Rule-Based No-Reference Video Quality Evaluation Using Additionally Coded Videos

Tobias Oelbaum, Christian Keimel and Klaus Diepold

***Abstract*—This contribution presents a no-reference video quality metric, which is based on a set of simple rules that assigns a given video to one of four different content classes. The four content classes distinguish between video sequences which are coded with a very low data rate, which are sensitive to blocking effects, which are sensitive to blurring, and a general model for all other types of video sequences. The appropriate class for a given video sequence is selected based on the evaluation of feature values of an additional low quality version of the given video, which is generated by encoding. The visual quality for a video sequence is estimated using a set of features, which includes measures for the blockiness, the blurriness, the spatial activity and a set of additional continuity features. The way these features are combined to one overall quality value is determined by the feature class, to which the video has been assigned. We also propose an additional correction step for the visual quality value. The proposed metric is verified in a process, which includes visual quality values originating from subjective quality tests in combination with a cross validation approach. The presented metric significantly outperforms PSNR as a visual quality estimator. The Pearson correlation between the estimated visual quality values and the subjective test results takes on values as high as 0.82.**

***Index Terms*—Visual quality, AVC/H.264, no-reference quality metric**

I. Introduction

Most humans can subjectively judge the visual quality of a processed and distorted video without ever watching the reference video. Performing subjective tests for assessing visual quality is time consuming, expensive, and cannot be easily included as part of practical applications. Video quality metrics are available, where a full reference video or a reduced reference information is accessible. Those metrics can deliver useful results. However, many systems or applications can not provide access to the reference video, and therefore require an evaluation without a reference video. In addition, the reference video may not even exist. This is for instance the case in applications where the user can select his own viewpoint on the scene, and not all viewpoints correspond to video footage recorded by a camera, but many of them correspond to views where the images are interpolated.

No-reference (NR) video quality evaluation has been the topic of many studies in the field of visual quality metrics. The results which have been presented so far have a few drawbacks. One of the major drawbacks of existing approaches is that the results have not been sufficiently verified such that the results do not allow to draw conclusions, which are meaningful beyond the set-up of the particular investigation (e.g. [1]), since the studies used the same data for the design as well as for the verification of the metric (e.g. [2], [3]). Other problems are in the context of using bit rate as quality indicator. There, one study uses only one encoder or just one setting for the encoder (e.g. [4]). One popular approach for no-reference quality evaluation is the use of watermarks (e.g. [5]). But those methods need access to the reference video, and therefore cannot be classified to be real no-reference metrics. Methods to predict the Peak-Signal-to-Noise Ratio (PSNR) from the coded bit-stream (e.g. [6]) have been shown to work very well. However, such an approach is limited to the prediction accuracy provided by the use of PSNR as a visual quality estimator.

Our metric is built on a set of simple measurements to detect certain features of a given video sequence. These features include blockiness and blurriness, two typical artifacts for video coding, spatial activity and a set of continuity features consisting of predictability, edge-continuity, motion-continuity and color-continuity. We extract those features from each video and combine them using different models. Each model corresponds to one of four different feature classes, to which the video is assigned. The feature classes are designed to capture video at very low rate (less than 100 kBit/s), video that is highly sensitive for blurriness or video that is highly sensitive blocking artifacts, and all videos that do not fall in one of these three classes. The appropriate feature class is selected using a low quality version of the video V that should be evaluated. This low quality version V_{low} is generated by encoding the video V at low bit rate, using a simple encoder with a fixed quantization parameter. We choose the appropriate feature class for V in a reliable fashion by evaluating this additional video V_{low}. The following example illustrates the problem of selecting the appropriate feature class: for a video V we measure a low blur value. This low blur value may result from two different reasons:

1) The particular video content is not very sensitive to blur (no matter at which quality level we encode the video, we will never get a high blur value). This is for instance the case for the video "Husky" used in our tests.
2) The particular video content is sensitive to blur, but was coded at a high quality level. Therefore, we detect only a low blur value.

Without additional information it is not possible to decide which of these two cases is present. This problem is solved by evaluating the low quality version of the video(V_{low}) such that

T.Oelbaum, C.Keimel and K.Diepold are with the Department of Electrical Engineering and Information Technology, Technische Universität München, Munich, Germany (e-mail:tobias.oelbaum@mytum.de, christian.keimel@tum.de, kldi@tum.de).

 Digital Object Identifier: 10.1109/JSTSP.2009.2015473

we can easily detect if the video is sensitive to blur or not. We can use the same idea for evaluating any other feature, such as the level of detail (spatial activity) or the amount of blocking that we measure in the video. We use the video V_{low} also to decide, if our quality estimation did over- or underestimate the visual quality of the video, in order to adjust the value for the estimated quality.

We verify the present approach using video sequences at CIF resolution and which we encode using a codec that complies to the AVC/H.264 [7] standard. We take special care to perform a proper verification of this new rule-based no-reference quality metric. The verification uses data from subjective tests which were carried out using a high number of test subjects and following the international recommendations for such subjective tests. In addition, we used a cross-calibration approach, which ensures that the quality prediction for a video is calculated using a model that was trained with a data set that does not contain this video.

The contribution is organized as follows. In Section II we give a short introduction to previous work, which forms the basis for the our approach. The new metric itself is described in Section III. The verification of the metric is discussed in Section IV. Section V presents some results, and concluding remarks are given in Section VI.

II. State of the Art and Previous Work

In this section we give a short introduction to no-reference video quality evaluation, and our previous work. So far, a relatively small number of no-reference quality metrics for video has been proposed. Most of the proposed metrics have not been sufficiently verified.

A. No-reference quality metrics

The first no-reference quality metric for video concentrated on measuring blocking artifacts. This method did not take into account other artifacts [8]–[10]. One of the first approaches, which did not focus on blocking artifacts was introduced by Gastaldo *et al.* in 2001 [11]–[13]. Gastaldo *et al.* propose not to process the decoded video, but to extract features from the compressed bit-stream, and to combine those features into one quality value using a neural network. The features that are extracted from the bit-stream include the number of bits per frame, information about the quantizer scale, and the motion vectors. Unfortunately, they do not provide information if they used different encoders to generate the bit-streams, or if they varied the encoder settings. Therefore it is difficult to judge the effectiveness of this approach in a more general context. The idea of using parameters from the compressed bit-stream was also used in [14], where bit-stream parameters are extracted in addition to features from the pixel domain. However, the article does not contain details about which features are extracted from either of the domains. Ries *et al.* propose to only use bit rate and frame rate to predict the quality of a video after having assigned a video to one special content class, such as sports or news. This study only uses one special encoder with fixed settings such that the results are of limited use if the encoder settings are changed, or if a different encoder is used [4], [15]. In addition, verification of this last proposal is done on very few different sequences only. For four different content classes only five different sequences were used for verification, and only ten people evaluated the video used for this verification.

