IMPROVING THE PREDICTION ACCURACY OF VIDEO QUALITY METRICS

Christian Keimel, Tobias Oelbaum and Klaus Diepold

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

ABSTRACT

To improve the prediction accuracy of visual quality metrics for video we propose two simple steps: temporal pooling in order to gain a set of parameters from one measured feature and a correction step using videos of known visual quality. We demonstrate this approach on the well known PSNR. Firstly, we achieve a more accurate quality prediction by replacing the mean luma PSNR by alternative PSNR-based parameters. Secondly, we exploit the almost linear relationship between the output of a quality metric and the subjectively perceived visual quality for individual video sequences. We do this by estimating the parameters of this linear relationship with the help of additionally generated videos of known visual quality. Moreover, we show that this is also true for very different coding technologies. Also we used cross validation to verify our results. Combining these two steps, we achieve for a set of four different high definition videos an increase of the Pearson correlation coefficient from 0.69 to 0.88 for PSNR, outperforming other, more sophisticated full-reference video quality metrics.

***Index Terms*—** PSNR, video quality metric, AVC/H.264, Dirac, temporal pooling.

1. INTRODUCTION

The vast majority of video quality metrics utilises a combination of features having a known or suspected relationship to the subjectively perceived visual quality. Differences are in the selected features, how they are measured, and how they are combined. Most video quality metrics either average all single feature measurements of each frame and then combine these mean values or first combine the features of each frame into one quality value per frame and then average these values over time [1, 2]. Hence one feature measurement results in only one parameter. Averaging, however, may not be sufficient to describe the distribution of this feature both in space and time.

In this contribution we therefore propose to improve the prediction accuracy of a video quality metric by taking more than one parameter derived from a single feature measurement. We propose to evaluate an extended set of parameters gained from one feature using different temporal pooling functions. Also the parameters should not be based on assumptions about the human visual system (HVS), but selected according to which parameter describes the statistical distribution of the feature best.

Secondly, we propose an additional correction step that is applied to the quality prediction gained by measuring and combining features [3]. It improves the prediction accuracy by estimating the relationship between the output of the metric and the perceived visual quality for the video. This estimation allows to precisely determine the quality of previously unknown videos, even if the model that is used is not very well suited for this video. The correction step is generic in the sense, that it can be applied to any type of visual quality metric. In [3] the most important restriction was that we had to generate additional videos using an encoder similar to the encoder used for the videos under test. Contrary to this, we suggest in this contribution that the restriction does not necessarily apply, and that the additional videos can be generated using different coding technology.

The basic feature we choose to demonstrate the effectiveness of the two proposed methods is the Peak Signal to Noise Ratio (PSNR). Since the beginning of video transmission PSNR has been in nearly universal use as a predictor for the visual quality of processed video even though it is well known that PSNR does not match the subjective visual quality of videos very well.

Our contribution is organised as follows: we explain pooling, selection and combining of the parameters, before introducing the correction step. Then we describe the subjective test and present the results. Finally we conclude with a short summary.

2. TEMPORAL POOLING: FEATURES TO PARAMETERS

Extracting a set of features from a video can be seen as a different representation of this video. The content of the video is not represented using the single values of each pixel or a combination of motion information and residual error, but by features that relate to visual quality. Due to the temporal dimension of video, features extracted for a certain time instance e.g. one frame, have to be combined into one quality value by temporal pooling. Most metrics do this by calculating the mean value over time, only few metrics use different pooling functions for the single features [4]. This pooling step results in a number of parameters which are derived from the extracted features. Calculating only the mean value for a feature is not sufficient to describe the statistical temporal and spatial distribution of this feature. Thus describing videos only with the mean is too coarse and hence more parameters are needed.

We use PSNR as a feature that can be extracted easily from videos. A simple adaptation to the HVS is made by evaluating the PSNR in the YUV color space, denoted as PSNR^Y. Using PSNR as a quality indicator requires the calculation of PSNR values on a frame-by-frame basis. The resulting series of values is then averaged to determine a single value for the entire video sequence. To simplfy matters this is usually only done in the luma component (PSNR$^Y_{Mean}$). Calculating PSNR$^Y_{Mean}$, however, does not result in a useful representation of the videos' visual quality.

