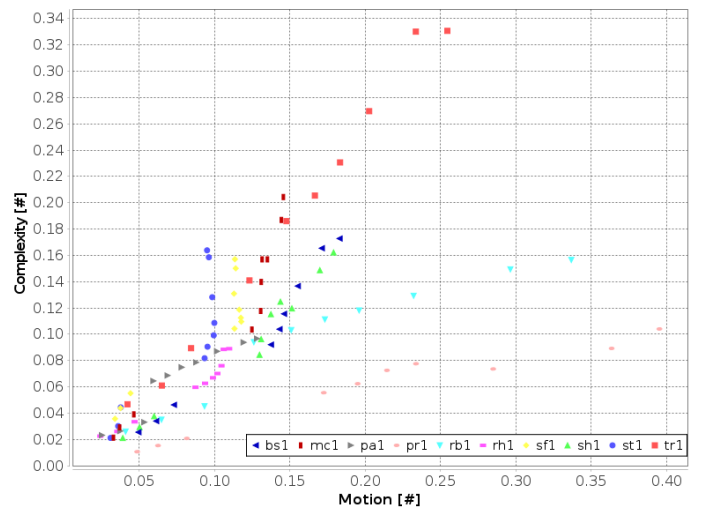


Fig. 4: Motion and complexity values for the training set.

Our goal for setting up the server’s ANN was to find the neural network that would converge near-optimally and have minimal error with the smallest possible network configuration. The total network error is defined as the difference between the value implied by the estimator and the quantity to be estimated.

In order to achieve our goal, there are several parameters that need to be selected. First, the number of neurons has to be estimated. Neurons are distributed in three types of layers: input, hidden and output. A network with too many neurons could lead to overfitting. On the contrary, a low number of neurons may not be enough for the model to fit the data. Furthermore, it is possible to add bias neurons, which are units not connected to the previous layer and which provide a constant output, useful to shift the activation function. Second, training the ANN with the chosen training set leads to the selection of its learning rate. Ranging between 0 and 1, it governs the learning speed of the network from the training set. A very low value would require a large number of training cycles, slowing down the process extremely. But, a learning rate close to 1 would make the weights to diverge and the objective error to heavily oscillate. Last parameter to take into account is the momentum, used to increase the learning speed of the system. Same as the learning rate, the momentum ranges between 0 and 1. A high value increases the speed of convergence. However, if the momentum is too high the system tends to overshoot the minimum, provoking instability.

Following the aforementioned rules, we started with a network with a small number of neurons, 8 + 1 bias, in one unique hidden layer, a learning rate of 0.4 and a momentum of 0.8. In an iterative process, we reduced the momentum and increased the learning rate and the number of neurons one at a time until obtaining convergence and minimum network error. After several configurations, we obtained our optimal structure: 1 hidden layer of 15 + 1 (bias) neurons, a value of 0.2 for the learning rate and a momentum of 0.7. After 2000 iterations, this structure outputted a total network error of 0.025. Further changes in the network did not lead to any measurable improvement.

V. EXPERIMENTS AND RESULTS

The network emulator generates impairments (such as delays, jitters or packet losses) in real time. In [28] we demonstrated that packet losses degrade the quality to a greater extent than any other factor. Thus, in these experiments, we assessed our method's performance in the presence of different levels of packet loss.

During a test session, one video is chosen and first transmitted without the influence of packet loss to verify our ability to analyze quality in real-time. The process is then repeated, increasing the packet loss level to 0.5%, 1%, 5%, 10% and 20%, respectively. After the session is finished, the client is allowed to access the server database to obtain the original material with the sole purpose of computing the SSIM metric for benchmarking.

Figures 5 and 6 show the test performed to the videos tr1 and sh1 encoded at 2048 kbps. The different characteristics in terms of C and M of these two videos, low values for sh1 (0.16 and 0.18) and very high values for tr1 (0.33 and 0.25), make them perfect candidates to thoroughly assess the performance of our method under extreme conditions. In both cases our video quality index (red line) degrades as the packet loss level increases, going from roughly 0.98 (when packet loss is set to 0%) to 0.75 for tr1 and 0.64 for sh1 (when packet loss is set to 20%). The difference on degradation between both cases is

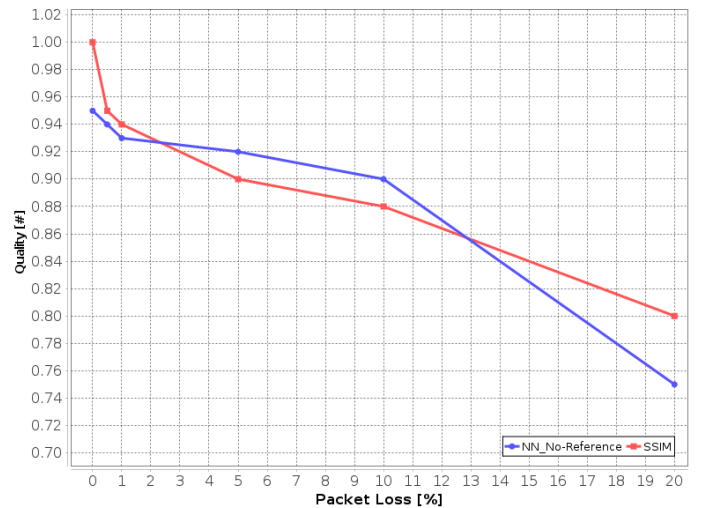


Fig. 5: Results on tr1 encoded at 2048 kbps, for different levels of packet loss.

