Design and Verification of Video Quality Metrics

Tobias Oelbaum

Technische Universität München
Lehrstuhl für Datenverarbeitung

Design and Verification of Video Quality Metrics

Tobias Oelbaum

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr.-Ing. Eckehard Steinbach
Prüfer der Dissertation:	1. Univ.-Prof. Dr.-Ing. Klaus Diepold
	2. Univ.-Prof. Dr.-Ing. habil. Jens-Rainer Ohm, Rheinisch-Westfälische Technische Hochschule Aachen

Die Dissertation wurde am 21.11.2007 bei der Technischen Universität eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 17.03.2008 angenommen.

für Johanna

Acknowledgments

Even if the ordering of the words in this thesis was done by myself, the words itself were brought and given to me by many people I was lucky enough to meet on my way. Therefore, I would like to thank all of you, who ever discussed with me, taught me, and provided me a new word or a new idea. Three of those people are Fernando Pereira, Thiow Keng Tan, and Mathias Wien, and here they are named as special representatives of all the others.

Among the many ones I met, there are the ones, without whom this work would not have been possible. Therefore, I want to thank Klaus Diepold, not only for providing me this topic, but much more for his guidance, advise, and inspiration. I do have to thank Vittorio Baroncini for undisclosing all his secret tricks on subjective testing, and teaching me how to conduct a subjective test. Special thanks also go to Jens-Rainer Ohm for reviewing this thesis, and for his fast and very helpful feedback.

As this thesis would not have been possible without having many colleagues being around just next door, helping out whenever needed, and being there for discussion, I want to thank all my colleagues I was happy to work with during my time at the Institute for Data Processing at the Technische Universität München. Very special thanks to Chun Hui Suen for proofreading and correcting my broken English.

Still, there would have been no way of writing such a thesis for me, if there would not have been a source of energy and love on which I lived on during all that time. Thank you Eva, Margarete, Peter, and Verena.

Abstract

This thesis introduces a new method for the design of video quality metrics. In addition, a set of generic guidelines for the verification of video quality metrics is described. Video quality metrics are an essential tool for the design of efficient video compression and transmission systems, but up to today the methods used for the task of evaluating the quality of a video that has been processed for transmission are very limited in their performance. The goal of visual quality metrics is to assign a quality value to a processed video that is equal, or at least similar, to the quality that would be assigned to the same video by human observers. To achieve this goal, a number of methods have been proposed so far, but none of those has proven to achieve the desired accuracy for a wide set of different sequences or processing steps. The main problem in the design of such quality metrics is the very limited knowledge about the way humans perceive what is shown in a video. Whereas many things can be measured in a video, and also some properties can be measured that are known to have an influence on the perceived visual quality, still the quantitative influence of this measured features on the visual quality is unknown. As it is currently not possible to model the human visual system to an extend that would allow designing a visual quality metric only based on such a model, it is proposed to treat this human visual system as a black box. This black box is then described only by its inputs, which is the video, or a set of measurements that describe this video, and its output which is the visual quality that is assigned to this special video. It is proposed to determine the relationship of the input values and the output visual quality using methods provided by multivariate data analysis. Using this approach, the quantitative influence of measurements like blur, blocking, detail or motion on the visual quality can be determined. It is shown, that accurate visual quality metrics can be built using the described method.

Zusammenfassung

In dieser Dissertation wird eine neue Methode für den Entwurf von Videoqualitätsmetriken vorgestellt. Zusätzlich werden generische Richtlinien für die Verifikation von visuellen Qualitätsmetriken beschrieben. Videoqualitätsmetriken sind ein essentieller Teil für den Entwurf von effizienten Systemen für die Kompression und Übertragung von digitalem Video. Allerdings sind die für die Bewertung der visuellen Qualität von Video verwendeten Methoden noch deutlich von der gewünschten Genauigkeit entfernt. Das Ziel von visuellen Qualitätsmetriken ist, einem Video das komprimiert und für die Übertragung bearbeitet wurde den gleichen oder zumindest einen sehr ähnlichen Qualitätswert zuzuweisen, den auch ein menschlichen Zuschauer für dieses Video vergeben würde. Um dises Ziel zu erreichen wurden bereits zahlreiche Methoden vorgeschlagen, bis jetzt konnte aber für keine dieser Methoden eine ausreichende Genauigkeit für eine Vielzahl von unterschiedliche Sequenzen oder Bearbeitungsschritten gezeigt werden. Eines der Hauptprobleme bei dem Entwurf von solchen Qualitätsmetriken ist das sehr beschränkte Wissen darüber, wie der menschliche Wahrnehmungsvorgang bezüglich der visuelle Qualität von Video funktioniert. Während viele Eigenschaften eines Videos gemessen werden können und von einigen dieser Eigenschaften auch bekannt ist, dass sie einen Einfluß auf die wahrgenommene Bildqualität haben, so ist weiterhin unbekannt wie groß dieser Einfluss im Einzelfall ist. Da es auch in naher Zukunft höchstwahrscheinlich nicht möglich sein wird den visuellen Wahrnehmungsprozess ausreichend genau zu modellieren um eine visuelle Qualitätsmetrik basierend auf einem solchen Model zu entwickeln wird dieser Prozess hier als 'Black Box Model' betrachtet. Diese 'Black Box' wird nur durch die Eingangswerte und dazu dazugehörigen Ausgangswerte beschrieben. In diesem Fall sind die Einganswerte dieser 'Black Box' das komprimierte Video oder eine Beschreibung dieses Videos, der einzige Ausgangswert ist die visuelle Qualität dieses Videos. Der Zusammenhang zwischen den Eingangswerten und dem Ausgangswert 'visuelle Qualität' wird mit Methoden der multivariaten Datenanalyse modelliert. Mit diesem Ansatz ist es möglich den quantitativen Einfluss von Variablen wie Unschärfe, Blockartefakte, Bewegung oder Details, die aus dem Video extrahiert werden können, auf die visuelle Qualität zu bestimmen. In dieser Arbeit wird gezeigt, dass mit Hilfe dieser Methode präzise visuelle Qualitätsmetriken für Video entworfen werden können.

Contents

Chapter 1

Introduction

Only 20 years ago, distribution of video or TV services was restricted to either analog terrestrial TV broadcasting or distribution via analog video tapes (VHS). The visual quality of the distributed material at the receiver was limited in a narrow range, being determined by the distance of the receiver from the sender or the number of times the same video cassette tape was already played. Today, the channel between the distributor and the receiver can be deemed to be error free and independent from the distance between those two instances for many cases. DVDs can be played as often as wished without any quality degradation, video distributed over the internet is aiming to replace classical TV broadcasting scenarios, and finally the video is played back not only on TV screens, but on any device ranging from mobile phones with very small displays up to digital cinema systems in big theaters. Now, video is first compressed using one of many established video coding technologies at bit rates ranging from several kBits per second to some 100 MBits per second. Afterwards, the video is possibly sent over error prone channels, and recoded several times to fit for the last transmission mile and to the specifications of the display device.

As a result of these many distribution possibilities and processing steps, the visual quality of a transmitted video ranges from "*not acceptable*" to "*perfect*". The quality that is delivered to the consumer is often not known by the sender, and in many cases this quality can not be determined by the sender alone, but also depends on many unknown processing steps in the distribution chain. Still, knowing the visual quality of such a processed video is essential for almost all applications dealing with digital video. Optimization concerning the transmitted quality is only possible if this quality can be measured in a meaningful way. Video encoders need to find the best trade-off between visual quality and bit rate and billing of the customers may

vary not only depending on the content, but also depending on the delivered visual quality.

The most accurate way to determine the visual quality of such processed videos would be to conduct subjective tests using human observers. But displaying carefully selected and processed video sequences in a controlled environment to a group of people, and asking for the opinion of these people about the visual quality of the video, is time consuming, expensive, and can never be part of any practical application.

1.1 Video Quality Metrics: State of the Art

For these reasons many objective visual quality metrics were proposed in recent years, all of them trying to produce the same quality ratings a human observer would give to the same video. The first metric that was used to accomplish the task to objectively rate the visual quality of a processed video was the PSNR (Peak Signal to Noise Ratio). Although this metric comes from the time of analog recording, encoding and transmission, and despite of many contributions that showed the limitations of PSNR to accurately predict the quality of a video that was processed in the digital domain (e.g. [1, 2, 3, 4, 5]), PSNR is still the visual quality metric that is most often used, and that can be deemed to be the de-facto standard for objective visual quality evaluation.

Since then a large variety of visual quality metrics was developed. Starting from extensions to PSNR ([6, 7, 8]), and proceeding with comparably simple error calculation models ([9, 10]). Other metrics try to measure the most common artifacts ([11]), special image features ([12, 13, 14]), or rely on parameters given by the encoded bit stream ([15, 16]). A number of metrics try to model the human visual system (HVS) ([17, 18, 19, 20, 21]), whereas others do not take into account the visual aspect of video, but try to measure the amount of distortion that is introduced on the data using watermarks ([22, 23]). Metrics that do work in the pixel domain have been developed, as well as metrics that use a transformation into a different domain such as DCT or wavelet domain before analyzing image properties (e.g. [24, 25, 26, 27, 28, 29]), or metrics that first split one image into different regions ([30, 31]). Whereas some metrics try to estimate if one special error can be actually seen ([11, 32]), others take into account any error that can be measured. Databases with impairment models are used ([31]), as well as learning systems such

as neural networks ([33, 15, 34, 35, 36]), and video is assigned to different content classes to be able to use slightly different metrics for different types of videos ([16]). One quite common classification for all those metrics follows the use of a reference video, which is the original video for most cases. Metrics that do need the complete reference video are called "Full Reference" (FR) metrics, those using only parts or features of the reference are called "Reduced Reference" (RR) metrics, and those working only on the processed version are "No Reference" (NR) metrics. As any image quality metric can be used as video quality metric by assuming video to be a series of images, those are at least partly included in this work. If no distinction between video and image quality metrics should be made, these metrics are referred to as visual quality metrics.

Although at least some of the many proposed metrics were adopted by standardization bodies [37, 38, 31], and although other metrics like the SSIM (Structural Similarity) [9] have gained a comparably high popularity, none of those metrics comes close at replacing PSNR being the reference metric for objective visual quality measurement of processed videos. The reasons for this ongoing importance of PSNR can only be guessed and are probably manifold:

- PSNR is a measurement that is easy to understand. It does not require any transform or special image analysis, but just sums up the error that appears between the reference and the processed frame.
- PSNR is old. PSNR was already used in times of analog transmission of videos. People active in the field of video encoding "grew up" using PSNR, whenever the visual quality of a video should be rated.
- PSNR values are well known. Experts in the field of video encoding and transmission can roughly guess the visual quality of a video if they know the respective PSNR value.
- PSNR does perform quite good for certain circumstances. As shown in [39], PSNR performs very good for distortions introduced by noise. As it will be shown in section 4.5.2, PSNR also delivers very good results if only one source sequence is regarded.
- Most other metrics are quite complex. It is interesting to see, that the objective metric that is most often used for comparison beside PSNR is the comparably simple SSIM metric. There were also quite some extensions proposed for the SSIM by different authors, whereas for other metrics extensions were proposed

only by the authors of the base metric.

- So far presented video quality metrics do not deliver a very high correlation to the results of subjective tests. For many metrics the reported gain compared to PSNR is not statistically significant.

- The vast majority of video quality metrics lack of proper and independent validation. Nearly all visual quality metrics results concerning the prediction accuracy are presented by the proponents of the metric only. In addition, the databases used for this validation are comparably small, more or less unknown and different for each single metric proposed.

1.2 Problems in Designing Video Quality Metrics

One main problem of designing visual quality metrics is the very limited knowledge of the relationship of what can be measured in a video or image, and what is perceived by a human observer. As shown in Fig. 1.1a to Fig. 1.1d, the same measured error may result in a broad range of perceived visual quality.

One approach for the design of visual quality metrics would be trying to rebuild the human visual system, or at least to build a model of the vision system. But those metrics face the problem, that what they try to model is very complicated and up to this moment, not well understood. While at least parts of the eye and the primary visual system can be modeled to some extent, it is especially difficult to model the process that leads from seeing to what we actually perceive.

A second approach would be to measure what is present in a video and representing the video using a set of features that do have a relationship to visual quality. Here the problem arises, that it is not known how big the influence of one certain feature of the video on the visual quality actually is. Although it is known, that some errors that can be measured do have a negative influence on the visual quality, and the same is true for some positive features, like the amount of details or the color richness, the quantitative influence of the single measurements and possible interrelationships are unknown.

(a) Lost Line (b) Blur (c) Noise (d) Coded

Figure 1.1: Distorted pictures of the author, all with the same measured difference sompared to the original image

1.3 Contributions of this Work

Accepting the inability to model the HVS, and accepting the limited knowledge about how to gain the quantitative influence of a set of features and/or errors, it is proposed to build new video quality metrics using methods provided by multivariate data analysis. Multivariate data analysis is a tool that is widely used in chemo metrics and food science, where the aim is, to find the value of a latent variable (e.g. taste), by analyzing some fixed variables (e.g. sugar, milk, cocoa) that can easily be measured. A regression model is built that describes the relationship between the fixed input variables and the latent output variable. For the field of video quality assessment, this translates into determining the latent variable "*video quality*" by measuring fixed variables such as blocking, blur, color or motion. The main advantages of this approach are twofold:

- The HVS is regarded as a black box model, and no attempt is made to model any aspect of the HVS. The "black box" has some inputs (the fixed variables), and hopefully the output (visual quality) can be somehow predicted using these input values. No assumptions are made on how the input and output are correlated. This information is extracted from the measured data only.
- The approach requires no previous assumptions about the influence of the fixed variables, and input variables that later turn out to have no influence on the output can be included without affecting the model building process.

The proposed method for the design of video quality metrics is similar to what has been proposed in recent years, in the sense, that it combines feature measurements from the video and calculates the visual quality using a linear combination of those measured features. Previous metrics did pay high attention to the question of which

features to extract, but only very few work has been published on how to combine these extracted features into one quality measurement. Quite often, the combination of those features was done in a suboptimal way. One main contribution of this work is to provide a stable and effective framework that can be used to determine the weights for calculating one single visual quality value using a linear combination of a set of measured features and errors.

It is not possible to prove, that a visual quality metric does actually work using mathematical formulas only. As stated before, a video quality metric *does* work, if the quality that is assigned to one video by the metric is equal (or at least very close) to the quality, human observers would assign to this video in a subjective test. Therefore, the performance of a video quality metric can only be measured in relationship to the results of subjective tests. For this reason, it is necessary to obey the following simple rules, when measuring the performance of a visual quality metric:

- The visual tests used for the performance evaluation have to be performed according to the rules for such visual tests. Of great importance is the use of a standardized viewing environment, following a precisely defined test procedure involving a large number of test subjects.
- A documentation of the performed subjective tests is necessary.
- It is not allowed to use the same data for performance evaluation and for the development of the metric.
- No fitting of the results to the data of this subjective test should be performed.

Unfortunately no metric presented so far was verified according to all four rules given above.

These simple rules are extended and described in detail in Chapter 3. The aim of this chapter is to set up a procedure that allows trustworthy verification and easy comparison between different video quality metrics. The proposed method for the design of video quality metrics is presented in Chapter 4 and 5. As a proof of concept, a set of reduced reference video quality metrics were developed, and the results for these metrics are given in Chapter 6.

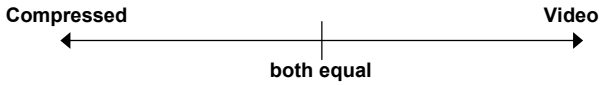
Chapter 2

Visual Quality Metrics

Many visual quality metrics have been proposed in recent years, and it is not the aim of this work to describe these metrics in detail. The goal of this work is not to provide one new quality metric that should replace existing metrics, but to provide methods to systematically build such metrics. Therefore, the metrics presented in this chapter are described from a function principle point of view. Instead of depicting a very high number of metrics, the aim is to describe only those metrics that differ significantly from others, and include other metrics only with its differences to these "prototypes". The goal of this chapter is to give an overview about how visual quality metrics work, and allow the reader to combine the design principles for visual quality metrics as presented in Chapter 4 with the functional principles of the described metrics to build new visual quality metrics.

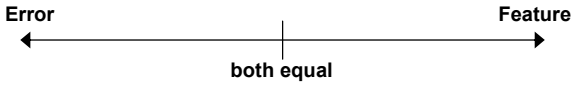
Analyzing the design of visual quality metrics on a higher abstraction level result in three different axes that can be used to describe the functional principle of most visual quality metrics. Those axes are:

- Processing domain. Visual quality metrics can work in the video/pixel domain, or in the compressed (bit stream) domain. Whereas some metrics do evaluate features or errors in both domains, most of the time it is possible to classify a visual quality metric as belonging to one of the two domains.

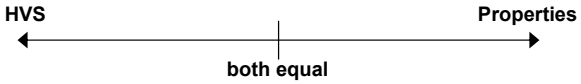


- Errors versus features. While one possibility for a quality metric is to sum up the error that can be measured (error based), also the opposite approach is

possible. Here one measures the "positive" features that are still existent in the processed video (feature based). The first option would result in subtracting the sum of errors from an ideal video, whereas the second option would result in adding some amount of quality starting from a "blank" video.



- HVS versus video properties. Metrics can be based on the knowledge of the HVS (what do humans see), or based on properties of the video (what does the video show).



These three axes can be used to build the *QualityMetricCube* (QMC) (see Fig. 2.1). With the exception of the metrics that evaluate the visual quality based on fragile watermarks, it should be possible to assign one data point in the QMC for each visual quality metric. Compared to the standard classification for video quality metrics of "Full/Reduced/No Reference" metrics, the location of a quality metric in this cube can give much more information about the functional principle and the possible strengths and weaknesses of this quality metric. In short, the advantages and disadvantages of (theoretical) metrics that lie *on* the cube (and therefore at the extremes of one axis) are given in Table 2.1, a more detailed discussion will be given in the following sections.

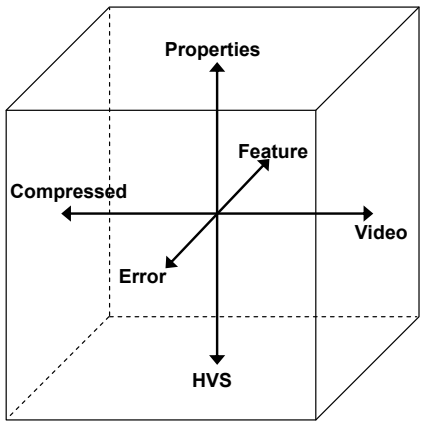


Figure 2.1: QualityMetricCube

The most prominent and some of the most interesting visual quality metrics are presented in this chapter. As the standard classification into "Full/Reduced/No

Reference" metrics provides a clear distinction between the different metrics, and coming from an application point of view this classification is very meaningful, it is preserved in this work. In addition, the (rough) position of the presented metrics inside the QualityMetricCube is given graphically. The size of the given area inside the QMC is a measurement of the uncertainty of this classification. A small area would show, that this classification is believed to be more accurate, whereas a large area would indicate, that the classification is more difficult for this metric and therefore no distinct point can be given.

Table 2.1: Advantages and disadvantages of (prototype) video quality metrics

	Advantage	Disadvantage
Processing Domain	no reference evaluation possible low computational complexity possible high correlation reported	tailored to specific codecs / processing steps tailored to specific transmission channels
Video Domain	not necessarily tailored to specific processing steps independent from different transmission channels	no reference evaluation difficult
Error	many algorithms to measure specific errors exist usually limited number of different errors	unknown how big the visual impact of one error is usually assumes the reference video to be perfect may be tailored to specific processing steps errors can be masked by features or other errors
Feature	many algorithms to measure specific features exist reference video does not need to be perfect	unknown how big the visual impact of one feature is unknown number of features that have a visual impact features can be masked by errors
HVS	theoretically allows to judge the video as a human observer would judge the video no reference quality evaluation possible independent of any specific processing step	can not be modeled to a sufficient level high computational complexity probably different metrics (and different results) for different viewing conditions
Video Properties	many algorithms to measure specific properties exist independent of viewing conditions	unknown how big the visual impact of a special property is unknown number of properties that affect the visual quality may be tailored to specific processing steps no reference evaluation difficult

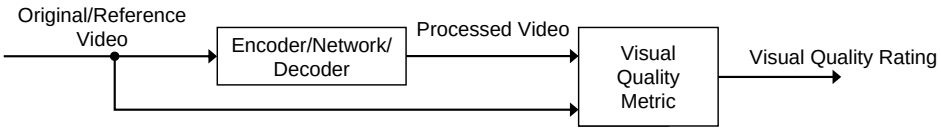


Figure 2.2: Full reference visual quality metric

2.1 Full Reference Quality Metrics

One straight forward way to build a visual quality metric is to compare the processed video to the reference video. This comparison can be done by calculating the pure difference of the pixel values, but also by extracting several features of the video and comparing the values of those features. Features of the video can be related to the amount of details, color properties, typical artifacts such as noise or blocking, motion present in a video, or the distribution of image frequencies. As a complete comparison between the reference video and the processed video is done, metrics using this method are denoted to be full reference metrics.

So far, most of the presented metrics are full reference metrics, including the metrics standardized in [37, 31]. The advantage of not only using the processed video, but also the reference video for quality evaluation, results in an easier and more accurate prediction of the visual quality. However, this comes on the cost of two major drawbacks:

- Access to this reference video is not possible in many application scenarios where the quality of the video should be measured at the receiver side.

- The quality of the processed video can only be given relative to the reference video. If the quality of the reference video is not perfect[1], those metrics can not deliver an indication about the absolute visual quality of the processed video.

In the following, a selected number of full reference metrics are introduced, starting with PSNR. PSNR is not only the first metric that was used for the task of visual quality evaluation, but still is the metric that is used most often.

[1] All full reference as well as reduced reference metrics assume the reference video to be of perfect quality.

2.1.1 Peak Signal to Noise Ratio

The Peak Signal to Noise Ratio (PSNR, see Equation 2.1) is a very simple, and quite old technology to measure the amount, to which a signal is distorted by noise. PSNR only calculates the mathematical difference between the reference images and the processed images[2]. Due to its logarithmic scale, very simple properties of the HVS are captured. For video, as well as for images, most of the time PSNR is calculated on the luminance part of the images only. However, correlation between PSNR values and subjective quality is quite limited, a fact that is well known and has been documented several times: [1, 2, 4, 3, 39]. In spite of this limitations, PSNR still is the only visual quality metric that is widely accepted and used, and therefore PSNR is the de-facto standard for measuring the visual quality of a video or an image.

$$\mathrm{PSNR} = 10 \log I_{max}^2 / \left(\frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} \left(I_{cod} - I_{orig} \right)^2 \right) \tag{2.1}$$

While overall correlation between PSNR values and subjective quality for videos or images compressed using up to date coding technologies is low, PSNR is a quite good visual quality metric if distortion caused by noise should be measured. In [39] a Pearson Correlation of 0.986 between PSNR and subjective ratings was reported for 174 images distorted by Gaussian noise. The maximum Pearson Correlation reached by one of the other nine tested metrics was 0.989, which is not significantly better than the performance of PSNR, whereas seven models perform significantly worse than PSNR. PSNR also does deliver good prediction results if only one sequence is coded to several quality levels (preferably using the same coding technology). This is shown in detail in Section 4.5.2.

Several extensions to PSNR were proposed to better take into account the peculiarities of human vision, and at the same time retaining a simple and easy to understand metric. [6] Lee *et al.* propose to calculate the PSNR only on those regions of the image that have been classified to contain edges[3]. In [8], the authors propose to perform the PSNR measurement on the subbands of a wavelet decomposition, whereas in [41] it is proposed to measure the PSNR on DCT values. Both authors propose to

[2] for PSNR calculation as for many other video quality metrics video is just a series of single images.

[3] Such a classification can be easily done using an edge detection algorithm, such as the one proposed by Canny in [40].

Table 2.2: Extensions to PSNR

Extension	Description	Pearson Correlation	Pearson Correlation PSNR
Edge-PSNR [6]	PSNR of edge regions	0.864	0.769[a]
Wavelet PSNR [8]	PSNR of wavelet subbands	0.691	0.648[b]
PSNR_{rf} [7]	PSNR reached by a certain percentage of frames	n.a.	n.a.

[a] Data taken from [3], averaged over 525 and 625 line data

[b] Averaged data

calculate the visual quality as a weighted prediction of the subband PNSR values. For assessing the quality of video transmitted using different channels, an extension to PSNR is proposed in [7]. Instead of measuring the PSNR, it is proposed to measure PSNR_{rf}, which is defined as the PSNR achieved by $f\%$ of the frames in each one of the $r\%$ of the channel realizations. Finally Tong *et al.* propose to use a weighted combination of PSNR and SSIM (Structural SIMilarity Index, see Section 2.1.2) in [36]. Weights between these two metrics are adjusted according to the distortion present in the video. Table 2.2 gives an overview of three extensions to PSNR together with the reported Pearson Correlation to subjective tests, compared to correlation values for standard PSNR.

PSNR Prediction Accuracy

To evaluate the prediction accuracy of PSNR as an objective visual quality metric, results from two subjective tests that were carried out as part of the standardization effort inside the Moving Picture Experts Group (MPEG) were analyzed. These two tests were the verification test for AVC/H.264 [42], and the entry tests that led to the scalable extension of AVC/H.264 [43]. For the latter one, Wien and Benzler showed in [4], that PSNR can not be used to compare videos processed using different coding technologies. Therefore, only results for AVC/H.264 were analyzed. Results as presented in Table 2.3 show the following:

- The overall Pearson Correlation between PSNR and the results of subjective tests is 0.629, which is a comparably low number.

Table 2.3: Pearson correlation for PSNR for [42, 43]

Codec	Resolution	Pearson Correlation[a]	Data Points	Slope	Offset[b]
All	All	0.629	197	13.22	22.99
AVC/H.264	All	0.636	118	15.07	22.01
MPEG-4[c]	QCIF, CIF	0.673	55	14.29	23.83
AVC/H.264, MPEG-4	QCIF	0.743	59	17.22	22.01
AVC/H.264	QCIF	0.718	35	16.48	21.99
MPEG-4	QCIF	0.767	24	22.04	21.25
AVC/H.264, MPEG-4	CIF	0.614	84	12.95	24.02
AVC/H.264	CIF	0.674	54	17.49	20.73
MPEG-4	CIF	0.571	31	10.38	25.98
AVC/H.264, MPEG-2	SDi	0.712	41	11.56	20.40
AVC/H.264	SDi	0.709	17	12.63	19.53
MPEG-2	SDi	0.685	24	11.27	20.63
AVC/H.264	4CIF	0.621	17	26.10	17.21

[a] For the Pearson Correlation see Section 3.4.1

[b] Visual quality can be computed as (PSNR-Offset)/Slope

[c] MPEG-4 ASP

- Pearson Correlation for AVC/H.264 is only slightly higher (0.636).