Most recent no-reference metrics combine a set of parameters that are extracted in the pixel domain. Typical representatives for those metrics are [1]–[3], [16]–[18]. In [3] the authors also use the bit rate of the compressed streams in addition to features extracted from the pixel domain. Again, only one encoder with fixed settings was used, which limits the more general usage of this metric.

The use of data hiding methods for the evaluation of visual quality was proposed in [5], [19], [20]. There, instead of extracting features from the bit-stream or the decoded video, the methods evaluate to which extent an inserted watermark can be recovered at the receiver. The distortion of this watermark is then set in relation to the visual quality. As those methods need access to the original video to insert the watermark in the video, they cannot really be classified as no-reference methods. Data hiding methods are also used to transmit properties of the original video within the video data stream, allowing to use this information at the receiver to calculate a quality prediction [21]. Again, this method requires access to the original video and hence this method should better be classified as a reduced-reference metric, where the reduced-reference data is transmitted within the video.

As indicated, the above mentioned metrics were not sufficiently verified. Weaknesses in the verification process are in procedures to perform the subjective tests, which did not follow international recommendations, and where a high number of votes was removed from the tests results [4], [15], or where only parts of the results were presented [1]. Only three methods were verified using data sets which differ from the data sets used for calibrating the quality metrics: [1], [4], [17]. We will show in Section V that using the same data for calibration and verification leads to over-optimistic conclusions about the prediction capabilities of the quality metrics. The no-reference metric that has been verified in an accurate way and where the complete results of the verification were presented, delivers a limited prediction accuracy [17]. The method provides a Pearson correlation value of 0.65 to measure the relationship between the computed predictions and the results of subjective tests. This values is comparable to the correlation that is accomplished by using plain PSNR values.

A special type of no-reference metric estimate the PSNR values from the compressed bit-stream [6], [22], [23]. Obviously, those methods provide the same accuracy as a visual quality estimators using standard PSNR values.

B. Reduced-reference metrics

In [24], we introduced a reduced-reference (RR) video quality metric for video that has been compressed using an AVC/H.264 compliant video codec. The approach presented in [24] uses a set of features to predict the visual quality of a compressed video. This RR quality metric can be summarized as follows:

- A value to express the visual quality is computed as a weighted sum of a set of simple no-reference feature measurements. The main features to be used in this metric are: 'blockiness', 'blurriness', 'spatial activity' and a set of 'continuity measures'.
- The weights of the individual features are calculated using a principal component analysis (PCA) and a partial least squares regression (PLSR) method. No special models for the human visual system are taken into account. The impact of the measured features to the subjectively perceived visual quality is determined using data modelling methods.
- The value for the predicted visual quality is corrected by comparing the value for the visual quality pertaining to the original video and the value for the visual quality pertaining to a low quality instance of the video.

In [25] we extended this correction step to estimate the visual quality of video in a more general setting. Our results showed that this RR metric outperforms PSNR significantly, and that it is slightly superior to two other full-reference (FR) metrics. Another advantage of our RR metric is the low data overhead of two bytes per sequence that needs to be transmitted along with the video. This correction step consists of transmitting additional quality-related information about the original video to the receiver, describing how the visual properties of this video will most likely change if it is compressed.

III. Rule-based no-reference metric

The no-reference metric presented here consists of the same three steps as the RR metric presented in [24], namely

1) Generate prediction models using a training data set and applying data analysis methods for finding the weights that are assigned to the different calculated features.
2) Calculate the features of the current video sequence and weight the features using an appropriate model.
3) Correct the value of the quality estimation based on an additional low quality version of the video.

We suggest two major modifications to our method in [24] to arrive at a NR system, that is

- One single model is not able to describe video sequences having very different properties. Therefore, different classes of video content are built, and each video sequence is assigned to one of those classes. Video sequences belonging to different content classes are then evaluated using different prediction models.
- The correction step is shifted to the receiver, where a low quality video V_{low} is generated by encoding the received video V. Details, on how this low quality video V_{low} is produced, and on the correction step itself, are given in Section III-D.

These steps are described in detail in the following sections. The overall system for the presented NR metric is shown in Fig. 1.

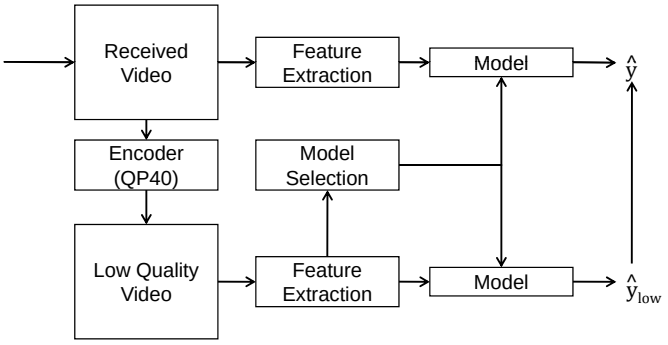

Fig. 1: Proposed no-reference video quality metric

A. Feature selection

The following features were used for the NR metric (as in [24]) such as blurriness, blockiness, spatial activity, temporal predictability, edge-continuity, motion-continuity and color-continuity.

The first three features are measurements performed on single frames of the video. The other four measurements are inter-frame features, later referred to as continuity measurements. The continuity measurements describe what happens between frames of a video. The introduction of these continuity measurements is based on the observation that visual quality is perceived differently if changes between neighboring frames appear abruptly or smoothly and hence describe the temporal aspects of video sequences. To calculate the value of a single-frame feature (e.g. blurriness) for the whole video sequence, the values for the individual frame are averaged. To reduce the computational complexity of the algorithm, all features except the color-continuity are calculated using the luma channel only.

1) Blurriness: The method to compute the blurriness (or blur) is described in [26]. The algorithm measures the width of an edge, and calculates the blur by assuming that blur is reflected by wide edges. As blur is something natural in a fast moving sequence (motion blur), this measurement is adjusted using a simple piecewise linear correction if the video contains a high amount of fast motion. The range of the blur values lies between 0, which indicates that only sharp edges appear in a frame, and a theoretical upper bound that is only limited by the size of the frame. For uncompressed frames the blur value was found to be approximately 1. For coded video sequences the detected blur values could be as high as 5.3.

