Framework for reproducible objective video quality research with case study on PSNR implementations

Framework for reproducible objective video quality research with case study on PSNR implementations

Ahmed Aldahdooh[1], Enrico Masala[2], Glenn Van Wallendael[3], Marcus Barkowsky[1]

[1]*LUNAM Université, Université de Nantes, IRCCyN UMR CNRS 6597, Nantes, France*
[2]*Control and Computer Engineering Department*
Politecnico di Torino, corso Duca degli Abruzzi 24, 10129 Torino - Italy,
[3] *Ghent University - imec - IDLab, Ghent, Belgium*

Abstract

Reproducibility is an important and recurrent issue in objective video quality research because the presented algorithms are complex, depend on specific implementations in software packages or their parameters need to be trained on a particular, sometimes unpublished, dataset. Textual descriptions often lack the required detail and even for the simple Peak Signal to Noise Ratio (PSNR) several mutations exist for images and videos, in particular considering the choice of the peak value and the temporal pooling. This work presents results achieved through the analysis of objective video quality measures evaluated on a reproducible large scale database containing about 60,000 HEVC coded video sequences. We focus on PSNR, one of the most widespread measures, considering its two most common definitions. The sometimes largely different results achieved by applying the two definitions highlight the importance of the strict reproducibility of the research in video quality evaluation in particular. Reproducibility is also often a question of computational power and PSNR is a computationally inexpensive algorithm running faster than

realtime. Complex algorithms cannot be reasonably developed and evaluated on the abovementioned 160 hours of video sequences. Therefore, techniques to select subsets of coding parameters are then introduced. Results show that an accurate selection can preserve the variety of the results seen on the large database but with much lower complexity. Finally, note that our SoftwareX accompanying paper presents the software framework which allows the full reproducibility of all the research results presented here, as well as how the same framework can be used to produce derived work for other measures or indexes proposed by other researchers which we strongly encourage for integration in our open framework.
Keywords: Video quality, large-scale database, objective video quality metric, video coding.

1. Introduction and Motivation

Objective Video Quality evaluation is used in many scenarios such as rate-distortion optimization of video encoding, prediction and replacement of subjective quality assessment, or improvement of video processing algorithms. Contrary to the continuous development in standardization seen for algorithms in video coding, the development of objective video quality evaluation is mostly advancing in individual research groups. A notable exception is the work of the Video Quality Experts Group (VQEG) and in particular its Joint-Effort-Group Hybrid (JEG-Hybrid) which supports and maintains this research.

The abovementioned isolation has three important impacts on the reproducibility of results: Firstly, the individual researchers, PhD students

in many cases, need to collect previous research individually, biasing their knowledge and leading to comparisons with outdated algorithms or using statistical measures for comparisons that are no longer state of the art. Secondly, implementations of existing algorithms, notably complex algorithms, are sparse or no longer maintained by their authors, notably after finishing their PhD work. For instance, the popular MetrixMux tool [1] is currently unavailable at its home page and only unofficial copies can be downloaded through Internet searches. Thirdly, textual descriptions of algorithms are often erroneous because no independent reimplementations are performed. The complexity required to reimplement complex algorithms has convinced the video coding community to accept the reference software being the ground truth rather than the textual description.

As an example, we mention the PVQM algorithm described in [2]. At a first glance, the paper seems to provide a detailed description of all the algorithms and formulas underlying PVQM. However, when dealing with all the details needed for the actual implementation, it becomes apparent that details are missing, e.g., some formulas are not coherent with the others so their output is not reasonable, or some existing algorithm that the calculation relies on may be implemented differently, such as histogram-matching that may be calculated from the center of the value range or from the extremes leading to slight differences that, in the course of the algorithm, are emphasized. In its re-implementation, made publicly available at [3] by one of the authors of this work as part of the activities in the JEG-Hybrid group, several parts of the source code contains comments where deviations from the paper statements have been recorded, supported by email communica-

tions with the authors of [2] who, inadvertently, introduced some errors in the published version of the formulas. Despite the kind help of the original authors, however, currently it is not possible to verify that the implementation is correct since the original implementation cannot be made publicly available and there is no conformance test dataset.

The fact that this is not an exception, is made plausible by the following experiment. We configured a query for the three terms video, quality, and prediction to appear in any order in the paper titles of the scientific publication search engine IEEEXplore leading to 59 hits. We manually screened each paper for the existence and reproducibility of a newly proposed algorithm. In 16 publications, it seemed that no new algorithm was proposed so we removed them from the analysis.

Figure 1 shows that only 9.31% out of the 43 papers (marked in green color shades) relevant to the search comes with associated source code that allows reproducibility of the techniques. 51.16% (red color shades) rely on some sort of learning technique (e.g., neural networks [4, 5], machine learning [6], regressions [7]) that would require the availability of the same exact dataset and the exactly same (potentially erroneous) version of the software implementation of the learning algorithm to re-train the system in the same way. None of the papers allows easy access to the dataset used in their experiments. Finally, 39.53% of the works (yellow shades) seem to provide reasonably detailed information about the techniques and formulas necessary to implement the proposed algorithms, as in, for instance, [8, 9, 10]. However, none provides access to source code or conformance test datasets, hence the same difficulties encountered with PVQM could happen in their case.

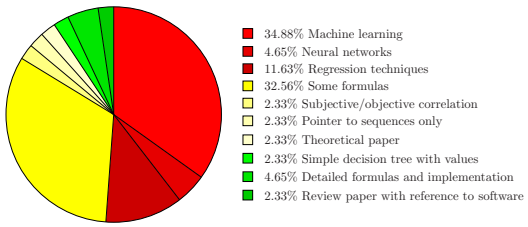

Figure 1: Reproducibility of the algorithms in IEEEXplore papers corresponding to the title search terms prediction, quality, and video

Instead of being able to use the most advanced algorithm as done in the video coding community, researchers often compare to simple algorithms such as Peak Signal to Noise Ratio (PSNR) or Structural Similarity Index (SSIM) and they are still the preferred algorithm for other communities such as digital signal processing and video coding in particular. While the problem of reproducibility is immediately evident for complex measures, even simple measures such as PSNR have been described inexactly in the literature and were thus be implemented differently. The first inexactness concerns the peak value. While most implementation use the value of 255 as the maximum of the data representation in eight bit, according to ITU-R BT.601 the luminance component of the YCbCr color space is limited to 235 leaving headroom for postprocessing. This definition is recommended by the ANSI/ATIS in [11]. The second inexactness deals with the requirement for temporal and color alignment of the reference and test sequence which was not present in the first versions of PSNR and may be considered as non-normative preprocessing steps [12]. However, the alignment has an important impact on the final result: When temporal mismatch occurs, PSNR without temporal alignment underestimates the quality because it uses the wrong reference for calculation [13]. PSNR with temporal alignment often

overestimates the quality as effects such as stalling, skipping, or reduced frame rate are ignored. For color alignment the situation may get even worse because some brightness, contrast, and color changes may even improve the perceived quality over the reference, a situation which PSNR is unable to cope with. The third inexactness is about temporal pooling. PSNR has been derived from the signal processing based measure Signal to Noise Ratio (SNR) noticing that the noise was equally disturbing in bright (high energy of the source signal) as in dark (low energy of the source signal) regions. Originally used for image signals, PSNR has been adapted to videos. Three temporal pooling strategies may be considered: Firstly, taking the video as a single dimensional signal (averaging the Mean Squared Error (MSE) values per frame), secondly, providing the average value of the quality measured per frame (calculating the mean of the PSNR values per frame), thirdly, emphasizing degradations by calculating the squared error of the PSNR values (calculating the squared mean of the PSNR values). It should also be noted that the scope of PSNR is limited to comparing the same content and the same type of degradation as described in [14].

The usual process that researchers follow to objectively verify and validate their newly developed objective measures is that they test on a particular transmission chain, referred to as hypothetical reference circuits (HRCs), notably choosing different quality levels (different bitrate budgets or different QPs). The main drawback of this procedure is that if another researcher selects different HRCs, different results may be obtained. On the other hand, using the large-scale database on their new objective measure often requires considerable computational effort and for verification and validation, subjec-

tive annotation is required as ground truth which is not feasible for large-scale databases as, for example, the abovementioned large-scale database contains seven days of 24 hours video sequences. Therefore, an algorithm that runs in realtime would require one week of calculation for each development cycle on a single computer system. Often algorithms are far more complex and are far from realtime execution and thus cluster infrastructures may be required even for the development. For reducing the complexity while still taking advantage of the large-scale database, a possible approach is to objectively annotate the database and then divide the measured quality in levels before, then, randomly selecting HRCs from each level. This process may suffer from instable results as different HRCs are selected. In order to guarantee the stability of the objective measure against different HRC sets, a representative set of HRCs has to be selected. According to the authors best knowledge, there are currently no algorithms, even simple ones, that let researchers select an HRC set that reflects the behavior of the whole large-scale dataset. Therefore, a novel approach is described. First, create an extensive dataset and then reduce its size by subset selection rather than doing an expert selection of parameters.

It is often stated that the research domain of video quality estimation requires a large initial effort compared to other domains of digital signal processing. There are several reasons, notably the required in-depth knowledge from various domains, ranging from signal processing, image processing, video coding, and network transmission to statistical modeling and analysis, perception, psychology, and user experience, to name a few. To summarize, this work aims at improving the last steps in developing a new objective pre-

diction algorithm, the reasonable comparison to other state-of-the-art measures, the statistical analysis, and the fair comparison. As these topics form the last part of the development of a new algorithm, they are often neglected or underestimated.