- The maximum correlation for any subset of this test data is 0.767, which is comparable to results reported for PSNR in [3].

- Slope and offset parameters that are needed to convert the PSNR values into visual quality in the range of 0 to 1 vary significantly both for different resolutions and for different coding technologies.

2.1.2 Structural Similarity Index

The Structural Similarity Index (SSIM) was first introduced as 'Universal Image Quality Index' (UIQI) in [44]. Instead of using traditional error summation methods, the SSIM models any image distortion as a combination of loss of correlation together with luminance and contrast distortions. The SSIM is not explicitly built on any HVS model, as the authors believe, that modeling a HVS leads to more accurate visual quality metrics only if the HVS could be modeled more precisely than currently done. The idea behind this approach is that

> The main function of the human eyes is to extract structural information from the viewing field, and the human visual system is highly adapted for this purpose. Therefore, a measurement of structural distortion should be a good approximation of perceived image distortion. ([44])

The SSIM is therefore defined as

$$\text{SSIM} = \frac{\sigma_{xy}}{\sigma_x \sigma_y} \frac{2\,\overline{x}\,\overline{y}}{\overline{x}^2 + \overline{y}^2} \frac{2\,\sigma_x \sigma_y}{\sigma_x^2 + \sigma_y^2} \tag{2.2}$$

with x and y representing the reference and the processed images, $\overline{x}$ and $\overline{y}$, representing the respective mean values, and the σ values representing the standard deviations. The first component in Equation 2.2 represents the degree of linear correlation between the reference and the processed image and is taken as a measurement for how much the structure is preserved ($s(x, y)$). The second component compares the mean luminance values ($l(x, y)$). The standard deviations σ_x and σ_y can be seen as contrast estimation, and therefore the third component could be seen as contrast comparison value ($c(x, y)$). In [45], the authors present a general form of the SSIM which is given as

$$\text{SSIM} = [s\,(x, y)]^{\alpha}\ [l\,(x, y)]^{\beta}\ [c\,(x, y)]^{\gamma} \tag{2.3}$$

So far, the authors did not use or propose other values than 1 for α, β and γ. The SSIM is not calculated globally for the whole image, but it is proposed to calculate the SSIM on a sliding window approach, calculating the SSIM for small local areas such as 8×8 pixels.

In [46] and [47], the authors present two video quality metrics that are based on the SSIM. Both proposals include motion as a kind of masking effect, meaning that the measurement for one single frame is adjusted if high motion is detected. In [46], the measurement is also adjusted if high blockiness or a lot of blur appears, and instead of working on the luminance channel only, they propose to also perform the

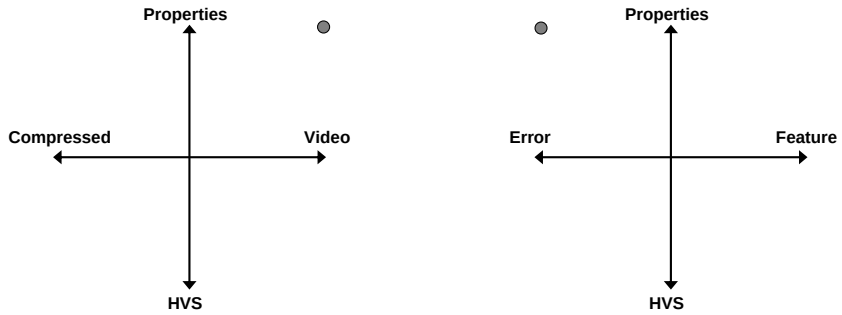


Figure 2.3: Location for PSNR and SSIM inside the QMC

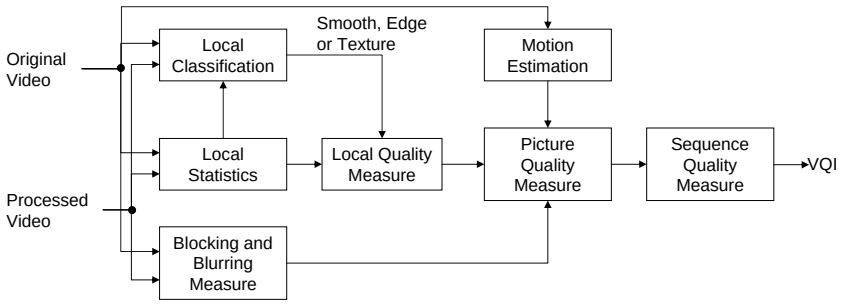


Figure 2.4: SSIM based video quality metric as proposed in [46]

measurement on the Cb and Cr channels. The system is shown in Figure 2.4. In their second proposal, the SSIM-based video quality metric includes an adjustment according to the luminance of the evaluated region, and the adjustment based on the motion estimation (see Figure 2.5). Comparing the presented results for the two video quality metrics shows, that the performance of both metrics seems to be very close to each other.

One main advantage of the SSIM is its simplicity and low computational complexity. On the other hand it is quite obvious, that the SSIM can be improved:

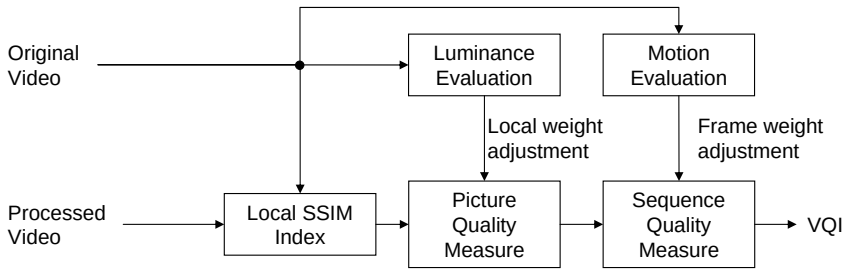


Figure 2.5: SSIM based video quality metric as proposed in [47]

- It is not obvious why α, β and γ are all equal to 1.
- The basic SSIM only works on the luminance plane, and does not consider color.
- No known features of the HVS such as different sensibility for errors that appear in different frequency bands are used.

Therefore it is not surprising that quite some extensions for the SSIM were proposed. The original authors extend their SSIM to a multi-scale approach in [48], where the SSIM is evaluated for different frequency bands of each image and the individual results are then combined into one measurement. In [49] they extend their basic idea by proposing to decompose the distortions of one image into a linear combination of structural and non-structural distortion components. There are also extensions of the SSIM by other authors: similar to the Edge-PSNR ([6]) Chen *et al.* propose to calculate the structural comparison $s(x, y)$ of the general SSIM as given in Equation 2.3 on the edges only to better gather the quality of badly blurred images in [50]. In [41] a very simple extension of the SSIM is proposed by evaluating the SSIM on different subbands of a wavelet decomposition of the image, and defining a linear combination of these subband SSIM values. Another very simple extension is presented by Medda and DeBrunner in [51], where it is proposed to evaluate the SSIM on the color channels in addition[4]. Again, a wavelet decomposition of the original images is used in combination with the SSIM in [25]. Here the moduli of the decompositions are compared using the SSIM. As already mentioned, Tong *et al.* propose to use a weighted linear combination of PSNR and the SSIM in [36].

2.1.3 ITU-T J.144

Beside national standards, such as PSNR, which is an ANSI (American National Standards Institute) standard [37], there is only one recommendation from an international standardization body targeting objective video quality metrics. The title of this ITU-T (International Telecommunication Union) standard is "Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference" [31]. This standard is also known as ITU-R BT.1683 "Objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference" [38], though most of

[4]It is quite interesting to see, that the last two contributions propose something very similar to what has been proposed by the original authors even after those have presented their extensions.

the time references are made to the original ITU-T document. The four video quality metrics included in this standard differ from other full reference video quality metrics for the following three reasons:

- These metrics are the only ones that are standardized by an international body.
- The performance of these metrics was evaluated by an independent group, and this evaluation was done in a very transparent way, based on a large database.
- As these metrics are standardized, a complete description is available that allows others implementing these metrics.

The four metrics included in this recommendation[5] deliver very similar prediction quality, though the metrics itself do vary substantially. It has to be noted that the call that lead to the standard (see [3]) required to perform the matching between the original and processed sequences, something that is normally out of the scope of pure video quality metrics (though a complete system of course would need such a matching step). In the following the four metrics are presented to an extent that the functional principle of each of the metrics can be understood.

British Telecommunications

The "British Telecommunications Full Reference Metric" (BTFR) follows the common principle of combining a set of parameters that were extracted from the processed and the reference video into one single quality value. Before extracting those parameters from both videos, the reference video is matched block by block to the processed video. The result of this step is a matched reference video, where every block in the reference video has minimum difference to the co-located block of the processed video. In particular the following features are analyzed to gain the desired parameters:

- PSNR (MPSNR, SegVPSNR): two PSNR values are calculated: MPSNR is the matched Y-PSNR between the processed image and the reference image that has been matched to the processed image. The SegVPSNR is calculated on the V color plane (upsampled to full resolution), using the matching vectors that have been found in the matching process. Here, the PSNR is calculated

[5]Two more metrics are included as annexes, but these two metrics were not part of the previous test effort, and are verified on completely different data sets. For this reason and as the status of these two metrics is somewhat unclear, those were not regarded.

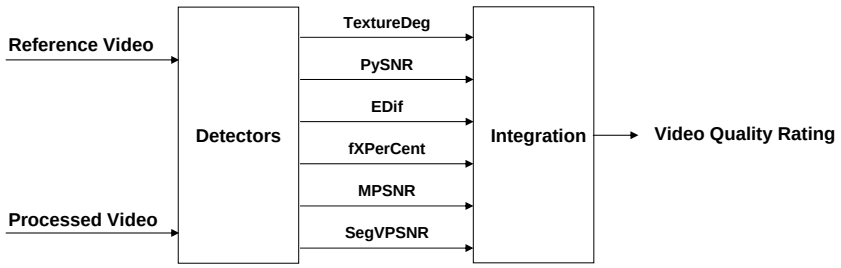


Figure 2.6: BT full reference video quality assessment model

separately for every block, and the SegVPSNR is the mean value of these block-wise V-PSNR values.

- Spatial frequency distribution (Pyramid SNR): the spatial frequency distribution is measured by a PSNR measurement that is performed on the result of a three stage spatial pyramid transform of the single frames.
- Edges (EDif): for evaluating how much edges are preserved, both videos are passed through an edge detection algorithm separately. The two edge maps are then mapped by a block matching algorithm, and the number of edge pixels in each corresponding block is compared.
- Texture (TextureDeg): the texture is measured on the processed image only by counting the number of turning points in the intensity values along the horizontal axis.
- Matching (fXPerCent): fXPerCent gives the maximum percentage of blocks, where the x-component of the matching vector that matches the reference image to the processed image is identical.

The basic principle of parameter extraction and parameter combination is shown in Fig. 2.6. A block diagram of the parameter extraction process is given in Fig. 2.7. These parameters are extracted for each frame separately and averaged over the whole sequence. The averaged parameters are then integrated into one quality measurement $PDMOS$ by building a weighted sum (see 2.4). It is quite interesting to see, that the weights given for the 625 line sequences (PAL) differ quite significantly from the weights given for the 525 line sequences (NTSC) (see Table 2.4).

$$\text{PDMOS} = \text{Offset} + \sum_{k=0}^{5} AvD\,(k)\,W\,(k) \tag{2.4}$$

Table 2.4: Integration parameters for the BTFR metric

k	Parameter Name	W(k)($525lines$)	W(k)($625lines$)
0	TextureDeg	0.043	−0.680
1	PySNR	−2.118	−0.570
2	EDif	60865.164	58913.294
3	fXPerCent	−0.361	−0.208
4	MPSNR	1.104	−0.928
5	SegVPSNR	−1.264	−1.529
	$Offset$	260.773	176.486

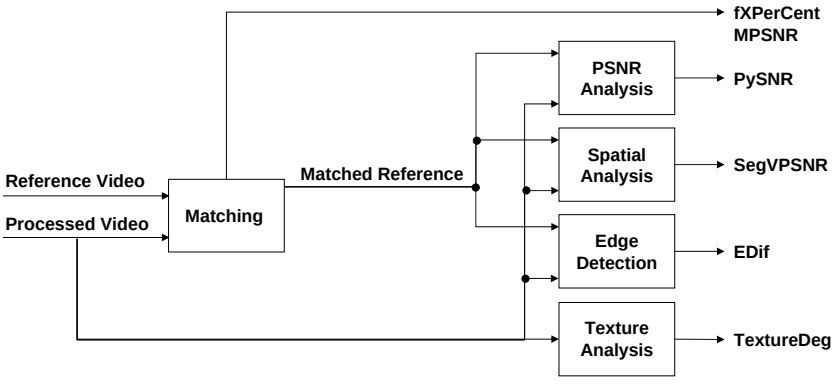


Figure 2.7: BT full reference parameter extraction

Edge-PSNR

The Edge-PSNR metric from Yonsei University is probably one of the simplest metrics that have been proposed so far. As mentioned before, this metric evaluates the PSNR only for those pixels that have been classified to belong to an edge region. The edge detection algorithm is performed on the reference video, and a mask sequence is created by thresholding this edge video sequence. In addition two post-adjustments are made:

1. For high quality videos, Edge-PSNR overestimates the visual quality, therefore a piecewise linear correction is made for high values.

2. For low quality videos (especially for videos with blurred edges) the Edge-PSNR is corrected taking into account the overall number of edge pixels, and the number of edge pixels that appear at the same location in both videos.

The design principle that is used for the Edge-PSNR metric is the same as the one for the BTFR metric, and is one of the most common design principles for visual quality metrics. This design principle can be described as a four step procedure:

1. Filter: the image or the video is first preprocessed using perceptual filters. For this special case this filter is an edge filter, another popular class of filters for this application are wavelet filters.

2. Measure: feature measurements are performed on the preprocessed images or videos. For the Edge-PSNR only one filtered version of the video exist, and the feature measured here is PSNR. The output of this process is a set of features.

3. Collapse/pooling: the measurements for one feature have to be collapsed in space and time (also known as pooling). The simplest pooling method is to calculate the average value, as it is done for the Edge-PSNR. Of course, different pooling methods can be used for different features, and several different pooling methods can be used for the same feature. The output of this pooling step is a number of parameters that is equal or bigger than the number of features measured times the number of filtered versions of the image or video.

4. Combine: if the output of the pooling step contains more than one parameter, these parameters have to be combined into one quality value. This combination can be done e.g. by building a weighted sum, or using a Minkowski summation (vector summation). For the Edge-PSNR metric this step is not necessary.

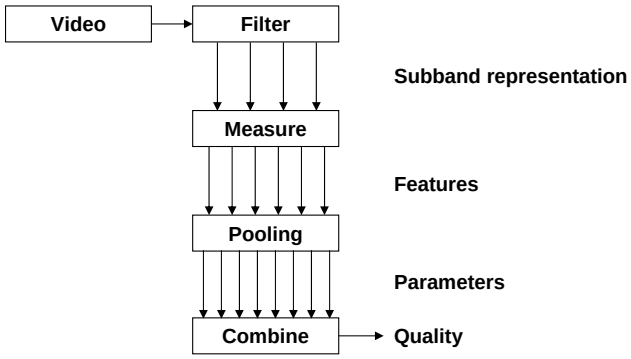


Figure 2.8: Block diagram for the design principle of "filter, measure, collapse and combine"

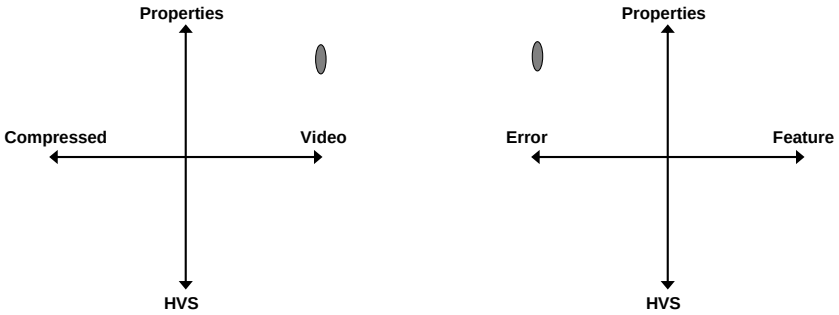


Figure 2.9: Location for Edge-PSNR inside the QMC

Image Evaluation based on Segmentation

The "Image Evaluation based on Segmentation" (IES) metric is based on three main parts:

1. Segmentation of the reference video into edge, plain and texture regions. The reference and the processed video are then compared for each of these three regions, and the result of this comparison are the objective measures m_i.

2. The use of a database of impairment models that is used to calculate two parameters F_i, G_i needed for transforming the measurements from the previous step into impairment levels L_i. The impairment levels are calculated as

$$L_i = 100/\left[1 + \left(\frac{F_i}{m_i}\right)^{G_i}\right]. \tag{2.5}$$

3. The use of additional instances of the original video that are generated using standard video encoding methods (with known settings, and therefore at least to some extent known quality), and that help in computing the variables F_i, G_i that are needed to transform the measurements from the first step.

So here the main part is not extracting the features from the video, but to "interpret" these extracted values according to

- The value of the same feature that is gained by analyzing two additional videos.
- One database that contains subjective ratings for a set of sequences.
- Temporal attributes of the database sequences, taking into account if this is a dynamic or more static scene.

The weights that are needed to combine the impairment levels into one quality prediction are gained by the use of the database and the additional coded instances. An overview of the metric is given in Fig. 2.10. This metric is quite complex, as it requires two additional encoded instances of the video (though the proposed encoders are very simple), and to calculate the objective measurement m_i for these additional videos. In addition, calculation of the variables F_i, G_i and the weights W_i requires to solve some minimization problems. Of course, this metric relies very much on the database itself, as well as on the ability of the two encoders used, to generate videos close to the video of interest. This makes the metric the least generic one out of the four included metrics.

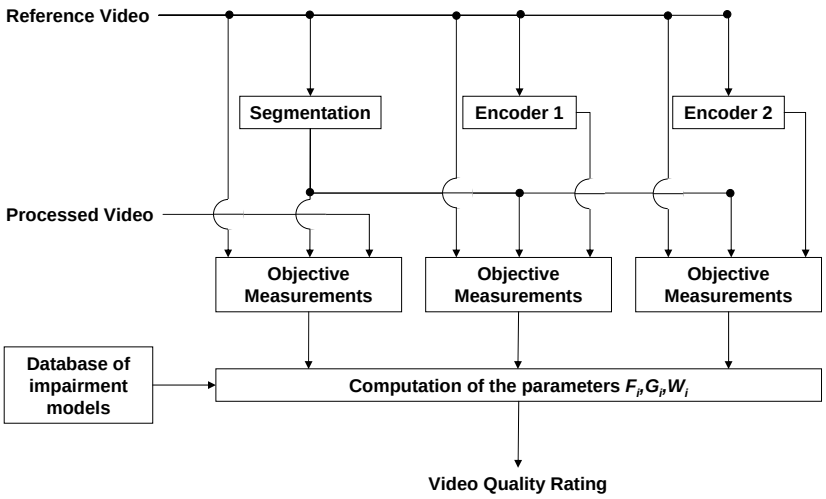


Figure 2.10: Overview of the IES metric

NTIA-VQM

The Video Quality Metric (VQM) presented by the National Telecommunications and Information Administration (NTIA) differs from the other three metrics, as it is a reduced reference metric. After calibration and matching of the video streams, the visual quality is computed by combining several different parameters that are computed from extracted features from the reference and the processed video. This metric again follows the basic principle that was used for the Edge-PSNR and the BTFR proposal. The main attributes of this metric are:

- Features are calculated separately for each spatio-temporal region (S-T region) of the reference and the processed video. No direct comparison of the reference and the processed video is necessary, but the extracted features for corresponding S-T regions are compared. The authors propose to use S-T regions of 8 pixels times 8 lines times 6 frames for 525 line videos ($8 \times 8 \times 5$ for 625 line videos).

- It is proposed to measure the following features:

  - spatial impairments

  - distortions in the chrominance channels

  - contrast

  - temporal information.

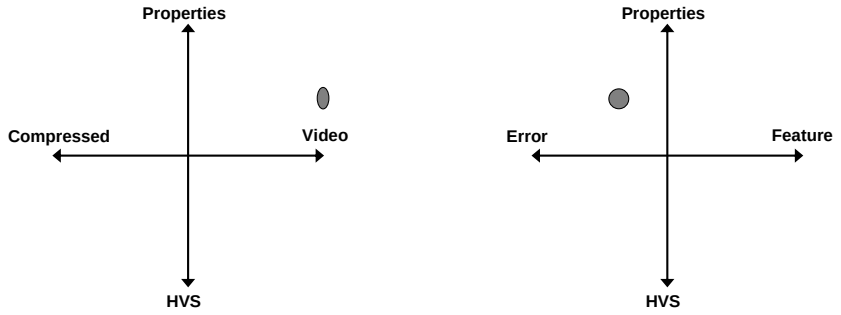


Figure 2.11: Location for BTFR, IES and NTIA VQM inside the QMC

- The features extracted from the reference video and the processed video are compared for each S-T region separately. For comparing the features several comparison functions such as error ratio or Euclidean distance are used.
- After comparing the features for each S-T region spatial followed by temporal collapsing is performed (pooling step). The authors propose to include collapsing functions that emphasis the strongest distortions. For each of the features more than one collapsing function may be used. The output of this step is a set of parameters, giving statistical information about how one feature is distributed over time and space.
- To take into account nonlinear relationships between feature measurements and perceived quality, a nonlinear mapping may be applied to the parameters.
- Finally the parameters are combined using a set of weights that were determined by regression between the parameters and subjective test results.

The framework of this metric was standardized by the ANSI in 1996 as ANSI T1.801.03-1996, 2003 [52].

2.1.4 And Beyond

Many more full reference visual quality metrics were proposed, most of them following the principle of "filter, measure, collapse and combine" as described in Section 2.1.3. Differences between those metrics can be found in the filters used for temporal and spatial filtering, features that are measured in the filtered subbands, pooling methods, and the combination models. Some metrics also model possible interactions between the different subbands, to take into account masking effects, and to better model the HVS. Out of the extensive list of proposed full reference

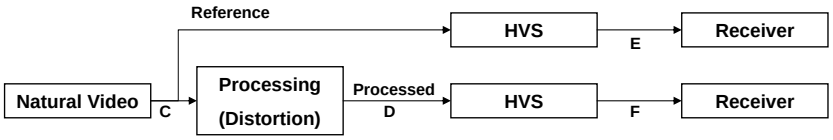


Figure 2.12: Mutual information concept (figure copied from [53]). Mutual information between C and E quantifies the information the brain can extract from the reference video, similar the mutual information between C and F gives the information that can be extracted from the processed video

.

metrics, one approach that differs from this design principle is worth mentioning.

Mutual Information and Motion Models

Bovik and Sheikh extended their information fidelity approach for still images introduced in [13] by including motion, and presented the first systematic approach to model motion for video quality measurement in [53], and later extended in [54, 55]. They propose to decompose a video using a set of Garbor filters that work in the spatio-temporal domain. They further assume, that the Garbor coefficients of the processed video can be modeled by applying a distortion operator on the reference video coefficients, and it is therefore possible to model the output of the Garbor filter $f_i(x, y, t)$ for the processed video, which is denoted as $D_i(x), i = 1, 2 \ldots N$ (N being the number of filters used), as $D_i(x) = G_i(x)C_i(x) + N_i(x)$ where $C_i(x)$ is the output of the Garbor filters for the reference video, $G_i(x)$ representing a gain field, and $N_i(x)$ is the Garbor decomposition of an AWGN (Additive White Gaussian Noise) channel. The video quality is then calculated as the mutual information between the coefficients of the reference and processed video sequences. Beside the assumption that the frequency components of a temporally filtered video follows certain rules, and that visual quality is closely related to the mutual information that is shared between the reference and the processed video, the crucial point in this work is the use of 3D bandpass filters to decompose the video.

2.2 Reduced Reference Video Quality Metrics

Reduced reference video quality metrics could be seen as something that lies on the path from full reference to no reference metrics. While for full reference metrics

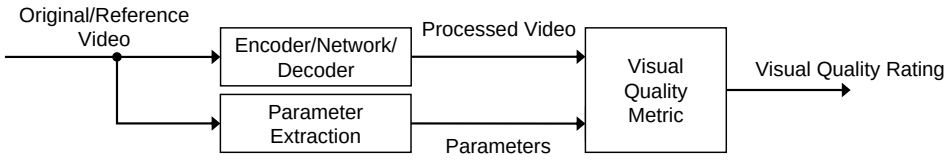


Figure 2.13: Reduced reference visual quality metric

the reference video has to be evaluated together with the processed video, reduced reference metrics only require the result from some parameter extraction process that was applied to the reference video to perform the quality evaluation for the processed video. Obviously any reduced reference metric has to follow two goals:

- Just as any full reference or no reference visual quality metric, the results of the metric should be close to the votes of human subjects.
- The amount of additional data that needs to be transmitted (often referred to as reduced reference bit rate) to the receiver should be as small as possible.

As the design principles for the quality metric itself do not significantly differ from those that are used to build full reference metrics, the following sections focus on the part of reducing the amount of reference data that has to be transmitted. So far two different approaches have been proposed: either the number of parameters that are calculated from the reference video is reduced, or a reduced version of the reference video itself is transmitted.

2.2.1 Comparing Parameters

One reduced reference metric was already introduced in Section 2.1.3: the VQM metric from NTIA included in ITU-T J.144 is the only reduced reference metric in this recommendation. Here, for every S-T region of the video a set of seven different parameters (gained from four different features) is extracted from the reference video, and is compared with the same set of parameters from the same S-T region in the processed video. However, the amount of reduced reference data that has to be transmitted in addition to the compressed video requires several Mbit/s. This may be more than the compressed video needs, and is far too much for a reasonable transmission system. In [56] the authors of the original VQM metric propose a reduced reference metric that uses less than 10 kbit/s for the reduced reference data. While the design principle is not changed in comparison to their original contribution, and also the same features are extracted, the new reduced reference

metric uses much larger S-T regions to extract those features, resulting in a reduction of S-T regions by a factor of 80. In addition, the pooling of the features is done on comparably large S-T areas (the authors suggest blocks of 96×96 pixels, and a duration of two seconds). As this very large feature extraction areas, and the even larger pooling areas, may not be able to capture distortions that appear only for a very short time period one reduced reference parameter was added to measure temporal artifacts. Meng *et al.* use the same original VQM metric, but define foveation regions inside one video, and adopt the size of the S-T regions according to this foveation regions in [57]. Using this step, they reduce the number of S-T regions by a factor of 8. The approach of transmitting the parameters for S-T regions of different sizes allows steering the reduced reference data rate, and therefore adapting the reduced reference metric to the bandwidth that is available.