Therefore we propose not only to evaluate the mean value, but also the minimum, maximum, standard deviation, the 90% and the 10% percentiles, denoted as PSNR$^Y_{Min}$, PSNR$^Y_{Max}$, PSNR$^Y_{sDev}$, PSNR$^Y_{90}$ and PSNR$^Y_{10}$, respectively. Temporal variations of PSNR can differ significantly between different sequences. Hence we also calculate the difference in PSNR between to consecutive frames dPSNR^Y and apply the same temporal pooling as for PSNR^Y. As

Table 1: Correlation values for PSNR based parameters to results of subjective tests

PSNR Parameter	Pearson Correlation
PSNR$^{Y}_{Mean}$	0.688
PSNR$^{U}_{Mean}$	0.588
PSNR$^{V}_{Mean}$	0.603
PSNR$^{Y}_{Min}$	0.753
PSNR$^{U}_{Min}$	0.606
PSNR$^{V}_{Min}$	0.635
PSNR$^{Y}_{10}$	0.720

a result, the feature PSNR^{Y} is represented by a set of 12 different values derived from PSNR. A similar process is applied to the two chroma channels resulting in PSNR^{U} and PSNR^{V}. Such a brute-force approach results in a total number of 36 PSNR parameters. Table 1 shows the Pearson correlation values of some of these PSNR based parameters to the results of the conducted subjective test. Take note that PSNR$^{Y}_{Mean}$ is in our test set not the best predictor for visual quality, but that PSNR$^{Y}_{Min}$ appears to be better suited.

3. SELECTING AND COMBINING PARAMETERS

Although a large number of parameters can be calculated only a few of those will be useful in predicting the visual quality of a video sequence. Many do not have any impact on a prediction model and some may even be harmful to prediction process. Using an extended set of parameters as described in section 2, we construct a data matrix $\mathbf{X}$. The rows correspond to data from individual sequences and the columns represent the parameters. The visual quality values that were determined in subjective tests are represented by the column vector $\mathbf{y}$. Assuming K sequences and L parameters, $\mathbf{X}$ has the dimension $K \times L$. In our example the matrix contains 36 parameters and 48 sequences, as four different videos were encoded using three different encoders at four different bit rates.

First we reduce the number of parameters to reach a robust prediction model by analyzing simple statistical properties: some parameters do not show significant variation across the different videos or are not correlated to the visual quality vector $\mathbf{y}$ at all. Thus we reduce the number of parameters from 36 to 24, omitting all parameters for dPSNR^{U} and dPSNR^{V}.

Then we analyse the contribution of the remaining parameters to the visual quality prediction. We use principal component analysis (PCA) to determine a more compact and stable representation of $\mathbf{X}$ and a partial least squares regression (PLSR) to find the relationship between $\mathbf{X}$ and $\mathbf{y}$. As we are interested in the variation of the parameter values, the values are centered around the mean and scaled to achieve a standard deviation of 1.0 in order to avoid that small, but important variation in one parameter is covered by large, but less important noise in a another parameter.

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 (PCs) 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 is done simultaneously on $\mathbf{X}$ and $\mathbf{y}$, ensuring PCs that explain the variance in $\mathbf{X}$ and $\mathbf{y}$ at the same time [5].

Table 2: Most relevant weights for the different models

Model[a]	None	CR	IT	OTC	PJ
dPSNR$^{Y}_{10}$	-1,391	-0,826	-0,932	-1,445	-0,942
PSNR$^{Y}_{Min}$	1,089	1,045	1,102	1,048	0,844
dPSNR$^{Y}_{Min}$	0,959	0,578	0,474	1,078	0,838
PSNR$^{Y}_{90}$	0,794	0,432	0,892	0,703	0,518
dPSNR$^{Y}_{90}$	0,787	0,619	0,276	0,706	0,857

[a] Excluded Sequence

We split the available data set into four different subsets and applied the PLSR on each of these subsets. Each subset consists of all data sets excluding the data set pertaining to one of the four sequences. 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 step. If we obmitted this cross calibration approach, it would lead to overly optimistic prediction models as we will show in section 6. Not only the weights for the selected PSNR parameters were determined using a cross validation approach, but also the selected parameters themselves.