In order to understand the importance of this topic, the paper takes the impact of small changes due to reasonable interpretations of the textual description in the most trivial video quality prediction algorithm as example.

The first step is to generate a common basis for the evaluation. As video coding is the most common degradation on which video quality measures are tested, the reproducible creation of a large-scale database is described in Section 2 that can either be identically computed in each lab or downloaded from the server. In both cases, hash value checksums assure that the same input sequence is used for testing objective algorithms. The second contribution in Section 3 is a comparison between the above-mentioned PSNR definitions on this large-scale database showing that even these slight implementation differences cannot be neglected. While PSNR is computationally inexpensive, it may not be feasible to calculate more complex algorithms on a huge dataset. Thus, the third contribution in Section 4 is a proposal for subset selection on large-scale databases in which the subsets are targeted to evaluate particular characteristics of the large-scale dataset that will be evaluated and compared to the full dataset in Section 5.2. We summarize our contribution in order to provide guidelines for reproducible publication of objective algorithms using our framework approach in Section 6. All software parts that are required for enabling our proposed reproducible research on objective measurement algorithms are made available in the associated SoftwareX publication [15].

2. Large-scale database description

The large-scale database [16, 17] used in this paper is designed to start from a reduced set of content types encoded using a very large set of encoding parameters leading to different processing chains, called Hypothetical Reference Circuits (HRC). More specifically, it is created using 10 source videos of 10 seconds long with a wide variation including a cartoon, sports content, nature, and user-generated content. The original High Definition (1920x1080) sources have also been downscaled to 1280x720 and 960x544 before further processing by a specific HRC .

As HRCs, only compression, so no packet loss, has been considered using a varied set of parameters. First of all, the bitrate has been fixed using two constant bitrate techniques (frame-based and coding unit based at 0.5, 1, 2, 4, 8, and 16 Mbps) and quantization parameter (QP) based (at QPs of 26, 32, 38, 46). Second, the Group Of Pictures (GOP) size has been varied between two (IBPBPBPBP) and eight (IBBBBBBBP) with additionally one low delay variation having a GOP size of four. Both open-GOP and closed-GOP structures have been considered at intra periods of 8, 16, 32, and 64. Finally, the number of slices has been varied (one, two, and four slices per picture) including a fixed slice size version providing 1500 bytes per slice. In total, 59520 sequences have been produced in this way, enabling a data analysis approach on video compression behavior.

In this work, the strategy has been to start with a limited set of sources and a large variety of compression parameters or HRCs in order to keep processing feasible. In a later phase, by identifying the most useful subset of HRCs an extension of the number of sources is planned against this re-

stricted set of HRCs. From all these encoded sequences, i.e., Processed Video Sequences (PVS), the frame-based and sequence average of Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index (SSIM) [18], and Visual Information Fidelity (VIFP) [19] have been calculated.

3. Video quality measures

Several objective video quality measures have been proposed in the scientific literature in the last decades. The first proposals relied on measuring the error introduced in the processed video with respect to a reference. This is the case of the Mean Squared Error (MSE), which measures the mean of the noise between a signal and a reference. More formally, for a single image of the video (i.e., a frame f) with $X \times Y$ pixels, indexed by i and j:

$$MSE_f = \sum_{i=1}^{X} \sum_{j=1}^{Y} (\hat{p}_{ij} - p_{ij})^2 \tag{1}$$

where p_{ij} is the value of the luminance component of the pixel in position i, j in the original reference image, and $\hat{p}_{ij}$ is the corresponding one in the processed image. For convenience, the MSE_f value is not used directly but often expressed in dBs through a logarithmic mapping, yielding the so-called $PSNR_f$ of the frame, commonly defined as:

$$PSNR_f = 10 \log_{10} \frac{\text{peak}^2}{MSE_f}. \tag{2}$$

The value of peak is commonly chosen as 255, the maximum value of the eight bit representation. When a sequence of frames is involved, as in a video sequence with N frames, two options are possible: either computing the mean of the noise over the whole sequence and doing the logarithmic

mapping at the end, or interpreting the $PSNR_f$ of each single frame as a quality indication and adopt a statistical approach, i.e., compute the first order moment (mean) of such values directly.

Therefore, even for such a simple measure such as a squared difference, different definitions are possible by just changing the temporal pooling strategy of the values for each frame. In the following, the term $PSNR_A$ (arithmetic mean) will be used when the MSE_f is averaged over all frames of the sequences, whereas $PSNR_G$ (geometric mean, here calculated as a sum in the logarithmic domain) refers to the mean of the $PSNR_f$ of each single frame k, indicated by $PSNR_{f_k}$. Formally:

$$PSNR_A = 10 \log_{10} \frac{255^2}{\frac{1}{N} \sum_{k=1}^{N} PSNR_{f_k}}, \tag{3}$$

$$PSNR_G = \frac{1}{N} \sum_{k=1}^{N} PSNR_{f_k}. \tag{4}$$

A clear mathematical definition of the measure in use as done in the previous equations would definitely help to solve ambiguities, but unfortunately the majority of the authors in the scientific literature just refer to "*PSNR*" without a clear reference to a well-defined formula or procedure. As a consequence, works from different authors cannot be easily compared even though all the other experimental parameters are the same. In the next section, we will investigate how one of the main sources of ambiguity, i.e., the temporal pooling strategy, might affect the final conclusions.

Even more, the constant for peak = 255 in the previous formulas is different in other definitions. For instance, in the ITU-R BT.601 recommendation the PSNR formulation requires to use the maximum brightness value of the

luminance equal to 235 and has been used for a *PSNR* definition in [11]. Just this simple uncertainty would immediately translate in a shift of all the previously defined *PSNR* values of $20 \log_{10} \frac{235}{255} \approx -0.71$ [20]. While this might have a limited impact when comparing results within the same research work that adopt a consistent definition, such uncertainty would immediately make all the results of one work look better or worse when compared to another one employing a different constant, even in absence of actual differences in the quality of the content itself.

3.1. Impact of different PSNR definitions

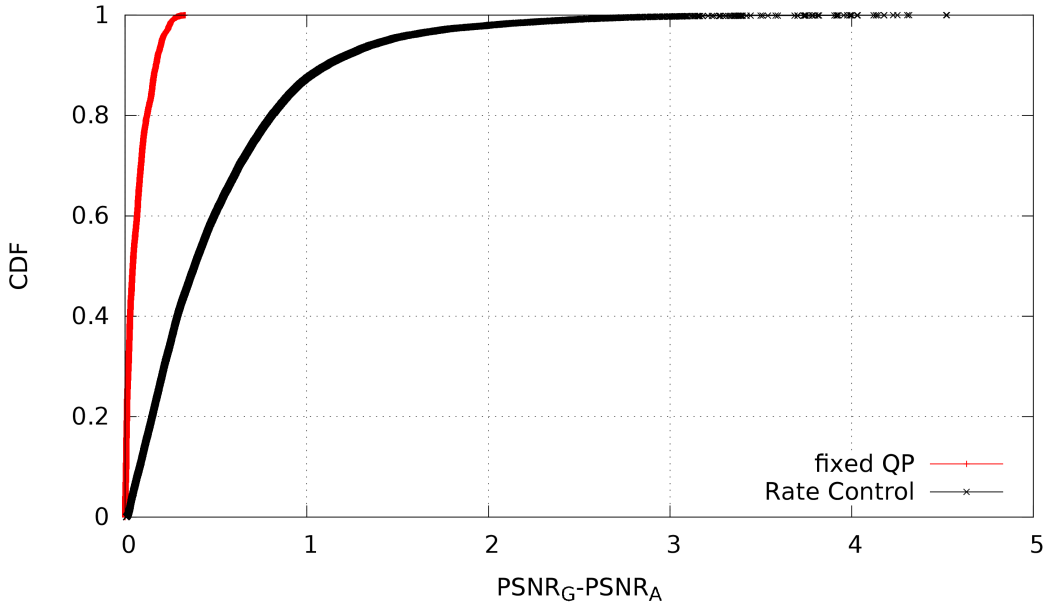

Figure 2: Cumulative distribution function of the $PSNR_G - PSNR_A$ difference.

The large-scale database provides an invaluable instrument to study such effect on a large scale. To this aim, we computed the $PSNR_G - PSNR_A$ value for each point, i.e., for each sequence, resolution and HRC available in the

database. Note that, by definition, $PSNR_A$ is always less than or equal to $PSNR_G$ due to the Jensen's inequality.

The cumulative distribution function (CDF) of such a difference is shown in Figure 2. First, note that the difference between the case of fixed QP encoding and the rate control is significant. This can probably be attributed to the fact that fixed QP encoding tends to keep the quality much more stable as the encoding progresses. Therefore, the MSE value presents less variability from frame to frame, hence the two different pooling strategies have a lower impact on the final result, the difference in terms of *PSNR* mostly stays below 0.5dB.

