NO-REFERENCE VIDEO QUALITY METRIC FOR HDTV BASED ON H.264/AVC BITSTREAM FEATURES

Christian Keimel, Manuel Klimpke, Julian Habigt and Klaus Diepold

Technische Universität München, Institute for Data Processing,
Arcisstr. 21, 80333 Munich, Germany
christian.keimel@tum.de, msk@mytum.de, jh@tum.de, kldi@tum.de

ABSTRACT

No-reference video quality metrics are becoming ever more popular, as they are more useful in real-life applications compared to full-reference metrics. Many proposed metrics extract features related to human perception from the individual video frames. Hence the video sequences have to be decoded first, before the metrics can be applied. In order to avoid the decoding just for quality estimation, we therefore present in this contribution a no-reference metric for HDTV that uses features directly extracted from the H.264/AVC bitstream. We combine these features with the results from subjective tests using a data analysis approach with partial least squares regression to gain a prediction model for the visual quality. For verification, we performed a cross validation. Our results show that the proposed no-reference metric outperforms other metrics and delivers a correlation between the quality prediction and the actual quality of 0.93.

Index Terms— H.264/AVC, HDTV, 1080p25, subjective testing, visual quality, video quality metric, no-reference metric.

1. INTRODUCTION

The current focus in the research on video quality metrics is shifting more and more to no-reference metrics as in many practical application scenarios undistorted references are not available due to overall bitrate constraints. Quite a few no-reference metrics have already been proposed so far to measure the visual quality of encoded video. Often, the aim is to develop a quality metric that works independently of a specific coding technology. With appropriate no-reference algorithms, features that describe certain visual properties of the video sequences, like blocking or bluring, are extracted and then combined in a no-reference metric. One example of such a metric, especially for HDTV, is our previous contribution in [1].

Even though these metrics work quite well, they have the disadvantage that the encoded video needs to be decoded before feature extraction algorithms can be used. This can lead for example to an increased overall computational complexity in a network if the quality should not only be determined at the terminating devices, but also at arbitrary points in between. Another approach is therefore to avoid the complete decoding and directly extract useful features for a video quality metric from the bitstream. Such metrics are of course limited in their scope due to the necessary adaption to a certain coding technology and its bitstream format. While this may seem to be a major shortcoming, one has to consider that in most of the current practical applications for HDTV, H.264/AVC coding technology is used predominantly.

In this contribution we will therefore present a no-reference video quality metric for HDTV based on H.264/AVC bitstream features. We use a data analysis approach with Partial Least Squares Regression (PLSR) to design a video quality metric with features extracted from the encoded H.264/AVC bitstream. We focus on the HDTV 1080p25 format, representing progressive video sequences in the full HDTV resolution of 1920×1080 pixels at 25 frames per second.

In related contributions to no-reference video quality metrics, Eden estimates the PSNR of interlaced HDTV video sequences with H.264/AVC bitstream features in [2] whereas Slanina et al. in [3] estimate the PSNR for videos in CIF resolution. Rossholm and Lövström not only estimate PSNR in [4], but also other video quality metrics for videos in CIF resolution from the bitstream. In [5], Lee et al. use bitrate, QP and deblocking filter parameters for quality prediction of QCIF resolution videos, but no different coding structures were considered. Another approach is the combination of bitstream features and features extracted from the decoded video sequences in hybrid metric as proposed for interlaced HDTV by Sugimoto et al. in [6].

This contribution is organized as follows: firstly, we will discuss the extraction of the features from the H.264/AVC bitstream, before introducing how the metric is built using PLSR. Then we will shortly describe the subjective testing. After presenting and discussing the results, we will conclude with a short summary.

2. FEATURE EXTRACTION

In order to build our metric, we first need to extract features from the H.264/AVC bitstream that describe the properties of the encoded video sequence. We assume in the following that the *byte stream* representing the Network Abstraction Layer (NAL) according to Annex B of the H.264/AVC standard is available and that any channel coding done for transmission has already been removed. Note that we do not have to further decode and reconstruct each frame: it is sufficient if we only reverse the entropy encoding of the bitstream, as we are not interested in the completely decoded frame, but rather in the properties of the bitstream. We then parse those NAL units (NALU) containing information about the coded frames, the so-called Video Coding Layer (VCL). Each VCL-NALU describes one slice of the current frame. A slice in turn is partitioned into multiple macroblocks, which again can be divided into submacroblocks. Hence we parse three successive layers as shown in Fig. 1.