2) Blockiness: For measuring the blockiness we use the method introduced in [27]. This algorithm calculates the horizontal and vertical blockiness by applying a Fourier transform along each line or column of an image. The unwanted blockiness can easily be detected in the frequency domain by comparing the measured spectrum with a smoothed version of it. Blockiness should appear as peaks in the spectrum located at distinct frequencies, which correspond to the size of the blocks. The frequency spectrum of a frame without blocks should be more smooth, and should continuously decrease with increasing frequencies. This blockiness measurement was originally designed for still images and MPEG-2 encoded video, where the size of the blocks does not change. Under those conditions it is easy to determine a value for the feature blockiness. In contrast, AVC/H.264 comprises different block sizes and therefore does not have a regular block structure.

The value for the blockiness feature for the AVC/H.264 coded video lies in a range between 0 - which is the theoretical lower bound - and 2. These values are actually significantly lower compared to the blockiness values detected using the same algorithm for video encoded according to MPEG-2. For MPEG-2, values up to 15 are detected. Even though the deviation of the values for the blockiness feature for AVC/H.264 encoded video is smaller, our experiments using AVC/H.264 encoded video showed that the value for blockiness increases with decreasing bit rate and decreasing subjective image quality. This blockiness measurement can therefore be used as one indication for the visual quality of the video, even if the video itself does not have a regular block structure. Note that for measuring the visual quality, it is not necessary to exactly detect the location in the image where the blockiness appears, but it is sufficient to measure the overall amount of blockiness.

3) Spatial activity: The spatial activity is measured by the amount of details that appear in a video frame. To measure the amount of details that are present in a video, the percentage of turning points along each line and each row is calculated. A turning point is given, if the sign of the intensity difference given by $I_n - I_{n-1}$ is different to the sign of the intensity difference given by $I_{n-1} - I_{n-2}$, with I_n being the intensity of the pixel located at position n. The two measurements for horizontal and vertical spatial activity are averaged to obtain one single value. This measurement is part of the BTFR (British Telecommunications Full Reference) metric included in [28]. As the amount of details that are noticed by an observer decreases with increasing motion, the spatial activity measurement is adjusted using a piecewise linear correction function if high motion is detected in the video.

4) Temporal predictability: For measuring the temporal predictability, a predicted frame is built. To this end a motion compensation using a simple block matching algorithm is performed, which is based on the approach presented in [29]. The motion compensation is done using the actual frame and the previous frame, the difference between the predicted frame and the actual frame is the residual error of the motion compensation step. The same motion-compensated prediction is used for calculating all the other continuity measurements.

To calculate temporal predictability, the actual frame and its prediction are compared block by block. A 8×8 block is considered to be noticeable different, if the Sum of Absolute Differences (SAD) exceeds 384[1]. To avoid that single pixels dominate the SAD measurements, both images are filtered using first a Gaussian blur filter followed by a median filter. The output of this process is the percentage of blocks that are not noticeable different. This percentage gives an indication of the temporal frame predictability.

5) Edge-continuity: The method to generate the edge-continuity measurement compares the actual image and its motion compensated prediction using the Edge-PSNR algorithm as described in [30]. This measurement should reflect how much the main structure of the image changes. The Edge-PSNR metric produces output values between 0 and 1, where the value 1 indicates that there is no difference between the two adjacent frames.

6) Motion-continuity: We usually assume that physical objects move along a smooth motion trajectory. However, there are reasons for motion trajectories to be not smooth due to objects that move in a chaotic way or due to transmission artifacts such as jitter. Changing the field order for interlaced video also results in non-smooth motion trajectories. To determine if motion is continuous throughout adjacent frames, two motion vector fields are calculated, one field between the current and the previous frame, and one field between the following and the current frame. The percentage of motion vectors, for which the difference between the two corresponding motion vectors does not exceed 5 pixels (either in x- or y-direction), determines a value for describing motion-continuity.

7) Color-continuity: The method for quantifying the feature color-continuity calculates a color histogram with 51 bins for each RGB channel for the actual image and its motion-compensated prediction. The value for color-continuity is given as the linear correlation between these two histograms. This method allows gradual changes in color, as they appear for illumination changes, but it will produce lower numerical values for situations where artifacts such as color bleeding appear.

B. Combining features

We used methods provided by multivariate data analysis to examine the calibration data base and to build a stable prediction model. This approach was proposed by Miyahara in [31], and was used in [32]–[34]. Multivariate data analysis is the method of learning to interpret a number of m input sensory signals $p_i, \quad i = 1, 2, \ldots m$ that contribute to a common output y. For the metric presented in this contribution, the input signals x_i are the feature measurements, which have been introduced before. The visual quality of the video, that has been determined by means of subjective visual tests serves as the output y.

We use a principal component analysis (PCA) to calculate a compact representation of the sequence description, and a partial least squares regression (PLSR) to establish a linear relationship between the principal components (PCs) and the visual quality. We already showed in [24] that this method leads to stable and useful prediction models.

Before building the prediction models, all $m = 7$ mentioned features p_i ($i \in [1 \ldots 7]$ are first centered around their corresponding mean values $\overline{p_m}$, since the interesting information does not lie in the absolute values, but in the variations of the feature measurements across different sequences. In addition, all feature values are scaled to have a standard deviation of 1, to avoid that some feature measurements that have only small absolute variations, are covered by some noise in features that have bigger absolute variations.

The visual quality prediction $\widehat{y}$ is calculated as

$$\widehat{y} = b_0 + \mathbf{p} \cdot \mathbf{b}. \tag{1}$$

where $\mathbf{p}$ is the feature vector of the single feature values feature p_i. The column vector $\mathbf{b}$ contains the individual

[1]This value was determined experimentally, and allows a mean difference of 6 for each pixel

TABLE I: Weights for the different feature classes (models including all calibration sequences)

	Low Rate	Blur	Blocking	General
Blur	-0.019	-0.040	-0.060	-0.030
Blocking	-0.046	-0.090	-0.114	-0.045
Spatial activity	0.024	0.078	0.060	0.067
Predictability	0.003	0.015	0.026	0.020
Edge Continuity	0.015	0.046	0.040	0.024
Color Continuity	0.006	0.029	0.014	0.019
Motion Continuity	0.034	0.031	0.022	0.074

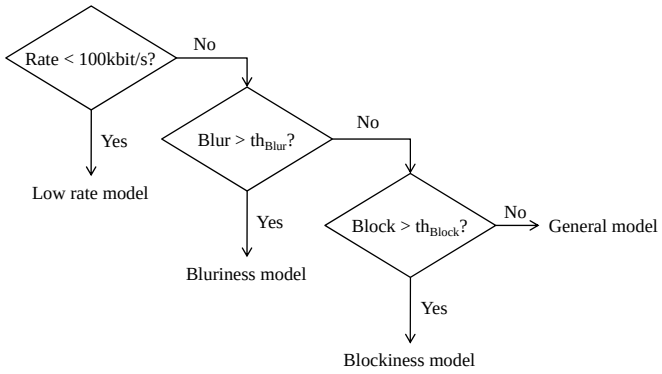

Fig. 2: Classification Process

estimation weights b_i for each feature p_i and b_0 is the model offset. A detailed description of PLSR can be found in Chapter 3.5 of [35].