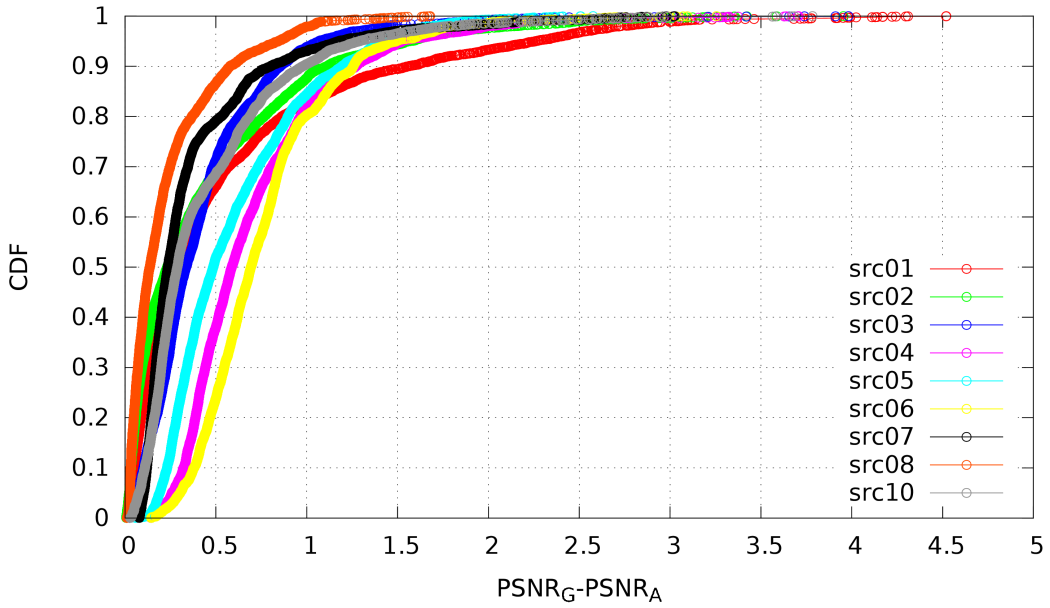

Figure 3: Cumulative distribution function of the $PSNR_G - PSNR_A$ difference (HRCs with active rate control only).

For the case in which a rate control algorithm is used, also the different content characteristics may play a significant role. This is shown in Figure 3

where the values are subdivided by content. Note that src09 is not included since its $PSNR_G$ values are infinite due to the presence of perfectly-coded black frames which yield zero MSE_f.

In case the rate control is used, the difference between $PSNR_G$ and $PSNR_A$ can be up to 4.5 dB on this large-scale database. This fact is extremely important since it shows that when comparing results among different research work it is absolutely necessary that the authors exactly define or reference the *PSNR* definition they employed for their analysis, otherwise there is a significant risk that the different results in the two works are simply due to the use of different definitions.

Another important implication is that when the quality as a function of the frame number for a given sequence is not almost constant, the temporal pooling strategy plays a significant role. In other words, it is possible that a sequence claimed to be better than another one on the basis of the sequence-level *PSNR* might present significant portions in which the reverse is true.

3.2. PSNR behavior as a function of the frame number

For such cases, and for any case in general, it would be interesting to provide additional information besides the sequence-level *PSNR*. For instance, just as an example, in this work we show how a simple indicator, namely the variance of the $PSNR_f$ of each single frame in the sequence, which will be referred to as σ^2_{PSNR} in the rest of the work, can be useful for this purpose. We computed this indicator for all the sequences, resolutions and HRCs available in the database, trying to correlate its behavior with the $PSNR_G - PSNR_A$ difference. While subjective quality assessment in general is outside the scope of this paper, it should be noted that a fluctuating

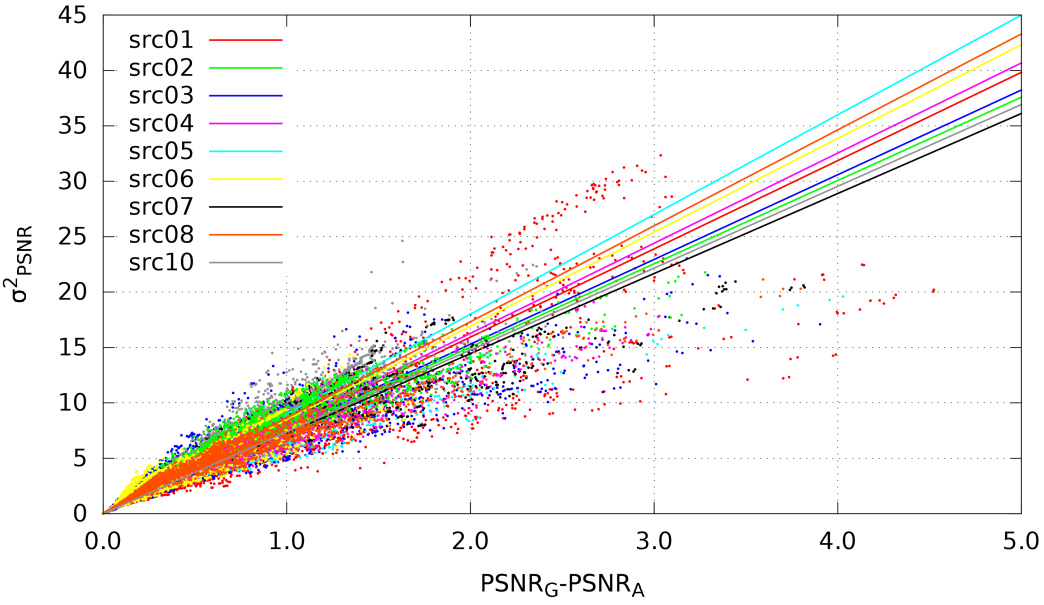

Figure 4: Variance of the PSNR of the frames in each sequence as a function of the $PSNR_G - PSNR_A$ difference (HRCs with active rate control only). The straight lines represent the interpolation of the points for each sequence.

temporal quality (higher σ^2_{PSNR}) usually annoys human observers. Results are shown in Fig. 4. As expected, higher σ^2_{PSNR} yields to higher difference. However, it is interesting to point out some notable points in the graph. For instance, if σ^2_{PSNR} it is lower than 2, in our database, which covers quite a wide range of coding conditions, there is no case which yields a *PSNR* difference higher than 0.5 dB. Conversely, for unlucky cases, if σ^2_{PSNR} it is just above 4, the $PSNR_G - PSNR_A$ difference can reach up to 1.5 dB. Therefore, depending on the application, a low σ^2_{PSNR} value could be used together with the sequence-level *PSNR* value to provide a further indication of the robustness of sequence-level *PSNR* comparisons regardless of the temporal pooling strategy.

For example, for the two extreme cases just considered, Fig. 5 and 6 shows

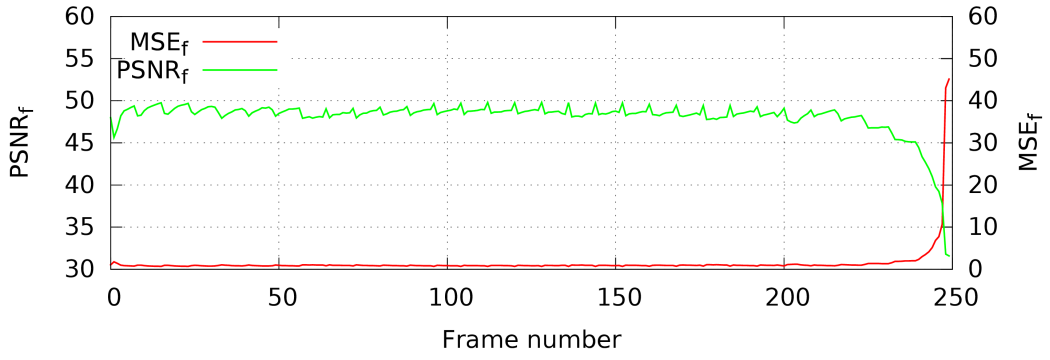

Figure 5: $PSNR_f$ and MSE_f as a function of the frame number for case $\sigma^2_{PSNR} < 2$ and $PSNR_G - PSNR_A \approx 0.5$ dB.

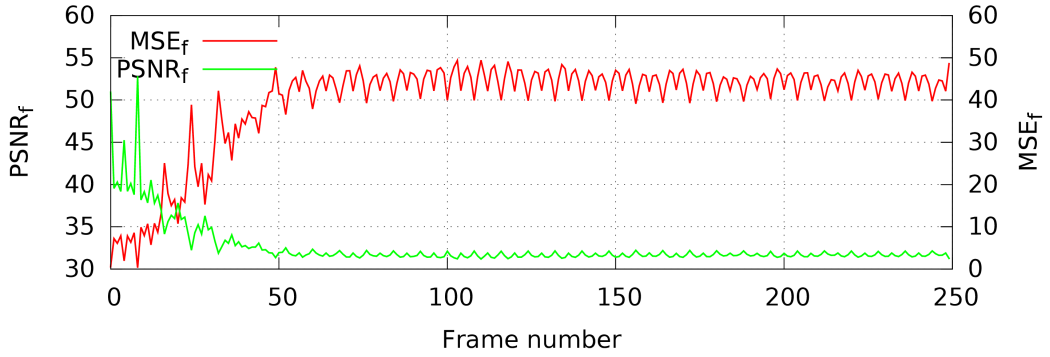

Figure 6: $PSNR_f$ and MSE_f as a function of the frame number for case $\sigma^2_{PSNR} > 4$ and $PSNR_G - PSNR_A \approx 1.5$ dB.

the behavior of the $PSNR_f$ as a function of the frame number. It is clear that for Fig. 6 there are sudden $PSNR_f$ variations at the end and at the beginning, which might signal that a sequence-level $PSNR$ value is probably not enough to perform quality evaluations over that particular sequence. A less extreme but equally interesting case is represented in Fig. 7, where large $PSNR_f$ variations are present from frame to frame, in addition to a sudden change of the rate control algorithm around frame 175.