Before descending to the slice level, we extract the *profile*, *level* and *entropy encoding type* for the complete video sequence. Then we extract the following features for each slice in the video

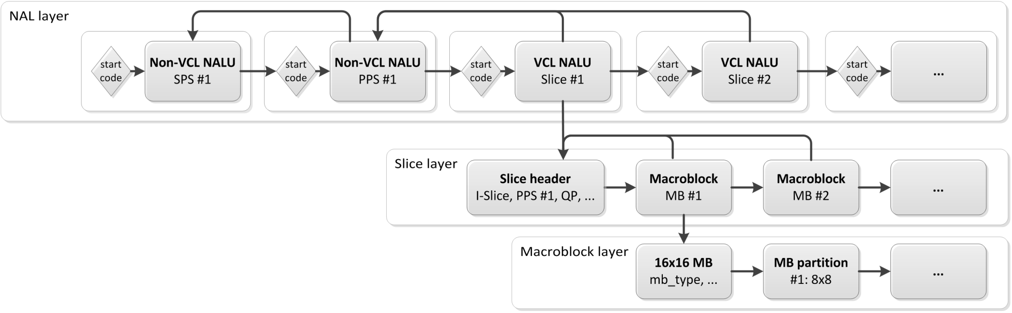


Fig. 1: H.264/AVC bitstream: overview over the different layers

sequence:

- bits per slice (*BPS*)
- average QP per slice (*QPA*)
- average, minimum and maximum motion vector length per slice (*MV, MVMin, MVMax*)
- average and maximum motion vector error per slice (*MVd, MVdMax*)

These features are then pooled temporally over all frames by calculating the average, median, standard deviation, minimum, maximum 10% and 90% percentiles. We denote this for each feature f as f_{Avg}, f_{Med}, f_{SD}, $f_{Min/Max}$ and $f_{10/90}$. Also we check if the QP changed at the macroblock layer from the initial QP in a slice. We then calculate the average difference qpd_{Avg} between initial and changed QP over all slices, but also the percentage of slices with constant QP over all macroblocks (*%qpd*).

Furthermore, we determine the percentage of the different slice types, but also of the different macroblock types and their subdivision over the whole video sequence:

- percentage of I-, P- and B-slices (*%I,%P, %B*)
- percentage of intra, inter and skip coded macroblocks (*%Intra, %Inter, %Skip*)
- percentage of intra macroblocks with 16×16, 8×8 and 4×4 subdivision (*%I16x16, %I8x8, %I4x4*)
- percentage of inter macroblocks with 8×8 and 4×4 subdivision (*%P8x8, %P4x4*)

All in all, we thus get 64 different features from each video sequence. While the large number of extracted features seem to imply an increased computational complexity, note that we only parsed the intrinsic parameters of the H.264/AVC bitstream and temporally pooled them. Some of these features were also used by Rossholm and Lövström in [4]. For feature extraction, we used a modified decoder of the H.264/AVC reference software [7].

3. BUILDING THE METRIC USING DATA ANALYSIS

After extracting the features from the bitstream, we employ data analysis methods to generate a model for predicting the perceived visual quality the features. In this approach, we do not assume a-priori specific relationships between the features and the visual quality, but rather gain the relationships by analyzing the available data. Firstly, we construct a data matrix $\mathbf{X}$, where the rows correspond to data from individual sequences or data points and the columns represent the features. The visual quality values that were determined in subjective tests are represented by the $n \times 1$ column vector $\mathbf{y}$. With n sequences and m features, $\mathbf{X}$ is a $n \times m$ matrix. In our example, the matrix contains 64 features and 32 sequences. Our aim is to find the unknown $m \times 1$ weight vector $\mathbf{b}$, mapping the features to the visual quality

$$\mathbf{y} = \mathbf{X}\mathbf{b}. \tag{1}$$

Although we extracted a large number of parameters, not all will be useful in predicting the visual quality of a video sequence, as we will see shortly. A principal overview of the model building with data analysis is given in Fig. 2.