C. Four different feature classes

It is not possible to represent all different sequences which were used for the calibration step and which exhibit corresponding variation of visual quality with one single prediction model. It turns out that in particular the sequences, which are coded with a relatively low bit rate fit to the model only, if a higher number of principal components (i.e., more than four PCs) is included. To reach a stable model, it is desirable to have a low number of PCs. Therefore, the low bit rate sequences were not included during the calibration step. For all sequences which are represented by a data stream with a bit rate below 100 kbit/s we create a special model. This particular model only needs two PCs to describe the variation of visual quality for the low bit rate sequences sufficiently well. The model which is built from the remaining sequences is dominated by the characteristics of video sequences which are either sensitive to blur or sensitive to blocking artifacts. For this reason, the remaining model is further split up into three additional models, namely, one model for the sequences with high sensitivity to blurring, one model for the sequences with high sensitivity to blocking, and a final model for all remaining sequences (general model). Each of these three models does not comprise more than two PCs, which leads to a compact and stable representation of all sequences.

The different models differ only in the weights b_i for the features p_i. The weights for the different features and corresponding with the different feature classes are denoted in Table I. These weights show the influence of the single features on the visual quality. Negative weights mean, that if the value for this feature increases, the visual quality decreases. This is the case for the two features blurriness and blockiness. This observation is in line with our expectation, which tells us that increased blurring or blocking leads to a lower visual quality. All other features show a positive value. We observe this also for the feature 'spatial activity'. A high number of details in an image should also result in a higher visual quality. This finding also backs the assumption that we made for inter-frame features. Increasing the similarity between adjacent video frames leads to a better visual quality. The values also show that the blockiness value is the most important value for three out of the four models, as the weight for this parameter has the highest absolute value. This observation is true even for the model that is used to evaluate videos which are sensitive to blurring. So even if the blocking artifacts are significantly reduced by the use of a de-blocking filter, as in AVC/H.264 [36], this artifact has still the highest impact on visual quality for coded video.

For a given video sequence V, the proposed method selects the appropriate model by analyzing the features of a low quality version of the video, V_{low}, which is produced by encoding the actual video V using a high quantization parameter (QP). Details on how to produce V_{low} are given in section III-D. Our method of a rule-based selection of the appropriate model is shown in Fig. 2. The threshold values th_{Blur} and th_{Block} refer to the blurriness and blockiness values of the low quality instances V_{low}. Their values were selected as the mean values $\overline{Blur}$ and $\overline{Block}$ over all video sequences included in the calibration database.

A low value for the blurring feature indicates either a video which is insensitive to blurring or a video sequence that is sensitive to blurring and which is encoded at a high quality level (i.e. high data rate). Using the low quality video V_{low}, which was generated by encoding the given video V, we can gain information about the 'sensitivity' of the video to coding artifacts. Instead of selecting the blurring model because we can detect a lot of blur in the video V, the blur model is only selected if the video is sensitive to blur.

There is a high correlation between the values for the blurring feature of the actual video V and the corresponding feature values for its low quality version V_{low}. However, in some cases our method selects a different model, if the model decision is made based only on the data for V. A similar statement is true for the feature blockiness and the selection of the blockiness model. The advantage of including the data for V_{low} for the model selection process is shown in Fig. 3 and in Fig. 4. For these figures, the values for the blurring feature of the distorted video sequences have been centered around the respective mean value. Obviously, distinction between video sequences that are sensitive to blurring is much easier and more robust using the low quality instances as here only very few values close to zero can be observed.

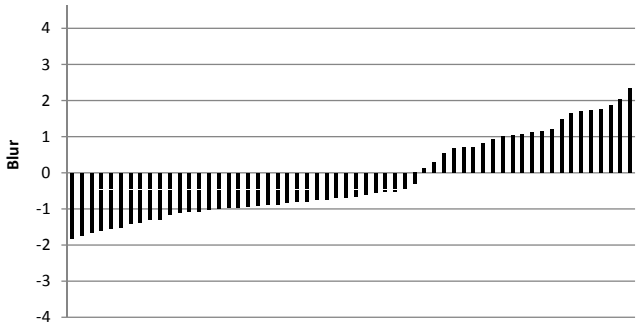

Fig. 3: Centered blur values for verification video sequences sorted by magnitude (each bar represents a different sequence)

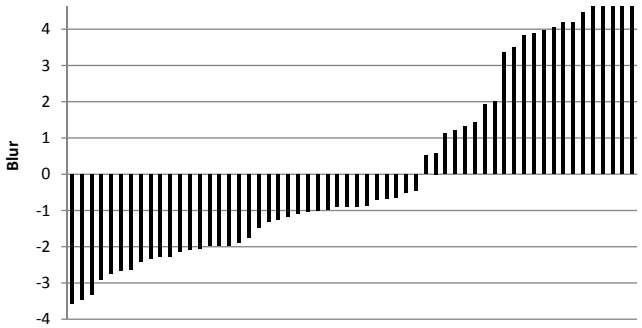

Fig. 4: Centered values for the blurring feature for low quality instances of the verification video sequences sorted by magnitude (each bar represents a different sequence)

Table II shows the model which is applied for the different sequences. As the low quality instances V_{low} are very similar for the same original video, and do not significantly vary for varying quality of V, different models are selected for different original video only, and this selection does not depend on the quality of the received video. This is different if the model selection process is done on the given video V.

D. Offset correction using the low quality video

The low quality video V_{low} is used to support the selection the appropriate feature class. In addition to this it is also used to determine a correction term for the final visual quality prediction.