The PLSR on the four subsets and on the complete data set reveals that the parameters PSNR$^{U}_{Max}$, PSNR^{V}, PSNR$^{V}_{Min}$, PSNR$^{V}_{10}$, PSNR$^{V}_{90}$, dPSNR^{Y}, dPSNR$^{Y}_{sDev}$ and dPSNR$^{Y}_{Max}$ have no relevance for the model, as the weights of these PSNR values are very close to 0. Still, 16 parameters are remaining that have a relevant influence on the variance of the entries of $\mathbf{X}$ and $\mathbf{y}$. We determined an optimal number of 5 PCs to efficiently describe the data in $\mathbf{X}$ and $\mathbf{y}$ simultaneously. The remaining error in $\mathbf{X}$ could be reduced further by including more PCs. But this would result in an overfitted model exhibiting reduced ability to predict the quality of unknown videos. The visual quality is then calculated according to

$$\text{PSNR}^{M} = b_0 + \sum_{j=1}^{16} w_j \text{PSNR}_j, \qquad (1)$$

where the weights w are provided by the PLSR. Table 2 shows the weights for the preprocessed parameters that do have the highest influence on the new quality metric PSNR^{M}.

Reaching a stable model using only four sequences is practically impossible if not all of the four sequences show similar content characteristics, as can be seen by the weights in Table 2 for the different models. A lower bound of 20 well selected sequences should be the minimum for a general model. Even if each of the models is suboptimal in predicting the respective verification sequence, the Pearson correlation coefficient for the PSNR^{M} metric is 0.80, outperforming not only standard PSNR$^{Y}_{Mean}$, but also PSNR$^{Y}_{Min}$ which showed the highest correlation to visual quality so far.

4. CORRECTING THE QUALITY PREDICTION

Most visual quality metrics exhibit an almost linear realationship between the estimated quality of the metrics and the actual visual quality if single source videos are considered [6]. This can also be seen exemplarily in Fig. 2 for one model. In [3] we introduced a generic method to increase the prediction accuracy of video quality metric by estimating the parameters of a linear model for this relationship.

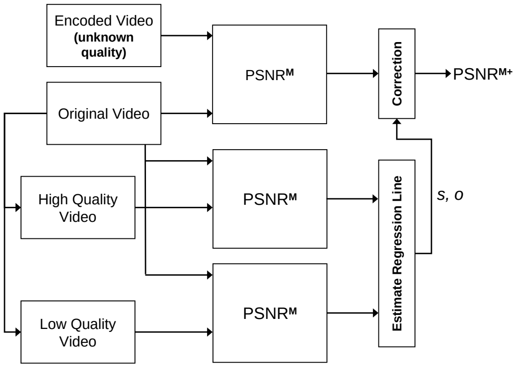

Fig. 1: Quality prediction system

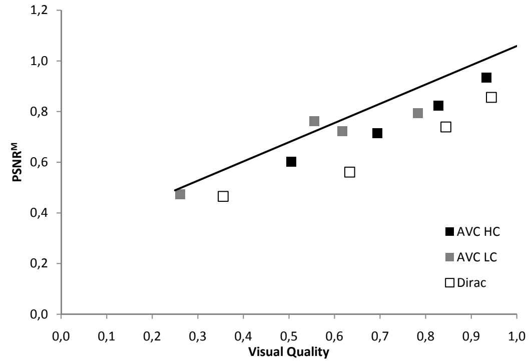

Fig. 2: Detailed results for CrowdRun, PSNR^M (Model "CR"); line shows the estimated regression line

This is done by generating two video sequences with a known visual quality. This is done by encoding the original video with a established encoder configuration producing videos at a known quality. A very simple encoder using a fixed quantization parameter (QP) is suitable for this task: a very high value of the QP results in a video having a very low visual quality v_{low}, a low QP in a video with high visual quality v_{high}. For simplicity the values for v_{low} are set to 0.25 and for v_{high} to 1 on a scale from 0 and 1. We then estimate the slope s and offset o of the regression line describing the almost linear relationship between the output of the metric and the visual quality using the metric output PSNR$^M_{low}$ and PSNR$^M_{high}$ for these two videos:

$$s = \frac{\text{PSNR}^M_{high} - \text{PSNR}^M_{low}}{v_{high} - v_{low}} \quad (2)$$

$$o = \text{PSNR}^M_{low} - v_{low}s. \quad (3)$$

In [3] this method was verified to work well for PSNR and is also an integral part of the reduced reference metric presented in [7]. Moreover, a modified version of this correction step was used for the no reference metric in [8]. Using this approach our example metric PSNR^M results in the improved metric PSNR^{M+}, calculated as

$$\text{PSNR}^{M+} = (\text{PSNR}^M - o)/s. \quad (4)$$

These additional instances v_{high} and v_{low} need not be generated with the same coding technology which was used for the videos with unknown quality as suggested in [3]: the prediction accuracy is improved even when using a substantially different coding technology for these additional instances as we demonstrate in this contribution. The final prediction system including this correction step is shown in Fig. 1. A regression line that was estimated using this simple method is shown for one model in Fig. 2. While the offset is slightly too high, the slope is very close to the regression lines of the actual data.