Our simple analysis shows that, with the help of a large-scale database

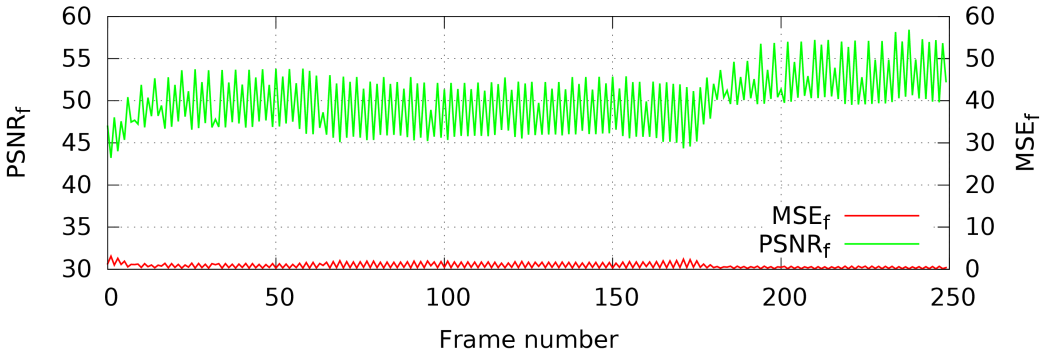

Figure 7: $PSNR_f$ and MSE_f as a function of the frame number for a case with strong $PSNR_f$ fluctuations. In this case, $\sigma^2_{PSNR} = 12.4$ and $PSNR_G - PSNR_A \approx 1.28$ dB.

representing a wide range of coding conditions, it is possible to define indicators and corresponding threshold that can suggest that sequence-level quality values, such as *PSNR*, have a reasonable reliability. However, building and analyzing such a large database requires a considerable computational effort.

The next section will focus on trying to find a methodology that allows to reduce the required number of samples in the dataset while not modifying the accuracy of the analysis. In other words, some more representative HRCs will be algorithmically identified so that the analysis can be performed only on that subset. The last part of the paper will discuss the effectiveness of this approach.

4. Goal-driven Large-scale Database Subset Generation

In Section 1, we discussed the limitations of the subjective experiments and the goals beyond the large-scale database. In this section, one goal beyond the generation of the large-scale database is discussed. Identifying target HRCs for a subjective experiment or for training a no-reference (NR) quality measure is challenging. Different correlation scores may be obtained

if one tests an objective video quality (VQ) measurement using two different databases. Table 1 shows an example. It shows the Pearson correlation coefficient for 25 experiments of 5 datasets that are selected from large-scale database. Three of them are randomly selected to cover different quality levels of PSNR. These dataset are used to train a model to predict the behavior of a full-reference quality measure (VQM) using pixel-based content features that are listed in [21]. A cross-testing experiment is conducted to evaluate the stability of each model. The stability is measured in terms of the performance of the validation with different datasets. As can be noticed from Table 1, Random datasets show unstable results. Random 1 based model shows unstable results for Random 3 in the testing. Random 2 shows unstable results for content-based dataset. Random 3 shows unstable results for the most data sets.

Table 1: The Pearson Correlation Coefficient for 25 experiments of 5 datasets that are selected from large-scale database.

		Tested on				
		Content-based	Quality/bitrate -based	Random 1	Random 2	Random 3
Trained on	Content-based	0.99	0.97	0.96	0.97	0.97
	Quality/bitrate -based	0.98	0.99	0.99	0.99	0.99
	Random 1	0.98	0.96	0.99	0.97	0.90
	Random 2	0.95	0.98	0.99	0.99	0.99
	Random 3	0.63	0.60	0.69	0.92	0.99

The reason could be the lack of content variety in the databases or the use of different HRCs in the experiments. Generally speaking, neither choosing different quality levels, i.e. different QPs or different bitrate budgets, nor

selecting different content types is the optimal way to generate the database. What we need is to choose the HRCs that cover a wide range of the targets. If the target is a quality measure, e.g. the PSNR, we need to select HRCs that cover all ranges of bitrate and quality. If the target is the content, we need to select HRCs that behaves differently with the contents. Dealing with the full set of 1984 HRCs for one resolution of a content is often computationally expensive. Therefore, in this section, a demonstration of two algorithms, Figure 8, to select a subset of the HRCs is discussed.

Figure 8 shows two flowcharts. Each elaborates the algorithm of selecting a subset of HRCs for a specific target. The left flowchart shows the selection that is optimized for the HRCs that cover different ranges of (PSNR, Bitrate). The right flowchart shows the selection that is optimized for the HRCs in terms of contents, i.e. those that behave differently with sources. The following subsections demonstrate the two algorithms.

4.1. Quality/Bitrate-driven HRCs Subset

In this subsection, the algorithm for selecting HRCs that cover a wide range of PSNR and bitrate values is demonstrated. Please refer to the flowchart in the left part of Fig. 8. At a specific quality level or in a specific quality range, the higher the quality the higher the bitrate.. This intuitive assumption is followed as the main idea of the selection process. On the other hand, this assumption might be deviated from this assumption when a specific encoding parameter is changed, such as slice parameters. This deviation is exploited to identify the behavior of each HRC in terms of quality and bitrate. The following steps are followed.

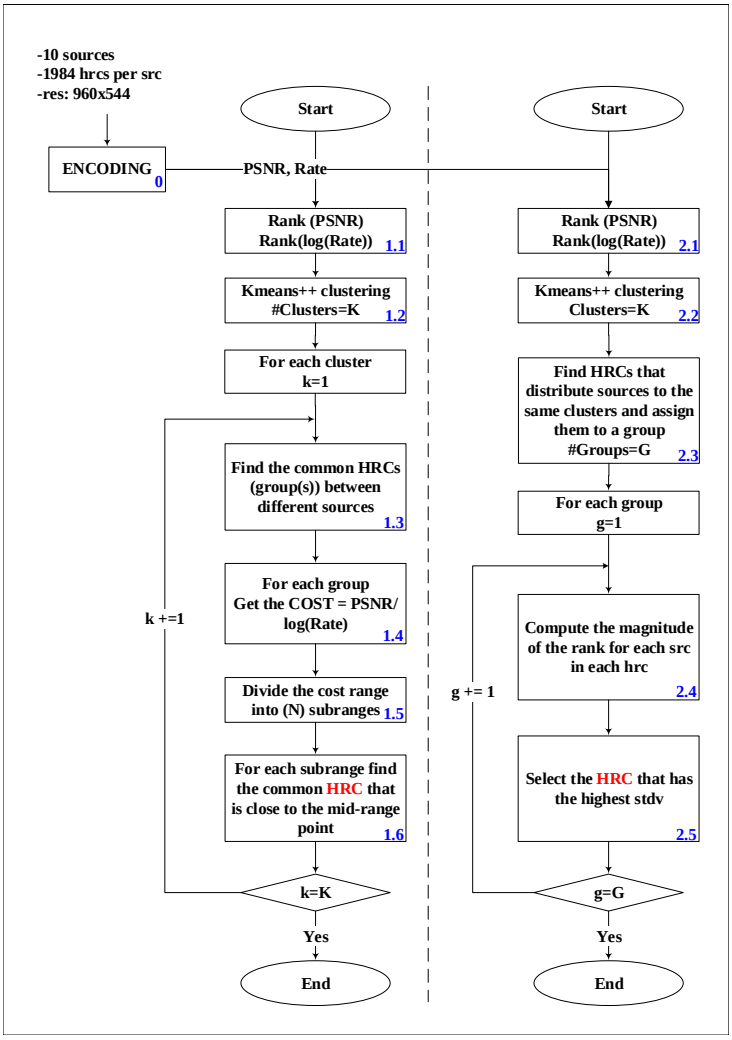

Figure 8: Two algorithms for selecting large-scale database subsets for different targets. Left) Selection is optimized on the HRCs that cover different ranges of (PSNR, Bitrate). Right) Selection is optimized on the HRCs in terms of contents (i.e. those that assign sources to different clusters)

- Step 0: all sources are encoded using all HRCs, then the quality measure and the bitrate are calculated for each HRC.