This correction term uses the mean estimated quality value $\overline{\widehat{y}_{low}}$ of the low quality version V_{low} of the video sequences V used for calibrating the model. In addition to this we use the standard deviation of the set of calibration video sequences of the single quality values $\widehat{y}_{low}$, σ_{low}. The set of calibration video consists of all coded video sequences except the ones that refer to one specific original video (this method is known as cross validation or 'leave one out' validation).

TABLE II: Models selected for the different sequences

Model[a]	Sequence
Blur	Crew, Foreman, Ice, Zoom
Blocking	Football, Harbour, Husky
General	Bus, City, Head, Mobile, Paris, Tempete

[a] Note, that one additional model is selected if the video has a very low bit rate

TABLE III: Selected settings for the low quality AVC/H.264 encoder

Feature[a]	Setting
Profile	Baseline
Reference Frames	1
Entropy Coding	CAVLC
R/D Optimization	Off
Rate Control	Fixed QP (QP 40)
Search Range	16
I-Frames	every second
B-Frames	0

[a] See the reference software manual [38].

To calculate the quality estimation $\widehat{y}_{low}$, we consider the same prediction model as for $\widehat{y}$, even if V_{low} has a very low bit rate and if originally the low rate model is selected for V_{low}. The quality prediction $\widehat{y}_{low}$ is first clipped to $\overline{\widehat{y}_{low}} \pm \sigma_{low}$ to avoid overcompensation and then a correction step

$$\widehat{y} = \widehat{y} - (\widehat{y}_{low} - \overline{\widehat{y}_{low}}) * 0.75 \tag{2}$$

is applied. The factor 0.75 was added to weight the original prediction more when compared to the correction step introduced by V_{low}.

The idea for this correction step is that V_{low} should always have roughly the same low visual quality for different content and different quality levels of the given video sequences V. This is true, as the encoder used to generate V_{low} uses a high QP, and the resulting quality is therefore low, no matter if V already had a low visual quality, or if it had a high quality. If $\widehat{y}_{low}$ for one video is different from $\overline{\widehat{y}_{low}}$, the value for $\widehat{y}$ needs to be corrected. The resulting value $\overline{\widehat{y}_{low}}$ is found to be very close to zero for all models.

The video V_{low} is generated using a very simple fixed QP encoder according to the AVC/H.264 video coding standard. This encoder is constrained in the sense that is uses only one reference frame, a prediction structure without B-frames, and it does not perform any rate distortion optimization. Using a very high value for the quantization parameter, $QP = 40$ say, ensures that the visual quality of V_{low} is reasonably low. For this task the we use the AVC/H.264 reference software version 11.0 [37]. The encoder settings are given in Table III. It has to be noted that the settings used to generate V_{low} differ significantly from the settings used to encode the video V, that were used for the verification process.

This correction step is not necessary if no cross validation approach is applied, and the same data is used for calibration and verification. Obviously, the information given by V_{low} allows to predict the quality of previously unknown sequences, which in this case were the sequences omitted in the 'leave one out' procedure. This is also reflected by the low prediction accuracy that is reached if this correction step is not applied (see Table V).

TABLE IV: Verification sequences

Sequence	Rates [kbit/s]
Bus	128, 256, 512
City	192, 384, 750
Crew	192, 384, 750
Football	96, 192, 256, 384, 512, 768, 1024
Foreman	96, 128, 192, 256, 384, 512, 768
Harbour	192, 384, 750
Head	96, 192, 384, 768
Husky	96, 192, 384, 768
Ice	192, 384, 750
Mobile	96, 192, 256, 384, 512, 768, 1024
Paris	96, 192, 384, 768
Tempete	96, 192, 384, 768
Zoom	96, 192, 384, 768

IV. Metric verification

Verification of a visual quality metric is done by comparing the metric's output to the results determined by subjective tests. For the verification of the metric proposed in this work, we use the results of two different subjective tests. The employed cross calibration/validation process avoided an overlap between calibration and verification data. For comparison purposes, we also provide results where no cross validation approach was applied, and where exactly the same data is used for the calibration step and for the verification step.

A. Calibration data base and verification test cases

The calibration data base consists of next to 300 video sequences at CIF resolution, encoded according to AVC/H.264 using 15 different original video sequences. These video sequences were generated using different encoders for AVC/H.264, including the reference software [37] and several proprietary encoders. The sequences were encoded to fit for certain scenarios, such as broadcasting or conversational applications.

The verification data consists of a subset of 56 video sequences originated from 13 different video sequences, for which accurate subjective test results were available. This verification data base spans a wide range of bit rates ranging from 96 kbit/s to 1024 kbit/s and includes sequences with different frame rates, prediction structures, and coding settings. The original video sequences are well known test sequences such as 'Foreman' or 'Mobile&Calendar', representing different scenarios including sports, news, or conversational applications. Again, we used different encoders to encode these sequences. The visual quality covered by these sequences ranges from 0.09 to 0.91 on a 0 to 1 scale. This shows that we actually cover the whole reasonable quality range for video at CIF resolution. Further details on the used sequences, including the video coding tools, bit rates, and frame rates, can be found in [39], [40]. Table IV provides an overview.

The verification of the metric produces results by calibrating one model using all data points except those that belong to one certain original sequence, and later predicting the data points, which have been left out using exactly this model (cross validation or 'leave one out'). Compared to the approach of dividing the available data base into one part that is used for calibration, and one part for verification only, this cross validation approach provides the advantage, that all available data can be used for verification. Therefore, the validity of the verification is increased even for a limited data base.

B. Subjective testing

The subjective results that were used for the verification of the proposed metric were generated for the verification tests on AVC/H.264 [39] and the entry tests for the scalable extension of AVC/H.264, SVC [40]. The tests were performed at the Fondazione Ugo Bordoni (FUB) in Rome and the Institute for Data Processing at the Technische Universität München in Munich. The test method used for the videos at CIF resolution was identical for both tests. Main attributes of these tests are:

- A room setup following ITU-R BT.500 [41].
- We use a DLP projector to display the progressive scanned CIF videos at their native resolution.
- The test method is based on a Single Stimulus procedure according to the Absolute Category Rating (ACR)(see [42]). To be able to specify comparably small quality differences we used an 11 grade discrete scale ranging from 0 to 10. For this work the votes were later rescaled to 0 to 1.
- At least 20 test subjects rated each single test case. All test subjects were screened for visual acuity and color blindness. The test subjects were students between 20 and 30 years old and all of them were naïve test subjects in the sense that none of them worked in the field of video coding or visual quality evaluation.
- The test subjects received an extensive training on the test method to guarantee stable results.
- To compensate contextual effects, which are known to be present in a Single Stimulus environment, all test cases were shown twice, and received two separate votes.