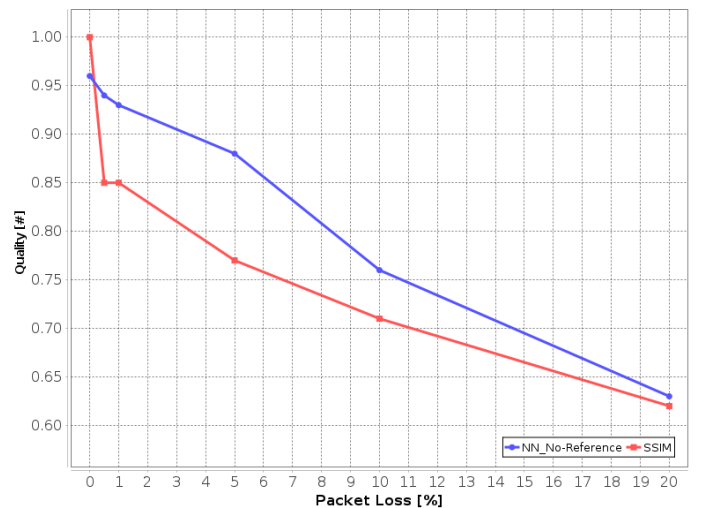


Fig. 6: Results on sh1 encoded at 2048 kbps, for different levels of packet loss.

expected, as these have diverse type of dynamics (M and C). As the packet loss degrades quality, C and M are very much affected. This provokes that, for same bitrate encoding, videos with lower C-M levels are affected more rapidly.

If we compare our method to the benchmark, SSIM (blue lines), we can observe that although there are some points in which the quality indexes differ, due to the fact that SSIM focuses on the structure while our method is more influenced by complexity and motion, our method correlates well with the SSIM benchmark. Furthermore, it counter-intuitively provides high correlation for high losses (20%), where the degraded dynamics of the video would be expected to make our assessment diverge greatly from the FR alternative. The reason for the high correlation even in extreme packet loss cases is the generality of our training set, which covers case-conditions with very low C and M (for example, very low bitrates

samples). Pearson and Spearman correlations as well as RMSE (Root-mean square error) were performed for quantifying of the correlation (table I). For both videos, the Pearson and Spearman correlations show higher values than 0.9 and the RMSE is lower than 0.08.

Correlation	tr1	sh1
Pearson	0.9073	0.9073
Spearman	1	0.9856
RMSE	0.0316	0.0716

TABLE I: Correlation values between SSIM and our method with and without ANN for videos tr1 and sh1.

From these results, we concluded that our hybrid ANN enhanced method is capable of assessing quality with results comparable with the state-of-the-art FR metrics. Hence our method allows for real-time analysis, reduced complexity and no need for the original material. Furthermore, thanks to its low complexity and run time it is fit for running in light-weighted mobile devices.

VI. CONCLUSION

In this paper we have presented a novel NR-video quality assessment method. More exactly, based on a linear blur-noise image assessment, we have designed and implemented a low complexity NR-Video Quality Assessment Method fit for real-time end-to-end service performance evaluation. An Artificial Neural Network is used to automatically adapt the noise and blurriness weights to diverse video characteristics. In an emulated network environment we have evaluated our NR method with the FR alternative SSIM, finding a strong correlation (RMSE of less than 0.07 in all evaluated scenarios).

Given the low computational complexity of our method, we can say that it is suitable for deployment on resource constrained mobile devices (such as smart phones), in real mobile network environments (like 4G-LTE) and in practical service provisioning systems (where the original videos are not available at the client side).

REFERENCES

- [1] A. Liotta, "The cognitive NET is coming," *IEEE Spectrum*, vol. 50, no. 8, pp. 26–31, Aug. 2013.
- [2] F. Agboma and A. Liotta, "Quality of experience management in mobile content delivery systems," *Journal of Telecommunication Systems, special issue on the Quality of Experience issues in Multimedia Provision*, vol. 49, no. 1, 2012.
- [3] A. Liotta, "Farewell to deterministic networks," in *in proc. of the 19 IEEE Symposium on Communications and Vehicular Technology in the Benelux*, November 2012.
- [4] M. Torres Vega, S. Zou, D. C. Mocanu, E. Tangdiongga, A. M. J. Koonen, and A. Liotta, "End-to-end performance evaluation in high-speed wireless networks," in *The Int. Conf. on Network and Service Management (CNSM)*, 2014.
- [5] V. Menkovski, G. Exarchakos, A. Liotta, and A. C. Sanchez, "Quality of experience models for multimedia streaming," *IJMCMC*, vol. 2, no. 4, pp. 1–20, 2010.
- [6] P. Le Callet, S. Moeller, A. Perkis, and eds., "Qualinet white paper on definitions of quality of experience," in *COST Action IC 1003*, Lausanne, Switzerland, 2012.
- [7] M. Shahid, A. Rossholm, B. Löfvström, and H. Zepernick, "No-reference image and video quality assessment: a classification and review of recent approaches," *Journal on Im. and Vid. Proc.*, vol. 40, no. 1, 2014.