- Step 1.1: rank the HRCs according to the quality measure and the bitrate in ascending order. Fig. 9 shows all pairs of rank(PSNR, Rate)

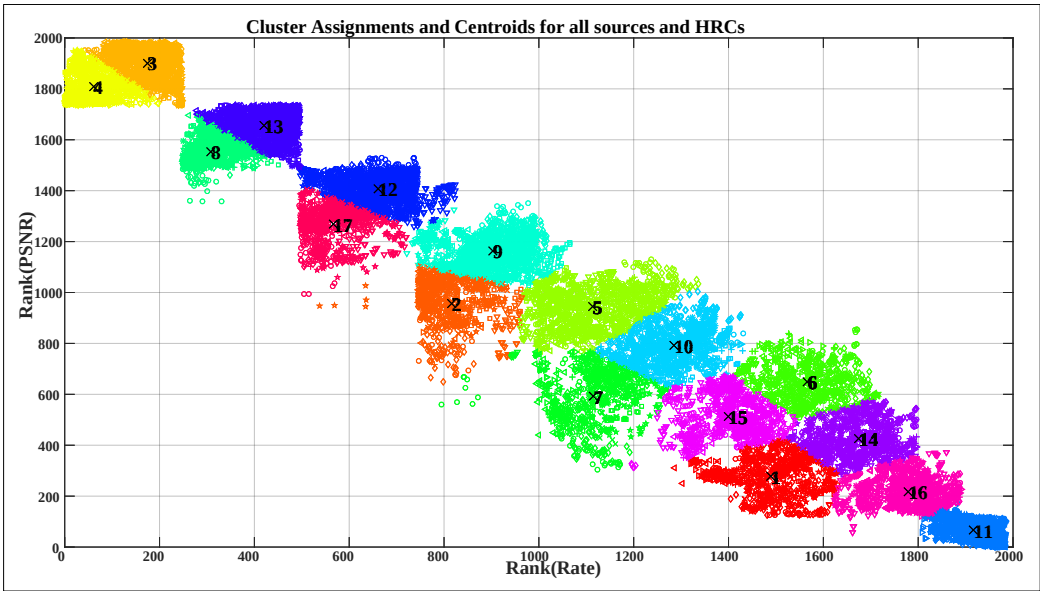

Figure 9: Rank(PSNR) against Rank(Rate) of all HRCs and contents. Numbers and colors indicate the cluster number.

of all sources while Figure 10 shows the pairs per content.

- Step 1.2: kmeans++ [22] clustering algorithm is used to cluster the HRCs according to their ranks in the quality measure. Different number of clusters are tested to select the optimal number of clusters. Figure 9 shows the 17 colored clusters and their centroids for all rank pairs for all HRCs while Fig. 10 shows the cluster assignments and their centroids per content. From these two Figures 9 and 10, one can observe the following. The intuitive assumption is stable in the very low quality and very high quality in all contents although there are changes in other encoding parameters. The deviation of this assumption in the middle range of quality is obvious and it points to the impact of other encoding parameters and to the content.

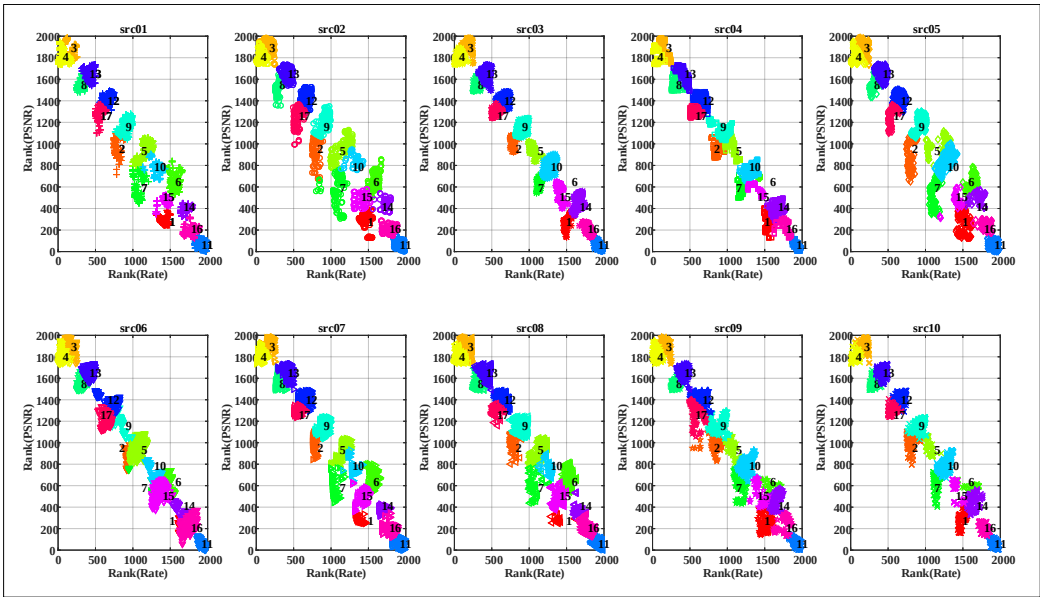

Figure 10: Rank(PSNR) against Rank(Rate) per content of all HRCs. Numbers and colors indicate the cluster number.

- Step 1.3: as it can be been observed from Fig. 10, each cluster has a different number of HRCs for different sources. For instance, SRC-03 does not have any HRCs that belong to cluster number 6 and has many of them in cluster 14. Therefore, in order to get all HRCs that cover a wide range of qualities and bitrates, each cluster is divided into groups. Each group represents the HRCs that are common between content sources. For instance, the first group contains the HRCs that are common between $1^{st}, 2^{nd}$, and 10^{th} content sources. The second group contains the HRCs of the 8^{th} source since there are no common HRCs with other sources. The third group contains the common HRCs of the rest of the sources.

- Step 1.4: for each group, the quality per rate ($Cost = PSNR/log(Rate)$)

is calculated to characterize each rank pair.

- Step 1.5: for each group, the *Cost* values are ordered and divided into N subranges. The value of N affects the number of HRCs to be selected for each group. The total number of HRCs is 32, 61, 83, and 109 if N equals to 1,2,3, and 4 respectively.

- Step 1.6: for each subrange in each group, compute the mid-subrange point and then select the closest HRC to this point. Therefore, all ranges of quality and bitrate values are covered.

4.2. Content-driven HRCs Subset

In this subsection, the algorithm for selecting the HRCs that behave differently with the contents is discussed, please refer to the flowchart in the right part of Fig. 8. The intuitive assumption that has already been discussed in the previous subsection, Section 4.1, is followed and exploited to identify the behavior of each HRC with different content sources. The following steps are followed.

- Steps 0, 2.1, and 2.2 are similar to steps 0, 1.1, and 1.2 of the quality/bitrate-driven HRCs algorithm respectively.

- Step 2.3: in this algorithm, we care about the behavior of each HRC with different contents. The HRCs that distribute source contents to same clusters are grouped. For instance, if one HRC distributes 3 contents out of 10 to clusters 2 and 5 respectively and another HRC distributes 4 contents out of 10 to clusters 2 and 5 respectively, then, the two HRCs belong to the same group. This decision is made because

it is observed that this can happen between neighboring clusters due to clustering error. In total, there are 97 groups for this dataset.

- Steps 2.4 and 2.5: for each group, in order to characterize each rank pair, the magnitude of rank of each content per HRC is computed and then the HRC that has the highest standard deviation is selected to represent the behavior of this group. Thereby, we reduce the effect of clustering error and ensure that redundant HRCs are avoided.

4.3. Selected HRCs for each subset

In this Section, the selected HRCs' qualities and bitrate(s) values are shown to confirm the output of the each algorithm of the subset generation. Figures 11, 12, and 13 show the quality measure (PSNR) against the logarithmic bitrate of all HRCs, quality/bitrate-driven HRCs, and content-driven HRCs per content source respectively. It can be observed that the quality/bitrate-driven HRCs cover the whole range of quality and bitrate values for each source content, while, on the other hand, the content-driven HRCs do not present the same behavior. Moreover, as it can be seen in Fig. 14 and 14, the distribution of quality and bitrate rank points are regularly distributed in quality/bitrate-driven subset over all source contents while, in content-driven subset, it can be noticed that the quality and bitrate rank points are not regularly distributed over all the contents and are distributed roughly in the area of middle qualities and middle bitrate(s). The standard deviation of the ranks' magnitudes of each HRC is another indicator that shows that the quality/bitrate-driven HRCs is not content representative. HRCs that have low standard deviation values in content-driven subset are

not selected, which means that there are similar-behavior HRCs of higher standard deviation that strongly distinguish the HRCs from others in terms of content.

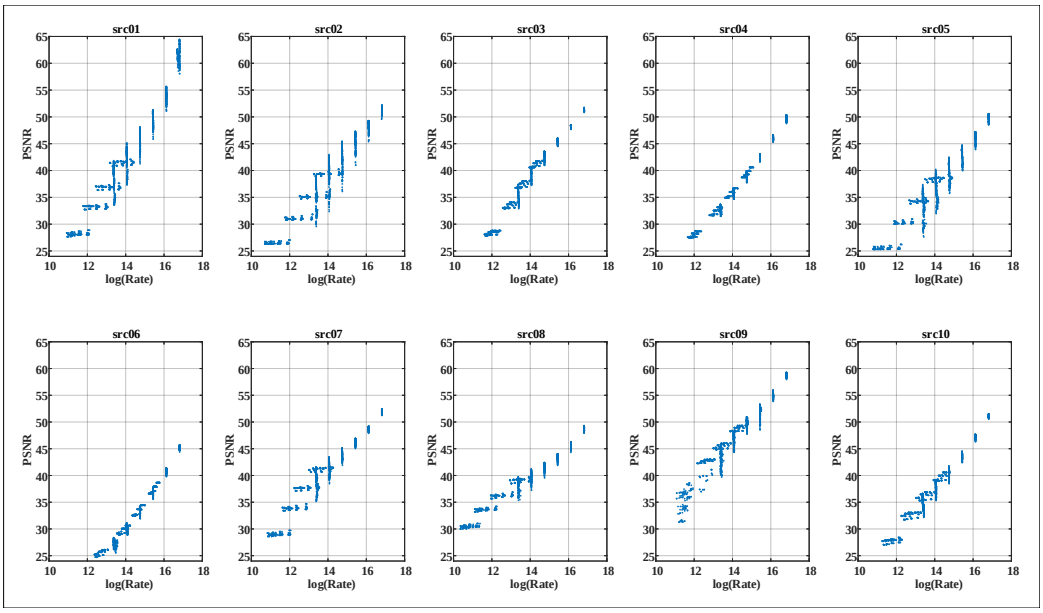

Figure 11: PSNR against log(Rate) of all HRCs per contents.