2.2.2 Reducing the Reference

A different approach to obtain a reduced reference video quality metric is proposed by Masry *et al.* in [24]. The authors propose to generate a subsampled version of the reference video that previously has been decomposed into several subbands, using a set of wavelet filters. The actual quality comparison is then done in a full reference way, using the subsampled reference video. Again, this approach is scalable in terms of reduced reference data rate, as the reduced reference bit rate can be adjusted by the extent to which the filtered video is subsampled. Their approach is motivated by the fact, that distortions in the video are correlated across space, scale, and time, and therefore it is not necessary to evaluate the whole video sequence. As an uncompressed version of the subsampled reference video is needed, again the amount of data that is needed to transmit this reference video may be higher than the amount of data that is needed to transmit the processed video. Fig. 2.14 shows the basic principle of this approach.

2.2.3 Features for Reduced Reference Metrics

Closely linked to the approach of reducing the amount of reference data, is the selection of features and the respective parameters that are evaluated. Some features and their parameters may be better suited for a reduced reference approach than others. However, this has not been investigated in video quality metrics so far. One reason - beside the fact, that the number of proposed reduced reference video quality

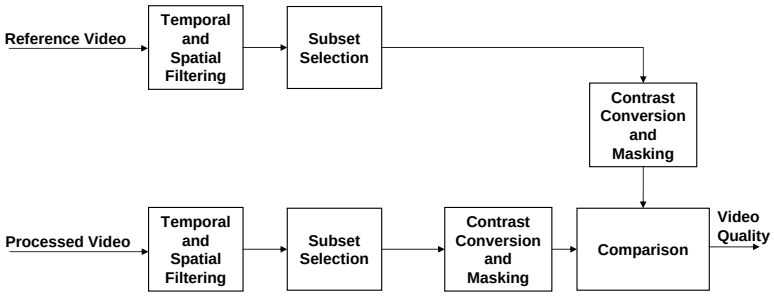


Figure 2.14: Basic principle of the reduced reference approach with subsampled reference

metrics is very low - may be that objective video quality evaluation has not reached a level where one could afford looking for anything else than features that deliver a high prediction accuracy. It is therefore worth taking a look at reduced reference image quality metrics, as objective image quality evaluation has reached a much higher level of prediction accuracy, and in this field it is therefore beneficial to look for other properties than pure prediction accuracy.

In [25] Zhai *et al.* decompose an image using a set of wavelet filters, and then search for edge regions across different subbands to build what they call "multi-scale edges". Their reduced reference image quality metric is based on comparing only these regions. Kusuma *et al.* propose to use a combination of blocking, blurring and "spatial activity", which should indirectly measure ringing artifacts, in [11]. This metric therefore captures the most common image distortions. One special attribute of this metric is, that it does not require any spatial registration between the processed and the reference images, as the pooling of the features results in one parameter for the whole image. Wang and Simoncelli compare the marginal probability distributions of wavelet coefficients of the reference and the processed images. Again no registration between the two images is needed. A systematic approach to select features useful for reduced reference evaluation is presented in [58]. Starting from a model of the HVS, the authors propose to use a combination of features that represent the structural information in an image.

2.2.4 A Reduced Reference Metric using Neural Networks

One approach for a reduced reference video quality metric that shares the common design principle of "filter and measure", but proposes a different way for the "collapse and combine" part is proposed by Le Callet *et al.* in [33]. The feature measurements

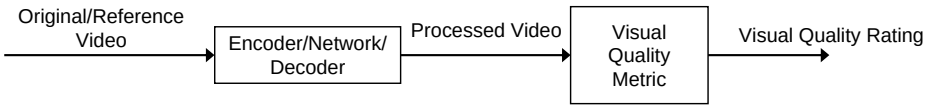


Figure 2.15: No reference visual quality metric

they use are blocking, a measurement for temporal changes, and two frequency content measures to evaluate blurring and tiling. These four features are collapsed spatially for the whole frame resulting in 12 spatial parameters for the three color planes of each image. But instead of proposing a set of fixed functions for temporal collapsing, and a set of fixed weights for the combination step, they propose to use a trained time delay neural network (TDNN) for this task. According to the authors such an approach has two main advantages:

- The TDNN mimics the temporal integration of distortions of the HVS.
- The TDNN can construct new distortion measurements by combining the initial features that are input to the TDNN.

2.3 No Reference Video Quality Metrics

The final goal in objective visual quality evaluation is to be able to judge the visual quality not only with the same accuracy as human observers, but to be able to do this without having access to the reference video, again just as human observers can do this. Any video application scenario that includes a receiver side would need a no reference approach for estimating the video quality. But as most human observers feel more confident about their quality judgment if they can compare one processed video to a reference, than if they should give a vote without seeing any reference, no reference visual quality measurement faces the problem of interpreting parameters extracted from the processed video without knowing the values for those parameters for a reference video (where the visual quality is inherently known). Obviously, features such as blur can result from compression, but at the same time they may represent properties of the reference video.

Due to the attractiveness and in spite of the problems involved, quite some no reference metrics for video quality evaluation were introduced. Beside the standard "filter, measure, collapse and combine", two more main approaches are proposed: the use of watermarks is shortly discussed in 2.3.2, and the evaluation of features

directly extracted from the bit stream (which could be seen as a special case of the standard method, where the filtering step is omitted) is introduced in 2.3.3. In addition, several no reference metrics exist that focus on one special type of distortion such as blur or blocking.

2.3.1 "Standard" No Reference Metrics

Most no reference visual quality metrics that follow the standard approach of combining several parameter measurements into one visual quality value, try to include the most prominent artifacts, such as blocking, blurring or noise. Obviously, whereas full reference and also reduced reference metrics most of the time try to measure what is retained in the image or video, no reference quality metrics focus on what is added (blur, blocking, ringing, noise), and what normally does not appear in natural images. A straight forward implementation of this approach for a video quality metric is presented in [59], where the authors combine noise, blocking, blurring, ringing and jerkiness.

2.3.2 The Use of Watermarks

The use of watermarks or data hiding methods is one popular method for a no reference image or video quality metric [23, 22, 60, 12]. In [23], [22], and [60] the authors propose to include a fragile watermark in the video at the sender, and rate the video according to the degradation of the embedded watermark. Differences between the proposed solutions can be found in the embedding algorithm and the parts of the video where these watermarks are embedded. Obviously, these metrics do not measure the image quality, but the distortion that can be found in some inserted data that is exposed to the same compression and transmission system as the image or video. A different approach is proposed by Wang *et al.* in [12]. Here, a (robust) watermark is used to transmit some extracted features of the original image in the image itself, thus avoiding the need of an ancillary data channel for the reduced reference features.

One restriction all watermark based metrics have in common is, that those methods are no real no reference methods as:

- Access to the original image or video is needed to embed the watermark. In most application scenarios for no reference visual quality measurement this is not possible.

- Embedding a watermark, may it be fragile or robust, does change the image or video that has to be encoded. This most probably leads to a bit rate slightly different compared to the bit rate that would be necessary to code the same image or video without that watermark (rate increase of 1% to 5% is reported in [23]). So whereas no ancillary data channel is needed, still some additional bit rate may be needed.

For these reasons, those metrics should not be accounted to be no reference metrics, but reduced reference metrics instead. Embedding some extracted features for a reduced reference evaluation as watermarks in an image or video may seem to be an elegant way, but how to transmit reduced reference features should not be part of a metric itself, as different transmission systems have different possibilities to transmit some small amount of additional data which is connected to the image or video. This is especially true for video, as video normally is not "silent movie", but does contain audio or timing information and all this "extra" information is coupled on a systems layer that also may include those reduced reference features.

2.3.3 Evaluating Bit Stream Features

A small number of contributions use features of the compressed bit stream for the estimation of the visual quality. Using such features as bit rate, the number of blocks that do not contain any DCT values, or the number of motion vectors is especially interesting from a complexity point of view. As there is no need for extra extraction algorithms to get those features, and the features can be read directly from the compressed bit stream, those metrics can be of very low complexity. In [16, 61] Rupp *et al.* evaluate only the bit rate and the frame rate, and in [62], they extend their approach by some motion models that work on the decompressed video. Gastaldo *et al.* use the bits per frame in addition to statistics concerning the type of used blocks, or the used motion vectors ([15, 63]). Features gathered directly from the bit stream are also used in [64], though it is not given which features are used. Those metrics can be characterized by the QMC given in Fig. 2.16.

The short list of features from the bit stream given in Table 2.5 should capture the most relevant features that can be extracted, and may give a starting point for quality metrics that can be built using such features.

Before starting to build metrics on features directly extracted from the bit stream, two major concerns for this method should be taken into account:

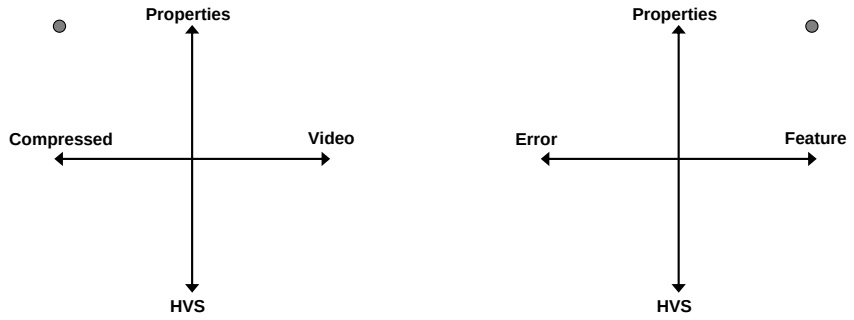


Figure 2.16: Location for metrics that analyze the bitstream inside the QMC

Table 2.5: Bit stream features for quality evaluation

Feature	Options
Bit rate	overall, separated for different frame types, over time
Frame rate	mean, over time
Motion vectors	number, length, direction in space or time, similarity in space or time
Block types	percentage of different types, clustering in space or time
DCT coefficients	percentage of blocks with at least n coefficients

- Many of those features are very much encoder dependent. This is especially true for features like “bit rate” (or the very closely linked quantization parameter “QP”). Of course, for the same encoder, the visual quality should be very much linked to the bit rate. But a metric built on this information, will only work for one very special encoder (maybe even with very special settings only). It is not too surprising, that none of the contributions that use some features like “bit rate” do state that a different encoder or at least different encoder settings were used for design and verification of the respective metric, leading to the conclusion, that only one encoder was used for all experiments (also see Section 3.2).

- In addition, all these features are very much content dependent. A high detail mostly static scene will not need much motion vectors, and the frames that are coded differentially may have many blocks with no additional DCT information (as this information already can be found in the reference frame). On the contrary, a low detail, high motion scene may need many motion vectors, and may have more blocks with additional DCT information. It is quite obvious, that for these two sequences no quality information can be given, based only on the number of motion vectors combined with the number of blocks that contain no DCT values.

As a consequence if features from the compressed bit stream are used, the selected features should be as generic as possible, and should not be the result of one special encoder, motion search strategy, or rate control approach. As proposed in [16], it may be beneficial to assign one sequence to one content class before calculating the visual quality. Obviously, special care should be taken not to use similar or even the same data for design and verification of such a metric. Especially, different encoders should be used, even though those encoders may produce bit streams that can be decoded by the same decoder.

One special case is the estimation of PSNR by evaluating the DCT coefficients as presented for AVC/H.264 encoded video in [65, 66]. Here, PSNR is estimated in a no reference approach using the decoded DCT coefficients, and it is shown, that this PSNR estimation does perform very well even for different encoders. However, as only PSNR is estimated, the performance of this method regarding real visual quality is probably quite limited, due to the limited prediction accuracy of PSNR. In addition, cascaded encoding, using different encoders may spoil the PSNR estimation process. A similar approach for PSNR estimation was presented in [67].

2.3.4 Single Distortion and Feature Measurements

Beside complete visual quality metrics, a number of no reference metrics exists that aim at judging one special artifact, or one special feature, like the amount of details. Again, most of these specialized metrics follow the standard design principle, only targeting a less general goal, and therefore working on a smaller number of features, and combining a smaller number of parameters. Examples for metrics that focus on special artifacts, are the blocking measurement introduced by Wang *et al.* in [68], the blur measurement proposed by Marziliano *et al.* in [69], and noise measurements such as [70]. Blocking artifacts, which are one of the most obvious artifacts for DCT block based image and video codecs, can be located easily in the spectrum of the image or video. While the spectrum of a natural image decreases monotonically, the spectrum of an image with blocking artifacts has distinct peaks at frequencies that are multiples of the block size. The spectrum of the processed image can also be used to get a measurement of the amount of spatial details that are present in an image. Sheikh *et al.* propose to use models for the frequency distribution of natural images for an overall quality metric [13, 71, 72, 73]. Blur is defined as a loss of detail, but as a no reference metric can not measure a loss of something (this would require to evaluate the original image), in [69] blur is defined as the width of edges which can be easily measured.

While these measurements are suited for still images and for video, the property that distinguishes video from still images, is motion that appears between consecutive frames. Metrics that measure how good motion is preserved in a processed video, are usually designed to measure the effect of lost frames or video at reduced frame rate ([74, 75, 76]). So far no metrics were proposed that are suited to measure motion artifacts that appear even if all frames are available.

Chapter 3

Metric Verification

The performance of a video quality metric can only be evaluated by the use of data obtained in subjective tests, and no theoretical proof is possible, that one quality metric does actually work. For this reason, verification of video quality metrics is not possible without performing carefully designed and conducted subjective tests. As the goal of an objective quality metric is, to produce results that are equal to the results gained in subjective tests, the performance can be measured by comparing the results of the metric with the results of subjective tests.

Subjective testing of visual quality is not a trivial task, but requires accurate design of the tests, and precise execution of these tests. In the following sections, the necessary steps for conducting precise subjective tests are described, starting from international recommendations that are available for subjective video testing, that precisely describe the environment and test procedures. In Section 3.1.2, guidelines for the design of subjective quality tests are given along with important details that have to be considered when conducting the tests. Section 3.1.3 describes the processing of the subjective results, and especially focuses on testing the validity of the gained results, and the subjective test.

After having calculated the objective visual quality, using the quality metric, and having measured the visual quality through subjective testing, those two results have to be compared with each other to measure the performance of the objective metric. While no special knowledge about statistical methods is necessary to measure the performance of a video quality metric, still some common problems can be found in many contributions to visual quality metrics. Quite often, the same data is used for development and verification, a problem that is described in detail in section 3.2.

Parts of the reported performance come from fitting the objective data to the subjective results. Why this is a problem, is demonstrated in Section 3.3. The correct use of simple statistical tools to measure the performance of objective visual quality metrics is described in Section 3.4. Statistics from [3] and [31] are used throughout this chapter to demonstrate the shortcomings of current verification procedures.

3.1 Subjective Testing

Subjective visual quality tests are still the only reliable source of funded knowledge about the visual quality of processed video sequences. In spite of recent and coming advances in the field of objective visual quality measurement, it is most likely, that subjective testing will not be replaced completely by objective metrics. One reason for the continuous need for subjective testing is, that visual quality itself is not a constant. The visual quality of the same processed video measured in different subjective tests can vary in quite a broad range, even if all those tests were performed correctly, and therefore all this different quality values have to be taken equally correct and meaningful. Having different correct quality values for the same processed video seems to be an antagonism at the first glance. However, this antagonism is rapidly resolved if we consider, that quality expectations vary not only over time, but also depend on the geographic region, the viewing environment, or the quality of the other videos that are shown in the test. The following short examples explain how perceived visual quality can vary depending on time, location, viewing environment, or expectations:

- Currently there is a change in Europe from TV broadcasted at "Standard Resolution" to "High Definition" TV (HDTV). In some years from now viewers might take HDTV to be the standard and what is currently widely accepted will then be "low resolution" and therefore "low quality" TV.

- Viewers will judge differently according to the quality they are used to so far, so most probably what would be "high quality" in some third world countries would be "low quality" for people in Japan, where HDTV is already standard for some years.

- Viewers will expect, and accept, lower quality if they see a video in a mobile environment than on a fixed high resolution screen.

The aspect of varying quality for one video being at least partly determined by the other videos in the test is also known as "contextual effect", an effect, that is inherently present in any subjective test. The biggest variation on the measured visual quality however is not caused by differences concerning the geographical location or time, but by differences in the test design. As it is quite easy to build a subjective test, which results are still "correct" but meaningless, subjective testing has to be performed with much care and in awareness of the limitations of such tests[1].

What can be measured is therefore no absolute quality number, but just a relative visual quality that can only be interpreted correctly if details about the test design and the conduction of the subjective test are available. As it is highly desirable to be able to produce results that are very close to each other across different tests (at least, if those tests are performed within a certain time slot, and within a more or less homogeneous group of people), methods for subjective quality evaluation are subject of standardization efforts for next to 30 years. The results of these efforts are described in the following section. Following international recommendations for a subjective test is a good start, and may be an essential basis for such tests, but still these recommendations only build the framework for subjective tests. What has to be done to perform an accurate and successful subjective test is described in Section 3.1.2. Section 3.1.3 treats the issue of processing the data gained through subjective tests and Section 3.1.4 highlights some common problems in reported subjective tests that are used to verify objective quality metrics.

The "problem" of the non existence of one "true" visual quality value results in the question how to measure or judge the quality of a visual test. This question can not be easily answered, but the result of the discussion in the following sections can be summarized as follows:

A good subjective test

- Has small confidence intervals for the subjective votes[2].
- has low percentage of votes that have to be removed as errors or outliers.
- is based on test procedures described in international recommendations.

[1] Correct but meaningless results would mean, that the subjective test lead to quality values that are only relevant for this special test scenario, but those results are not helpful in real world scenarios as here the perceived quality will be completely different e.g. due to significant differences in the viewing conditions.

[2] For a good subjective test the 95% confidence interval should be below 0.08 for each single test case, and the mean of the confidence intervals should not exceed 0.06.

- comes with a detailed description.
- produces results that are reproducible if the same tests would be performed a second time at at different place using different people.
- produces results that are meaningful to real world scenarios.

3.1.1 International Recommendations

In 1974, the CCIR (Consultative Committee for International Radio or Comite Consultatif International des Radiocommunications), which was the predecessor of what is now the ITU (International Telecommunication Union), released the first version of what is currently known as ITU-R BT.500 [77]. As video at this times was inherently coupled to TV systems, the title of this important recommendation was and still is "Methodology for the Subjective Assessment of the Quality for Television Pictures" (the word video does not even appear in the title). The latest version of this document is revision 11 from 2002, and even if the recommendation still focuses on the quality of TV pictures, and does not explicitly mention video presented on different displays, or transmitted over different channels than those used for TV, this recommendation is the basis for any subjective visual quality test. This does not only cover those involving any kind of video, but tests for still images are also based on this recommendation.

The recommendations of this document are very clear and do provide a good abstract about what is the content of this recommendation (quoting from [77]):

> The ITU Radiocommunication Assembly ... recommends
>
> 1. that the general methods of test, the grading scales and the viewing conditions for the assessment of picture quality, described in the following Annexes should be used for laboratory experiments and whenever possible for operational assessments;
> 2. that, in the near future and notwithstanding the existence of alternative methods and the development of new methods, those described in § 4 and 5 of Annex 1 to this Recommendation should be used when possible; and
> 3. that, in view of the importance of establishing the basis of subjective assessments, the fullest descriptions possible of test configurations, test materials, observers, and methods should be provided in all test reports;
> 4. that, in order to facilitate the exchange of information between different laboratories, the collected data should be processed in accordance with the statistical techniques detailed in Annex 2 to this Recommendation.

So this recommendation does not only describe the subjective test methods, but also defines appropriate viewing conditions, and requests a precise report of the conducted tests. The text also states, that visual quality assessment is still developing.

Further standardization work resulted in ITU-R BT.710 [78], which especially focuses on high definition TV services, and ITU-T P.910 [79] from 1999, which was the result of an emerging need for subjective testing methods for multimedia content. Also aiming at testing multimedia content is a method developed by the EBU called SAMVIQ (Subjective Assessment Methodology for Video Quality), which is currently under standardization as ITU-R BT.700. Current research work in this area includes simultaneous evaluation of audio and video [80, 81, 82], and first attempts have been made to use the displays of mobile phones for the evaluation of video in mobile environments [83]. Although, the ITU decided in 1999 that the combined measurement of audio and video quality should be studied, and that this studies should be completed by 2006 ([84]), so far no recommendation for the combined evaluation does exist.

The first test method that was used for subjective visual quality tests is the DSCQS (Double Stimulus Continuous Quality Scale) test method. DSCQS still is the primary choice for subjective tests in TV environment, or when high quality video is evaluated. Being time consuming, DSCQS is believed to deliver more accurate results, and allows the test subjects to better differentiate between subtle quality differences. The title of this method already tells two important characteristics:

- DSCQS is a double stimulus method. Double stimulus means, that the processed video and a reference video are shown. In most cases the reference video is the unprocessed original video.
- A continuous quality scale is used. Contrary to a discrete quality scale, the used grading scale is continuous for the test subject, and will be sampled only for processing the data.

ITU-R BT.500 proposes two different variants of the DSCQS method, of which Variant II is used in most cases. In short, the DSCQS method is characterized by the following additional properties:

- For Variant I, the test subject switches between the processed video and the reference video as often as he desires before making a decision on the quality of both videos. For Variant II, the processed video and the reference video are

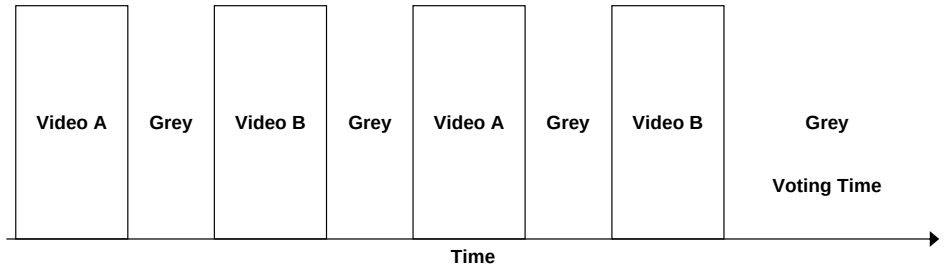


Figure 3.1: DSCQS presentation structure

shown twice before the subjects are asked to vote (see Figure 3.1).

- The presentation order of processed and reference video varies randomly.
- Subjects are asked to vote for the quality of the processed video and the reference video. The final quality is then calculated as the difference between those two votes (see Fig. 3.2).

Alternatively to DSCQS, the DSIS (Double Stimulus Impairment Scale) test method can be used. This method is also part of ITU-R BT.500, and differs from DSCQS in the following aspects:

- The reference sequence is always displayed first.
- Test subjects are asked to vote only for the processed video sequence.
- This vote should be relative to the reference sequence, and should express the level of impairment.
- A discrete scale with only five impairment levels is used for voting.

The DSIS method is also included in ITU-T P.910, here it is called Degradation Category Rating (DCR). In [85] a new test method was proposed that can be seen as a combination of the DSCQS method and the DSIS method. While the grading scale was the same as used for DSIS (five grade impairment scale), the random order from DSCQS (Variant II) was retained. The first presentation of the pair of reference and processed video is therefore used by the subjects to guess which of the two sequences is the reference sequence, whereas the second presentation is used for grading the quality of the processed video.

In addition, ITU-R BT.500 includes several single stimulus methods that differ mainly in the number of repetitions one test case is presented, and the grading scale that is used. Also included are stimulus comparison methods, where two videos are

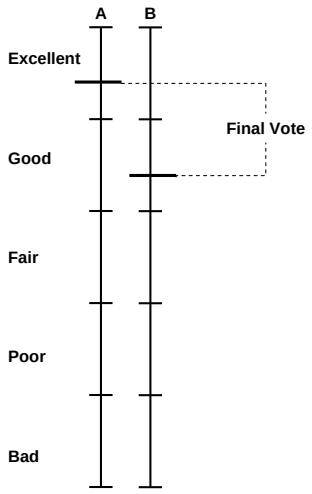


Figure 3.2: DSCQS rating

displayed simultaneously. Especially interesting for time variant transmission channels are methods that allow a continuous quality evaluation over time instead of providing just one vote. Those are the SSCQE (Single Stimulus Continuous Quality Evaluation, also see [86, 87, 88]), and the DSCQE (Double Stimulus Continuous Quality Evaluation [86, 89]).

ITU-R BT.500, as well as ITU-T P.910, also define the viewing conditions that have to be set up for conducting a visual test. This includes the room setup, display type and setup, lighting conditions and the distance of the observers to the screen. Some details on the setup of the room are given in Section 3.1.2.

The SAMVIQ test method, first introduced by the EBU, aims at evaluating video in a multimedia environment. The main difference to other test methods previously standardized, is the amount of interaction that is required from the test subjects. For all other test methods (except DSCQS Variant I), the test subjects are only asked to vote and the time line of the test, or the presentation order of the sequences are controlled by the person who designed the test, and no influence of the test subjects is desired or possible. In contrast, SAMVIQ allows the test subject to see one video as often as desired, change votes previously made, and to compare two (or more) processed instances of the same video directly. In addition, an explicit reference video is provided to the test subjects. Whereas some contributions report a better stability of the results for SAMVIQ ([90]), this could not be verified in other tests ([91, 92]). It seems, that this method does produce results similar to those that can be gained using established methods. As SAMVIQ is very time consuming, this

method does not seem to be suitable if a large number of videos should be tested. One additional concern regarding SAMVIQ is, that the test method is much more artificial than common single stimulus testing, as under normal conditions, no video will be watched several times and compared to the same video at different quality levels.

3.1.2 Designing and Conducting a Subjective Test

The available international recommendations do set the framework, and provide the tools necessary for a successful subjective test, but using these tools in an improper combination would most probably lead to results that can not be used in a meaningful way. Before starting a subjective evaluation, many questions have to be answered:

- Which test method should be used?
- What type of display is appropriate?
- Should the the recording of the votes be done with paper or electronic devices?
- Which test sequences and which coding conditions should be used?
- how to order the sequences inside the test?
- One person per display or more at the same time?

The above list is not complete, but it captures the main questions that arise for any subjective test. This section tries to discuss the above questions, and provides help in designing and conducting a successful subjective test.

Even before asking the above questions, and trying to find an answer (and the best suited test setup), one main question has to be posed: what should be the outcome of this subjective test? Possible answers to this question could be:

- Codec A performs better than codec B for the selected coding conditions.
- Codec C delivers a certain visual quality for the selected coding conditions.
- Codec D is more robust against transmission errors than codec E.
- Changing the coding settings of codec F, results in a visual quality that is better (by x%) than codec F with standard settings.
- A target visual quality can be reached with codec G at bit rate x.

- For reaching the visual quality of codec H at bit rate x, codec J needs a bit rate of y.