A Single Stimulus procedure was selected to allow the subjects to distinguish between different quality levels even for comparably low visual quality. As the test included video sequences at comparably low bit rates, the ability to differentiate between different levels of low visual quality would be compromised if the high quality original is displayed as a direct anchor. Displaying the CIF sequences using a DLP projector provides two advantages compared to the use of professional (interlaced) TV monitors or (LCD) computer screens. First, no upsampling filters have to be used, and the video sequences can be displayed at their native resolution. Second, this method allows to fix the viewing distance in relation to the height of the displayed video. Using computer screens, even small movements of the viewers changes the relative viewing distance significantly if the sequences were displayed at their native resolution. In contrast, the projector setup allows to easily fix the viewing distance to 6 times the picture height.

This design of the subjective tests yields low confidence intervals for the test cases. The 95% confidence intervals lie between 0.03 and 0.08 on a 0 to 1 scale with a mean confidence interval of 0.05. This shows that the results of the subjective tests are reliable. More details on the subjective test methodology can be found in [39], [40].

V. Results

The performance of the proposed NR metric is measured by comparing the results of the metric to the results of the subjective tests. We make sure that the results for every sequence are generated using a model that was calibrated using a data base that does not contain this sequence by following a cross validation approach. Using previously unknown data for the verification step is of utmost importance when trying to assess the prediction capabilities of a video quality metric. Not using a cross validation approach, and using the same data for calibration and verification leads to overoptimistic prediction results. For comparison purposes we chose a PSNR scale, as PSNR values can be estimated quite well in the no-reference case, where the correlation between PSNR and the estimated NR-PSNR takes on values of more than 0.95 [6]. Therefore PSNR can be taken as a benchmark for NR metrics. Unfortunately, no implementation of other NR metrics was available for generating comparison data.

A. Performance Metrics

The metrics most often used to measure the performance of an objective quality metric are the Pearson linear correlation coefficient (PLC), the Spearman rank order correlation coefficient (SRO), and the outlier ratio (OR). The Pearson linear correlation (Eq. 3) gives an indication about the prediction accuracy of the model. The Spearman rank order correlation performs a similar task (Eq. 4). The rank order correlation gives an indication about how much the ranking between the sequences under test changes for the model's values compared to the subjective values (prediction monotonicity), given by

$$r^p = \frac{\sum_k (q_k - \overline{q})(MOS_k - \overline{MOS})}{\sqrt{\sum_k (q_k - \overline{q})^2}\sqrt{\sum_k (MOS_k - \overline{MOS})^2}}. \quad (3)$$

Here q_k is the predicted value for the video under test, and $\overline{q}$ is the mean value of all predictions. The symbols MOS_k and $\overline{MOS}$ represent the respective subjective values. For the Spearman rank order correlation

$$r^s = \frac{\sum_k (\chi_k - \overline{\chi})(\gamma_k - \overline{\gamma})}{\sqrt{\sum_k (\chi_k - \overline{\chi})^2}\sqrt{\sum_k (\gamma_k - \overline{\gamma})^2}}, \quad (4)$$

where r^s, χ_k is the rank of q_k, and γ_k is the rank of the respective subjective value MOS_k. The symbols $\overline{\chi}$ and $\overline{\gamma}$ denote the corresponding midranks.

For outlier calculation, the individual 95% confidence intervals ci_k of the subjective votes were obtained. The quality estimation for one video sequence is defined to be an outlier if

$$|MOS_k - q_k| > ci_k \quad (5)$$

is satisfied.

For our proposed method no data fitting was applied during outlier calculations, whereas first order data fitting was applied for PSNR (fitting has to be applied for PSNR as otherwise the outlier ratio can not be calculated). A subsequent data fitting is often applied for the presentation of visual quality metrics. Data fitting is the method of finding the function $q_k = f(MOS_k)$, which minimizes the distance between the vector $\mathbf{q}$, which contains all predicted values q_k and the vector $\mathbf{MOS}$, which contains all subjective quality values MOS_k. First order data fitting was chosen to fit the predicted values to the actual given data. Although higher order fitting is sometimes proposed for this purpose [28], higher order fitting always carries the danger of over-fitting the model to the actual data, and jeopardizing the predictability of the model for unknown data.

B. Prediction accuracy

If we drop the cross validation step, the Pearson correlation coefficient between the model's output and the subjective ratings turns out to have a values of 0.88. For this case the correction step, which has been described in Section III-D is not necessary. If a proper verification is done, this high value for the correlation is significantly reduced. For this case, the Pearson correlation between predicted quality and actual visual quality drops to 0.63 and the prediction accuracy is even below the value that we achieve using PSNR (correlation for PSNR is at 0.67). Introducing the correction step as described in Section III-D, the value for the correlation increases back to 0.82. The reader can find more detailed results in Table V, Table VI and Fig. 5 to Fig. 8.

TABLE V: Prediction results for the verification video sequences

	PLC[a]	SRO[b]	OR[c]
NR, no cross validation	0.88	0.85	0.54
NR, no correction step	0.63	0.62	0.77
NR, as proposed	0.82	0.75	0.75
PSNR	0.67	0.65	0.75

[a] PLC: Pearson linear correlation
[b] SRO: Spearman rank order correlation
[c] OR: Outlier ratio

The high prediction accuracy for the case of no cross validation in combination with a low prediction accuracy for the case when cross validation is used, but the correction step is skipped, shows that the pure NR model (without the correction step) can only predict the quality of sequences included in the calibration data base. Introducing the correction step only allows to accurately rate the quality of previously unknown sequences. This demonstrates the importance of this comparably simple correction step.