3.1. Partial Least Squares Regression

In this contribution we use the Partial Least Squares Regression (PLSR) to estimate the weights $\mathbf{b}$. PLSR is an extension of the principal component regression method (PCR). For PCR, the data matrix $\mathbf{X}$ is first subjected to a PCA, and then for selected principal components (PC) a regression on $\mathbf{y}$ is done. The disadvantage of PCR is that the PCs best suited to represent $\mathbf{X}$, carrying the structure of the videos, are not necessarily the same PCs best suited to explain the variance in $\mathbf{y}$, describing the quality variation of the videos. Therefore, the modeling with PLSR is done simultaneously on $\mathbf{X}$ and $\mathbf{y}$, ensuring PCs that explain the variance in both $\mathbf{X}$ and $\mathbf{y}$ best.

By using PLSR, we obtain an estimation $\hat{\mathbf{b}}$ of the weight vector $\mathbf{b}$ and thus can write the quality estimation $\hat{\mathbf{y}}$ for $\mathbf{y}$ as

$$\hat{\mathbf{y}} = \mathbf{1}\hat{b}_0 + \mathbf{X}\hat{\mathbf{b}} + \mathbf{e}, \tag{2}$$

where $\hat{b}_0$ describes the offset and $\mathbf{e}$ the estimation error of the model. The video quality of unknown video sequences with a $1 \times m$ feature vector $\mathbf{x}_u$ can then be predicted as

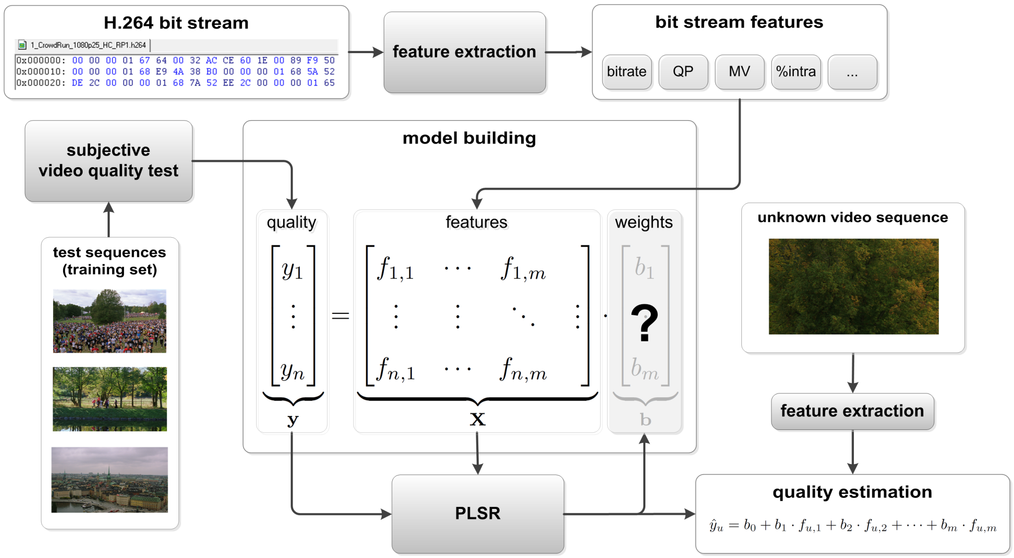
$$\hat{y}_u = \hat{b}_0 + \mathbf{x}_u\hat{\mathbf{b}}. \tag{3}$$

For more information on PLSR, we refer to [8].

3.2. Cross Validation

It is important to use separate data sets for training and validation of the designed metric. If we used the same data for training and validation, it would lead to overly optimistic prediction models as discussed in [1].

Therefore we perform a cross validation and split the available data set into four different subsets and apply the PLSR on each of these subsets. Each subset consists of all data sets excluding one data set of the four video sequences introduced in the following section. Consequently, we compute four different PLSR models, allowing us to verify the results for each sequence using a model that did not include this particular sequence during the calibration.



```
0x000000: 00 00 00 01 67 64 00 32 AC CE 60 1E 00 89 F9 50
0x000010: 00 00 00 01 68 E9 4A 38 B0 00 00 00 01 68 5A 52
0x000020: DE 2C 00 00 00 01 68 7A 52 EE 2C 00 00 00 01 65
```