5. Analysis on reduced sets

5.1. Using Reduced sets in building prediction models

In Section 4 and in Table 1, we show the instability of the random-based datasets. In this subsection, we show the stability of the proposed subset selection. The table shows that the quality/distortion and content-based subsets are stable and have a high correlation. The quality/distortion-based subset covers a wide range of quality/bitrate values while this is not the case for content-based subset. Therefore, the training model has a better ability

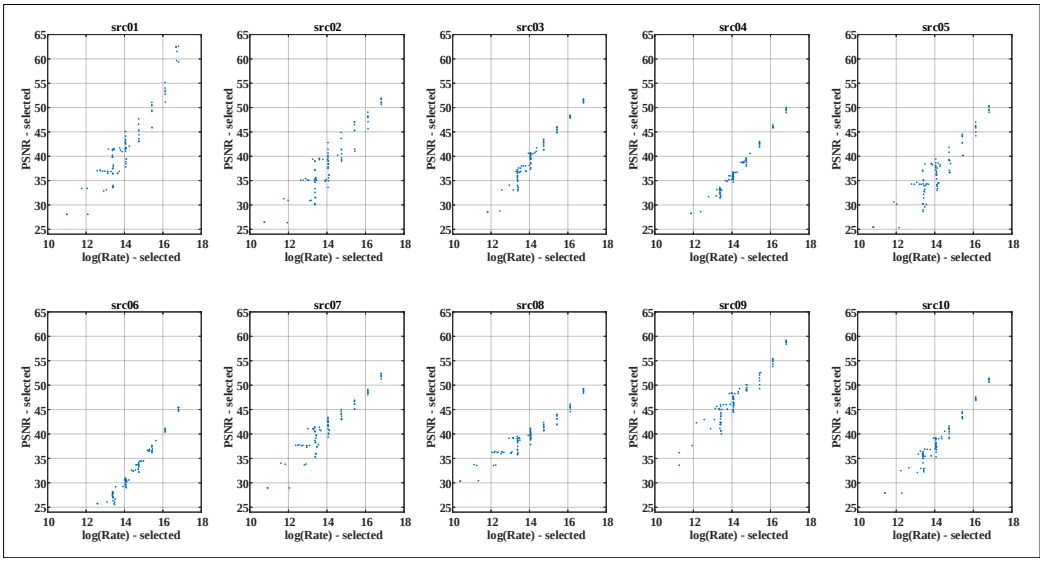

Figure 12: PSNR against log(Rate) of all HRCs per contents of selected HRCs for the quality/bitrate-driven subset.

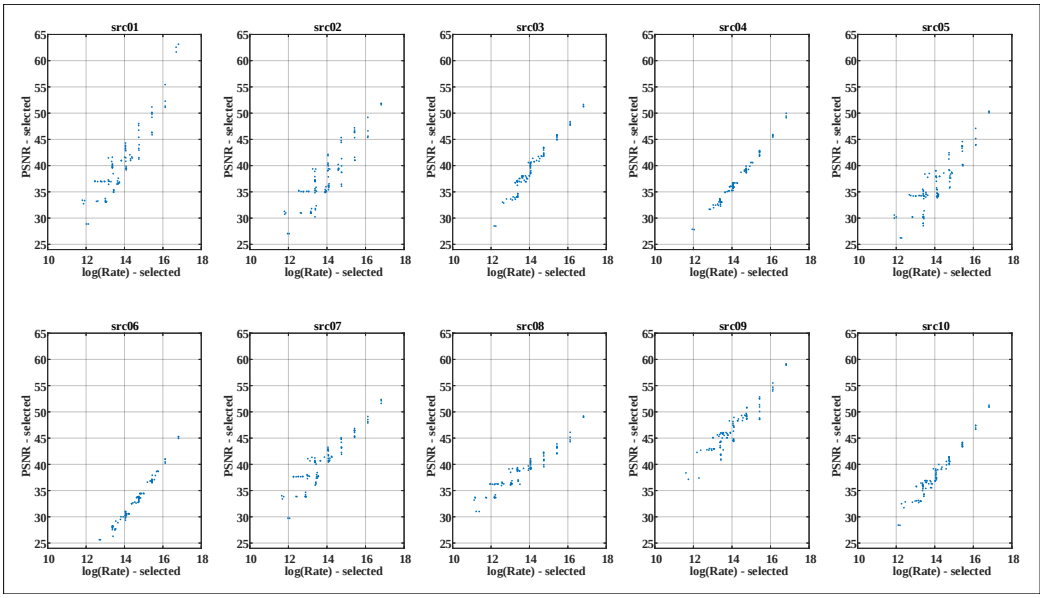

Figure 13: PSNR against log(Rate) of all HRCs per contents of selected HRCs for the content-driven subset.

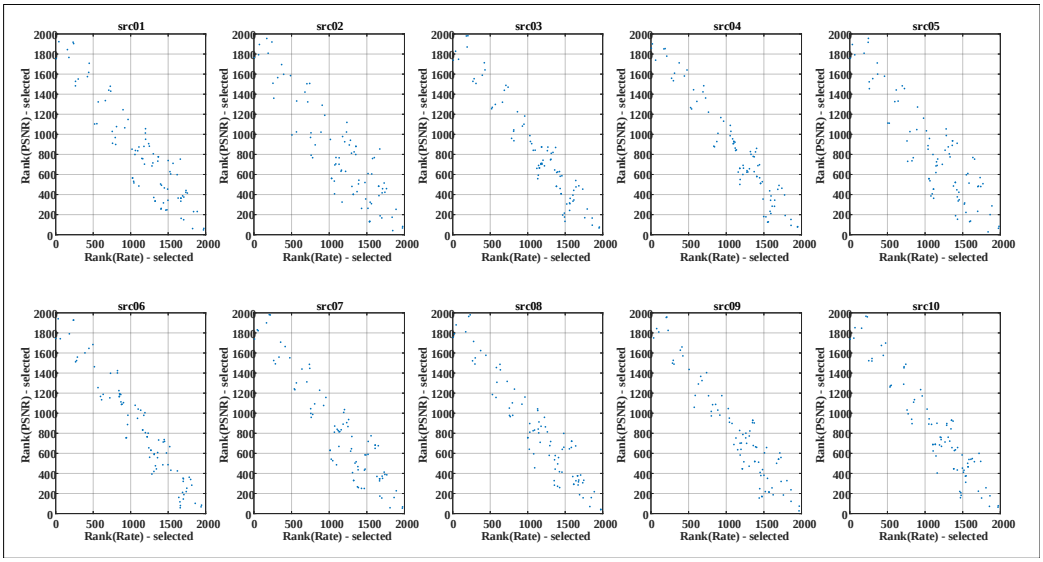

Figure 14: Rank of PSNR against Rank of log(Rate) of all HRCs per contents of selected HRCs for the quality/bitrate-driven subset.

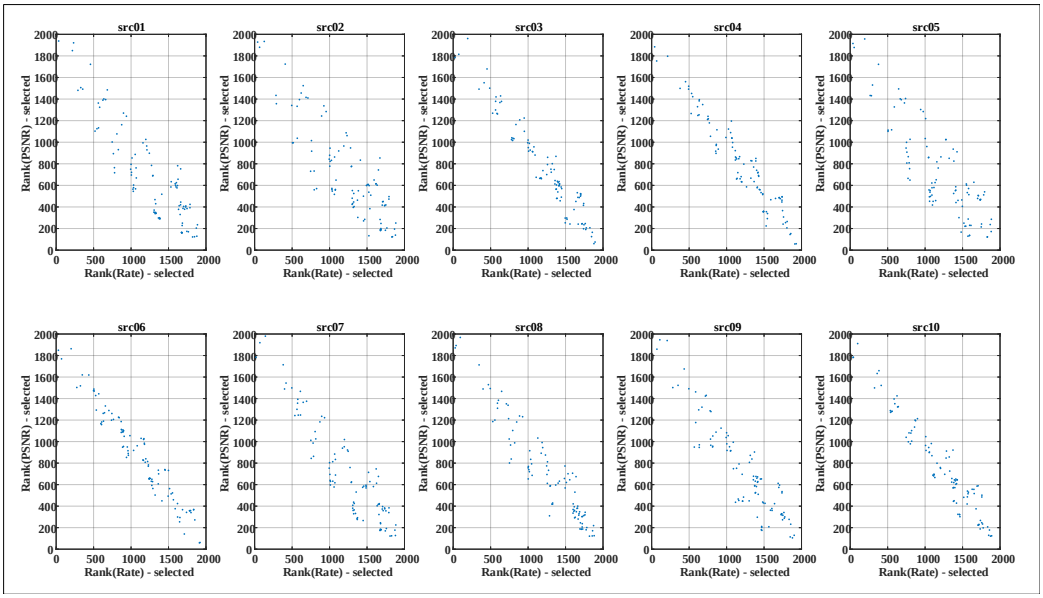

Figure 15: Rank of PSNR against Rank of log(Rate) of all HRCs per contents of selected HRCs for the content-driven subset.