Knowing what should be the result of the subjective tests already restricts the number of possible test methods and variations. If the relative quality should be evaluated (performance of codec A compared to codec B), then a stimulus-comparison method could be used, which is not very time consuming, easy to understand by the test subjects, and can provide a precise answer to the given question. But if the absolute visual quality of (at least) one codec should be evaluated, this would require a test with an absolute quality or impairment scale, and the use of a reference is not of final importance.

The selection of the appropriate test method can be broken down into two main decisions: Single Stimulus or Double Stimulus, and the selection of the grading scale that best fits.

Double Stimulus and Single Stimulus

Double Stimulus (DS) methods (two different videos are presented to rate the quality of only one of these two videos) are preferred if:

- The relative visual quality between two different videos should be evaluated.
- The visual quality should be measured relative to a reference video.
- Videos with a medium to high visual quality (videos with only few distortions) should be evaluated.
- Robustness concerning transmission errors should be evaluated.

On the contrary, Single Stimulus (SS) methods (only the processed video is presented to rate the quality of this video) are preferred if videos with low to medium quality (videos with severe distortions) should be evaluated. Of course, SS methods are also used if no (common) reference is available. This is the case for scalable video coding, where only for the highest resolution layer a reference is available, and the visual quality of the lower resolution layers can only be evaluated in a SS test. The use of SS methods for low to medium quality videos is motivated by the fact, that compared to a (high quality) reference video, all low quality videos will have more or less similar low visual quality, and therefore the ability of the test subjects to discriminate between the different levels of low visual quality decreases.

For evaluation in a TV environment, DS methods are still believed to be more reliable and more accurate, however for a multimedia environment (where the visual quality is comparably low), no disadvantage for SS methods compared to DS methods were discovered ([93, 94]).

For DS methods, the reference video could be the original video or a processed version. If the question is nothing more than "is codec A better than codec B?" the video processed by one of the two codecs will be treated to be the reference. Of course this is only appropriate if not more than two codecs or coding settings are compared. If the error robustness of a video should be evaluated, the reference video will be a processed video. Otherwise, it is difficult to separate the quality degradation introduced by transmission errors, from the quality degradation introduced by coding. Going to the other extreme of the scale, if small quality differences for high quality videos should be rated, the reference video will be the original video as this allows judging the level of impairment introduced by comparably few artifacts.

For DS methods, a reference video is presented to the viewers in addition to the video that should be evaluated. therefore, those methods do need more time to be performed. But this extra time provides two major advantages compared to SS methods:

- DS test methods do not suffer from context dependent votes (also known as contextual effect).
- Results from different DS tests using the same reference, but different processed videos can be compared directly[3].

The contextual effect is *"a variation in the evaluation of video"* (quoted from [85]). It occurs, because no human observer can possibly assign pure absolute quality ratings independently from what was presented before (the context). For DS methods, the reference video is shown, and therefore, no absolute quality rating is required or expected, but also the context is explicitly known. But for SS methods an absolute quality rating is required. Thus this especially affects SS test methods. The effect can be easily demonstrated with just three different videos, (those may even show different content) having low, medium and high visual quality. The rating for the video with medium visual quality will depend on the presentation order. Lower votes will be given for the video with medium visual quality if the presentation order is high - medium - low, as compared to the presentation order being reversed (low

[3]This only holds under the assumption of comparable test conditions, including test subjects and room setup.

- medium - high). If the sequence that is shown just before the medium quality sequence is of high quality, the medium visual quality will be judged comparably low. In contrast, the visual quality will be judged higher if the previous video, which acts as some kind of (unintended) anchor, is of low visual quality. Even if the test subjects know, that an absolute quality rating is required, the medium quality video will always be at least partly judged compared to what was shown before. To minimize the influence of this contextual effect, special care has to be taken when designing a SS test. This includes:

- Having a pseudo random order of the videos under test, ensuring that not two different processed versions of the same source video are shown twice in a row.
- Having a pseudo random order to avoid showing two different videos processed by the same processing steps (coding, transmission) in a row.
- Having every video under test be presented twice in the test and receiving two votes. The final vote for this video can be calculated as the mean of those two votes. This allows to partly compensate the contextual effect. Of course, the (quality) context in which this video is presented should be different the second time. It is obvious, that showing every video more than once increases the time needed for a SS test.
- Having a different pseudo random order of the videos under test, for different groups of people, or even for every test subject.

This contextual effect is the main reason for the difficulties that appear when results from two different SS tests are compared against each other. Only by including several identical videos in both tests, and careful analysis of the results for this overlapping test set would allow comparing the results from two tests, or combining the results.

While the rating procedure used in a DS test is more natural for human subjects, as most quality ratings performed in daily life compare the quality of one item to the quality of another similar item, the presentation procedure of a SS test is more similar to the common viewing experience, where a video is shown once at one distinct visual quality without the choice to see the same video at different qualities. As mentioned, most test subjects feel more comfortable with DS test methods as they can rate the processed video in comparison to a reference. The different viewing and rating experience of a DS test compared to a SS test also influences the result of the test. This is especially true, if a trade off between sequences with more artifacts

but more details, and sequences with fewer artifacts but also fewer details should be made. In a SS test, most test subjects will favor the videos with fewer artifacts (and less details), as it is not known how many details did exist in the video before processing, and nothing will be missed that is not known to be existent. On the other hand, artifacts may be clearly visible, and those artifacts may be very disturbing. In a DS test the original video will exhibit all the details that are available in the video, and now the same test subjects will actually miss those details in the low artifact, low detail video, resulting in lower quality ratings. In this case, the strong artifact, high detail video may get a benefit for showing all the details.

If the general question "DS or SS" is answered, still some parameters within those two methods can be varied. For SS methods, only the question about the number of repetitions before voting applies, and for most cases the decision will be to show each sequence only once before voting. For DS, some more options beside the number of repetitions are available: the two videos could be shown side by side, or sequentially, the reference and the processed video could always be in the same spatial or temporal order, or randomly change places, and a decision about which video to use as reference video has to be made. Most of the time, a time sequential presentation with random order of reference and processed video, and one repetition will be used. Randomly changing the presentation order avoids fatigue of the test subjects, and ensures high concentration level. A side by side presentation of reference and processed video is only useful if the video is of low spatial resolution. Otherwise ,it is not possible to actually watch the two videos at the same time. Still, it is quite difficult to look at two different versions of the same video simultaneously, therefore, this option is only used, if the video under test is too long to be evaluated in a time sequential order (more than 10 to 12 seconds). For a side by side comparison, most of the time the positions of the processed and the reference video are fixed, and known by the test subjects.

Rating Scales

In general, any scale can be used for rating the quality of a video. The scale may be continuous, or discrete, awith numerical labels, verbal expressions, or both. Rating scales may either aim at rating the remaining visual quality (quality scale), or rating the impairment introduced by the processing steps (impairment scale). Table 3.1 and Fig. 3.3 give an overview about the most commonly used scales.

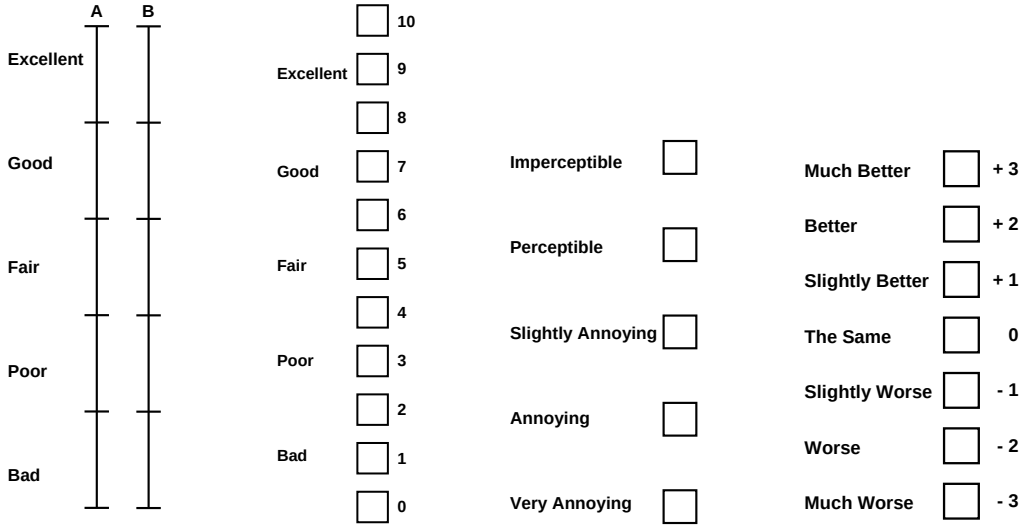
The scale has to be selected in a way, allowing the test subjects to discriminate

Table 3.1: Commonly used quality scales

Scale	Levels	Labeling
(Continous) Quality Scale	100 (continous) 5,11	Excellent Good Fair Poor Bad [a]
Impairment Scale	5	Imperceptible Perceptible, but not annoying Slightly annoying Annoying Very annoying
Comparison Scale	7	Much worse Worse Slightly Worse The Same Slightly Better Better Much Better

[a] Labeling is optional and for instance is not used for continuous quality evaluation

between different quality levels, which would favor a continuous scale, or a discrete scale with many levels. On the other hand the voting procedure should be made as easy as possible, which would favor a discrete scale with only few quality levels. One reason for the introduction of the DSUR (Double Stimulus Unknown Reference) test method in [85] was, that it is probably not possible for the test subjects to make meaningful use of the continuous scale used for DSCQS, which allows for up to 100 different quality levels. For continuous quality evaluation, performed to measure the error robustness of a processed video, a scale similar to the Continuous Quality Scale is usually used. As the video presented in such a test normally does not reach "excellent" quality, common quality scales are not useful, and therefore a continuous scale without verbal labeling is used.



(a) Continuous quality scale (b) Quality Scale (c) Impairment Scale (d) Comparison Scale

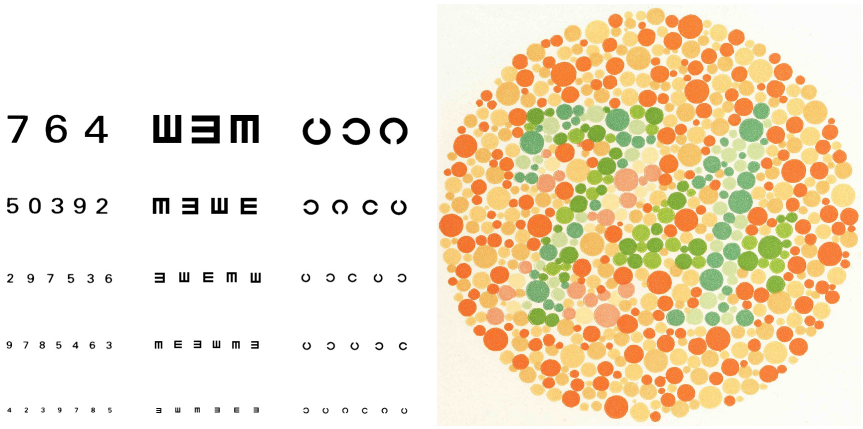
Figure 3.3: Commonly used quality scales

Running a Test

The subjective test can be split up into three phases: screening of the subjects, followed by a training phase, and finally the real test.

All test subjects should be screened for visual acuity, and color blindness. This is normally done using Snellen or Landolt charts for acuity testing, and Ishihara charts for testing color vision (see Fig. 3.4 for examples of these charts). The training phase is used to make the subjects familiar with the testing procedure, the impairments that appear in the videos, and the overall quality of the videos during the test session. More details on the selection, screening, and training of the subjects are given in 3.1.2.

The subjective test should not take longer than 20 to 25 minutes, as it is not really possible for the subjects to maintain a high concentration for a longer time. ITU-R BT.500 suggests test sessions up to half an hour, but experience showed, that more than 20 minutes of testing is perceived to be stressful to the test subjects, especially for SS tests. Assuming video sequences of 10 seconds and a five second voting time, 25 minutes testing results in about 100 votes for a SS test with no repetition. If a single test would last more than 25 minutes, it should be split up into two shorter tests with a short break in between the two tests, even if this lengthens the time



(a) Snellen test chart (b) Ishihara color chart

Figure 3.4: Visual test charts

for the whole testing procedure. At the beginning of each test, a stabilization phase of at least three (and normally not more than five) test cases should be inserted. This stabilization phase consists of test cases from the test set that should represent the minimum and maximum quality that is present in the test, and are meant to allow the test subjects to adopt their own scale of what they perceive as good or bad quality. The votes for this stabilization phase are later discarded, and the subjects should not be made aware of this stabilization sequences.

Votes of the subjects can be collected on paper or using special devices implemented in hardware or software. Collection of votes on paper has two main advantages: no new device has to be handled by the subjects, and if necessary one can easily proof that the votes were actually not tampered. Drawbacks are, that this requires a separate processing step on the votes, a process that is not error free, even if the errors can be detected and corrected. If the test is a continuous quality test, and votes have to be collected continuously, this can only be done automatically. For this case, at least two votes per second should be recorded ([86]). Care has to be taken that recording of the votes is synchronized with the video, to accurately measure the actual quality, as required in a continuous quality evaluation. The time for voting if a single vote should be given after having seen the sequences, depends on whether the test is a DS test or a SS test. For a SS test, where absolute quality votes are required, the voting time should be long enough to allow the subjects to make the vote, but not long enough for the subjects to start thinking about the individual votes and its relationship to previous votes, which is something to be avoided. The decision about the quality of the video should be made while watching the video, and

this decision should only be noted down during the voting time. For this reason, a voting time of not more than five seconds is proposed, a time long enough to switch concentration from the screen to the voting sheet (or voting device), make the vote and again concentrate on the screen for the next video. For DS tests, since a vote relative to a reference is needed, a bit more time should be allowed (five to six seconds). But still, the time should not be long enough to allow the subjects to start thinking about the vote after having seen the videos. If votes for the reference video and the actual video have to be made, a voting time between eight and eleven seconds is proposed.

Room and Equipement

The room used for subjective testing should allow the test subjects to distinguish between subtle quality differences. Therefore, a controlled setup concerning lighting and viewing conditions, as well as listening conditions is needed. This includes low reflecting gray walls, noise protection to protect the test subjects from disturbance by outside noise, and the possibility to have the test room completely darkened.

ITU-R BT.500 precisely defines the background chromaticity, room illumination, and display setup concerning brightness and contrast. Even if the values given in ITU-R BT.500 were selected for TV displays, and testing in a TV environment, those settings also provide a guideline for tests that are planned and conducted for different application areas, that use other displays than professional TV monitors. In general, lighting should be as continuous as possible, have daylight characteristics, and should not interfere with the refresh rate of the display. The lighting should be indirect, and should be comparably low. Depending on the type of display, the used lighting will only be in the back of the observers, or also used as background illumination behind the display. Two exemplary setups for the use of (CRT) monitors and projection on a screen are shown in Fig 3.5.

Today, video is not only displayed through TV monitors only, but also through a wide variety of displays ranging from small LCD displays in mobile devices such as mobile phones, to large LCD screens, computer displays or different types of projectors, using back projection, LCD or DLP technology. For this reason, video testing is no longer restricted to (professional) TV monitors. Computer displays, projectors, and high resolution LCD displays are used as well. For testing in a TV environment, ITU-R BT.500 suggests to use professional grade monitors that do not only have a much higher resolution than consumer devices, but are also

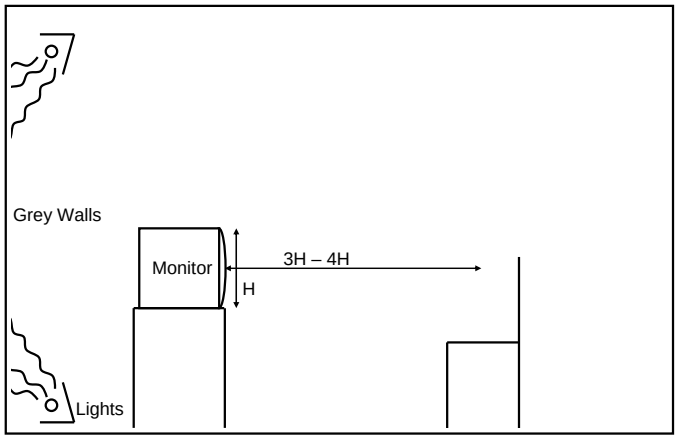
more accurate in color reproduction. Following the same argument, in order to set up critical viewing conditions for the detection of small quality differences, all other types of display that are used for testing should be designed for professional use, and should allow accurate adjustment of relevant display parameters such as brightness, contrast, or color. All displays should be used at their native resolution. The size of the displayed video at the native resolution of the screen, should be large enough to allow good viewing conditions. If the display is not completely covered by the video, the rest of the display should be set to a mid gray level. For projection, DLP projectors are preferred compared to LCD projectors due to their superior reproduction of black. For LCD displays, LCD/DLP projectors as well as CRT computer displays, color calibration, setting of black level and gamma is of special importance (see [95, 96]). In [97], Pinson and Wolf showed, that a professional CRT monitor, as normally used for subjective testing, is also suited to emulate lower resolution LCD consumer monitors, as long as no special impairments appear that are visible in interlaced mode only, and that are not visible on a progressive screen (such as incorrect field order). Despite of recent advances in display technology, LCD displays as well as LCD or DLP projectors still face the problem of having insufficient reproduction quality of motion, which results in obvious motion blur. In [98], Tourancheau *et al.* showed, that the negative effect of this motion blur can not compensate the advantages compared to CRT displays, such as a higher luminance range, or reduced flickering, and that up to today, CRT can still deliver superior quality when it comes to video, especially, if this video contains a lot of motion.

Test Sequences

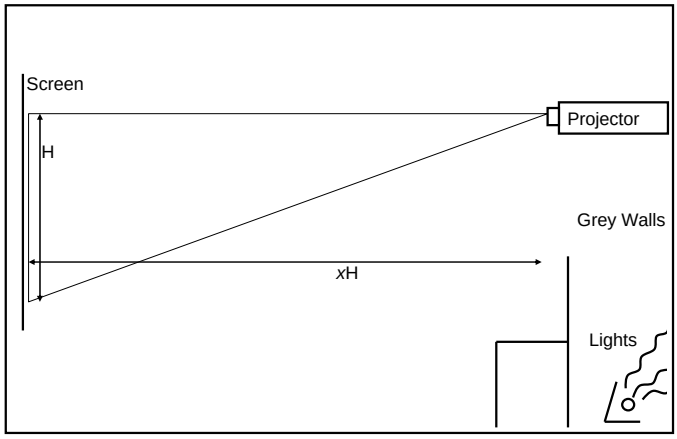
Selecting suitable video test sequences is a difficult task, and frequently sequences that were thought to be useful, turn out to be not appropriate when used in a real test. A good test sequence should fulfill the following criteria:

- It should be discriminating for videos processed using different processing steps or processing parameters. Fig. 3.6 shows quality versus bit rate plots for sequences that are not equally well suited.

- For video coding, an increase of the bit rate of a factor of two should result in a quality difference that roughly correspondents to one quality level on a 5 grade scale.



(a) Monitor setup



(b) Projector setup

Figure 3.5: Test room setup

- The length of the sequence should be 8 seconds to one minute. For tests where one single vote should be given for each sequence after having viewed it, the length of the sequence should be not more than 12 seconds. For continuous tests, the length of one sequence (and therefore the length of one voting period) should be preferably between 30 seconds to one minute.

- The sequence should not contain too many scene changes. For sequences up to 12 seconds, not more than one scene change is recommended.

- The content of the sequence should be easy to understand by the test subjects.

- The sequence should not contain humor or anything that possibly involves the test subjects emotionally. This includes not only sexual content, or violence, but also well known actors, important sport moments, or other scenes that can be expected to be known to the test subjects. Thus instead of selecting a soccer scene from the world cup (possibly even the match winning goal in the final), an unimportant scene from some lower soccer league should be taken.

A single test should contain as many different test sequences as possible, to avoid loss of concentration of the test subjects due to too many repetitions of the same sequences. The quality range for one single test session should not be too big. If very high and very low quality videos should be evaluated, it may be beneficial to split this test into two different test sessions, one for the high quality part, and one for the low quality part including an overlapping part. If the application, which is targeted by the tests covers only a comparably narrow range of sequences, for instance in the case of video conferencing, where the camera is typically fixed, and one can see one or more people in front of a static background, and the whole sequence contains only low motion, care should still be taken to select test sequences that are as different as possible. Before using a new test sequence in a test effort, this sequence should be tested internally to evaluate the usability of this sequence for subjective quality tests. This internal test may be comparably informal, but should at least be performed in a common test environment and with processing conditions that are similar to those that will be present in the real test. The use of standard test sequences such as the famous “Mobile&Calendar” or “Foreman” sequence comes with the advantage, that these sequences are known to work for subjective testing, but it can be expected that video codecs are already fitted to work especially well for those sequences. Therefore, a test should contain at least some more recent or less common test sequences.

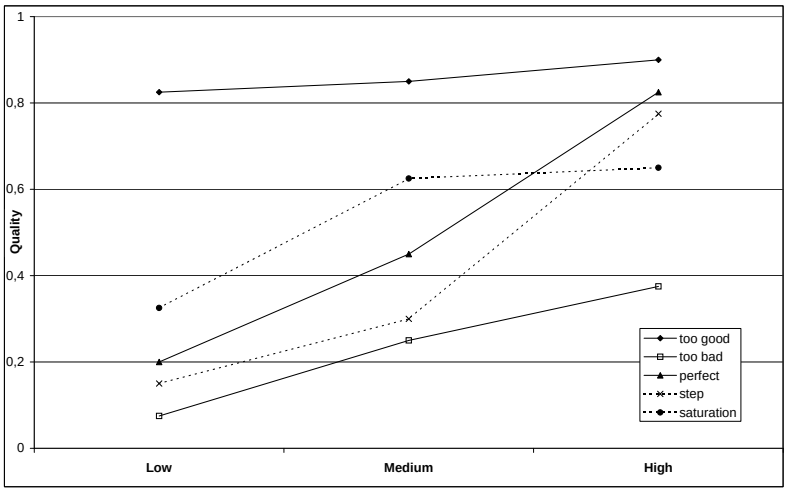


Figure 3.6: Quality versus bit rate for different sequences

Test Subjects

Any test, even if prepared with utmost care, can be compromised if the test subjects are not selected with the same care, and are not prepared to perform the test. For competitive video quality tests, naïve viewers are preferred compared to expert viewers, whereas for tests that are used for the calibration of video quality metrics, the use of expert viewers is proposed in [99][4]. Naïve viewers are not only preferred because they better represent the standard viewer, but they also do not have any technical preferences: for an expert it is comparably easy to detect which codec, or at least which coding technology, was used, and experts are usually not agnostic to these technologies, leading to results that are not representative. In addition, most of the tests that are conducted using expert viewers are tests, where the experts are asked if they could spend some time doing these tests, when most of them would prefer doing other things. In contrast, naïve viewers are usually paid to do the tests, and this difference in motivation has shown to result in more reliable results if naïve viewers are employed. Due to superior visual sensivity, younger people are preferred. This is especially due to a better ability to adapt the focal length when switching between the display and the voting sheet. As already mentioned, all viewers have to be screened for corrected to normal visual acuity and correct color vision. In addition, the test subjects should be able to understand the task of the test, and the comments made by the presenter during the training phase without loss of detail. This means, that they should understand the language in which the tests is held

[4]For the verification of those metrics, it is again suggested to use naïve viewers.

without any problems. To reach statistical significance, at least 15 valid viewers (viewers, from which the votes could be included in the results) should be available. The question when and why the results from one viewer should be discarded is discussed in the next section. For competitive tests, it is proposed to have at least 20 test subjects to increase the stability of the results. In [79], an unreasonable large interval of 4 to 40 subjects is proposed. But with too few test subjects, confidence intervals will be too large, and results will possibly depend too much on the votes of one single person. On the other hand, no real increase in stability of the results has been observed, when using more than 25 test subjects for the same test.

Instructions and Training

Even more important than the selection of the test subjects are the explanations of the presenter to the test subjects, explaining the meaning of the test, and the task the test subjects have to fulfill. Those explanations and instructions should be given together with the training session that is held before the actual test is started. The test subjects should be informed about some background of the tests as long as it does not compromise the votes of the test subjects[5]. This includes background information about the application scenario of the test, such as video over mobile, standard TV, or high quality HDTV. Explanations should include if this is a competition test for new codecs, or a verification test that should evaluate the quality of an existing codec and all other information that can be given away freely. The more the test subjects know about the test, its meaning and importance, the more they will be involved in the test, and aim to produce accurate results. The training phase aims at making the test subjects familiar with the test procedure, the kind of video they will see and the range of quality they can expect in the test. First, a verbal explanation of the test procedure should be made, which includes information about:

- Test method: whether this is a single stimulus test or a double stimulus test. For naïve test subjects these terms should not be used, but instead it should be explained if they should make a relative vote or an absolute vote, and how many votes they should make.
- Timing of the test: how often one sequence is shown before voting, duration of a standard video sequence, time available for voting, overall time for the tests.

[5]If students from one university are involved, they should not be informed about the fact that one of the codecs under test was developed at an institute from the same university.

- Voting scale: the voting scale should be explained in advance, and the subjects should be encouraged to make use of a broad range of the voting scale. If an absolute vote is requested, the test subjects should be encouraged to use the extremes of the scale if they feel that it is appropriate, and do not "spare" some parts of the scale for possibly better or worse videos that might (or might not) appear later in the test.

- Voting procedure: the test subjects should be reminded not to make the vote before the video is finished, and the appearance of the voting prompt. If votes are collected on paper, and the test subjects therefore have different possibilities to mark their vote (line, dot, cross, circle etc.), it should be made clear how a valid vote should look like, and how one vote can be corrected in a way that there are no uncertainties about the correct vote when processing the votes[6]. The test subjects should also be advised to make their votes without too much thinking involved by following their first impression. It has been observed, that most subjects can reproduce their own results very well when following their first impression. This is also in line with responses of test subjects when asked about the test procedure after they have performed the test. Most test subjects state, that their confidence with one vote they made decreases with time when they think about the correct vote. If subjects are not certain about their vote, or can not give a vote because they lost concentration for one sequence, they should preferably leave a blank vote. If the votes are collected on paper, it also would be possible to make one vote, or correct one vote quite some time after the vote should be given. Obviously, this is not allowed, and the subjects should be advised not to correct one vote at a later time instance, but at most make the vote invalid.

The training session should be similar to the final test. That means, that not only the same testing procedure should be used, but also the quality of the presented videos should be similar to the quality that will appear in the test. Care should be taken that the lowest and the highest visual quality that will be presented in the real test should be included in the training session. Of course, the types of artifacts that will appear in videos during training should be similar to those present in the real test. But while the same testing procedure, and videos at the same quality level should be used for the training session, it is highly recommended to use different video sequences than those in the test. Of course, the sequences used for training should come from the same application area, but to avoid training the test subjects

[6]For this case it is proposed to have one line for a valid vote and one cross for an non valid one.

on special distortions in special parts of one special sequence, the used sequences for training should be sufficiently different from those used in the actual test.

