OpenVQ - A Video Quality Assessment Toolkit

Kristian Skarseth¹, Henrik Bjørlo², Pål Halvorsen^{3,4}, Michael Riegler^{3,4}, Carsten Griwodz^{3,4}

¹LABO Mixed Realities, Norway ²Sopra Steria, Norway

³Simula Research Laboratory, Norway ⁴University of Oslo, Norway

skarseth@mixedrealities.no, henrbjor@ifi.uio.no, {paalh, michael, griff}@simula.no

ABSTRACT

This paper presents an open source video quality assessment tool called OpenVQ and OPVQ, an objective quality metric based on work of the Video Quality Experts Group. It was primarily developed to satisfy the need of the community for a freely accessible and easy-to-use quality assessment tool, which would remove the need for using thoroughly outdated methods that is still prevalent because only those are easily available.

OpenVQ is implemented as a toolbox of full-reference quality assessment methods, which can be extended with improved methods in the future. The primary method provided by OpenVQ at this time is OPVQ, the Open Perceptual Video Quality metric. It is very much inspired by the formal model of ITU-T Rec. J.247 Annex B, however with many modifications to deal with errors and inconsistencies in formulas and pseudo code, and restrictions to avoid violating known patents. OPVQ has been validated with several IRRCyN datasets and was found to yield results similar to those reported in ITU.T J.247.

OpenVQ implements also PSNR (Peak signal-to-noise-ratio), because it is in high demand in spite of its bad performance, and SSIM (Structural similarity index), an image quality metric that is known to perform quite well in many cases.

1. INTRODUCTION

Researchers working with video must frequently assess the visual quality of the video that they have processed. Performing large-scale subjective studies under controlled conditions is currently the only way to achieve truly valid answers concerning the quality of the video as perceived by humans. Doing this is, however, rarely affordable.

It is much more desirable to compare original and processed video objectively, using deterministic mathematical operations, preferably a metric, which can express distance and thereby also order. What remains elusive is how such a metric is related to the human perception of video quality when watching the video, i.e., which Quality of Experi-

ence [7] is achieved. Its performance is defined by its ability to accurately predict quality ratings given by human test subjects.

The prevalent metric in research articles is PSNR (peak signal-to-noise ratio), which is a pixel-by-pixel mean square error (MSE) between an original and a compressed video. The limited precision in using PSNR for estimating video quality as a human perceives it has been shown time and again [13, 1]. Efforts to create objective metrics for digital video that correlate better with subjective assessments have been made since the early days of digital transmission [14, 2]. The Alliance for Telecommunications Industry Solutions (ATIS) performed the first large scale validation test of such metrics in 1994–95, and since the late 1990s the Video Quality Experts Group (VQEG) has performed several similar tests resulting in various standards [8].

2. THE TOOL

OpenVQ can be used to implement a variety of full-reference objective quality estimation algorithms, and provides unified base classes, I/O functions and statistical functions. The 3 existing metrics have also been structured in a modular manner to allow the re-use of algorithms like spatial alignment in future algorithms. Currently, the central algorithm of OpenVQ is OPVQ, but it also supports the PSNR and SSIM algorithms. With OPVQ, we attempt to provide an independent and free implementation of the ITU.T standard for full-reference objective video quality assessment, J.247. Within the constraints discussed in Sections 2.2 and 2.3, we have achieved this goal.

2.1 Algorithm selection

Choosing the algorithm that we would implement was not a straight-forward decision. The OPVQ algorithm is inspired by ITU-T J.247 Annex B [4], which appears to be a description of PEVQ (Perceptual evaluation of video quality). J.247 is supposed to be more up-to-date than J.144 [3], which targeted MPEG-1 and H.263-encoded videos. Software for J.144 Annex C (VQM) is freely available as Matlab code¹, and sometimes used for video quality assessment.

J.247 was the result of a competition initiated by VQEG, which should identify the most promising objective quality metric for MPEG-2- and H.264-encoded videos with resolutions up to VGA. Several competitors are included in the standard, with Annex B yielding the lowest average error rate with respect to subjective assessments, while Annex C

¹<http://www.its.bldrdoc.gov/resources/video-quality-research/software.aspx>

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored.

MM '16 October 15–19, 2016, Amsterdam, Netherlands

© 2016 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-3603-1/16/10.

DOI: <http://dx.doi.org/10.1145/2964284.2973800>

This work is licensed under a Creative Commons

Attribution-NonCommercial-ShareAlike International 4.0 License.

Figure 1: Overview of the PEVQ model (from J.247 [4])

yielded the best average difference from subjective assessments.

J.341 [5] is supposed to be more up-to-date again than J.247, targeting H.264-encoded videos of resolution up to HD. J.341 reports the results for a single contender.