5. SUBJECTIVE TESTING

We used the sequences 'CrowdRun'(CR), 'ParkJoy'(PJ), 'IntoTree'(IT) and 'OldTownCross'(OTC) from the SVT high definition multi format test set [9] with a spatial resolution of1920 × 1080 pixel and a frame rate of 25 frames per second. Each sequence was encoded at four different bit rates, from 5.4 Mbit/s to 30 Mbit/s resulting in a quality range from 'not acceptable' to 'perfect', corresponding to a mean opinion score (MOS) between 0.19 and 0.96 on a scale from 0 to 1. The sequences were encoded using the AVC/H.264 reference software [10], version 12.4. Two significantly different encoder settings were used: a low complexity (LC) and a high complexity (HC) setting representing a 'Main' and 'High' profile, respectively, to assure that our model is independent of specific coding structures and settings. Additionally we used the 'Dirac' encoder [11] version 0.7 in order to investigate, if it is possible to build a model that is useful for different coding technologies. The additional instances of the original videos to estimate the regression lines were generated using simple encoder settings for the AVC/H.264 reference encoder in order to keep the additional computational complexity caused by the correction step within an acceptable limit.

The subjective tests were performed in compliance with ITU-R BT.500 [12] at the video quality evaluation laboratory of the Institute for Data Processing at the Technische Universität München. In total 17 naïve viewers and one expert viewer participated. All of them were screened for visual acuity and color blindness. The distance between the screen and the test subjects was set to three times the picture height. The test was carried out using the Double Stimulus Unknown Reference (DSUR) [13], which is a variation of the standard DSCQS test method, and a discrete voting scale with eleven grades ranging from 0 to 10 later rescaled to the interval from 0 to 1. The 95% confidence intervals of the subjective votes are below 0.07, the mean 95% confidence interval is 0.04.

6. RESULTS

The PSNR^{M+} metric is compared to PSNR$^Y_{Mean}$ and to three more full reference video quality metrics. These are: SSIM [2], the VQM described in [14] and the VQM according to Annex D of ITU-T J.144 [4]. For the VQM as part of ITU-T J.144, the general model was used. The SSIM was evaluated on all three channels of the YUV color space.

The high prediction accuracy for the example metric PSNR^{M+} is shown by a Pearson correlation coefficient of 0.88, which is significantly above the correlation achieved for PSNR$^Y_{Min}$ and the reference metrics. No data fitting was performed for PSNR^{M+}. The correction step is not necessary for the case where the same data is used during calibration and validation. For this case the relationship between the parameters and the visual quality for each single video is already part of the model. The PSNR^M model without the correction step can only be used to give correct quality estimates for the four videos included in the model. It is the subsequent correction step that allows us to rate previously unknown videos. Table 3 shows

Table 3: Prediction Results

Metric	Pearson	Spearman	RMSE$^{(a)}$
PSNR$^{Y}_{Mean}$	0.69	0.61	0.22
PSNR$^{Y}_{Min}$	0.75	0.70	0.18
PSNR^{M}	0.80	0.71	0.14
PSNR^{M+}	0.88	0.87	0.17$^{(b)}$
PSNR$^{M\ (c)}$	0.97	0.97	0.05
VQM [14]	0.78	0.66	0.13
SSIM [2]	0.85	0.79	0.11
VQM Annex D of [4]	0.85	0.78	0.11

$^{(a)}$ After first order fitting for all comparison metrics, no fitting for PSNR^{M}, PSNR^{M+}

$^{(b)}$ If first order fitting would be applied, this is reduced to 0.11

$^{(c)}$ No cross validation

the correlation values for the Pearson correlation, the Spearman rank order correlation and the Root Mean Squared Error (RMSE) for the different quality metrics. Also a overview of the prediction results for PSNR^{M+} is shown in Fig. 3. We see that our example metric PSNR^{M+} delivers an improved prediction accuracy compared to the reference metrics. Especially the Spearman rank order correlation is improved, whereas for the VQM according to Annex D of [4] and the SSIM [2], the Pearson correlation coefficient is only slightly worse and the RMSE is nearly identical.