- [8] V. Menkovski, G. Exarchakos, and A. Liotta, "The value of relative quality in video delivery," *J. Mob. Multimed.*, vol. 7, no. 3, pp. 151–162, Sep. 2011.
- [9] T. Zinner, O. Hohlfeld, O. Abboud, and T. Hossfeld, "Impact of Frame Rate and Resolution on Objective QoE Metrics," in *International Workshop on Quality of Multimedia Experience, 2009. QoMEX 2009.*, Trondheim, Norway, Jun. 2010.
- [10] B. Staehle, A. Binzenhöfer, D. Schlosser, and B. Boder, "Quantifying the influence of network conditions on the service quality experienced by a thin client user," in *14. GIITG Konferenz Messung, Modellierung und Bewertung von Rechen- und Kommunikationssystemen (MMB 2008)*, Dortmund, Germany, Apr. 2008.
- [11] V. Menkovski, G. Exarchakos, and A. Liotta, "Tackling the sheer scale of subjective qoe," in *MobiMedia*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 79. Springer, 2011, pp. 1–15.
- [12] S. Winkler and P. Mohandas, "The evolution of video quality measurement: From PSNR to hybrid metrics," *IEEE Transactions on Broadcasting*, vol. 54, no. 3, pp. 660–668, Sep. 2008.
- [13] D. C. Mocanu, G. Exarchakos, H. B. Ammar, and A. Liotta, "Reduced Reference Image Quality Assessment via Boltzmann Machines," in *in proc. of the 1st. IEEE/IFIP IM 2015 International Workshop on Cognitive Network & Service Management*, May 2015.
- [14] D. C. Mocanu, G. Exarchakos, and A. Liotta, "Deep learning for objective quality assessment of 3d images," in *IEEE International Conference on Image Processing (ICIP)*, 2014.
- [15] M. Torres Vega, D. C. Mocanu, R. Barresi, G. Fortino, and A. Liotta, "Cognitive streaming on android devices," in *in proc. of the 1st. IEEE/IFIP IM 2015 International Workshop on Cognitive Network & Service Management*, May 2015.
- [16] M. G. Choi, J. H. Jung, and J. W. Jeon, "No-reference image quality assessment using blur and noise," 2009.
- [17] A. Reibman, S. Sen, and J. der Merwe, "Analyzing the spatial quality of internet streaming video," in *in Proceedings of International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, January 2005.
- [18] T. Oelbaum, C. Keimel, and K. Diepold, "Rule-based no-reference video quality evaluation using additionally coded videos," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 3, no. 2, 2009.
- [19] G. Valenzise, S. Magni, M. Tagliasacchi, and S. Tubaro, "No-reference pixel video quality monitoring of channel-induced distortion," *IEEE Trans. Circuits Syst. Video Techn.*, vol. 22, no. 4, pp. 605–618, 2012.
- [20] T. Brandão and M. Queluz, "No-reference quality assessment of h.264/avc encoded video," *IEEE Trans. on Circuits and Systems for Video Tech.*, vol. 20, no. 11, pp. 1437–1447, November 2010.
- [21] V. Menkovski, G. Exarchakos, and A. Liotta, "Machine learning approach for quality of experience aware networks," in *in proc. of Computational Intelligence in Networks and Systems*, November 2010.
- [22] M. Shahid, B. Rossholm, and B. Löfvström, "A reduced complexity no-reference artificial neural network based video quality predictor," in *Proc. of the Int. Congress on Image and Signal Processing*, 2011.
- [23] T. Shanableh, "Prediction of structural similarity index of compressed video at a macroblock level," *IEEE Sign. Proc. Lett.*, vol. 18, no. 5, 2011.
- [24] C. Keimel, J. Habigt, and K. Diepold, "Hybrid no-reference video quality metric based on multiway pls," in *EUSIPCO 2012: 20th European Signal Processing Conference*, Sep 2012, pp. 1244–1248.
- [25] A. Liotta, D. C. Mocanu, V. Menkovski, L. Cagnetta, and G. Exarchakos, "Instantaneous video quality assessment for lightweight devices," in *Proc. of Int. Conference on Advances in Mobile Computing*, ser. MoMM '13, New York, NY, USA, 2013, pp. 525:525–525:531.
- [26] F. Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Washington DC, USA: Spartan Books, 1961.
- [27] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, "Study of subjective and objective quality assessment of video," *Trans. Img. Proc.*, vol. 19, no. 6, pp. 1427–1441, Jun. 2010.
- [28] D. Mocanu, A. Liotta, A. Ricci, M. Vega, and G. Exarchakos, "When does lower bitrate give higher quality in modern video services?" in *Network Operations and Management Symposium (NOMS), 2014 IEEE*, May 2014, pp. 1–5.