3.1.3 Processing of the Results

After having performed the subjective tests, the results have to be processed, outliers removed, and the reliability of the test results assessed.

Detecting, eliminating and correcting outliers

The first step in processing the results is the removal of outliers, and checking if all subjects were able to conduct the test in a meaningful way. Before going into the task of manually detecting and either eliminating or correcting possible erroneous votes, a formal assessment of the test subjects as proposed in Annex 2 of ITU-R BT.500 is suggested. Here, an analysis is done on how often a subject gave a vote that can be classified to be an outlier, and if these outliers are high or low outliers (corresponding to a comparably high or comparably low vote). For this calculation, a vote for one test case v_i is classified to be an outlier if the following equation is fulfilled:

$$v_i \geq a * S * \sqrt{(v_i - \overline{v})^2} \tag{3.1}$$

$\overline{v}$ is the mean value for all observers, S is the standard deviation and the parameter a is set to 2 if the distribution of the votes v is a normal distribution, and set to $\sqrt{20}$ otherwise. The number of of total outliers Out is calculated, and the ratio between the number of outliers and the number of total votes for one observer is calculated resulting in out. A second ratio $out_{\pm}$ is built by $out_{\pm} = \frac{Out_+ - Out_-}{Out_+ + Out_-}$ with Out_+ and Out_- being the number of high and low outliers. This second ratio tells if the outliers are more systematic or more random. ITU-R BT.500 suggests to remove one test subject if out exceeds 0.05 and $out_{\pm}$ is below 0.3. It is also suggested to run this procedure only once (calculation of the mean values and standard deviation is done with the data of all subjects only), and to completely skip this procedure if 20 or more test subjects were involved.

This first formal assessment of the test subjects can help remove test subjects that produce non-systematic errors, but cannot help to detect more systematic errors. Outliers are votes that differ significantly from the mean votes and can be considered to be erroneous. Such errors can appear for three main reasons:

- The test subject did not understand the testing procedure.
- The test subject did not understand what should be rated.
- The test subject made a singular error.

In the first two (quite similar) cases the error should be systematic, and therefore comparably easy to detect. Common errors are:

- The use of only very few levels of the scale. This could be rating every video to have "medium" quality, or the extensive use of the highest possible vote (extensive compared to other test subjects).
- Quality of the processed video rated higher, than the quality of the original video in a standard DSCQS test. Of course this could happen a few times if the quality of the processed video is very high, but if this happens too often or for cases, where other test subjects rated the processed video to have significantly lower quality, this is most possible an error.
- Regular patterns. This can be observed for continuous quality tests, as well as for the "one test case, one single vote" tests. Fig 3.8a and Fig. 3.8b show an obvious pattern for a continuous quality test, where one test subject just slowly moved the slider back and forth between the two extremes, no matter what the actual quality was. This case can be detected comparably easy, as the pattern for different test cases is exactly the same, and even the time between two peaks is identical.
- For continuous quality tests, a pattern that differs significantly from the mean pattern of the other test subjects.
- The inability to detect quality differences. This also can be detected quite easily in a graphical representation comparing the results from one subject to the mean of the other subjects. One example can be found in Fig. 3.9a. Three different quality levels for four test sequences appeared in the test: two higher quality levels and one lower quality level. Two aspects are clearly visible: the single subject did use only a small quality range, and for none of the three sequences the third quality level that was rated significantly lower by the other subjects, was rated to have the lowest quality. Another example is shown in Fig. 3.9b. Here, the subject basically rated every test case to have the highest possible quality, whereas the mean opinion score gained from the other subjects shows, that there are actually quality differences (though those

may be small).

- The inability to assign the same quality value to the same sequence shown at different time instances. This can be controlled very easily, especially for single stimulus tests, where every test case is presented twice.

All of these systematic errors should be detectable by simply looking at graphical representations of the gained data. If a systematic error is detected, the safest option is to remove this person from the list of test subjects. Even if it is obvious, that the respective person solely used the provided voting scale in reversed order, it is better to discard the votes of this person. One person that did not understand the voting scale might also not not have understood some other aspects of the testing procedure. Of course, such systematic errors could appear for parts of the tests only (e.g. for only one special sequence). Again, the safe way would be to discard all votes of this test subject, as this obvious error for one part of the test may be an indication of more errors in the remaining test that are just less visible.

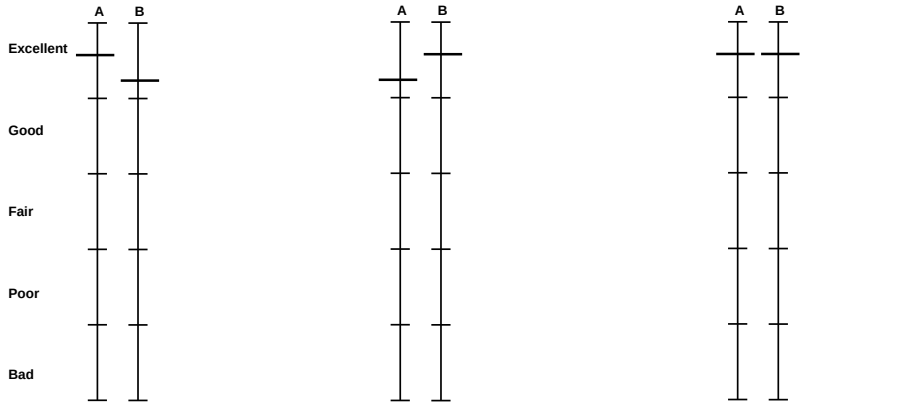
Singular errors are much harder to detect, as the test subject may in fact have intentionally chosen to give this special vote which looks like an error. Most of the time, one vote is regarded to be suspicious if it differs too much from the mean vote that was given by all other test subjects. But before classifying this vote to be an error, the following options have to be evaluated:

- The respective person could in general give votes that are not close to the mean vote, but are closer to one of the extremes (e.g. the person tends to give lower votes). If this effect could explain some part, why this vote was classified to be suspicious, the likelihood increases, that this is not an error, but the vote was given intentionally.

- The respective person could in general give more extreme votes. In this case, looking at the mean vote of this person may not give any valuable information, as it could give in the previous case. Again, if this could explain the suspicious vote, the vote should be kept.

Votes that seem to be erroneous should be removed or corrected only if an error is obvious, and should be kept if it seems to some extent reasonable, that this vote was given intentionally. Otherwise, there is a danger, that the true results are corrected to better match the expected outcome. The most critical option is always to correct one vote instead of deleting it.

For the standard DSCQS test one special “error” could happen, which is an inversion



(a) Original rating (b) Assuming inversion (c) Assuming equal quality

Figure 3.7: DSCQS rating error, and possibilities to correct this error. In this case, B was the uncoded reference video, and A was the processed video.

between the reference and the processed video sequence. This could be due to three different reasons:

- The test subject did give the right votes, but did not remember which one was the reference, and which one was the processed video, even if he or she knew just seconds before. This case could be foreseen, and the test subjects could be asked to indicate this case e.g. by placing a question mark between the two votes.
- The test subject did give the right votes, but reversed the order of the processed and the reference video. If the quality for the processed video is very close to the quality of the reference video, this is hardly detectable.
- The test subject did actually prefer the processed video compared to the reference video (though by definition this case is not possible).

The difficult cases are those, where the quality of the processed video is close to the quality of the reference video. Here the following two options apply in addition to simply deleting the votes: the vote is either reversed (which does not capture the intention of the test subject, who would have rated "equal quality" if this would have been the best vote), or the vote is clipped to "equal quality" (which would not catch the intention of the test subject if the reversion was not made intentionally). None of the three options is really satisfying, therefore the action taken is often selected intuitively.

One additional possibility to assess the ability of one test subject to correctly perform

the subjective test, is to check the consistency of votes on the same test case. To be able to check consistency of votes, at least 10% of the test cases should be repeated, even for DS tests. Of course, these test cases should cover a wide quality range, and different sequences. Again, no strict rule can be given about the allowed difference between two votes for the same sequence, so it is up to the experience of the experimenter to decide if two different votes are within an allowed limit. The decision will most likely be different for each test case, and will depend on the mean quality for this test case, as well as on the variance in the votes of the regarded test subject, and if a systematic difference can be found (e.g. if the second vote was always lower).

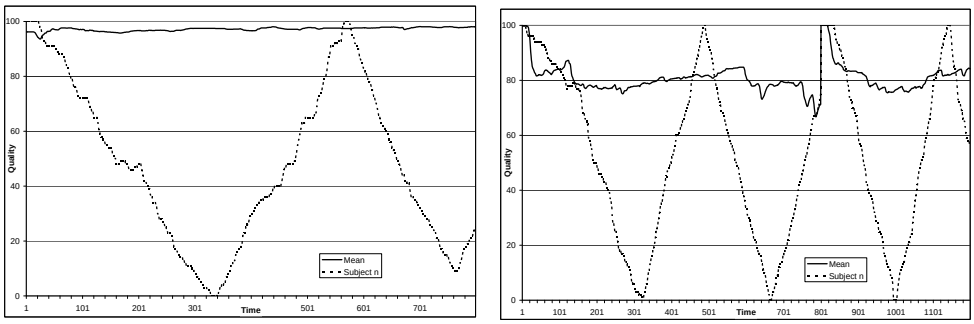
Finally, a decision has to be taken if all votes from a test subject should be removed, depending on the number of detected errors for this test subject. Again, this decision cannot be described using a mathematical formula, but rather it depends on the type of errors, or the mean errors among all participants. The most significant and alerting errors are inconsistencies between votes for the same sequence. The votes of one test subject should be removed if it can be concluded, that the votes are not completely consistent.

The number (or percentage) of errors and outliers that have been detected and removed can again be used to give an indication about the reliability of the gained results, and the quality of the test. While again, no general rule can be given, the number of outliers that come from subjects that are considered for the final results (not including the votes from subjects that were removed completely), should be significantly below 5%. For a SS test, where each test case is repeated, both votes are usually discarded if one vote has been classified as an outlier. Therefore, the percentage of removed votes can be higher, but should be clearly below 10% even for this case.

Analyzing test parameters with ANOVA

After having removed all outliers and errors, the raw data should be examined by an analysis of variance (ANOVA). ANOVA is a statistical tool that allows to check the null-hypothesis if the result of the test does not depend on a certain variable in the test design [100]. The main variables in the test design are

- test subject
- sequence



(a) Test Case 1 (b) Test Case 2

Figure 3.8: Error patterns in continuous quality tests

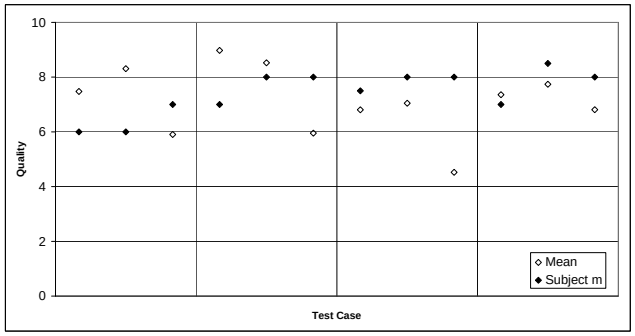
- processing (coding conditions)
- time one test case is shown in the test
- seat position.

Others variables could be gender, age of the subjects, or the time the test is performed (early in the morning or late in the evening). Obviously, the result should not depend on factors such as the seating position, or the time one test case is shown during the test[7]. To be able to test this latter case, a certain number of test cases should appear several times. For the variables "test subject" "sequence", and "processing" the null-hypothesis (the result is independent from one of these variables) should not be met. If those assumptions could be verified, the final processing of the results can be done. This normally includes the calculation of the mean quality of the single test cases, and calculating the 95% confidence intervals.

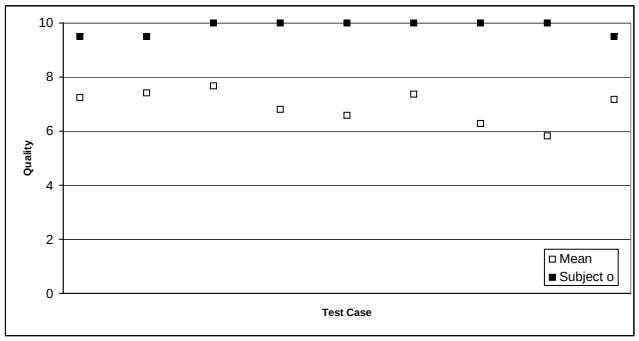
3.1.4 Identified Shortcomings of Published Test Data

Many proposals for visual quality metrics fail at this very first step of producing trustful data that can be used for verification. Most common shortcomings of the subjective tests, which are performed to verify visual quality metrics, are a too low number of test subjects, or not following some standard test procedure. An insufficiently low number of people took part in the tests for [7, 101, 102, 103, 104]. In [105, 106], a new subjective quality method is proposed using only eight different people, and later objective methods are compared to results of this test. Non-

[7]This last aspect is true for DS testing only. Due to the contextual effect the time one test case is shown may have an effect for SS tests.



(a) Low quality is not detected



(b) No quality differences

Figure 3.9: Error patterns in subjective results

standard test procedures are used in [107], where the authors use printouts, showing all distortions for the same image before they proceed to the next image. Printouts are also used in [29], instead of showing the images on a screen. The danger of using different test setups than recommended by the standardization groups is also shown in [108]. The authors use hand held devices to display the videos under test, but the reported number of votes that were removed as outliers is extremely high (more than 12%), and also the reported 95% confidence intervalls are bigger than what can be found for other tests.

A notable number of contributions in this field does not give any detailed information about subjective tests, among those are [109, 110, 15, 111, 28, 112]. Sometimes, no information about subjective tests is given, although the authors state that tests were done, or that their metric correlates well with human quality experience ([27, 36, 70]). Other authors do not test their methods against subjective ratings, but compare themselves against PSNR [60, 23], bit rate [113], or set up assumptions such as "low-pass filtering increases blur", and check if their metrics correlate with these assumptions [114, 115].

3.2 Calibration and Verification Data

If one video quality metric that does not work solely on the approach to build a mathematical model of the HVS needs to combine two or more measurements (features or errors or both) into one quality level, the question of how to weight the different measurements arises. The weighting can either be done by guessing, simple assumptions, or by exploiting the correlation between results from subjective tests and the measured features and errors. If the weights are obtained with the help of data from subjective tests, the resulting metric is inherently trained with the videos used for this test. Careful research would require using different data for training and verification of objective quality metrics. Therefore no videos used for training should be used for verification.

However, many contributions in this field do not take care of this important issue, and do not differentiate between data used for development of visual quality metrics, and data used for verification only. Contributions from the following (incomplete) list have most likely overlaps between calibration or training data on the one hand, and verification data on the other hand. The proposed metrics do rely on the weights gained from the training data, and may be tailored to specific sequences, processing steps, or test conditions. For this reason, using completely different data for training and validation would be of special importance.

- In [15], Gastaldo *et al.* presented a no reference approach using a neural network, evaluating only parameters from the bit stream (such as bit rate, quantization parameter, or numbers of motion vectors). No information is given if the same encoder was used for training the neural network, and producing the reported test results.

- Ong *et al.* present their "Colour Perceptual Video Quality Metric" in [32]. This metric uses a combination of blocking measurement, and "content richness" measurement in conjunction with visibility thresholds. The authors say, that they use a weighted average of the results from each color plane, but do not give any indication about the used color space, or the weights of the results from the different color planes. The metric was tested by the authors using six different video sequences at two resolutions, but no indication is given if any of these sequences was used for gaining any weights.

- Carli *et al.* proposed to use watermarks placed in "perceptually important areas" [22]. In their contribution, the authors selected those "perceptually

important areas" by analyzing color, motion, and contrast of the video. However, results are reported for only one sequence, and no indication is given, if a different sequence was used to develop the algorithm for finding the important areas, where watermarks should be included, and for developing the relationship between visual quality and watermark distortion.

- In [61], the authors proposed to use a combination of bit rate, frame rate, spatial activity, and motion activity. Again, the modeling was done after performing the subjective tests (calibration and verification data are identical). In addition, only one special encoder is used, which results in a metric that is tailored to this special encoder.
- An extension to the SSIM is proposed in [51]. This extension combines the basic SSIM that is normally only performed on the luminance channel, with two more SSIM values that are calculated on the two color difference channels. The weighting coefficients for the three different SSIM values are gained through regression using exactly the same database that is then used to show the superiority of the proposed method, compared to standard SSIM. This approach is very similar to the metric that is built in Section 3.2.1, which shows the danger of overlapping verification and calibration data.

Other works where there is at least a partial overlap between calibration data and validation data can be seen in [11, 30, 23, 109, 101, 104, 8, 116, 117, 41, 118, 119].

The danger of overlapping (or identical) data for development and verification is shown with one very simple example in the following section. In Section 3.2.2, data from the ITU-T recommendation J.144 is analyzed for the effect of possible overlaps between training data and verification data. A very simple and common method to overcome the problem of overlapping data is discussed in Section 3.2.3.

3.2.1 PSNR[3]

In the following example, the effect of using the same data for calibration and verification of one very simple visual quality metric is demonstrated. Following the results from a subjective test, a new simple full reference image quality metric is proposed that outperforms the standard $\mathrm{PSNR_Y}$ in terms of correlation to subjective results. Similar to [51], where the SSIM is extended to include the two color difference channels, it is proposed to evaluate PSNR not only on the luminance channel, but to calculate a combined PSNR using a weighted sum of the PSNR on all three color

Table 3.2: Pearson correlation coefficients for $PSNR^3$ and $PSNR_Y$

	Calibration Data	Verification Data	All Data
$PSNR_Y$	0.644	0.684	0.533
$PSNR^3$	0.953	0.709	0.835
$PSNR^{3*}$	0.864	n.a.	0.864

channels of the YCbCr color space. The corresponding subjective test included four different sequences at bit rates between 96 kbit/s and 1024 kbit/s resulting in 15 data points, involved more that 20 students following the recommendations given in ITU-R BT.500. The weights for each channel are determined by multivariate regression on this test data (calibration data) and the simple new metric that was gained, which is called $PSNR^3$, can be calculated according to the following formula:

$$PSNR^3 = 0.9887 + 0.108 * PSNR_Y - 0.211 * PSNR_{Cb} + 0.108 * PSNR_{Cr}$$

According to the Pearson Correlation coefficients as reported in Table 3.2, the new $PSNR^3$ metric performs significantly better[8] than standard $PSNR_Y$, and the very high correlation coefficient of 0.95 would indicate, that this new metric is closer to a perfect model than any other video quality metric before. (Un)fortunately, the same tests were also done on four more standard test sequences (resulting in 14 data points), encoded using the same coding conditions (verification data). Correlation for these four unknown sequences (sequences that were not used for building the model), is only slightly higher than for standard $PSNR_Y$ which shows, that the $PSNR^3$ metric was tailored to the four videos used for the regression step, but does not really fit other sequences. As the correlation coefficient for the calibration data is very high, the overall correlation for $PSNR^3$ is significantly above the correlation for $PSNR_Y$.

Performing the regression for $PSNR^3$ on the complete data set results in $PSNR^{3*}$, and leaves no data available for verification. The overall correlation for $PSNR^{3*}$ could be increased up to 0.864, which would be significantly above the value for $PSNR_Y$. Without verifying a new metric on previously unknown data, the danger of having a metric (even if it is as generic as the proposed $PSNR^3$) that is fitted to special sequences is quite high.

[8] 95% significance level was reached without problems, even for the low number of only 15 data points.

Table 3.3: Correlation comparison IES and Edge-PSNR

	Sequences	Test Points	IES	Edge-PSNR
All 525	13	64	0.830	0.857
Phase 1 525	3	12	0.945	0.856
Unknown 525	4	16	0.946	0.913

3.2.2 ITU-T J.144

Even for [3], the comparison that lead to ITU-T J.144, the majority of the used sequences was known in advance to the test. Though it was not known explicitly which test sequences were later used for this evaluation, it is worth taking a closer look at the statistics from this test.

A candidate for a very generic method from this recommendation is the Edge-PSNR method as also presented in [6]. This method should perform equally well on all subsets of test sequences. Two out of the four standardized metrics are typical candidates for metrics, tailored to special sequences or impairments. The "Image Evaluation Based on Segmentation" (IES) as presented in Annex C of [31] uses a database of impairment models, and tries to match unknown impairments to those inside this database. Unfortunately the only information given by the authors about how this impairment database was built, is that they used 12 different videos with 525 lines each. Comparing the subset of 525 lines sequences that were already used in the previous effort [2], with the complete set reveals, that the correlation for the known sequences is noticeable above the correlation for the whole set of sequences, whereas this is not the case for the more generic Edge-PSNR metric (see Table 3.3). But looking at the correlation for the unknown sequences only, reveals that the higher correlation may result from different reasons.

The "British Telecommunications Full Reference Metric" (BTFR) presented in Annex A of [31] uses a combination of texture, activity and several PSNR measurements, but no information about how the integration parameters were gained is given. It is also not clear, why these integration parameters differ significantly for the 625 and 525 line sequences. For the set of 625 line sequences, differences in the correlation between the known and the unknown sequences, and the complete set are much higher than for the Egde-PSNR method (Table 3.4).

In many cases, it is easier to gain high correlation for a lower number of test points,

Table 3.4: Correlation comparison BTFR and Edge-PSNR

	Sequences	Test Points	BTFR	Edge-PSNR
All 625	13	64	0.779	0.867
Phase 1 625	3	12	0.861	0.871
Unknown 625	4	16	0.546	0.806
All 525	13	64	0.932	0.857
Phase 1 525	3	12	0.914	0.856
Unknown 525	4	16	0.964	0.913

therefore higher correlation for the known test sequences alone does not give an indication. The data for the IES proposal and the Edge-PSNR suggests, that the higher correlation for the known 525 line sequences may be caused by this effect. However, for the BTFR proposal, the differences for the 625 line data are big enough to support the suggestion that the verification data should be unknown. For the case of a standardization effort, this would mean to use a completely unknown set of test sequences[9]. Seeing this numbers, one has to remember, that for the 625 line sequences, all proposed metrics (including the reference metric PSNR) have an overall correlation between 0.703 and 0.898, and only additional statistical processing beside the Pearson correlation lead to the decision of selecting only four metrics for ITU-T J.144.

3.2.3 Cross Validation

As conducting subjective tests for video is very time consuming and expensive, most video quality metrics proposed are developed and tested with not more than six different sequences. This is obviously too small a number, especially if this six sequences should be further split up into a set used for verification and another set for development. But even if the subjective tests span more than six different sequences, it is desirable to use the whole available test set for the design and calibration of new quality metrics, as it can be expected, that modeling will be more accurate if more sequences and test points are involved. One possibility to overcome the problem of

[9]The VQEG currently suggests 20% of unknown combinations of sequences and encoders/channels in the current version of their HDTV test plan, so it is even not necessary that any of the source sequences is unknown. The test plan of the multimedia group from VQEG does not require any unknown video sequence.

too few available data for development and verification would be to build not only one general metric, but one metric for each single sequence, using all data points from the other sequences for the calibration step ("leave one out" - LOO). This method is a straight forward extension to the common approach of splitting one data set into two separate sets and using one data set to verify the metric gained with the other data set. This is often referred to as cross validation or cross calibration. The only (theoretical) disadvantage of this method is, that one can not present or sell one single quality metric, but just a number of different metrics. However, if the gained metrics would be very similar, this would show, that no single sequence has a strong influence on one metric, and that a general metric could be built, given that a high enough number of sequences for calibrating the metric would be available. On the contrary, if one of the gained metrics differs significantly from the majority of the other metrics, this shows, that the sequence that was not used to calibrate this special metric has some special properties that are not present in any of the other calibration sequences. This information can then be used for a refinement of the design of the visual quality metric. The performance of this set of metrics can be evaluated by predicting each sequence with the metric that was calibrated without using this special sequence. If the number of available sequences is high enough, it is also possible to omit one or more sequences for the calibration step of all metrics. These verification sequences can then be used to show the variance in the prediction of the different metrics.

This method that allows using the whole available data set for calibration and verification is used for the example metrics that are presented in Chapter 6. So far, this method is used very rarely in conjunction with visual quality metrics, the only contribution that could be identified is [33].

3.3 Data Fitting

Data fitting is the task of finding the function $y' = f(y)$ that minimizes the error between two vectors $\mathbf{v}$ and $\mathbf{y}'$. For visual quality metrics, $\mathbf{y}$ would be the vector of the predicted objective visual quality, whereas $\mathbf{v}$ would be the vector of the visual quality measured in subjective tests. Fitting the predicted values to the ratings measured in subjective test, decreasing the mean error or the outlier ratio, is common practice in the field of visual quality metrics. This is in spite of the fact, that such a fitting step (that happens after having performed a subjective test) is not possible in a real

world scenario. While first order data fitting at least does not have an influence on the correlation coefficients, higher order fitting (such as sigmoid or logistic fitting) can change correlation values in a significant way.

As long as the fitting function $y' = f(y)$ does not vary for varying content or processing steps, data fitting is not a problem, and the fitting function can become a part of the quality metric. But if this fitting function is not independent from the used sequences, data fitting is nothing more than a method to make results look better than they actually are. In the following sections, it will be shown, that fitting functions often do vary for varying content, and that higher order data fitting does not provide any benefits for a careful evaluation of visual quality metrics, but rather increases the uncertainty about the real performance of such quality metrics.

3.3.1 First Oder Data Fitting

In the case of first oder data fitting ($y' = a + by$), no differences will appear for the correlation values between $\mathbf{y}$ and $\mathbf{v}$ and between $\mathbf{y}'$ and $\mathbf{v}$. But data fitting decreases the mean error between the two vectors, and also decreases the outlier ratio, which is the number of predicted values y_i that differ significantly from their corresponding subjective value v_i.

The problem of data fitting after having done the subjective tests is again depicted by the example of the PSNR^3 metric from Section 3.2.1. For the calibration data, the first oder fitting line should have no offset and a slope of 1 as this was the goal of the bilinear regression applied to find the weights for the three PSNR values. This is perfectly met for the above example. On the contrary, the fitting line for the set of validation sequences has a offset of about 0.2 and the slope is not as steep as desired (0.7). As a result, the error between the predicted quality and the actual quality increases significantly if no final fitting step is allowed, and the visual quality for the unknown sequences would be underestimated as long as the visual quality is below 0.75. One has to remember, that in a real world scenario, one would have to use the fitting parameters of the calibration dataset, as these are the only known values. Fig. 3.10 shows the prediction result for the PSNR^3 metric.