Obviously, we intended to provide a more recent objective quality metric than J.144, which already exists. J.341 would be desirable for two reasons. First, because it has been validated for HD videos, and second, because the dataset including subjective assessment scores that was used in its evaluation has been released to the public. The major problem with implementing J.341 is that large parts of the standard itself are formulated as simplified C++ code. A direct implementation of the standard would, consequently, result in a violation of copyright.

This problem did not exist for J.247, because the contenders were in this standard described in a mix of pseudo code and formulas. A major problem with an independent implementation of J.247 is that the dataset used for its assessment has not been released to the public. Still, we chose J.247 Annex B as the basis for OPVQ.

2.2 Constraints

We studied J.247 Annex B, which comprises a series of steps that are best illustrated by Figure 1. In trying to understand whether we would be able to publish our work as source code, we tried to identify patents that the code could violate, and found a series of patents held by Opticom related to the temporal alignment of video sequences (in particular US Patents 8,228,385 [6] and 9,037,435 [11]).

These seem to affect the use of any implementation of J.247 Annex B that includes the coarse and detailed temporal alignments steps. To provide a free objective quality assessment tool, we could either design our own temporal alignment mechanism or leave temporal alignment out all together. We removed this part from our implementation, and discuss the impact of this decision.

J.247 aims at quality assessment of sequences with "transmission errors with packet loss" and "temporal errors (pausing with skipping) of maximum 2 seconds". Clearly, the situation will occur in every setting that makes use of UDP-based streaming, and occurs also in HTTP adaptive streaming when a buffer under-run occurs.

We find, however, that a playback device is fully aware of the difference between the progression of its own playback clock and the clock of video stream itself, because the latter is carried in timestamps of the transport stream. The majority of research testbeds will not require a blind analysis of independently captured and decoded original and modified video. They will have the temporal alignment information directly available, and have access to the compensation decisions made by the decoder. The resulting, simplified, structure is shown in Figure 2. Still, by making this decision, OPVQ departs from J.247 Annex B in a major way.

Figure 2: OPVQ model

2.3 Further modifications

In analyzing the formulas in J.247 Annex B, we deviated further from the standard. The spatial alignment step was simplified without any change in semantics.

The clipping of ranges for the Luma indicator in distortion analysis is incorrect in the standard, because clipping occurs far outside the possible value range. However, it cannot be discarded because it reflects the strength of an impact that incorrect edges have on human perception. The impact is clearly capped at a maximum disturbance, after which a stronger color difference does no longer matter. Figure 3 illustrates this point. We could only repair the problems with a best-guess decision.

The Chroma indicator has a similar problem, but there are also indices referring to Luma and Chroma planes in the original and modified images that appeared unreasonable. We modified the indices to copy the behaviour of the Luma indicator, and added a best guess in edginess as before.

In the Omitted Component indicator, we found a negation of differences that appeared unreasonable, whereas it was missing in the Introduced Component indicator. We chose to remove the negation.

Finally, J.247 Annex B defines a table of mapping coefficients to create an overall DMOS from the various indicators. We found it curious that some of these indicators improve the resulting DMOS value when a disturbance increases between original and modified image. All other parameters remaining equal, this can obviously yield valid results, as illustrated in Figure 2.3, but this is not generally true.

We decided therefore to restrict the coefficient for each indicator to the sensible range, so that more disturbance would also reduce DMOS, and to create a new table of coefficients based on the freely available IRCCyN [9] datasets.

Figure 3: More pronounced error, but equal disturbance

Figure 4: Equal curves with different parameters

The final DMOS score is defined as in the J.247 Annex B:

$$Score = \epsilon + \sum_{i=0}^3 \frac{\omega[i]}{1 + e^{\alpha[i]L_{lim}[i] + \beta[i]}} \quad \begin{matrix} \alpha, \omega \in \mathbb{R}_{>0} \\ \beta, \epsilon \in \mathbb{R} \end{matrix} \quad (1)$$

$$DMOS_P = \begin{cases} 1 & \text{if } Score < 1 \\ 5 & \text{if } Score > 5 \\ Score & \text{otherwise} \end{cases} \quad (2)$$

where each index i refers to one of the 4 main indicators. The final result is a value between 1 and 5, where 1 represents the lowest and 5 the highest quality.

3. PERFORMANCE EVALUATION AND COMMAND LINE EXAMPLES

The OpenVQ software is meant for video quality evaluations in general. The primary goal of this software is to implement a video quality metric that performs significantly better than PSNR.

3.1 OPVQ Evaluation

One of the goals of OPVQ is to provide a video quality assessment metric that performs significantly better than PSNR. We therefore ran tests on multiple datasets to compare PSNR and OPVQ using various IRCCyN [9] and ETFOS [10] datasets. In summary, our experimental results show a clear difference in performance in all configurations, and strongly indicate that OPVQ significantly outperforms PSNR [12]. However, since the amount of subjective test data available is small, we encourage further validation of the metrics performance using new datasets.