Fig. 2: Model building with bitstream features and PLSR

Table 1: Weights for selected features and different models

Feature	CrowdRun$^{(a)}$	ParkJoy	InToTree	OldTownCross
QPA$_{SD}$	-0.016	-0.011	-0.020	-0.029
MVd$_{10}$	-0.018	-0.004	-0.005	-0.007
qpd$_{Avg}$	-0.041	-0.026	-0.043	-0.053
%Inter	0.003	0.001	0.003	0.003

$^{(a)}$ Video sequence left out during the model building step

The PLSR and cross validation on the four subsets reveal that only 48 of the features are relevant for the model and thus $m = 48$, as the weights of the other 12 do not have any significant influence on the predicted quality. In particular, we can exclude the feature *MVMin* completely. We determined an optimal number of 3 PCs to efficiently describe the variance in both $\mathbf{X}$ and $\mathbf{y}$ at the same time. Table 1 shows the weights for selected parameters.

3.3. Sigmoid Correction

On the extremes of the voting scale in subjective testing, at very good or bad quality, the test results exhibit a nonlinear nature. Thus ratings do not reach the boundaries of the scale, but saturate much earlier. Therefore, we correct the prediction values $\hat{y}$ slightly, using a fixed sigmoid nonlinear correction in order to emulate this behavior [1]. The sigmoid correction of $\hat{y}$ is given as

$$\hat{y}_S = 1.0/(1 + e^{(-(\hat{y}-0.5)/0.2)}). \tag{4}$$

This function is not adapted to the actual data, but is rather a fixed part of the quality metric. Hence, $\hat{y}_S$ represents the final prediction result of our video quality metric.

4. SUBJECTIVE TESTING

For subjective testing we encoded four different video sequences with the H.264/AVC reference encoder at multiple bitrates. We used two significantly different encoder settings, each representing the complexity of various devices and services. The first setting is chosen to simulate a low complexity (LC) H.264/AVC encoder: many tools that account for the high compression efficiency are disabled. In contrast to this, we also used a high complexity (HC) setting that aims at getting the maximum possible quality out of this coding technology. We used the H.264/AVC reference software [7] version 12.4. Selected encoding settings are listed in Table 2. We used the test sequences *CrowdRun*, *ParkJoy*, *InToTree* and *OldTownCross* from the SVT high definition multi format test set in the 1080p25 HDTV format. We selected four bitrates from 5.4 Mbit/s to 30 Mbit/s. This resulted in a quality range from ‘not acceptable’ to ‘perfect’, corresponding to mean opinion scores (MOS) between 0.19 and 0.96 on a scale ranging from 0 to 1. In total, we have thus 32 different data points. The tests were performed in the video quality evaluation laboratory of the Institute for Data Processing at the Technische Universität München in a room compliant with recommendation ITU-R BT.500 [9]. A professional, 24 inch LCD reference display with a native resolution of 1920 × 1080 pixels was used. We used a variation of the DSCQS test method, the Double Stimulus Unknown Reference (DSUR) test method with a scale from 0 to 1, representing the worst and best quality. For more details on the subjective testing, we refer to [10].

Table 2: Selected encoder settings

	LC	HC
Encoder	JM 12.4	
Profile&Level	Main, 4.0	High, 5.0
Slices per Frame	1	1
Reference Frames	2	5
R/D Optimization	Fast Mode	On
Search Range	32	128
B-Frames	2	5
Temporal Levels	2	4
Intra Period	500 ms	
8x8 Transform	Off	On

Table 3: Performance of the quality prediction

Metric	Pearson	Spearman	RMSE$^{(a)}$
Proposed metric	0.93	0.95	0.08
No-reference metric [1]	0.91	0.85	0.09
PSNR	0.72	0.69	0.15
SSIM [11]	0.85	0.82	0.12
VQM Annex D of [12]	0.84	0.78	0.11

$^{(a)}$ After first order fitting for all comparison metrics, no fitting for both no-reference metrics

5. RESULTS

The prediction results of our metric are presented in Fig. 3 and Table 3. Besides the Pearson and Spearman rank order correlation coefficents, we also provide the root mean squared error (RMSE) between predicted and actual visual quality. For comparison, we included the results of our no-reference metric presented in [1], but also the results of two well-known full-reference video quality metrics: SSIM [11] and the VQM according to Annex D of ITU-T J.144 [12]. For the latter, the general model was used. SSIM was evaluated on all three channels of the YC_BC_R color space.