Quite often, PSNR is fitted to visual quality in the range of 0 to 1 by calculating $PSNR_{fit} = (PSNR - 15)/30$. To evaluate how much the fitting parameters for PSNR are content or sequence dependent, data from subjective tests containing 13 different sequences was analyzed. Building three different sets of four sequences

Table 3.5: Calibration and validation sets for PSNR[3]

Test Set	Slope	Offset	Mean Absolute Error data fitting	Mean Absolute Error no data fitting
Calibration	1.00	0.00	0.05	0.05
Validation	0.70	0.22	0.10	0.13
All Data	0.85	0.11	0.08	0.09

Table 3.6: Fitting parameters for PSNR

Test Set	Slope	Offset[a]
1	13.76	24.16
2	24.69	16.01
3	19.94	17.61
All Data	17.49	20.73

[a] Visual quality is computed as (PSNR-Offset)/Slope

each, and calculating the fitting parameters for the PSNR values to the visual quality ratings shows, that these fitting parameters vary significantly. The sequence "Foreman" was not included in any of the three sets. It is used to test the variation in prediction accuracy, which is caused by using different data to calculate the fitting parameters. The mean deviation in the prediction results for different processed versions of this is 0.059, and the predicted quality can vary as much as 0.22 (on a 0 to 1 scale).

To see how much the fitting values that were gained using the whole data set depend on the influence of single sequences, the fitting values for the whole data set were again calculated according to the method of "leave one out". As the data from Table 3.8 shows, fitting values are much more stable if at least twelve instead of four sequences are regarded. It can be expected, that the fitting parameters gained from the complete set are very similar to those one would gain in a different test using different sequences, as long as the same coding technology is used, and the video is encoded at a similar spatial resolution[10].

[10]For this case this was video encoded at CIF resolution using the AVC/H.264 video coding standard.

Table 3.7: Prediction differences for sequence 'Foreman'

		Prediction using fitting data set			
Visual Quality	PSNR	1	2	3	All
0.260	28.93	0.347	0.523	0.568	0.469
0.533	32.14	0.580	0.653	0.729	0.652
0.561	32.28	0.590	0.659	0.736	0.660
0.679	34.94	0.783	0.767	0.869	0.812
0.788	36.40	0.890	0.826	0.942	0.896
0.913	35.08	0.794	0.772	0.876	0.820
0.913	38.04	1.009	0.892	1.025	0.990

Table 3.8: Fitting values for PSNR

	Slope			Offset		
	Min	Mean	Max	Min	Mean	Max
Leave one out	14.44	17.46	18.91	19.54	20.75	23.19
All data		17.49			20.73	

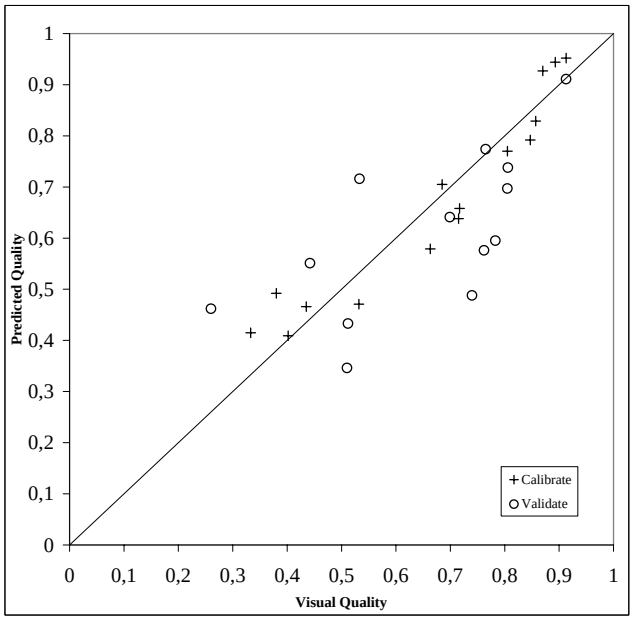


Figure 3.10: Predicted versus measured visual quality for PSNR[3]

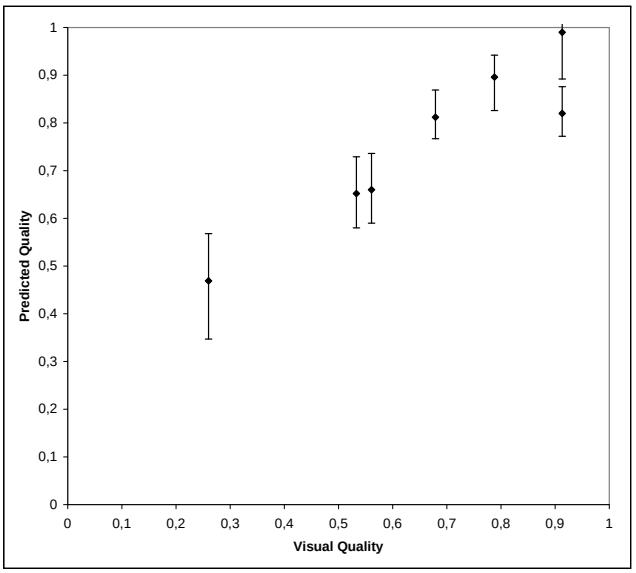


Figure 3.11: Error bars show the range of predicted values depending on the fitting function for the example shown in Table 3.7.

3.3.2 Higher Order Data Fitting

Sigmoid (or logistic) fitting was proposed in [2] and [3]. Since then this method was widely used in the field of visual quality metrics. The reasons behind proposing sigmoid fitting instead of first order fitting were, that subjective tests themselves do not produce results that are linear over the whole range. A typical property of the results from subjective tests is, that compression appears at the very ends of the quality range (very good and very bad quality), and the extremes of the quality scale are never reached. Following this argument, the sigmoid fitting function should be more or less constant for one subjective test, meaning that different metrics should have the same sigmoid fitting function, and differences between the fitting functions of different subjective tests should be very small[11]. In addition, the shape of the sigmoid fitting function should have the following characteristics: saturation toward the ends of the quality range with a large middle section that is close to being linear (perfectly with a slope of 1.0 and 0 offset). Such a sigmoid function is shown in Fig. 3.12. The general sigmoid function is given as

$$y' = a/\Big(1 + e^{(-(y-b)/c)}\Big). \tag{3.2}$$

For the proposed function the variables are set to $a = 1.0$, $b = 0.5$ and $c = 0.2$, which results in a sigmoid function that is very close to be linear over a wide range,

[11] At least, if the videos tested in those different tests do lie in the same quality range

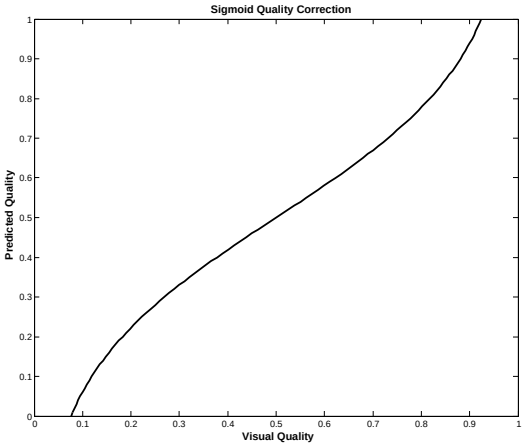


Figure 3.12: Sigmoid correction function

and does vary significantly from the function $y' = y$ only at the very extremes.

However, the data from ITU-T J.144 does not really support the idea of using sigmoid fitting functions. For the BTFR proposal, included as Annex A of [31], the two fitting curves for the 525 line data and the 625 line data vary, and are completely off from the required shape. For the Edge-PSNR metric from Yonsei University, included as Annex B, the two functions are very similar, but suggest that the lower end of the quality range is actually reached. For the IES proposal from CPqD (Telecommunications Research and Development Center, Annex C) both curves again vary, and do not have the required shape. The two curves from the NTIA proposal (Annex D) are quite similar, and are at least close to the required shape. Summarizing the results, one could say, that half of the fitting curves are completely off, two more violate at least one assumption that was raised as an argument for the introduction of logistic fitting, and only two out of eight curves do have a shape that is at least similar to what is required. Fig. 3.13 shows the sigmoid fitting curves for the data from ITU-T J.144. One also has to keep in mind, that some metrics already included a nonlinear correction to take into account the differences in subjective ratings for different quality ranges (e.g. the Edge-PSNR metric, Annex B, has four different linear mapping functions for low, medium, high, and very high quality).

If the subjective tests for the 525 line data and the 625 line data was done properly, and the assumption holds, that the perceived quality for the original video is not different for those two different video standards[12], then there is no real reason,

[12]In contrast, the perceived quality for a video at CIF resolution should differ from the perceived

Table 3.9: Data from ITU-T J.144 (525 line data and 625 line data)

		BTFR	Yonsei	CPqD	NTIA
no fitting	RMSE	0.202	0.135	0.109	0.157
	Outlier Ratio	0.977	0.789	0.586	0.852
	Pearson Correlation	0.865	0.844	0.850	0.899
first order fitting	RMSE	0.112	0.122	0.119	0.094
	Outlier Ratio	0.609	0.688	0.609	0.532
	Pearson Correlation	0.865	0.844	0.850	0.899
sigmoid fitting	RMSE	0.095	0.099	0.100	0.097
	Outlier Ratio	0.587	0.563	0.539	0.469
	Pearson Correlation	0.870	0.856	0.853	0.911

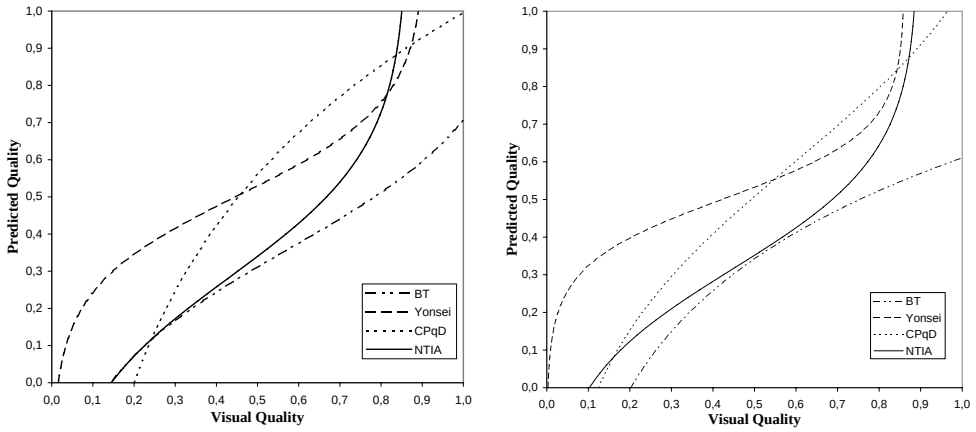
why there should be two different fitting curves. In addition, three out of the four metrics included in ITU-T J.144 use the same metric for the 525 line data and the 625 line data. This would suggest using only one fitting function. For the following examples, the data from these two tests were combined, and fitting was performed on the combined data. Table 3.9 shows, how much the root mean squared error (RMSE) and the outlier ratio changes if the final data is fitted either with first order fitting, or sigmoid fitting. Note, that for this calculation, a data point is categorized as an outlier if the difference between the predicted value and the actual value exceeds 0.05, which is different to the way outliers were calculated in [31], where the individual 95% confidence intervals for each sequence were taken into account.

The above example may exaggerate the differences between fitted and non-fitted data, as all the proponents knew, that they do not have to take care of a 1:1 relationship between the results of the metric and subjective results[13]. But this clearly shows, that the final fitting step is of absolute importance. Fitting does make the difference between a metric that is of very limited value (an outlier ratio of 0.977 is equal to 3 correct values out of 128), and a metric where more than 40% of the data points are correctly predicted, as it is the case for the BTFR metric.

Checking sigmoid fitting functions that are reported for different visual quality metrics, reveals, that quite some of them also do not show the shape that would be expected from the starting assumptions. Among those is the well known SSIM ([45]),

quality of a video having 525 or 625 lines.

[13] Agreement on those fitting methods was reached before the tests.



(a) 525 line data (b) 625 line data

Figure 3.13: Sigmoid fitting functions for metrics included in ITU-T J.144

others are [107] and [11]. In [120], the authors show, that different types of distortion need different fitting functions. Again, only some of the curves do have a shape that is similar to the desired one.

3.4 Statistical Evaluation

As no (theoretical) proof is possible that a proposed visual quality metric actually works, the only possibility to show that the metric produces the right results is comparing the output of the metric with subjective results using statistical tools. Evaluation of visual quality metrics does not require very special statistical tools. For most cases, the Pearson and Spearman correlation values, together with an outlier ratio or a mean squared error (MSE), are sufficient if used properly.

3.4.1 Pearson and Spearman

The statistical tool that is used most often to demonstrate the performance of a visual quality metric, is the Pearson correlation as given in 3.3. The Pearson correlation gives an indication about the prediction accuracy of the metric. A similar task is solved by the Spearman rank order correlation (3.4). This rank order correlation gives an indication of how much the ranking between the sequences under test changes for the metric's values compared to the subjective values (prediction monotonicity). Both statistical metrics should be calculated for the whole verification

data set, and not for each sequence (or each coding settings) separately. If parts of the verification data set are left out of this calculation, this should be documented. Reporting Pearson or Spearman correlation values separately for each sequence (as it is done in [16] or in [61]) does not give much information, as slope and offset of these single regression lines may vary significantly, and therefore the overall prediction accuracy may lie much below the reported values for the single sequences. Even for PSNR the correlation values for one sequence and one codec (or one codec family) lie above 0.9 in most of the cases. It should be noted, that it is easier to gain high correlation values if the data points are not evenly distributed over the whole quality range but mainly found in the low and high quality regions.

$$r^p = \frac{\sum_k (q_k - \overline{q}) \left(MOS_k - \overline{MOS}\right)}{\sqrt{\sum_k (q_k - \overline{q})^2} \sqrt{\sum_k \left(MOS_k - \overline{MOS}\right)^2}} \tag{3.3}$$

Here q_k is the predicted value for the video under test and $\overline{q}$ is the mean value of all predictions. MOS_k and $\overline{MOS}$ are the respective subjective values. For the Spearman rank order correlation $r^s \chi_k$ is the rank of q_k and γ_k is the rank of the respective subjective value MOS_k. $\overline{\chi}$ and $\overline{\gamma}$ are the respective midranks.

$$r^s = \frac{\sum_k (\chi_k - \overline{\chi}) (\gamma_k - \overline{\gamma})}{\sqrt{\sum_k (\chi_k - \overline{\chi})^2} \sqrt{\sum_k (\gamma_k - \overline{\gamma})^2}} \tag{3.4}$$

As neither the Pearson correlation nor the Spearman rank order correlation give an indication about the absolute error between the predicted and the actual values, those metrics are supported by the error between the subjective data and the objective values. This may either be the mean squared error (MSE) or the root mean squared error (RMSE). As these numbers are often hard to interpret, an outlier ratio may also give an indication about the accuracy of a metric. Calculation of the outlier ratio may be based on the confidence intervals of the single data points (e.g. consider every data point that does not fall into the 95% confidence interval as an outlier). For simplicity, a fixed deviation may be allowed. On a 0 to 1 scale, it is proposed to allow a maximum deviation of 0.05.

Taking a closer look at the formula for the Pearson correlation reveals, that the if the difference between the single values q_k and the mean value $\overline{q}$ increases, this part of the numerator grows faster than the same part of the denominator (of course the same is true for MOS_k and $\overline{MOS}$). As a consequence, the correlation coefficient increases if a larger quality range is covered. This effect can be shown by a simple

example, again using data from ITU-T J.144. A subset of the 525 line and 625 line data was created using only processed videos that had a visual quality (MOS) in the range of 0.4 to 0.6. The correlation for this 37 data points for the BTFR proposal (Annex A) is 0.680[14]. A second data set is created using the 14 videos that do have the best visual quality (here the MOS ranges from 0.710 to 0.951). As a quality predictor for those videos, a random number generator is used that is forced to produce values in a range between 0.6 and 1.0. As expected, the correlation coefficient for this random quality predictor is very close to 0 (-0.047). Due to the nature of the Pearson correlation coefficient, combining these two data sets results in an overall correlation of 0.819. This is something not expected at the first glance. One rather would expect that the overall correlation decreases instead, because completely uncorrelated is added.

To overcome this problem, one could give correlation values for different quality ranges instead of giving only one correlation value. This would allow detecting, that the only indication the random metric can give is that the quality of the video is equal or above 0.6. Again taking a look at the data from ITU-T J.144[15], and defining only three (slightly overlapping) quality ranges (low, medium and high quality) results in the data given in Table 3.10. As one can easily see, not all proposals are equally suited for all different quality ranges, but some do deliver better results (at least from a correlation point of view) for the low quality part, whereas others have a better performance for high quality video sequences. It is also interesting to see, that the overall correlation is significantly higher than the correlation values for the single quality ranges. Of course, this behavior does not disqualify the Pearson correlation coefficient for testing the prediction accuracy of a visual quality metric, but at least one should keep this in mind, when dealing with correlation values[16].

3.4.2 Statistical Significance

It is obvious, that the more data points used for verification, the more reliable are the numbers used to support a new visual quality metric. A hypothesis test using Fisher's Z transformation could help to determine how many data points are actually needed to see if two visual quality metrics, one with a Pearson correlation coefficient of r_1^p, and a second one with the correlation coefficient r_2^p, are significantly different

[14]The same example could be made with any other metric included in ITU-T J.144.

[15]525 data and 625 data combined in one data table, raw data fitted using only one sigmoid fitting function for each metric.

[16]As the formula for the Spearman rank order correlation is very similar the same problems apply.

Table 3.10: Pearson correlation coefficients for different quality ranges

		Visual Quality			
		Low	Medium	High	Overall
		0.0–0.4	0.3–0.7	0.6–1.0	
Metric	BTFR	0.621	0.679	0.636	0.870
	Yonsei	0.637	0.710	0.515	0.856
	CPqD	0.572	0.655	0.655	0.853
	NTIA	0.581	0.817	0.709	0.911

from a statistical point of view. These two models differ statistically significant at the 95% confidence level, if r_2^p is not included in the interval $[r_{low}^p, r_{high}^p]$ that is calculated by the following formulas (n being the number of data points):

$$0.5 * \ln \frac{1 + r_1^p}{1 - r_1^p} + 1.96 * \sqrt{\frac{1}{n-3}} = 0.5 * \ln \frac{1 + r_{high}^p}{1 - r_{high}^p} \tag{3.5}$$

$$0.5 * \ln \frac{1 + r_1^p}{1 - r_1^p} - 1.96 * \sqrt{\frac{1}{n-3}} = 0.5 * \ln \frac{1 + r_{low}^p}{1 - r_{low}^p} \tag{3.6}$$

So if for one metric a Pearson correlation coefficient of $r_1^p = 0.8$ was calculated on the basis of 20 data points, one could not claim superiority on a 95% confidence level compared to a second metric if the correlation coefficient of this second metric (r_2^p) is above 0.553[17]. Even including 40 data points raises the lower bound not higher than 0.651.

However, looking at the data from ITU-T J.144, and taking into account the 128 data points from both tests (525 line data and 625 line data) reveals, that the lower bound for the metric with the highest correlation coefficient (the NTIA metric included as Annex D, the correlation coefficient after sigmoid data fitting is 0.911) is higher (0.876) than the correlation coefficient of the second best metric (BTFR metric, Annex A, 0.870).

[17] Usually the correlation coefficient for PSNR already is above this level.

Chapter 4

A New Approach for the Design of Video Quality Metrics

This chapter presents a design proposal for new video quality metrics. The proposed method is built on the assumption, that the visual quality of a video can be calculated as a weighted sum of parameters that can be calculated from features measured in a video. Obviously, the problem of such an approach is to select the right parameters and features, so as to assign the correct weights to these parameters. It is proposed to solve the problem by methods of machine learning. Instead of setting some restrictions that are based on models of the HVS, machine learning algorithms are used to learn the influence of parameters of the video on the visual quality. For the learning process, subjective quality values are needed in addition to the extracted parameters. The learning method proposed is multivariate data analysis. Multivariate data analysis is explained in more detail in Chapter 5.

The metric itself consists of four different steps:

1. Calculation of selected features from the video.
2. Pooling of those features to get a set of parameters.
3. Combination of these parameters into one quality value, by building a weighted sum of the parameters.
4. Correction of the gained quality estimation, by estimating the regression line for predicted versus actual quality of the respective video.

The latter two steps are described in detail in Sections 4.4 and 4.5, Fig. 4.1 shows

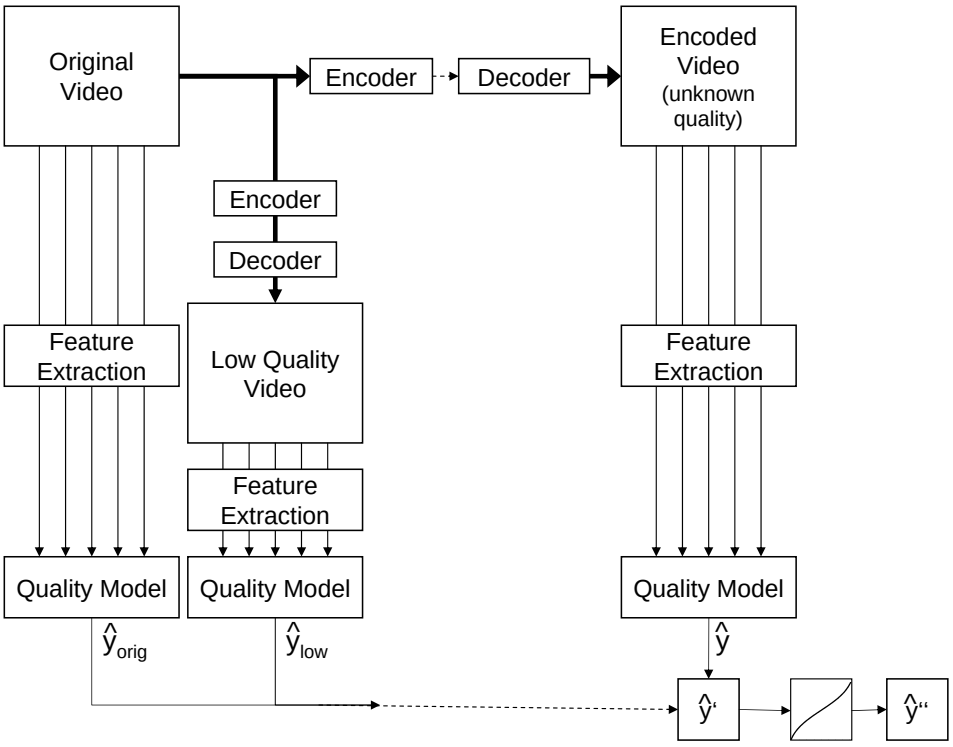


Figure 4.1: Quality prediction process

the complete quality prediction process for a metric that is based on evaluating a set of no reference features.

The process of designing a new video quality metric can be split up in the following steps:

- Select a set of feature measurements. This part is discussed shortly in Section 4.2.
- Create a set of parameters by spatial and temporal pooling (collapsing) of the features.
- Perform subjective tests (see Section 3.1), to gain the subjective results needed for the multivariate calibration.
- Perform the multivariate calibration to gain the weights for the relevant parameters. This multivariate calibration is discussed in detail in Section 5.2.

4.1 Black Box Human Visual System

Many visual quality metrics are built on the basis of a model of the HVS. Metrics that follow a HVS model very closely are [18, 17, 121, 122]. As knowledge about the process of vision is still very limited, and concepts like visual masking are only available for low level features of an image (like the amount of details), but cannot capture the semantic meaning of an image, such models can only be a very rough approximation of the real process. Building a visual quality metric on such a model may not be the optimal way, as the model most likely does not capture all relevant parts of the vision process, exaggerates some parts and underestimates the importance of other parts. Whereas the eye itself can be modeled to a certain extent[1], and even some parts of the higher level vision process can be modeled, (it is known that artifacts are masked in high detail areas), questions like "which attributes or parts of an image do we actually recognize?" or "how much does one object or part of an image influence the attention we pay to other parts or objects?" cannot be answered in a satisfactory way. In "Gorillas in our midst", Simons and Chabris showed, that we may not see objects in a video even if they appear in saliency centers [123]. This effect is called "inattentional blindness". One similar effects is "change blindness", which describes the effect, that we may not notice obvious changes between two images [124]. Although these effects can be shown in experiments with human observers, those effects can only be described. So far, no explanation for this kind of selective viewing (that can be described as "looking but not seeing") can be given, and not even a formal description of the circumstances that are needed for such an effect to appear is possible.

Obviously, any HVS model that includes only low level parts of vision ("looking") but does not take into account the higher level parts ("seeing"), cannot precisely model the HVS. Recognizing that it is not possible to model the HVS, it is proposed that the HVS is treated as a black box. The output of this black box is a visual quality index that can be measured in subjective tests. The input to this black box is a video, and the video itself can be described in many ways, using a wide variety of parameters and features describing low level attributes such as frequency distribution, color, motion, or more high level features such as blur, edges, or blocking. The black box itself can then be described by the function that transforms a set of input parameters, describing one video, into one output value that describes the visual quality of that

[1] Most video codecs make use of the fact that human eyes have three different receptors for light, and that those receptors are not evenly distributed by using a color subsampling scheme.

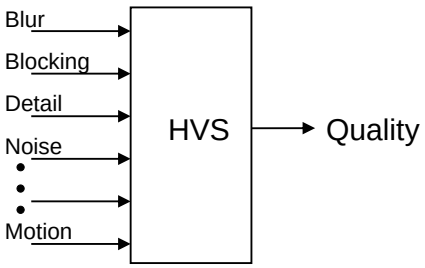


Figure 4.2: Black box HVS

video.

4.2 Feature Measurement and Feature Selection

The input of the black box HVS is the processed video, that can be represented by a time series of 2D matrices of pixels. A video can also be represented by the bit stream of the encoded video, or by image features such as frequency distribution, or color histograms, by more high level features like blur, noise, or blocking. Instead of using three different, equally sized 2D matrices, representing the color planes of each image, one image could also be represented using a series of 2D matrices, each carrying the coefficients of different wavelet subbands. Instead of representing the images in the pixel domain, a common approach for visual quality metrics is to first transform the images into a different domain. The idea behind those transforms is, that the selected representation better fits to the process of vision, and allows better selecting what is actually visible. One example would be the masking of errors in high detail areas, as it is known that distortions are more visible in areas with few details corresponding to errors in frequency subbands that contain low spatial frequencies. Common transforms used are

- Cosine transforms [41]
- Wavelet transforms [125, 110, 29, 8]
- Sobel's gradient images (Annex C of [31])
- Singular value decomposition [107]
- Garbor transform [26, 54].

The number of features that can be used to describe a (processed) video can be arbitrary. For any practical visual quality metric this number should be as small as

possible. The goal is therefore to find the features that have the highest influence on the attribute "visual quality". This selection of features was found to be one main problem when developing visual quality metrics.