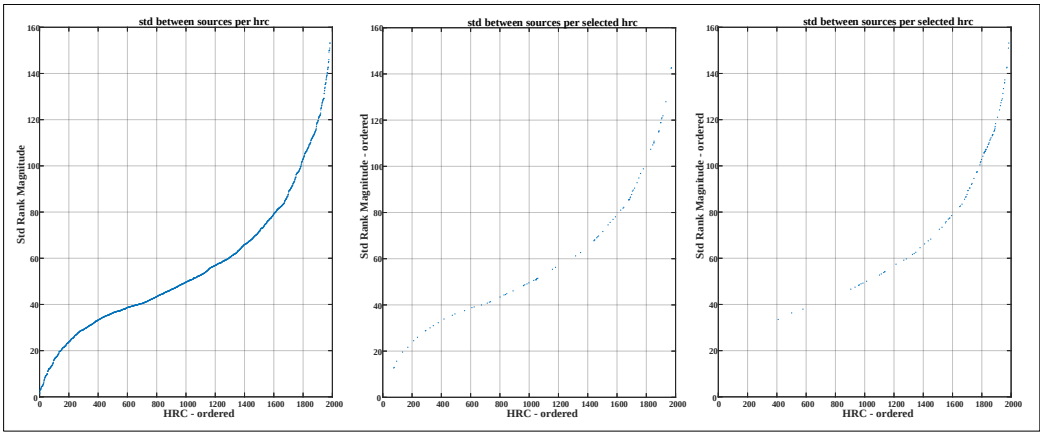

Figure 16: Standard deviation of rank magnitudes for each HRCs. left) all HRCs. center) Selected HRCs of quality/bitrate-driven subset. right) Selected HRCs of content-driven subset.

to predict the VQM value. Hence, this is an indication that the selection algorithm for quality/bitrate-driven works well.

5.2. PSNR analysis on reduced sets

As anticipated at the end of Sec. 3.2, it would be useful to find a representative subset of the database, and the HRCs in particular, that can allow to achieve most if not all the conclusions presented in the analysis of Sec. 3.1, in particular considering the σ^2_{PSNR} and the $PSNR_G - PSNR_A$ values.

The procedures highlighted in the previous sections have been applied to identify a subset of the HRCs in the original database. Two ideas have been pursued: one is based on representing HRCs that behave differently with different contents, the other one is based on representing all ranges of PSNR and bitrates. As a reference, we also considered random selections of the original HRCs instead of the ones provided by the analysis.

Results are shown in Fig. 17, 18 and 19 through scatter plots as already

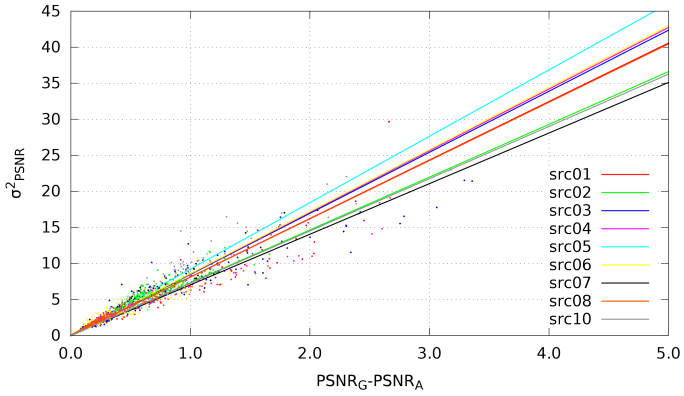

Figure 17: Subset of HRCs based on content.

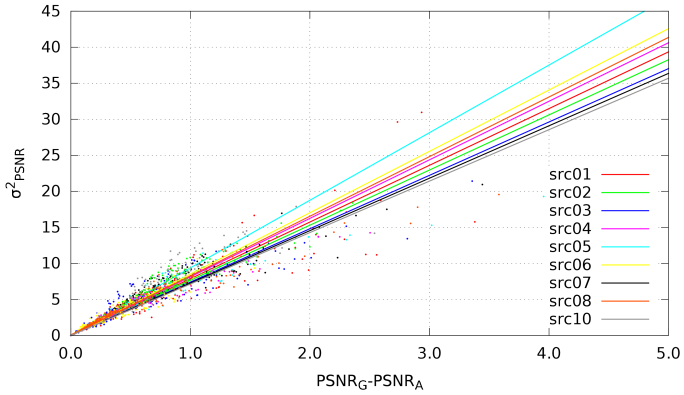

Figure 18: Subset of HRCs based on rate-distortion analysis.

done for the whole database. For better visualization, an interpolating line has been plotted for the points belonging to each source sequence.

In order to quantify the difference between the various subsets, we tried to match each point in the subset to be analyzed with all the points in the full database, shown in Fig. 4. Each single point in the full database has been assigned to the closest one in the subset on the basis of the distance on the graph. Therefore, for each point in the subset it is possible to compute an average distance from all the represented points, as well as the average

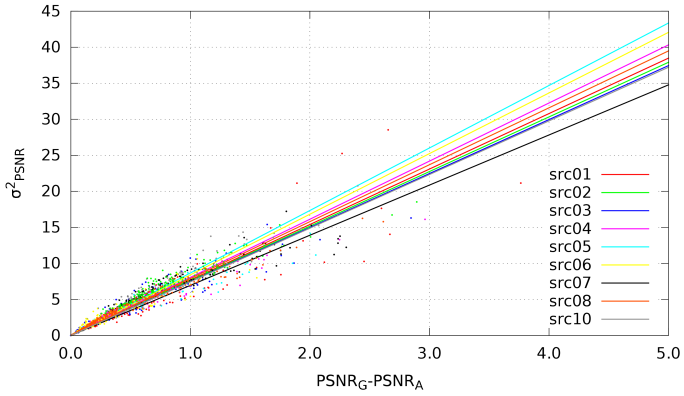

Figure 19: Random subset #1 of HRCs.

Table 2: Representativeness of different subsets: average distances.

Subset	Avg distance
Rate-Distortion	0.027156
Random #1	0.027994
Random #2	0.030429
Random #3	0.025605
Content	0.029160
Random #1	0.025195
Random #2	0.026157
Random #3	0.024695

distance considering all the points in the original graph. The latter can be interpreted as a sort of quantization noise of the original set of points onto the new ones in the subset.

Values are reported in Table 2. Note that, to ensure fairness in the comparison, the number of HRCs in the selected subset (either Rate-Distortion or Content) is the same as in the randomly selected subset.

On the basis of the average distance value, there seems to be no clear indication on a preferred subset. This might be due to the fact that the

original large database spans over a wide range of values in terms of rate (from 0.5 Mbit/s to 16 Mbit/s) and distortion (25 to 60 dB), therefore with a relatively low number of randomly sampled HRCs it is possible to cover a large variety of conditions.

It might happen that specifically focusing on selecting a subset with the average distance as the ultimate performance metric results could be better (i.e., the average distance could be lower), but the aim of this work has been to focus on a more general case to show the usefulness of the large dataset and how, in principle, this set of data can be reduced without sacrificing data representativeness.

6. Discussion and guidelines on publishing objective quality estimation algorithms

It has become evident that the research on quality estimation algorithms may be improved by the availability of implementations that accompany the textual description. Furthermore, performance evaluation on large-scale databases may allow for more fine-grained analysis of the the performance. During the development and in the verification and validation phase, subsets may need to be used due to computational complexity issues and a careful selection may be required that can be aided by appropriate subset clustering algorithm such as the one described in this publication. As validation requires subjective ground truth data, the subset selection may be used in order to reduce the subjective assessment burden. This approach may be generalized as for new technologies often experts select appropriate content for subjective testing using their experience in the field instead of algorithmic methods.

Their experience may however be biased or important observations may be ignored. A practical example from our dataset is that one video sequence contained black frames in the cross-fade of two shots that were perfectly reconstructed and therefore led to an infinite frame PSNR, therefore leading to an infinite sequence PSNR when averaging over all PSNR values. These black frames are visually unobtrusive and the content may have been removed from the database because similar contents were present.

In order to render all results as reproducible as possible, the following guidelines should be respected when developing and publishing algorithms:

Textual description must be as precise as possible, e.g. referring to other algorithms requires a reference to another publication of its complete description or to a software package that should be stated with the exact version number.

Test vectors on a validation dataset have to be published. This dataset may be any publicly available dataset that promises a longterm availability such as the VQEG datasets including the large-scale dataset described in this publication. Configuration parameter values shall be given and the output shall be recorded.

An implementation of the algorithm has to be made publicly available unless prohibited by circumstances that need to be described in the paper. The executable or source code shall preferably be submitted to a scientific journal such as SoftwareX or be made available on several different platforms such as the institutional and private homepage, VQEG's JEG group, software repositories such as GitLab or Source-

Forge, or storage spaces such as EUDAT [23]. The software should be made available both as source code and compiled version. If possible, a Virtual Machine, such as VirtualBox or VMWare including the software and accompanying packages and libraries should be deposited such as to allow for reproducibility for a duration of at least ten years.