7. CONCLUSION

We proposed two simple methods that can help to improve video quality metrics: extending the set of parameters that are gathered from a single feature by temporal pooling and introducing a simple correction step which estimates the relationship between the output of a metric and the visual quality for a video. Both improve the prediction accuracy considerably. In particular the correction step allows us to predict the quality of previously unknown videos, even if the base metric itself is not very well adapted for this type of video sequence.

The example metric PSNR^{M+} was developed to demonstrate the effectiveness of these two methods. A Pearson correlation of 0.88 is remarkably high for this rather simple metric especially compared to the results of more sophisticated full reference metrics. Due to our cross-validation approach, this value is not the result of data fitting, but shows the real capability of this visual quality metric for high definition video.

The presented gains may not hold completely for a larger data set. Still, we strongly believe that the general principle provides relevant improvements for predicting the visual quality.

8. REFERENCES

[1] C. Lee, S. Cho, J. Choe, T. Jeong, W. Ahn, and E. Lee, “Objective video quality assessment,” *SPIE Optical Engineering*, vol. 45, p. 7004, Jan. 2006.

[2] Z. Wang, L. Lu, and A. C. Bovik, “Video quality assessment based on structural distortion measurement,” *Signal Processing: Image Communication*, vol. 19, no. 2, pp. 121–132, Feb. 2004.

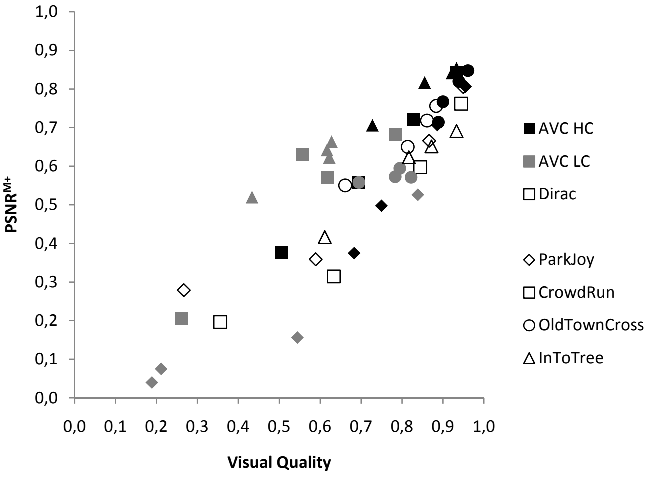

Fig. 3: Prediction results for PSNR^{M+} (Models “CR”, “IT”, “OTC”, “PJ”)

[3] T. Oelbaum, K. Diepold, and W. Zia, “A generic method to increase the prediction accuracy of visual quality metrics,” in *Proc. Picture Coding Symposium*, Nov. 2007.

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

[5] H. Martens and T. Naes, *Mutltivariate Calibration*. Wiley & Sons, 1992.

[6] J. Lubin, “A human vision system model for objective picture quality measurements,” in *Proc. IBC International Broadcasting Conference*, Sep. 1997, pp. 498–503.

[7] T. Oelbaum and K. Diepold, “A reduced reference video quality metric for AVC/H.264,” in *Proc. European Signal Processing Conference EUSIPCO*, Sep. 2007, pp. 1265–1269.

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

[9] SVT, “The SVT high definition multi format test set,” Feb. 2006. [Online]. Available: http://www.ldv.ei.tum.de/lehrstuhl/team/Members/tobias/sequences

[10] K. Sühring, “H.264/AVC software coordination,” 2007. [Online]. Available: http://iphome.hhi.de/suehring/tml/index.htm

[11] C. Bowley, “Dirac video codec developers’ website.” [Online]. Available: http://dirac.sourceforge.net

[12] “ITU-R BT.500 methodology for the subjective assessment of the quality for television pictures,” Jun. 2002.

[13] V. Baroncini, “New tendencies in subjective video quality evaluation,” *IEICE Transaction Fundamentals*, vol. E89-A, no. 11, pp. 2933–2937, Nov. 2006.

[14] F. Xiao, “DCT-based video quality evaluation.” [Online]. Available: http://compression.ru/video/quality_measure/vqm.pdf