To investigate if OPVQ performs significantly better than PSNR, we calculated PSNR ratings for the applicable datasets, and used cubic polynomial fitting to best fit the scores to the subjective scale of each dataset. We used our own implementation of PSNR with spatial alignment enabled to ensure optimal conditions for PSNR. We also calculated the final performance evaluation data for OPVQ with cubic polynomial fitting, using unaltered PEVQ coefficients as well as

optimized coefficients using both schemes described in more detail in [12].

The results in table 1 show a clear difference in performance between PSNR and OPVQ in all configurations, and strongly indicate that we have achieved our goal of implementing a video quality metric that significantly outperforms PSNR. An interesting observation is that the best performance for OPVQ after polynomial fitting is achieved when using the unaltered PEVQ coefficients — if only slightly better than with trained coefficients. The coefficients from the PEVQ description are not logically sound for some indicators, as they reward increase in amount of errors with higher contribution to the final score. Because of this, and the fact that the difference recorded is only minimal, our recommendation is to use optimised coefficients adhering to the constraints defined for OPVQ.

As we have not had access to any software or hardware that computes PEVQ scores, and the dataset that was used for the analysis in J.247 is unavailable, we do not have the means to investigate the subsequent goal of keeping OPVQ's performance on par with PEVQ. The performance data we could produce shows correlation and error values for the IRCCyN and ETFOS datasets that are close to the values reported in J.247. However, since these results are not produced using the same dataset, this is at most a hint that performances are comparable.

	AVC/SVC		Infl.cont.		Dataset		QP0	QP1 (A)	QP0	QP1 (B)	EVVQ	Average
OPVQ w/coeffs unaltered from PEVQ												
Pearson R	0.943	0.907							0.873		0.890	0.897
RMS error	0.201	0.454							0.578		0.283	0.416
Outlier ratio	0.000	0.521							0.665		0.078	0.381
OPVQ w/coeffs optimised using all other datasets												
Pearson R	0.943	0.907							0.869		0.890	0.896
RMS error	0.201	0.454							0.574		0.283	0.419
Outlier ratio	0.000	0.529							0.649		0.078	0.386
OPVQ w/coeffs optimised using datasets of same test methodology												
Pearson R	0.943	0.907							0.867		0.871	0.886
RMS error	0.201	0.454							0.576		0.583	0.288
Outlier ratio	0.000	0.525							0.655		0.671	0.111
PEVQ as reported in J.247 – not tested with the same data												
Pearson R	-	-							-		-	0.825
RMS error	-	-							-		-	0.571
Outlier ratio	-	-							-		-	0.502

Table 1: Overall performance evaluation after cubic fitting (best results in bold)

The evaluation shows that for the subjective data available to us and within an expected range of quality degradations, OPVQ performs significantly better than PSNR. Because the amount of subjective test data available is small, we encourage further validation of the metrics performance using other datasets. We have not been able to conclusively compare the performance of OPVQ with PEVQ.

3.2 Command Line Examples

In order to use the program, OpenVQ provides a simple command line interface. Running the program with `-help` or without any arguments displays the help screen. First,

```
openvq [global options] \
command [command specific options]
```

shows the overall command format where *[global options]* are `-help` and `-log-level`, *command* is video quality metric to use (`opvq`, `psnr` or `ssim`) and particular *[command specific options]* are options for the individual quality eval-

uations like frame number limits (`-max-frames`), path to source (original) and processed video files (`-src` and `-pvs`), number of threads to use (`-num_threads`), etc.

The OPVQ algorithm may be used like this:

```
openvq opvq -src src01.avi -pvs pvs01.avi
```

Here, the quality of the processed video file `pvs01.avi` will be evaluated compared to the original source video `src01.avi`, and a final `opvq` score between 1 and 5 is returned. Replacing the command `opvq` with `psnr` or `ssim` gives similar quality evaluations using the respective metrics.

4. INSTALLATION AND LICENSE

OpenVQ is meant as a tool for researchers. It is licensed under the terms of the GNU Affero General Public License (AGPL) version 3, as published by the Free Software Foundation. The decision to use AGPL was made to ensure that OpenVQ can fulfill its intended role in the research community, while avoiding use in a commercial service. The primary dependencies of OpenVQ are OpenCV (3-clause BSD), Libav (LGPL v2.1) and Boost (Boost license).

OpenVQ is distributed as a CMake project. For UNIX systems, CMake can be invoked from the command line. If no other project type is specified, CMake will generate a Makefile that can be invoked with Make. If no errors are reported, OpenVQ was built successfully and is ready to be installed into the system path. The install command may have to be invoked with `sudo`, depending on the user's privileges and the permissions of the install prefix.