Comparing Fig. 6 and Fig. 7 reveals that with the help of this correction step the number of quality ratings with a large prediction error (predicted quality is far too low or far too high) is reduced. This is also reflected in Table VI. Here the percentage of data points is given where the prediction

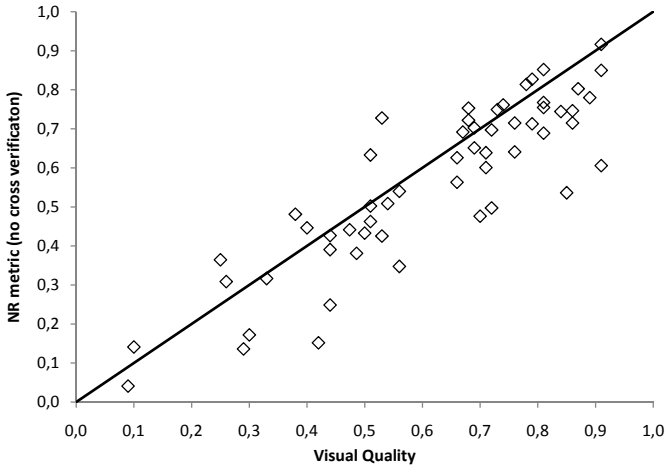

Fig. 5: Prediction results for the proposed NR metric - no cross validation

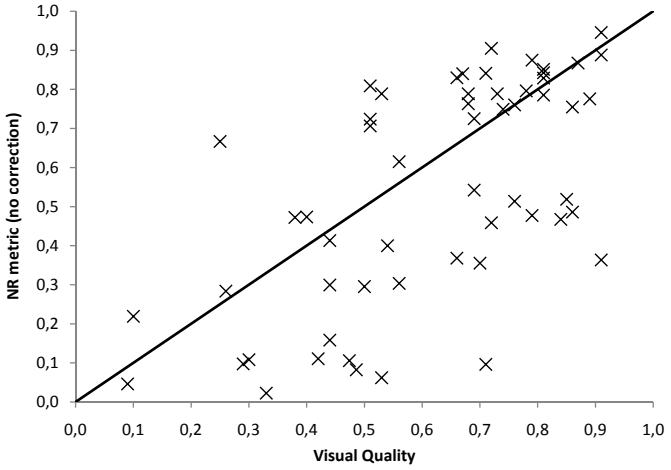

Fig. 6: Prediction results for the proposed NR metric - no correction step

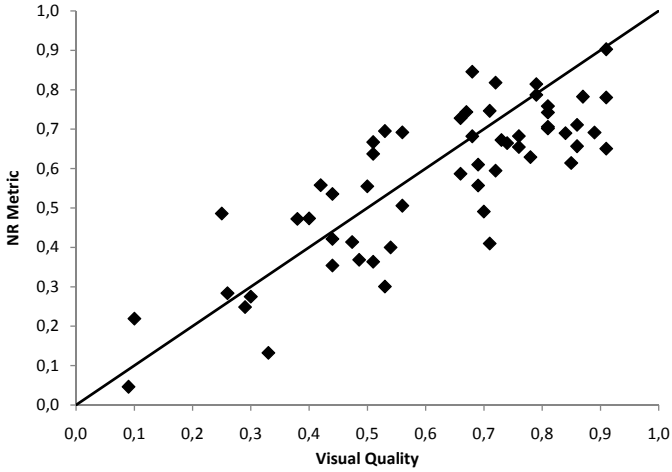

Fig. 7: Prediction results for the proposed NR metric - no data fitting

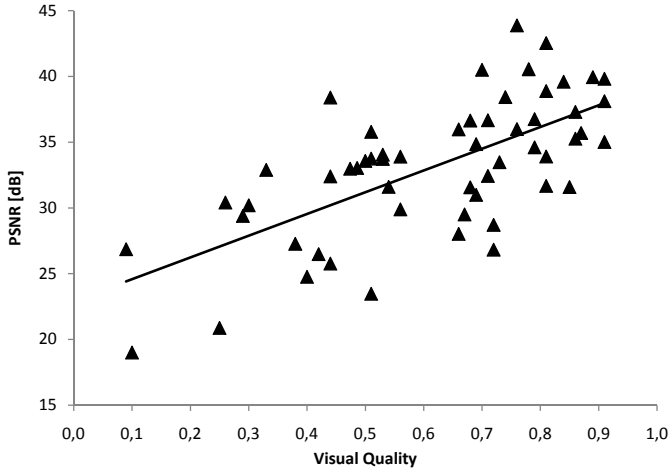

Fig. 8: Prediction results for PSNR - fitted regression line

error exceeds a certain threshold. Reducing the number of data points where there is a large prediction error, is of special importance for practical applications, as those cases would be especially noted either by the consumer (who would get a low quality, whereas the quality metric used by the provider would indicate a high quality), or the provider (who spends additional data rate on a video that is rated to have low quality, whereas in fact the visual quality would be already sufficient).

The outlier ratio is identical for PSNR and the proposed NR metric, although the overall prediction quality is significantly lower for PSNR. This shows that for some sequences PSNR actually does deliver good results. But even for the NR model where the correction step is skipped, the outlier ratio is within the same range. This shows that the outlier ratio does not tell too much about the prediction capabilities of the different metrics in this case. The low significance of the outlier ratio may be caused by the small confidence intervals already present in the results of the subjective tests that we used to determine the prediction accuracy of PSNR and our proposed NR metric.

TABLE VI: Percentage of data points where the prediction error exceeds a certain threshold

Threshold:[a]	0.1	0.2	0.3
NR, no cross validation	38%	11%	4%
NR, no correction step	64%	39%	23%
NR, as proposed	50%	13%	0%
PSNR	55%	21%	4%

[a] On a 0 to 1 quality scale

VI. Conclusion

We present a no-reference video quality metric that is based on a set of features, which are extracted from the pixel domain of a given video. The measured features are combined using different models, which have been calculated using a large calibration data set of video sequences. The selection of the appropriate model is done using an additional version of the video sequence, which is coded to exhibit a lower visual quality and is based on features of the video, not on its content.

We showed that even in a no-reference environment, a correction of the predicted visual quality is possible by introducing an additional coded instance of the video. This correction is not as powerful as in the case where two instances with known quality are available, but allows a simple, yet important correction. We showed that this correction is of essential importance for predicting previously unknown sequences. Without this correction step the prediction accuracy is significantly reduced.

To the best of our knowledge, this is the first NR video quality metric for AVC/H.264 that was verified on a comparably large data base, and that explicitly takes care to avoid overlaps between data used for training and verification. The results show that a high prediction accuracy can be reached, and a clear advantage compared to PSNR is achieved.

REFERENCES

[1] H.R.Wu and K.R.Rao, Eds., *Digital video image quality and perceptual coding*. CRC, 2006, ch. No-Reference Quality Metric for Degraded and Enhanced Video, pp. 305–324.

[2] R. R. Pastrana-Vidal and J.-C. Gicquel, "A no-reference video quality metric based on a human assessment model," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[3] S. Péchard, D. Barba, and P. Le Callet, "Video quality model based on a spatio-temporal features extractions for H.264-coded HDTV sequences," in *Proc. Picture Coding Symposium*, Nov. 2007.