Versioning is required for both the source code and the executable. Often bug-fixes, library or system updates, or changes to the algorithm's trained parameters after the publication change the results on the validation data, affecting the reproducibility of the algorithm and potentially retrograding the reliability of the algorithm.

Competitor's algorithms, i.e. algorithms that are used for performance evaluation and comparison have to be cited with the exact version, configuration parameters, and any other required information for reproducing the results.

Cross-checking of the algorithm's correctness by an independent organization is strongly encouraged. Similar to core experiments in the video coding community [24] the cross-checking organization should only use the textual description and should at least verify that the provided executable is capable of reproducing the test vectors. Such cross-checking should be stated in the publication.

When respecting such guidelines, continuous improvement of algorithms in video quality prediction becomes feasible. The framework published in the accompanying SoftwareX part of this publication allows for straight forward implementation regardless of whether the algorithm concerns video pre- or

postprocessing, isolated quality indicators such as framerate, combination algorithms of existing indicators, complete prediction algorithms, sequence subset selection algorithms, or performance measures for comparing objective measures.

Similar approaches have been used in other communities. The most notable example is the video coding community that has, since the 1990s continuously improved the block-based hybrid video coding scheme and achieved, from H.262 to H.265 (HEVC), a reduction of the datarate by $\left(\frac{1}{2}\right)^3 = \frac{1}{8}$ at the same visual quality [25]. The domain of depth reconstruction from stereoscopic images has largely benefited from the effort of the Middlebury College where verification datasets, performance results and publication pointers are stored [26]. In the same direction, competitions or grand challenges are organized by conferences [27], workshops or independent organization [28]. However, these efforts are often time limited and may not be suited for continuous long-term improvements.

7. Conclusion

In this work reproducibility of objective video quality measures has been tackled in several steps. Firstly, a large scale database containing about 60,000 HEVC coded video sequences has been employed to investigate how different implementations of textual definitions may affect the reproducibility of performance measures. This has been exemplified with commonly used variations of PSNR. In a detailed analysis, this difference has been put in relation with the variance of the PSNR computed for each frame. Hence, the work showed the paramount importance of having strict reproducibil-

ity of the research in video quality evaluation since even small uncertainties in the exact measure definitions may yield completely different conclusions when comparing different research works. Secondly, since performing such an analysis for a large-scale database might be impractical, techniques to select significant subsets of the coding parameters have been introduced. The results showed that an accurate selection can significantly reduce the complexity while preserving the variety of the results seen on the large database. The subset selection algorithm has been described in detail and its implementation has been made available in order to allow for reproducibility. Improved algorithms that use more sophisticated clustering criteria or clustering algorithms may therefore compare results to our approach using the same or a different large-scale dataset. Thirdly, we proposed a software framework for reproducible research in video quality evaluation which has been presented in our SoftwareX accompanying paper [15]. This framework allows for isolated improvements in each step without requiring in-depth knowledge of the other parts of the processing chain. This enables experts from various domains to contribute. For example, an expert in perceptual modeling may evaluate the performance of his algorithm on the large-scale database or a data mining specialist may improve the subset selection algorithm, or a statistician may add further performance measures in the future.

Acknowledgment

Some of the computational resources have been provided by HPC@POLITO (http://www.hpc.polito.it). Some parts of this work are supported by the Marie Skłodowska-Curie under the PROVISION (PeRceptually Optimised

VIdeo CompresSION) project bearing Grant Number 608231 and Call Identifier: FP7-PEOPLE-2013-ITN. The research activities described in this paper were partially funded by Ghent University, imec, Flanders Innovation & Entrepreneurship (VLAIO), the Fund for Scientific Research Flanders (FWO-Flanders), and the European Union. Some aspects of this work were carried out using the STEVIN Supercomputer Infrastructure at Ghent University.

References

[1] M. Gaubatz, metrix_mux v.1.0 (2007).
URL http://ollie-imac.cs.northwestern.edu/%7Eollie/GMM/code/metrix%5Fmux/

[2] A.P. Hekstra, J.G. Beerends et al., PVQM – a perceptual video quality measure, Signal Processing: Image Communication 17 (10) (2002) 781–798.

[3] E. Masala, PVQM video quality measure (2014).
URL http://media.polito.it/jeg

[4] T. Ghalut, H. Larijani, A. Shahrabi, Content-based video quality prediction using random neural networks for video streaming over LTE networks, in: IEEE Intl. Conf. on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), IEEE, 2015, pp. 1626–1631.

[5] T. Ghalut, H. Larijani, Non-intrusive method for video quality prediction over lte using random neural networks (RNN), in: 9th International

Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP), IEEE, 2014, pp. 519–524.

[6] H. Chen, X. Yu, L. Xie, End-to-end quality adaptation scheme based on QoE prediction for video streaming service in LTE networks, in: 11th International Symposium on Modeling & Optimization in Mobile, Ad Hoc & Wireless Networks (WiOpt), IEEE, 2013, pp. 627–633.

[7] B. Feitor, P. Assuncao, J. Soares, L. Cruz, R. Marinheiro, Objective quality prediction model for lost frames in 3D video over TS, in: IEEE International Conference on Communications Workshops (ICC), IEEE, 2013, pp. 622–625.

[8] A. G. Davis, D. Bayart, D. S. Hands, Hybrid no-reference video quality prediction, in: IEEE International Symposium on Broadband Multimedia Systems and Broadcasting BMSB, IEEE, 2009, pp. 1–6.

[9] J. Wang, S. Wang, Z. Wang, Quality prediction of asymmetrically compressed stereoscopic videos, in: IEEE International Conference on Image Processing (ICIP), IEEE, 2015, pp. 3427–3431.

[10] S. Yoo, D. Kim, D. S. Kim, H. Choo, Quality prediction of mobile video service based on radio access network log data, in: 11th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), IEEE, 2015, pp. 599–605.

[11] NTIA / ITS, A3: Objective Video Quality Measurement Using a Peak-Signal-to-Noise-Ratio (PSNR) Full Reference Technique, ATIS T1.TR.PP.74-2001.

[12] ITU Study Group 9, J.340 : Reference algorithm for computing peak signal to noise ratio of a processed video sequence with compensation for constant spatial shifts, constant temporal shift, and constant luminance gain and offset, Recommendation ITU-T J.340.

[13] S. Wulf, U. Zlzer, About the imperfection of objective quality metrics on high-definition video content, in: 2013 IEEE 15th International Workshop on Multimedia Signal Processing (MMSP), 2013, pp. 384–389. doi:10.1109/MMSP.2013.6659319.

[14] Q. Huynh-Thu, M. Ghanbari, Scope of validity of PSNR in image/video quality assessment, Electronics Letters 44 (2008) 800–801.

[15] A. Aldahdooh, E. Masala, G. Van Wallendael, M. Barkowsky, Reproducible research framework for objective video quality measures using a large-scale database approach, Elsevier SoftwareX.

[16] G. Van Wallendael, N. Staelens, E. Masala, M. Barkowsky, Full-HD HEVC-encoded video quality assessment database, in: Ninth International Workshop on Video Processing and Quality Metrics (VPQM), 2015.

[17] M. Barkowsky, E. Masala, G. Van Wallendael, K. Brunnstrom, N. Staelens, P. Le Callet, Objective video quality assessment – towards large scale video database enhanced model development, IEICE Transactions on Communications E98-B (1) (2015) 2–11.

[18] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assess-

ment: from error visibility to structural similarity, IEEE Transactions on Image Processing 13 (4) (2004) 600–612.

[19] H.R. Sheikh, A.C. Bovik, Image information and visual quality, IEEE Transactions on Image Processing 15 (2) (2006) 430–444.

[20] M. Barkowsky, J. Bialkowski, A. Kaup, Subjective Video Quality Assessment for Low Bitrate Multimedia Applications (in German), in: ITG Fachbericht 188: Elektronische Medien 2005, VDE-Verlag, 2005, pp. 169–175.

[21] A. Aldahdooh, E. Masala, O. Janssens, G. Van Wallendael, M. Barkowsky, Comparing simple video quality measures for loss-impaired video sequences on a large-scale database, in: Eighth International Conference on Quality of Multimedia Experience (QoMEX), 2016, pp. 1–6. doi:10.1109/QoMEX.2016.7498941.

[22] D. Arthur, S. Vassilvitskii, k-means++: The advantages of careful seeding, in: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms, Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[23] Eudat collaborative data infrastructure.
URL `https://www.eudat.eu/`

[24] M. Wien, High Efficiency Video Coding: Coding Tools and Specification, Springer, Signals and Communication Technology (2015) p. 19.

[25] J. R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, T. Wiegand, Comparison of the Coding Efficiency of Video Coding Standards;Including

High Efficiency Video Coding (HEVC), IEEE Transactions on Circuits and Systems for Video Technology 22 (12) (2012) 1669–1684. doi:10.1109/TCSVT.2012.2221192.

[26] D. Scharstein, H. Hirschmller, Y. Kitajima, G. Krathwohl, N. Nesic, X. Wang, P. Westling, High-resolution stereo datasets with subpixel-accurate ground truth., in: X. Jiang, J. Hornegger, R. Koch (Eds.), GCPR, Vol. 8753 of Lecture Notes in Computer Science, Springer, 2014, pp. 31–42.

[27] 2017 ieee international conference on image processing: Grand challenges.
URL http://2017.ieeeicip.org/GrandChallenges.asp/

[28] Kaggle: Your home for data science.
URL https://www.kaggle.com/