The results show that the proposed metric outperforms our previous no-reference metric in [1] slightly with respect to the Pearson correlation and the RMSE, but especially well with respect to the Spearman rank order correlation. This is not surprising, as we build our model only for H.264/AVC, compared to [1], where we also considered alternative, wavelet-based coding technologies and therefore the metric is not optimized only for H.264/AVC. Also, we achieve better results than the metric in [5]. Moreover, our metric outperformed all full-reference metrics. However, note in Fig. 3 that the prediction quality is worse at the lower end of the quality scale, caused by the lack of data points in the training set in this area.

6. CONCLUSION

We presented a no-reference video quality metric based on H.264/AVC bitstream features for HDTV and used PLSR to build a prediction model in combination with subjective tests.

Our results show that the metric not only outperforms common full-reference metrics, but also our previously presented, more universal no-reference metric. Although this metric is only applicable to H.264/AVC encoded video, it covers most practical applications. Still, the metric can be further improved in future work by either including a larger data set or by considering a larger set of different encoding parameters.

The set of H.264/AVC bitstreams, the modified decoder for feature extraction and additional data is available at `www.ldv.ei.tum.de/videolab`.

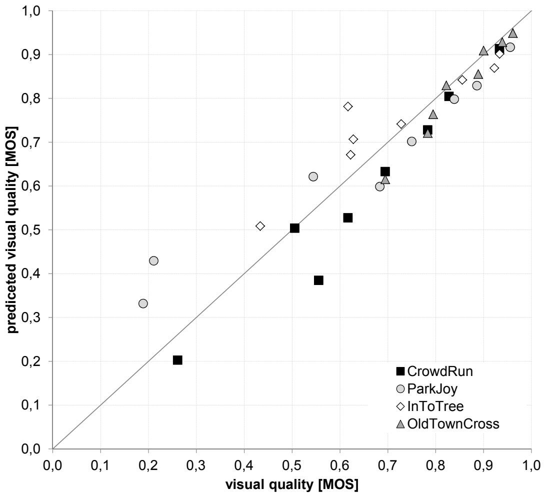


Fig. 3: Prediction results of the proposed no-reference metric

7. REFERENCES

[1] C. Keimel, T. Oelbaum, and K. Diepold, "No-reference video quality evaluation for high-definition video," *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pp. 1145–1148, Apr. 2009.

[2] A. Eden, "No-reference estimation of the coding PSNR for H.264-coded sequences," *IEEE Trans. Consum. Electron.*, vol. 53, no. 2, pp. 667 –674, May 2007.

[3] M. Slanina, V. Ricny, and R. Forchheimer, "A novel metric for H.264/AVC no-reference quality assessment," in *EURASIP Conference on Speech and Image Processing, Multimedia Communications and Services*, 2007, pp. 114 –117.

[4] A. Rossholm and B. Lövström, "A new video quality predictor based on decoder parameter extraction," in *International Conference on Signal Processing and Multimedia Applications*, 2008, pp. 285–290.

[5] S.-O. Lee, K.-S. Jung, and D.-G. Sim, "Real-time objective quality assessment based on coding parameters extracted from H.264/AVC bitstream," *IEEE Trans. Consum. Electron.*, vol. 56, no. 2, pp. 1071 –1078, May 2010.

[6] O. Sugimoto, S. Naito, S. Sakazawa, and A. Koike, "Objective perceptual video quality measurement method based on hybrid no reference framework," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*, 2009, pp. 2237 –2240.

[7] K. Sühring. (2007) H.264/AVC software coordination. [Online]. Available: http://iphome.hhi.de/suehring/tml/index.htm

[8] H. Martens and M. Martens, *Multivariate Analysis of Quality*. Wiley & Sons, 2001.

[9] *ITU-R BT.500 Methodology for the Subjective Assessment of the Quality for Television Pictures*, ITU-R Std., Rev. 11, Jun. 2002.

[10] C. Keimel, J. Habigt, T. Habigt, M. Rothbucher, and K. Diepold, "Visual quality of current coding technologies at high definition IPTV bitrates," in *Multimedia Signal Processing (MMSP), 2010 IEEE International Workshop on*, 2010, pp. 390 –393.

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

[12] "ITU-T J.144. objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference," Mar. 2004.