4.2.1 Distortions Present in Coded Videos

One straight forward way of selecting features for visual quality metrics, would be to concentrate on the main types of distortions that can be found in encoded videos. Video distortions (or artifacts) can be found either in the spatial domain, or in the time domain. The most prominent ones for the spatial domain would be blocking, blur, noise, and ringing, and for the time domain the main artifacts would be flickering and poor reconstruction of motion.

Blocking

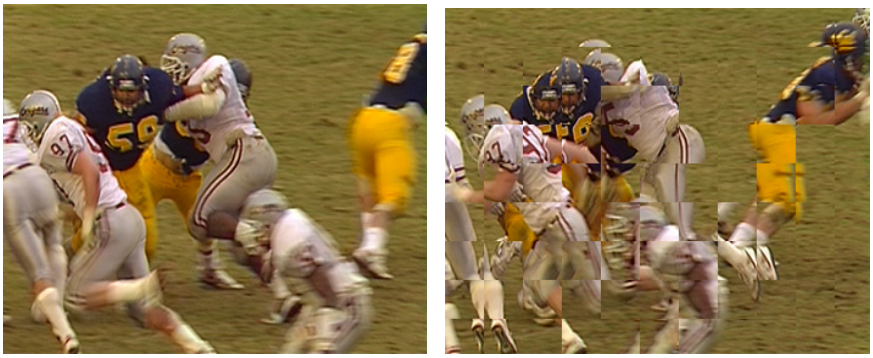
Most video codecs process each image on a block basis. The size of these blocks varies depending on the coding technology. While for most codecs, including MPEG-1 and MPEG-2, only blocks of 16×16 pixels were used, for AVC/H.264 variable blocksizes from 16×16 to 4×4 pixels are allowed, including non-square blocks such as 4×8 pixels. This block-wise processing is used in conjunction with two different aspects of the video codec:

- Motion prediction is done on a block basis, meaning that no motion inside these blocks can be modeled. As motion of these blocks is restricted to translational displacement, more complex motion such as rotation or change of size (zoom) cannot be modeled to a high accuracy. If the bit rate does not allow transmitting a residual signal that compensates this modeling error, these blocks will be easily visible.
- The transform of the video (may it be the Intra-frame or the residual signal) from the pixel domain into the frequency domain, is done for each block separately e.g. by using a DCT. After quantization, discontinuities between block boundaries can become visible. If the bit rate allows keeping only the DC value of each block, the blocking structures become the most disturbing artifact.

The overall blocking effect is shown in Fig. 4.3, the effect of block-wise motion prediction and block based transform and quantization are shown in Fig. 4.4 and Fig. 4.5.

(a) Original frame (b) Severe blocking

Figure 4.3: Severe blocking as a result of insufficient reproduction of details and block wise motion compensation



(a) Frame to be predicted (b) Predicted frame with visible blocks

Figure 4.4: Blocking due to block based motion compensation

Blur

Blur can be described as a loss of detail. The main reason for blur is the quantization of the DCT-values that sets the higher frequency values to zero. To reduce the effect of blocking an inloop filter was introduced in AVC/H.264. This inloop filter is a low pass filter that filters the pixels around the borders of the blocks. The strength of this filter depends on the quantization factor. This filter is one of the main reasons for the reduced blocking in AVC/H.264, but it comes at the cost of additional blur. This blur effect is especially visible in regions, where fine details are wiped out e.g. grass that is turned into one uniform green area.

(a) Original frame (b) DCT coded frame

Figure 4.5: Blocking due to block based DCT coding and quantization. As a result of the high quantization factor, many blocks are represented only by its DC-value.

(a) Original frame (b) Frame with high amount of blur

Figure 4.6: Blur can be detected best in uniform high detail areas such as grass.

Ringing, Mosquito Noise

Ringing artifacts are introduced at sharp edges and jump discontinuities. The inability of the transform function to model this frequency change that causes these types of artifacts is also known as Gibbs phenomenon. Ringing can be easily seen around the border of objects, as the discontinuities that are needed for this effect to appear can be found here. Mosquito noise could be seen as ringing artifacts over time. As the ringing artifacts are not stable, but are moving around the edges (like a mosquito flying around somewhere), these artifacts are especially visible.

Figure 4.7: Obvious ringing around the borders of objects

Flicker

With the exception of the mosquito noise, which is ringing over time, the previous artifacts are artifacts that can be found in the spatial domain, and that do also appear in still image coding. As video has a temporal dimension, additional artifacts appear that only can be seen when watching the video at the intended frame rate (or at least close to this frame rate). The most well known and disturbing temporal-artifact is flicker. Flicker appears if neighboring frames of a video are somehow different. This difference may caused by compression (e.g. high quality I-Frames are followed by low quality P or B-Frames). Flicker could also appear if the single frames are of the same visual quality. For this case, typical reasons for flicker are:

- Frames, which are not spatially aligned.
- Fields of an interlaced video, which are displayed in the wrong field order.
- Frames, which do differ in brightness or color.

Obviously, no still image metric that is used for video without adaptations, can discover such a flickering effect. This problem is shown in [126], where this flicker effect is introduced by a special error control strategy. Overall PSNR would suggest that this error control strategy is beneficial to the video quality, but in fact, the flicker effect that appears in the video, is perceived as very disturbing by human observers.

Motion

The negative effect of poor motion reproduction can be easily shown by producing a video where the single frames are of perfect visual quality, but the frame rate is reduced substantially by frame dropping. The video will appear snatchy and the

overall quality will be comparable low. But problems in the motion reproduction are not only recognized if single frames are dropped. Other examples are wheels that stand still, whereas observers know, that they should rotate (because they can see the car moving), or soccer balls that do not fly a curve as expected, but rather exhibit a some stop/go behavior, or other similar situations. As most motion models allow only translational movement, and cannot directly model rotation or change of size, such effects can be especially noticed for rotating objects.

4.2.2 A Selection of No Reference Features

There are many things that can be measured in a single video in a no reference manner, starting from measuring simple objective image properties such as the spatial resolution of the video, the numbers of different colors, or the distribution of spatial frequencies in the image. As video is not just a series of arbitrary images, motion plays an important role in the perception of video, and therefore this property should be assessed through means such as measuring the length and direction of the motion vectors. Regions or pixels of every image can be classified if they belong to an edge, a flat region or a region with high amount of details. These features correspond to higher level impressions, and a wide variety of different algorithms exist to perform such a classification. Even higher level features such as blur, blocking, noise, or ringing can be measured by analyzing some of the already mentioned features. Methods to measure those last features can already thought of a visual quality metric. Table 4.1 gives an overview about typical features and distortions for visual quality metrics, that can be measured in a no reference way. In addition to the static measurement, a temporal component can be introduced for every single feature, measuring how much one feature changes over time. As all those features (and their temporal derivate's) can be measured in the processed video and the reference video, these no reference feature measurements can be extended to reduced reference and full reference measurements. A reduced reference feature measurement would apply if the result of the feature measurement in the reference video is transmitted to the receiver, where the processed video is evaluated. As a result, one single no reference feature measurement can deliver at least six different values: the pure no reference measurement, working only on one single image of the processed video, a reduced reference value, which can be calculated taking into account the result of the feature measurement in the reference video, and a full reference value can be gained (by e.g. working on the difference image between the reference and the

Table 4.1: No reference features and distortions

Feature	References
Blur	[114, 69]
Blocking	[114, 68, 113, 127, 128]
Details	[31]
Noise	[70]
Ringing	[129]
Fluidity	[117, 76]
Frequency distribution	[13]
Luminance	[45]
Contrast	[45]

processed single images). For all of these three measurements, a temporal dimension can be introduced.

One example where no reference measurements are used in a reduced reference fashion is the SSIM [45]. The SSIM measures luminance and contrast in the processed and the reference image, and compares the results of these two measurements.

4.2.3 Full Reference Features

Taking into account not only the coded video, but assuming that one could also have access to the original video, the possibility of producing a large amount of full reference measurements appears. Possibly, the most simple full reference measurement is the SAD, the Sum of Absolute Differences, between the processed image and the reference image. As discussed in Section 2.1.1, a variety of PSNR measurements can be performed that operate not only on the normal image planes, but PSNR could also be measured on preprocessed images such as edge images, the different decompositions of a wavelet transformation of the images, on the luminance part only, or on all possible color planes.

A very popular approach for full reference metrics is to perform the comparison on pre-processed versions of the images that contain only the edges. Similar thinking leads to a comparison of structure. A proposal for a full reference comparison of structure is given in [45], where this is part of the SSIM. Here structural comparison

between two images x and y is defined as

$$s(x,y) = \frac{\sigma_{xy} + C}{\sigma_x \sigma_y + C} \tag{4.1}$$

with

$$\sigma_x = \left(\frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_x)^2 \right)^{\frac{1}{2}} \tag{4.2}$$

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_x)(y_i - \mu_y) \tag{4.3}$$

and μ being the mean intensity values of the images x and y respectively. C is just a very small constant to avoid instability if $\sigma_x \sigma_y$ is close to zero.

Of course, all features that can be measured in a no reference manner can also be measured by comparing the reference image or video and its processed version. For most cases, this comparison will be more accurate, and the comparison process will be simpler compared to a no reference evaluation of this feature. One example would be blur, that can be evaluated very easily in a full reference environment as just the loss in detail (which corresponds to the shift to lower spatial frequencies) needs to be calculated.

4.2.4 Feature: Motion

What distinguishes video from still images, or an arbitrary series of still images, is the existence of motion between consecutive images. Therefore, it is quite surprising, that most objective video quality metrics do not take into account the temporal properties of video, but just perform measurements on single frames and calculate the overall quality as an average or at best as a weighted average, over the single image quality values. With only few exceptions [102], motion was only considered as a masking parameter, taking into account that fast motion masks some distortions, or the ability to detect fine details, and only recently were motion effects studied more deeply [54]. What has been studied to a certain extent is the effect of video at reduced frame rate, or video that contains dropped frames [75, 74]. But these studies did not involve any coding or other distortions beside the dropped frames. As dropped frames or jerkiness usually are caused by problems of the transmission network, and those non-intentional artifacts[2] are not a result of the normal coding,

[2] In contrast artifacts such as blocking, loss of detail or blurring are result of a rate/distortion optimization process and are somehow intentional.

these severe distortions are normally not regarded when evaluating the quality of a video codec for many transmission systems. Whereas jerkiness and lost frames are quite rare artifacts for transmission of video via classical broadcast networks, such as cable or satellite networks, such artifacts do appear in networks with substantially lower bandwidth, such as IP networks or mobile networks. Therefore, these artifacts are mostly taken into account for video at low bit rates, transmitted over such networks [62, 117, 76].

4.2.5 Features for the Proposed Metric

To show the feasibility of the proposed approach, a set of visual quality metrics are developed. For these metrics, a set of simple no reference feature measurements was selected, representing the most common kind of distortions namely blocking, blurriness, and noise. One feature measurement was added to measure the amount of detail present in the encoded video. To take into account the temporal dimension of video, four different continuity measurements were performed. All feature measurements are done for each frame of the video separately, and the mean value of all frames is then used for further processing, not taking into account the variation of one feature over time, and not including any other pooling methods than simple averaging. In the following paragraphs, a short description of the used feature measurements is given.

Blur

The blur measurement used is described in [69]. 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 by a simple piecewise linear correction if the video contains high amount of fast motion. The range of the blur value lies between 0 (which would indicate that only sharp edges appear in one image), and a theoretical upper bound that is only limited by the size of the image. For uncompressed images the blur value was found to be approximately 1.

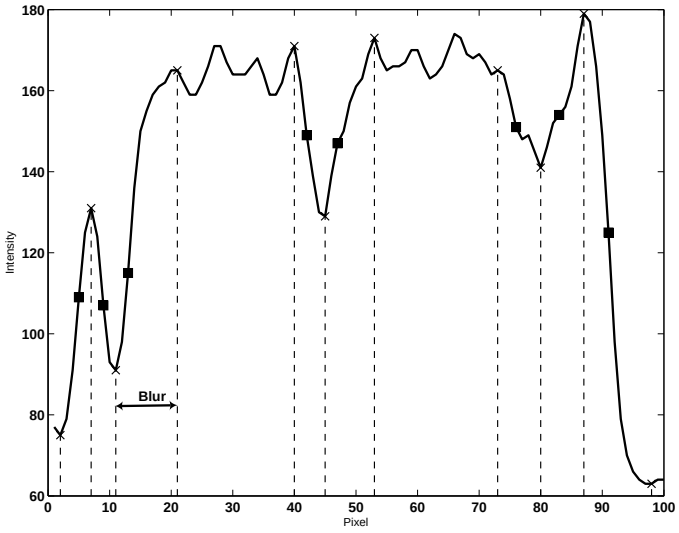


Figure 4.8: Blur measurement in one image. The small boxes show the edges found in the image, the dashed lines give the positions of the collocated minima and maxima that determine the width of the edges, and therefore the amount of blur.

Blocking

For measuring blockiness, the algorithm introduced in [68] is used. This algorithm calculates the horizontal and vertical blockiness by applying a Fourier transform along each line or column. The unwanted blockiness can be easily detected by the location in the spectra, using the measured spectrum compared to a smoothed version of the spectrum. Blockiness should appear as peaks at distinct frequencies (corresponding to the size of the blocks), whereas the frequency spectrum of an image without blocks should be more or less smooth, and should continuously decrease with higher frequencies. For this measurement, as well as for the blur measurement, it is sufficient to take into account the luminance channel only. Again, the theoretical lower bound is 0, indicating no blocking, a value that was actually reached for some of the original sequences. Values above 20 were found for low quality MPEG-2 encoded videos, whereas for AVC/H.264 encoded videos no values above 2.5 were found. Fig. 4.9 shows the algorithm used. The spectrum of a (very) blocky image is given in 4.10, plotting the power of the spectrum versus the relative frequency l/N where N is the maximal frequency that could be achieved.

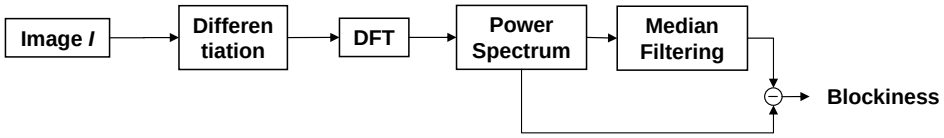


Figure 4.9: Blockiness detection algorithm

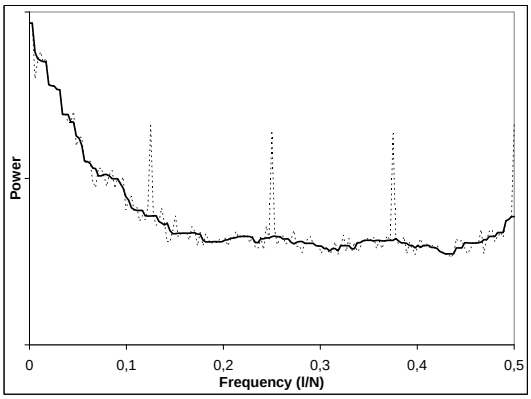


Figure 4.10: Detected power spectrum of a blocky image (dashed line) and the smoothed power spectrum that estimates the original spectrum (solid line)

Noise

To detect the noise present in the video, a very simple noise detector was designed. Noise in video can be easily noticed, if the noise is not only random in space, but also random in time ("mosquito noise"). Therefore, a prediction of the actual image I is built by motion compensation, using a simple block matching algorithm. Second, a difference image D between the actual image and its prediction P is calculated, and a low pass version of this difference image is produced by first applying a median filter and a Gaussian low pass filter afterward. A pixel is classified to contain noise if the difference value between the original difference image and the low pass difference image D_{lp} exceeds a threshold of 25 (assuming 8 bit values ranging from 0 to 255) for one of the three color planes. A block diagram of the noise detector is given in Fig. 4.11. Output of this process is the percentage of pixels that are classified to contain noise. Whereas significant noise was found in MPEG-2 encoded videos, noise does not seem to be a problem for AVC/H.264, which is most probably a result of the inloop filtering.

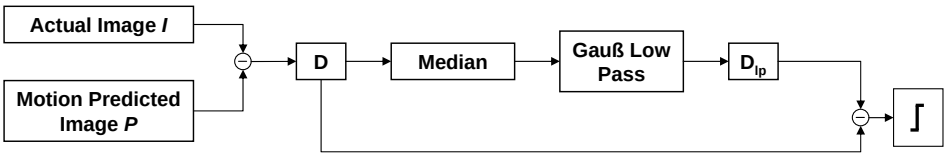


Figure 4.11: Noise detection algorithm

Activity

The activity is measured by evaluating the amount of details. To measure the amount of details that are present in a video, the percentage of turning points along each line and each row are calculated. The two measurements for horizontal and vertical activity are simply averaged to obtain one single value. This measurement is part of the BTFR metric included in [31]. For simplicity, this measurement is performed on the luminance channel only. As the amount of details that are noticed by an observer decreases with increasing motion, the activity measurement is adjusted if high motion is detected in the video, using a piecewise linear correction.

Predictability

The following four measurements are included to capture the temporal dimension of video. The underlying assumption behind these measurements is, that natural video contains mostly "smooth" transitions between successive frames. The following measurements therefore try to capture how noticeable the change between two successive images in a video actually is.

For the predictability measurement, a predicted image is built by motion compensation using a simple block matching algorithm. The actual image and its prediction are then compared block by block. A 8×8 block is considered to be noticeable different if the SAD exceeds 384[3]. To avoid, that single pixels dominate the SAD measurement, both images are first filtered using a Gaussian blur filter and a median filtering afterward. The output of this process is the percentage of blocks that are not noticeable different.

[3]This value was determined though experiments and allows a mean difference of 6 for each pixel

Edge Continuity

The actual image and its motion compensated prediction are compared using the Edge-PSNR algorithm as described in [6]. This measurement should reflect how much the main structure of the image changes. The Edge-PSNR metric produces and output value between 0 and 1, with 1 indicating no difference.

Motion Continuity

Most objects usually follow a smooth motion trajectory. Reasons for motion trajectories that are not smooth may be chaotic object movement, or artifacts like jitter. Changing the field order for interlaced video would also result in non smooth motion trajectories. To detect if motion is continuous for two adjectant frames, two motion vector fields are calculated: between the current and the previous frame, and between the following and the current frame. The percentage of motion vectors where the difference between the two corresponding motion vectors exceeds 5 pixels (either in x- or y-direction), determines the motion continuity.

Color Continuity

A color histogram with 51 bins for each RGB channel is calculated for the actual image and its prediction. Color continuity is given as the linear correlation between those two histograms. This would allow for gradient changes in color (as it could appear for illumination changes), but would show if artifacts such as color bleeding appear.

4.3 Pooling Features to Calculate Parameters

The aim of pooling or collapsing process, is to gain meaningful parameters from the measured features. These are those parameters that are best suited to describe the special visual properties of the video. The most simple pooling process is to calculate the overall mean value for each feature, which would result in exactly one parameter per feature. However, this may not be the optimal method for collapsing the features, as strong artifacts and errors may have a larger influence, and building only an average value may underestimate short errors that appear only in a small area, but are clearly visible. In addition, this would ignore the temporal component,

and treat a time series of images as if those were simply aligned to form one large single image. To better represent the distribution of the measured feature values in space and time, the following set of collapsing or pooling functions may be useful:

- Deviation of the feature measurements over space and time.
- Minimum, maximum, mean and median values.
- 10% and 90% percentiles of the measurements.

The number of parameters that can be gained from one feature is not limited, and finding the best collapsing functions may be as difficult as finding the best features. Sheikh *et al.* avoided this difficulties for the feature "frequency distribution", by modeling the histogram of the frequency distribution in one image using a Gaussian distribution with two parameters that steer the width of the distribution in [73]. So instead of using a whole set of different parameters to describe the distribution of the frequencies, just the two parameters that model the distribution are needed.

4.4 Combining Parameters to get Visual Quality

In [130], Farias *et al.* showed, that if a video is distorted by different types of artifacts (blur, blocking and noise in this case), the overall perceived distortion of the video can be constructed as a weighted sum of the perceived distortions as if only one single type of artifact is present in the video. Following these findings, and assuming, that one processed video could be modeled as a video that has been subject to a number of different single processing steps, it should be possible to model the perceived quality of any processed video by a weighted sum of the visual impact of the single processing steps. Further assuming, that the selected features and their parameters do have a relationship to perceived quality the second main problem in building visual quality metrics remains: which parameters should be used, and how to find the weights for each of these parameters. As already mentioned, it is proposed to use a multivariate regression model to gain these weights. The multivariate regression is described in more detail in Chapter 5, a short overview about the complete process of how to combine feature measurements to get a visual quality prediction is given here.

The building process of the regression model (also referred to as calibration of the model) includes four steps:

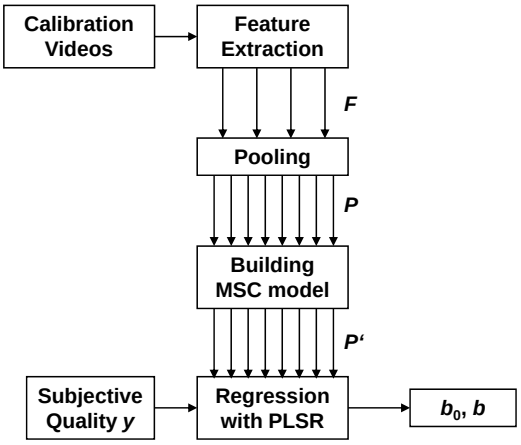


Figure 4.12: Model building process

1. Selection and measurement of the features that possibly contribute to the visual quality of the video.

2. Pooling of these features to generate parameters that describe the statistical distribution of the measured features in space and time.

3. Correction of the calculated parameters to compensate multiplicative and additive effects. The result of this step is a correction model that is needed later as a part of the quality metric. This step is described in detail in Section 5.1.1.

4. Calculation of the regression model (which is given by the weights of the selected parameters), using a multivariate regression method. At this stage, not only the measured parameters, but also the results of subjective tests are needed as input to the process.

A block-diagram of this process is given in Fig. 4.12. The output of this process is a set of weights **b** for the selected parameters, and the predicted visual quality of one video $\hat{y}$ can be calculated as

$$\hat{y} = b_0 + \sum_{i=1}^{n} b_i x_i. \tag{4.4}$$

p_i is the result of the parameter calculation of feature j and the pooling process k. In addition, a Multiplicative Signal Correction (MSC) model is built and applied to the measured parameter data before the regression step.

Contrary to many proposals in the field of objective quality evaluation, the HVS is not modeled, but the model that fits best from a statistical point of view is selected, and the underlying mechanisms are not regarded. The multivariate regression just

selects those parameters that are best suited to explain the variance in the input parameters, and at the same time can explain the variance in the output "visual quality". The main advantage of the use of multivariate regression is twofold:

1. No prior assumptions about the relevance of the features and parameters need to be made.

2. All parameters that have been calculated can be included in the regression step. The multivariate regression will assign weights very close to zero to those parameters that do not have an effect on the visual quality, and those can then be removed before recalibrating the system.

For these reasons, any feature, and any deviated parameter that may be linked to the visual quality of the video or the image can be included in the regression step, and no previous modeling of the HVS is required. Of course, care must still be taken in selecting the features and parameters to avoid including parameters in the model that are statistically linked to "visual quality" but do not have any influence on true visual quality. One simple example for such a misinterpretation would be if different sequences are coded at the same bit rate and the sequences are grouped according to their content in "conversational", "news", "sports" and "movies". Using this content classes as input parameters to the regression step would show that the content classes actually do have a strong influence: sequences from the content classes "conversational" and "news" would have a much higher visual quality as sequences from the other two content classes. Similar effects would appear, if the sequences would be encoded using two different codecs but with one codec always using a higher bit rate. Again the feature "codec" would turn out to have an effect on the visual quality. In the image domain, one could think of using the feature "spatial resolution" as input, and most probably the sequences with the higher spatial resolution would have a better visual quality (at least if all sequences are encoded at reasonable bit rates), but still "spatial resolution" would not be a good predictor for visual quality, nor would spatial upsampling of a low quality video increase the quality of this video (as increasing the bit rate would most probably do). So while this method releases the designer of visual quality metrics from building incomplete and inaccurate models of the HVS, still the selection of the features and the collapsing functions needs to be made with special care.

4.5 Correcting the Quality Prediction

Most objective quality metrics deliver next to perfect prediction results if only one sequence (in the most simple case processed by the same processing steps, only with different settings - e.g. encoded by the same video encoder at different bit rates) is regarded. For this special case, the Pearson correlation between subjective results and the objective prediction is very high (usually above 0.9). This high correlation decreases with every different video sequence that is added to the test set. This behavior is also used in some contributions for visual quality metrics: to increase correlation numbers, correlation is given for each sequence or image separately, instead of reporting the overall correlation in [16] or [61]. The overall correlation can therefore be increased if the parameters of the regression lines for each single image or sequence can be estimated. A method is proposed that tries to estimate slope and offset of these regression lines by introducing two additional (processed) instances of the respective image or video, and making a quality prediction not only for the video or image that should be evaluated, but also for these additional instances.

This very generic method is described in the following section, and a proof of concept is made by extending the PSNR to PSNR$^+$.

4.5.1 High and Low Quality Video

It is proposed to estimate the regression line for one special sequence by producing two additional instances of the original video, and using these two instances to calculate the slope s and offset o of the linear regression line. The visual quality of these two additional instances should be inherently known (e.g. by using encoders that produce a video in a certain - preferably small - quality range). The gained parameters s and o are then used to correct the original quality prediction. An overview of the overall system is given in Fig. 4.13.

The accuracy of the proposed correction step mainly depends on three attributes:

1. Difference between the actual visual quality and the assumed visual quality of the additional instances.

2. Sensitivity of the parameters s and o to the error between the actual visual quality and the assumed visual quality of the additional instances.

3. Ability of the additional instances to represent the regression line for the given

sequence.

For these reasons, the following is proposed for the generation of these additional instances:

- The visual quality of the two additional instances should be very different. These instances can be produced e.g. by encoding the original video using a fixed quantization parameter (QP).
- If the visual quality metric is a no reference metric, one instance can be the uncoded original, otherwise it is proposed to generate a coded version of the video that most likely has no or only very few impairments. The visual quality v_{high} of this instance will be assumed to be in the range of 0.8 to 1.0 on a 0 to 1 scale.
- The low quality instance should be of low visual quality, but should not contain artifacts that are not present in the video of interest, e.g. should not contain dropped frames if the video of interest does not contain dropped frames. The visual quality v_{low} of this instance will be assumed to be in the range of 0.1 to 0.3 on a 0 to 1 scale.
- The encoder used to encode the additional instances should be close to the encoder used to encode the image or video of interest. If the encoder and it's settings are unknown, at least the same coding technology should be used.