[4] M. Ries, C. Crespi, O. Nemethova, and M. Rupp, "Content based video quality estimation for H.264/AVC video streaming," in *Proceedings of IEEE Wireless and Communications & Networking Conference*, 2007.

[5] Y. Fu-Zheng, W. Xin-Dai, C. Yi-Lin, and W. Shuai, "A no-reference video quality assessment method based on digital watermark," in *Proc. 14th IEEE Personal, Indoor and Mobile Radio Communications 2003*, vol. 3, Sep. 2003, pp. 2707–2710.

[6] 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.

[7] *ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG4-AVC), Advanced Video Coding for Generic Audiovisual Services*, ITU, ISO Std., Rev. 4, Jul. 2005.

[8] T. Vlachos, "Detection of blocking artifacts in compressed video," *IEEE Electron. Lett.*, vol. 36, pp. 1106–1108, Jun. 2000.

[9] K. Tan and M. Ghanbari, "Frequency domain measurement of blockiness in MPEG-2 coded video," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 3, Sep. 2000, pp. 977–980.

[10] Rohde&Schwarz. DVQ digital video quality analyzer. [Online]. Available: http://www2.rohde-schwarz.com/product/DVQ.html

[11] P. Gastaldo, S. Rovetta, and R. Zunino, "Objective assessment of MPEG-video quality: a neural-network approach," in *Proc. IEEE International Conference on Neural Networks*, vol. 2, Jul. 2001, pp. 1432–1437.

[12] ——, "Objective quality assessment of MPEG-2 video streams by using CBP neural networks," *IEEE Trans. Neural Netw.*, vol. 13, no. 4, pp. 939–947, Jul. 2002.

[13] P. Gastaldo, G. Parodi, and R. Zunino, "DSP-based neural systems for the perceptual assessment of visual quality," in *Proc. IEEE International Conference on Neural Networks*, vol. 1, Jul. 2005, pp. 663–668.

[14] N. Suresh, O. Yang, and N. Jayant, "AVQ: A zero-reference metric for automatic measurement of the quality of visual communications," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[15] M. Ries, O. Nemethova, and M. Rupp, "Performance evaluation of video quality estimators," in *Proc. European Signal Processing Conference EUSIPCO*, Sep. 2007, pp. 159–163.

[16] M. Montenovo, A. Perot, M. Carli, P. Cicchetti, and A. Neri, "Objective quality evaluation of video services," in *Second International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2006.

[17] F. Yang, S. Wan, Y. Chang, and H. R. Wu, "A novel objective no-reference metric for digital video quality assessment," *IEEE Signal Process. Lett.*, vol. 12, no. 10, pp. 685–688, Oct. 2005.

[18] M. C. Farias and S. K. Mitra, "No-reference video quality metric based on artifact measurements," in *Proc. IEEE International Conference on Image Processing*, vol. 3, Sep. 2005, pp. 141–144.

[19] M. Carli, M. C. Farias, E. Drelie Gelasca, R. Tedesco, and A. Neri, "Quality assessment using data hiding on perceptually important areas," in *Proc. IEEE International Conference on Image Processing 2005, (ICIP2005)*, vol. 3, Sep. 2005, pp. 1200–3.

[20] M. C. Farias, M. Carlib, A. Nerib, and S. K. Mitraa, "Video quality assessment based on data hiding driven by optical flow information," in *Proc. SPIE Image Quality and System Performance*, vol. 5294, Dec. 2003, pp. 190–200.

[21] B. Hiremath, Q. Li, and Z. Wang, "Quality-aware video," in *Proc. IEEE International Conference on Image Processing*, Sep. 2007.

[22] A. Ichigaya, M. Kurozumi, N. Hara, Y. Nishida, and E. Nakasu, "A method of estimating coding PSNR using quantized DCT coefficients," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 2, pp. 251–259, Feb. 2006.

[23] T. Brandão and M. P. Queluz, "Blind PSNR estimation of video sequences using quantized DCT coefficient data," in *Proc. Picture Coding Symposium*, Nov. 2007.

[24] 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.

[25] 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.

[26] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "A no-reference perceptual blur metric," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 3, Sep. 2002, pp. 57–60.

[27] Z. Wang, A. C. Bovik, and B. L. Evans, "Blind measurement of blocking artifacts in images," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 3, Oct. 2000, pp. 981–984.

[28] *ITU-T J.144. Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference*, ITU-T Std., Mar. 2004.

[29] T. Zahariadis and D. Kalivas, "Fast algorithms for the estimation of block motion vectors," in *Proceedings of the Third IEEE International Conference on Electronics, Circuits, and Systems, ICECS*, Oct. 1996, pp. 716–719.

[30] 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.

[31] M. Miyahara, "Quality assessments for visual service," *IEEE Communications Magazine*, vol. 26, no. 10, pp. 51–60, 1988.

[32] M. Miyahara, K. Kotani, and V. Algazi, "Objective picture quality scale (PQS) for image coding," *IEEE Trans. Commun.*, vol. 46, no. 9, pp. 1215–1226, Sep. 1998.

[33] Y. Horita, M. Katayama, T. Murai, and M. Miyahara, "Objective picture quality scale (PQS) for video coding," in *Proc. IEEE International Conference on Image Processing (ICIP)*, Sep. 1996, pp. 319–322.

[34] K. Kotani, Q. Gan, M. Miyahara, V. Algazi, and I. Jaist, "Objective picture quality scale for color image coding," in *Proc. IEEE International Conference on Image Processing*, vol. 3, Oct. 1995, pp. 133–136.

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

[36] P. List, A. Joch, J. Lainema, G. Bjøntegaard, and M. Karczewicz, "Adaptive deblocking filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 614–619, 2003.

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

[38] A. M. Tourapis, K. Sühring, and G. Sullivan, "H.264/MPEG-4 AVC reference software manual," ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Tech. Rep. JVT-X072, Jul. 2007. [Online]. Available: http://iphome.hhi.de/suehring/tml/

[39] MPEG Test Subgroup, "Report of the formal verification tests on AVC (ISO/IEC 14496-10 ITU-T Rec. H.264)," ISO/IEC JTC1/SC29/WG11, Tech. Rep. N6231, Dec. 2003. [Online]. Available: {http://www.chiariglione.org/mpeg/quality_tests.htm}

[40] ——, "Subjective test results for the CfP on scalable video coding technology," ISO/IEC JTC1/SC29/WG11, Tech. Rep. N6383, Apr. 2004.

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

[42] *ITU-T P.910 Subjective video quality assessment methods for multimedia applications*, ITU-T Std., Rev. 1, Sep. 1999.