We have tested our software on *Linux*, *Mac OS X* and *Windows*. The following instructions are meant for Ubuntu Linux:

- Clone the *OpenVQ* source code from https://bitbucket.org/mpg_code/openvq.
- Install the pre-requisites that are listed on the repository web page. You can also find them in the cloned repository in `README.md`.
- Use CMake and Make to compile the source code. The installation directory is chosen at this time, with a default installation in `/usr/local`.
- Optionally install the program `openvq` after compilation.
- Provide the program with an original video using `-src` and a processed video using `-pvs` and an algorithm to run. It will output the final score, which is algorithm-dependent.

5. RELATED WORK

PSNR is frequently used because freely available quality estimation tools are not discovered, although they do exist. Beyond PSNR and SSIM, NTIA VQM was mentioned in Section 1, the free version of MSU VQMT¹ provides MSU's own metrics, MMSPG VQMT² provides a Visual Information Fidelity (VIFp), while LIVE MOVIE³ makes use of motion trajectories in its quality assessment. The machine-learning based VMAF Development Kit (VDK)⁴ is provided by Netflix and based on their own (partly public) datasets. The VQEG Joint Effort Group (JEG)⁵ has been established to collect implementations of separately developed metrics, but has not released code yet.

- 1 http://compression.ru/video/quality_measure/index_en.html
- 2 <http://mmspg.epfl.ch/vqmt>
- 3 <http://live.ece.utexas.edu/research/Quality/movie.html>
- 4 <http://techblog.netflix.com/2016/06/toward-practical-perceptual-video.html>
- 5 http://vqegjeg.intec.ugent.be/wiki/index.php/Main_Page

6. CONCLUSION

The video quality is highly important for the users' perceived experience of the video service. However, it has for a long time been discussed how to evaluate the quality. The OpenVQ open source software provides a general tool to easily calculate objective metrics for video quality evaluation in general. It provides algorithms for PSNR and SSIM, but also the new full reference objective video quality assessment metric OPVQ, which is an independent implementation of the ITU-T Rec. J.247 for a full-reference objective quality metric.

We are searching for freely available datasets to determine a parameter set that applies to HD and higher resolutions videos. We are also planning to create and assess a spin-off from OPVQ by integrating considerations that are found in standards J.341 and J.343.

7. REFERENCES

- [1] Q. Huynh-Thu and M. Ghanbari. Scope of validity of PSNR in image/video quality assessment. *Electronics Letters*, 44(13):800–801, 2008.
- [2] ITU-R. Methods for objective picture quality assessment in relation to impairments from digital coding of television signals. Technical report, ITU, 1992.
- [3] ITU-T. J.144 : Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference, 2004.
- [4] ITU-T. J.247: Objective perceptual multimedia video quality measurement in the presence of a full reference, 2008.
- [5] ITU-T. J.341: Objective perceptual multimedia video quality measurement of HDTV for digital cable television in the presence of a full reference, 2011.
- [6] M. Keyhl, C. Schmidmer, and R. Bitto. Apparatus for determining information in order to temporally align two information signals, July 2012. US Patent 8,228,385.
- [7] F. Kuipers, R. Kooij, D. De Vleeschauwer, and K. Brunnström. Techniques for Measuring Quality of Experience. In *Proc. of WWIC*, 2010.
- [8] M. Pinson, N. Staelens, and A. Webster. The history of video quality model validation. In *Proc. of IEEE MMSP*, 2013.
- [9] Y. Pitrey, M. Barkowsky, P. L. Callet, and R. Pepion. Subjective quality assessment of MPEG-4 Scalable Video Coding in a mobile scenario. In *Proc. of EUVIP*, 2010.
- [10] S. Rimac-Drlje, M. Vranješ, and D. Žagar. Foveated mean squared error—a novel video quality metric. *Multimedia Tools and Applications*, 49(3):425–445, 2010.
- [11] C. Schmidmer, R. Bitto, and M. Keyhl. Device and method for determining a sample rate difference, May 19 2015. US Patent 9,037,435.
- [12] K. Skarseth and H. Bjørlo. OPVQ and OpenVQ – Creating free software tools for video quality assessment. Master's thesis, Department of Informatics, University of Oslo, Norway, 2015.
- [13] S. Winkler and P. Mohandas. The Evolution of Video Quality Measurement: From PSNR to Hybrid Metrics. *IEEE Trans. on Broadcasting*, 54(3):660–668, 2008.
- [14] S. Wolf. NTIA Report 90-264: Features for Automated Quality Assessment of Digitally Transmitted Video. Technical report, National Telecommunications and Information Administration, 1990.