The additional instances are rated by the same visual quality metric that is used to gain the prediction $\widehat{y}$. The gained values $\widehat{y}_{high}$ and $\widehat{y}_{low}$ are used to predict the slope s and offset o of the regression line:

$$s = \frac{\widehat{y}_{high} - \widehat{y}_{low}}{v_{high} - v_{low}} \tag{4.5}$$

$$o = \widehat{y}_{low} - v_{low} * s. \tag{4.6}$$

In the following section the method is shown by extending PSNR, but the same method can be used in conjunction with any other visual quality metric, regardless if this is a full reference, reduced reference, or no reference metric, and regardless if this metric tries to model the HVS, or uses a combination of artifacts such as blocking, blurring, and noise. The only restriction that can be seen is, that a no reference metric will not be a no reference metric any more, as the correction parameters s and o have to be transmitted to the receiver, and access to the original video is

needed to generate and evaluate the additional instances. As the additional data load that has to be transmitted for such a reduced reference metric is only two values per sequence, this extra data can be easily embedded in the bit stream, or in the image itself by the use of watermarks.

In addition to an increased accuracy of predicted quality values, the proposed method has the advantage, that it delivers a 1:1 relationship of predicted quality to visual quality. So no more correction for slope and offset has to be made, and instead of asking which visual quality is equal to a PSNR of 32dB (which is not independent of the regarded sequence), a number in the required range is given[4].

The proposed method is very generic, and follows a very simple principle, but the complexity of the overall quality estimation system increases substantially. This does not apply to the quality calculation at the receiver; here the estimated result only has to be corrected according to slope and offset. The increased complexity appears at the sender, where not only two additional runs of the quality metric itself are required, but also the two additional videos[5] have to be generated, which may be even more complex than calculating the quality for these two videos. To limit the additional computational complexity at the sender, the following actions can be taken:

- The encoder used to produce the additional videos can be very simple. As shown in the next section, a very simple fixed QP encoder without rate control, and even without a rate distortion optimization can deliver very good results. As the additional videos do not have to be transmitted and therefore do not come with bandwidth constraints, no optimization regarding the bit rate has to be made.

- As it will be shown in Section 4.5.3, it is not always necessary to encode the complete original video to get the additional videos. It may be sufficient to encode only a part of this original video, and use this (shorter and smaller) videos for the estimation of slope and offset.

[4]The required range can be selected by selecting different values for v_{low} and v_{high}.

[5]If the base metric is a no reference metric only one additional video is sufficient as the original video itself can be used.

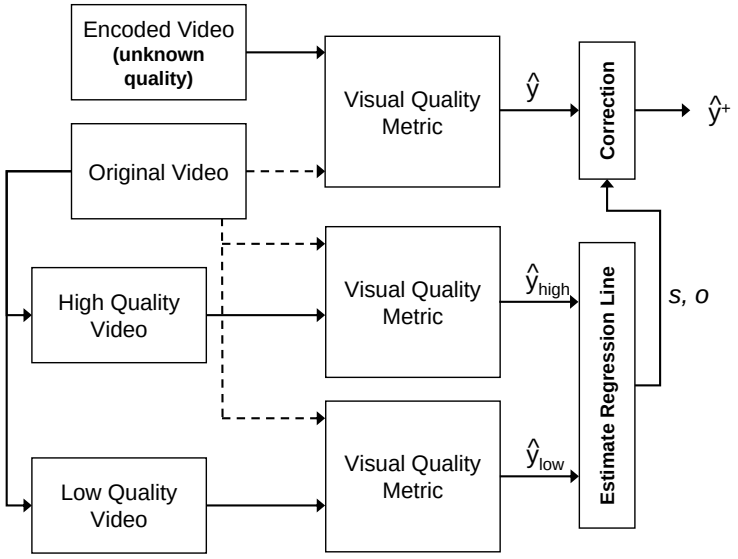


Figure 4.13: Block diagram for the quality correction step

4.5.2 PSNR$^+$

As a proof of concept, PSNR is extended to PSNR$^+$ using the above described method. A data set of 13 different videos coded using AVC/H.264 [131] at CIF resolution, and at bit rates ranging from 96 kbit/s to 1024 kbit/s, resulting in 54 data points, was used for verification of the new PSNR$^+$ metric. The subjective results were gained by carefully designed and conducted tests that followed the recommendations given in ITU-R BT.500 [77]. A detailed description of this dataset is given in Section 6.1.2.

Table 4.2 gives the Pearson correlation results for the whole data set and for 8 out of the 13 sequences where at least four different data points were available for the same sequence (a minimum of three different data points were available for each sequence). The last row gives the overall correlation, if the parameters s and o of the regression line for each sequence were actually known.

The data in table 4.2 shows, that for one sequence coded at bit rates which differ by a factor of more than 10 (96 kbit/s compared to 1024 kbit/s), and coded using different settings concerning frame rate, or the number of I-frames inserted, PSNR can give very good quality prediction, even if the overall correlation is comparably low. This is also illustrated in Fig. 4.14, which shows the different regression lines for three sequences: for each of the single sequences, the values can be predicted very well using a linear regression line, whereas trying to fit one line for all data

Table 4.2: Correlation values for PSNR

Sequence	Data Points	Pearson Correlation
All	54	0.665
Football	6	0.968
Foreman	7	0.940
Deadline	4	0.870
Husky	4	0.995
Mobile	6	0.933
Paris	4	0.924
Tempete	4	0.944
Zoom	4	0.913
All, known parameters	54	0.991

points would result in large errors between the regression line and the actual values. Assuming one would actually know the parameters of the regression lines of all sequences, PSNR would be a perfect model[6].

As the parameters of the regression lines are not known, they were estimated using two additional instances of each sequence. This was done using the AVC/H.264 reference software in the version JM11 [132] using very simple settings:

- All sequences were coded at original frame rate (30 fps) with one I-frame every 15 frames.
- Only one reference frame was used and no B-frames were allowed.
- Instead of CABAC, CAVLC was used as entropy coding scheme.
- The encoder was set to the "low complexity mode", meaning that no extensive rate-distortion optimization was performed.
- The QP was fixed to 20 for the high quality instance, and fixed to 40 for the low quality instance.
- All other settings were set to default values.

Note, that not only different encoders were used to generate the sequences of interest,

[6] A perfect model would be able to predict the visual quality within the variation of the votes human subjects are able to give.

Table 4.3: Comparison of PSNR$^+$

Metric	Pearson Correlation	Outlier Ratio	*Slope*	$Offset^a$
PSNR$^+$	0.820	0.685	0.899	0.018
PSNR	0.665	0.907	17.488	20.728
Edge-PSNR	0.763	0.778	0.397	0.331
SSIM	0.763	0.741	0.092	0.834

[a] Visual quality is computed as (Prediction-Offset)/Slope

but also the coding settings used differ quite significantly. For a description of the settings see the reference software manual for the JM software [133].

The visual quality of the high quality instance was set to be 1.0, and for the low quality instance, a visual quality of 0.25 was assumed. After calculating $\text{PSNR}_{\text{high}}$ and PSNR_{low}, PSNR$^+$ can be computed as

$$PSNR^+ = (PSNR - o)/s \tag{4.7}$$

$$\text{with } s = \frac{PSNR_{high} - PSNR_{low}}{1.0 - 0.25} \text{ and } o = PSNR_{low} - 0.25 * s \ .$$

Table 4.3 shows, that compared to PSNR, PSNR$^+$ delivers a significant better correlation to results of subjective tests. To see how good the performance of PSNR$^+$ is compared to other well known FR visual quality metrics, prediction values were calculated for the SSIM as introduced by Wang *et al.* in [9] and for the Edge-PSNR metric as described in [6]. In addition to the Pearson correlation coefficient, an outlier ratio is given. A data point is considered to be an outlier if the difference between the predicted value and the actual value exceeds 0.05 on a 0 to 1 scale. As already mentioned, PSNR$^+$ delivers a quality prediction that does not need to be transformed into the correct range. This is not true for PSNR as well as for most other visual quality metrics including the SSIM and the Edge-PSNR. The last two columns of Table 4.3 show the slope and offset for the regression line that expresses the relationship between predicted and measured visual quality.

As the correction parameters s and o can only be estimated, the correlation result for PSNR$^+$ is still far away from the optimal model. But compared to comparably new FR metrics, the developed PSNR$^+$ metric seems to be slightly superior, and the gap between the result that can be achieved with PSNR and the exact correc-

Table 4.4: Edge-PSNR$^+$ and SSIM$^+$

Metric	Pearson Correlation	Outlier Ratio	*Slope*	$Offset$
Edge-PSNR	0.763	0.778	0.397	0.331
Edge-PSNR$^+$	0.812	0.685	0.949	0.077
SSIM	0.763	0.741	0.092	0.834
SSIM$^+$	0.845	0.667	1.010	0.034

tion parameters and pure PSNR is halved by the introduction of this very simple correction step. Detailed results for all metrics can be found in Fig. A.2 to Fig. A.5. Note, that for plotting the three comparison metrics and calculating outlier ratios, linear data fitting was applied, while for PSNR$^+$ no data fitting is necessary and therefore is not used.

Correcting Edge-PSNR and SSIM

Applying the same method of estimating the regression lines as shown in Fig. 4.13 for the Edge-PSNR and SSIM metric results in Edge-PSNR$^+$ and SSIM$^+$. Table 4.4 shows the results for these two metrics. These results show, that the correlation can be increased, and the outlier ratio decreases even though the gain is not as high as for PSNR, which may be due to the already higher correlation of the base metrics. The effectiveness for the method can be also shown by the ability to estimate the correct regression parameters. For both metrics the slope of the regression line is estimated with an error of not more than 5% and also the offset is very close to 0, while especially the SSIM is not able to deliver a 1:1 relationship between predicted values and visual quality.

4.5.3 Reducing the Complexity Overhead

The main drawback of this approach is its high complexity due to the process of encoding the two additional instances and evaluating these instances with the respective quality metric. It is therefore proposed to not encode and evaluate the complete video, but only parts of it to estimate the regression parameters. Table 4.5 shows, that encoding only the first 25% of the frames, and evaluating this part only, does not change the prediction accuracy of SSIM$^+$ at all. The effect on PSNR$^+$ and

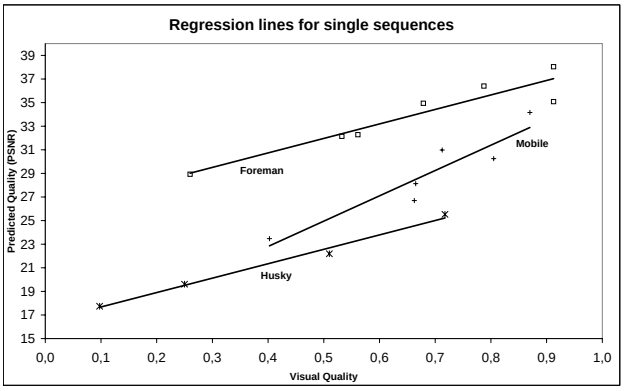


Figure 4.14: Regression lines for three single sequences

Table 4.5: Prediction accuracy (Pearson correlation) with reduced complexity

Frames	100%	50%	25%
PSNR$^+$	0.820	0.823	0.807
Edge-PSNR$^+$	0.812	0.822	0.820
SSIM$^+$	0.845	0.845	0.845

Edge-PSNR$^+$ is very small and not statistically significant, and prediction accuracy is even increased for Edge-PSNR$^+$. As encoding the additional instances at different (lower) frame rates would most probably result in a video that is quite different from the video of interest (due to very different motion vectors), this method is not recommended for video quality metrics that evaluate motion, whereas this would be possible for the three metrics used here.

4.6 Quality Prediction

To actually estimate the quality of an unknown video, the above steps are combined and the quality prediction process can be split up into five different steps:

1. Extraction of the features from the video.
2. Calculation of a set of parameters from those features.
3. Correction of the parameter values, using the obtained MSC model.

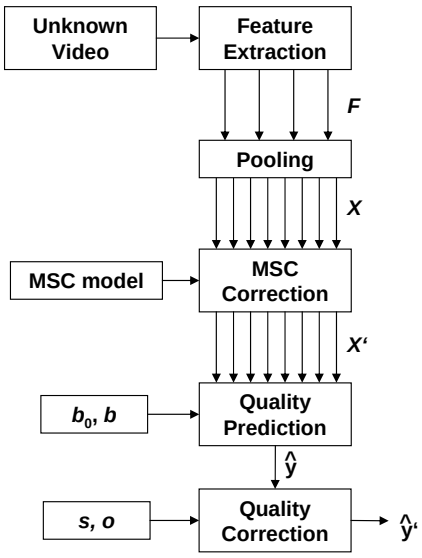


Figure 4.15: Using the model to predict the quality

4. Calculation of a quality prediction, combining the parameters using the weights gained by the multivariate regression step.

5. Correction of the quality prediction, using the correction values gained by the process as described in the previous section.

Fig. 4.15 shows the block diagram for this process. For the quality prediction, only those features which result in parameters that have weights not close to zero are extracted from the video. The main complexity for the calculation of the objective quality lies in the extraction process and, as already mentioned, in the calculation of the correction parameters s and o. Depending on the complexity of the feature extraction process, the actual quality prediction could still be performed in realtime with a delay, depending on the temporal pooling functions that are used.

Chapter 5

Multivariate Data Analysis

In this chapter, the tools used to gain the weights for the measured parameters are introduced. The method of multivariate data analysis, using Principal Component Analysis (PCA) and Partial Least Squares Regression (PLSR), is described using the same data that is later used to actually build visual quality metrics. To increase readability, only plots from four different sequences are presented (bus, city, crew and football - for a description of the used sequences see 6.1.3). The whole process of calibrating the system, that results in a set of weights for the extracted parameters, is described in this example, together with the necessary tools needed for the analysis of the data. For convenience, the notation used in this chapter follows what is used in standard textbooks on multivariate data analysis.

As described in the previous chapter, it is possible to extract a very high number of parameters from a video that describe the video, that are possibly linked to visual quality. But some of these parameters will not actually be needed to describe the properties of the video, and some of the remaining parameters may turn out to have no influence on the visual quality. Therefore, the goal is to find those parameters that are not only needed to describe the video, but actually *do* have an influence on the visual quality.

To learn, which parameters do have an influence on visual quality, one would have conducted subjective tests to gain the column vector $\mathbf{y}$ that contains one visual quality value for each video that was tested. Assuming that K videos were included in the test, the size of $\mathbf{y}$ would be $K \times 1$. At the same time, those videos would be analyzed and the selected parameters would be extracted extracted, yielding one row vector $\mathbf{x}$ with size $1 \times L$, and L being the number of extracted parameters for

each video. Together, the K row vectors $\mathbf{x}_i$ ($i \in [1 \dots K]$) would form the parameter matrix $\mathbf{X}$ with size $K \times L$.

In a first attempt, one could try to find a one-to-one relationship between any of the parameters x_j ($j \in [1 \dots L]$) and the visual quality by linear regression of the column vectors $\mathbf{x}_j$ that contain the measurements of parameter j for any of the K videos, and the visual quality vector $\mathbf{y}$, and hope that one of the many parameters allows to directly measure visual quality. But most likely this parameter does not exist, and instead of a one-to-one relationship, it is more likely, that visual quality can be measured by evaluating a combination of different parameters. Table 5.1 shows the regression values for the seven parameters that were selected in this example for 54 different videos that form this data set. The tested videos were gained by encoding 13 different source videos with different coding settings. Whereas all parameters can be used to predict the visual quality to a certain extent, obviously none of the parameters can be used to directly predict the visual quality with the required accuracy. The highest correlation between visual quality and a single parameter is achieved by the blockiness measurement, having a Pearson correlation of 0.618, which is a very good value for one single parameter. None of the other six parameters reached a correlation above 0.46 and some of the parameters even show a correlation that lies below the significance level of 0.27, among those is the variable "blur". The high correlation for the variable "blockiness" is unexpected, as the sequences contain quite some blur and other degradations besides blocking, and in some sequences, the blockiness is not even clearly visible. One attribute that differentiates blocking from all other measured parameters is, that blockiness is a pure distortion, which does not appear in the original video. A perfect blockiness measurement would show 0 blockiness for the unprocessed video[1]. In contrast, all other features have a measured value different from zero for a visually perfect video, such as blur, which comes close to representing pure distortion, also appears in natural videos.

One single parameter obviously can not capture the visual quality directly, but maybe a combination of the parameters can accomplish this task. Allowing combinations of two out of L parameters would result in $(L \times (L-1))/2$ possible combinations. For the seven parameters that were finally selected for the example metrics, this would result in 21 combinations if combinations of two parameters would be allowed. Checking all possible combinations of seven parameters would mean, that one would have to check 127 different models.

[1]For the selected algorithm a blockiness of 0 is not reached perfectly for all source videos, but the blockiness values for the source videos were always below 0.1.

Table 5.1: Correlation values for single parameters

Feature	*Pearson Correlation*
Activity	0.254
Blocking	−0.618
Blur	−0.191
Color Continuity	−0.020
Edge Continuity	0.356
Motion Continuity	0.456
Predictability	0.417
Significance Level	0.265

In the following sections, it is assumed that the $K \times L$ parameter matrix $\mathbf{X}$ and the column vector $\mathbf{y}$ are available. To show the effectiveness of the multivariate analysis in removing variable measurements that do show a variation across the K different videos, but do not have an effect on the visual quality, one parameter measurement x_{rand} is added, which contains only random values in the range of 0 to 1 (the dimension of $\mathbf{X}$ now will be $K \times L + 1$).

Following the common naming conventions for multivariate data analysis, the rows of the matrix $\mathbf{X}$ that represent the different videos will be referred as "samples", the parameter measurements are denoted as "variables".

5.1 Preprocessing of the Data

At the very first stage, the matrix $\mathbf{X}$ contains absolute values in different ranges. The single variable measurements are possibly not independent from each other, as certain effects in the video may result in changes in more than one variable. In addition, the interesting information is not so much reflected in the absolute values, but in the variation of the variable values across the different samples. Finally, all measurements may contain some noise, and care has to be taken to prevent important, but small, variation of one variable measurement from being suppressed by large unimportant noise in a different variable.

For these reasons, some simple preprocessing steps are recommended before processing the data. The absolute values for the subset that was selected for the plots in

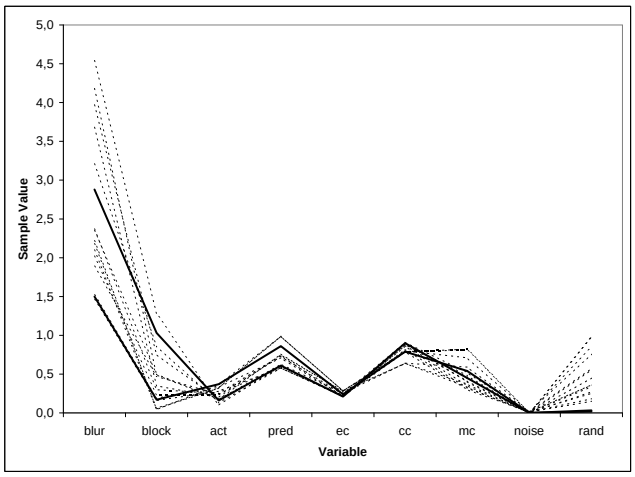


Figure 5.1: Variable values for the selected subset

this chapter are shown in Fig. 5.1. The lines for the two samples, which show a quite different behavior, are plotted in bold. Obviously, not all the selected parameters are in the same range, and have a quite different variation. Looking at the raw data could already lead to some expectations of the different influence of the various variables on the overall variance in $\mathbf{X}$: it can be expected, that the variables "blur" or "blocking" are better suited to distinguish the different samples, and to better describe the main components of $\mathbf{X}$ than "edge continuity", "color continuity", or the activity measurement as the variation in the latter variables is comparably small. It is also quite obvious, that the noise measurement did not find any real variation of noise among the different sequences. Therefore, this variable was removed at this stage. This data also shows, that the variable "random" may cause some troubles, as apart from this variable, a clear structure can be seen that is more or less similar for all samples.

5.1.1 MSC: Removing Multiplicative and Additive Effects

The measured data may not be free from additive or multiplicative effects. A certain change in the structure of the video may result in changes in more than just one variable. Applying for instance a low pass filter on the single frames of a video may result in different measurements for

- Blur: this value should be increased if the blur detection algorithm works as intended.
- Blocking: the additional low pass filtering might decrease the blockiness of the

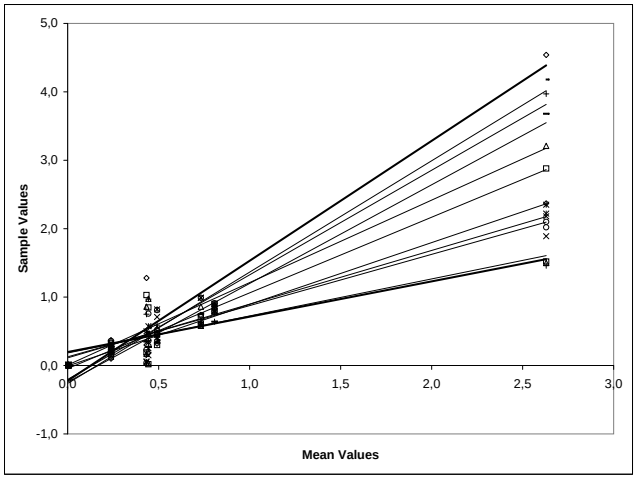


Figure 5.2: Absolute parameter values plotted versus mean parameter values

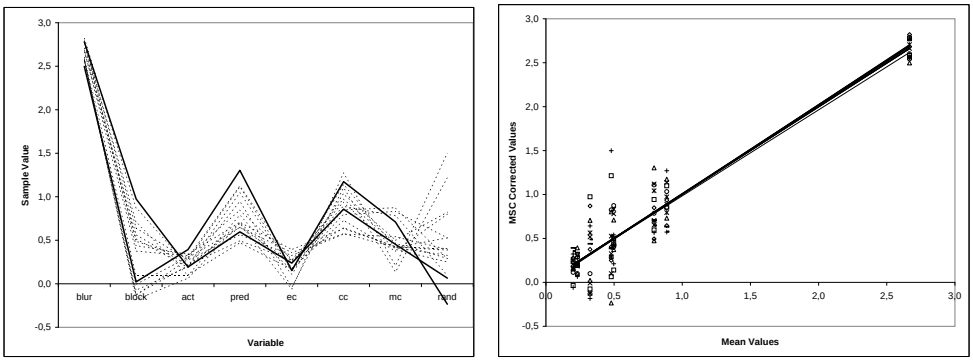
image.

- Activity: details may be reduced by low pass filtering.
- Predictability: reducing the number of details may again result in a better ability to predict one image from the previous one.

Even the two last parameters, "edge continuity" and "motion continuity"[2], may be somehow affected by this simple processing step, and only the variable "random" should not be changed. Plotting the actual values x_{ij} for each sample against the mean values of each variable $\overline{x}_j$, and adding the regression lines for the single samples as shown in Fig. 5.2, shows that these regression lines have different offsets and slopes. This is an indication for interdependencies between the different measurements. Calculating slope s and offset o of these regression lines, and correcting the variable measurements x_{ij} by $x'_{ij} = (x_{ij} - o)/s$ forces all regression lines to be close to the diagonal (see Fig. 5.3b). Still, plotting the measurements for the different samples will result in more or less the same structure as before, only having a different deviation for the different parameter measurements (see Fig. 5.3a).

This multiplicative signal correction (MSC) method was originally invented as "multiplicative scatter correction", to correct measurements in reflectance spectroscopy. While some analogy to the interference of different light spectra can be made, it is still not evident, why this method should also work for this case. As pretreatment of the parameter values using this technique actually led to more stable models,

[2]The parameter "noise" was already removed and is not considered but of course also this parameter would be affected.



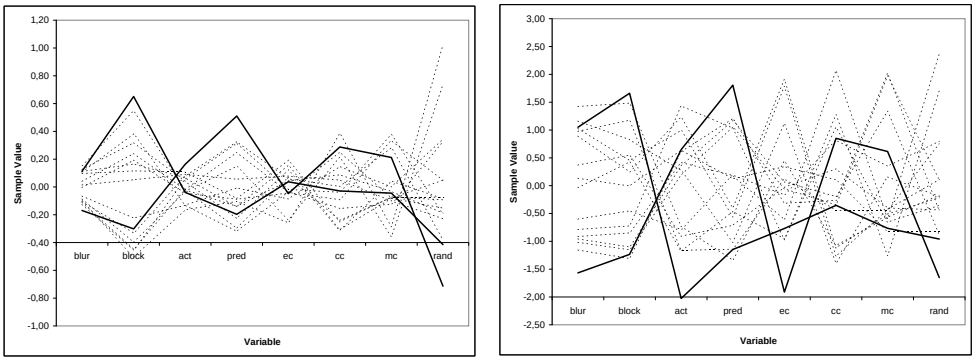
(a) MSC corrected parameter values (b) MSC corrected values versus mean values

Figure 5.3: Data after MSC correction

providing a higher prediction accuracy, this method was adopted here.

5.1.2 Center and Weight the Data

The interesting information does not lie in the absolute values of the variables (which do have a more or less arbitrary offset that is only determined by the person who designed this special variable measurement), but in the variation of this variable across the different samples. Therefore, it is recommended to center the data before further processing by calculating $x'_{ij} = x'_{ij} - \overline{x}'_j$. This corrected and centered data shows, that largest variance can be found in the variable "blocking" (and of course in the variable "random"). The data also shows, that some variables are correlated with each other (e.g. higher than average blur measurement comes with a higher than average blocking measurement), whereas others are not (the blur measurement is not correlated at all with the amount of activity measured). This structure was not visible when using the absolute values (e.g. the absolute values for blocking and blur were slightly anti-correlated). Again, the limitations of classical expectation driven modeling show up: it can be expected, that low quality sequences come with high amount of blur and high amount of blocking at the same time. But what about the activity measurement? it is not correlated to any of those. So does a high amount of details increase or decrease the visual quality? If the assumption holds, that there is a correlation between blur/blocking and visual quality and high amount of blur, and high amount of blocking artifacts results in low visual quality (and vice versa), it seems as if the activity measurement does not tell anything about the visual quality (but correlation between the activity measurement and visual quality



(a) Centered values

(b) Centered values with equal standard deviation

Figure 5.4: Preprocessing of the data

was shown before). Checking all options again would mean dealing with a very high number of possible models, even if the clear starting assumption concerning blur and blocking reduces the number of models one could take into account.

The centered data may still contain one problem: the range of the values for different variables is quite different, which again may be the result of different ranges used for the variable measurements. This may lead to the case, that noise in one variable that has a higher standard deviation, covers the important, but smaller, variation in another parameter that has a significantly lower standard deviation. If the assumption holds, that all variables have approximately the same relative error level (meaning that no variable measurement is affected by a high noise level, while others are not), scaling of the variables by dividing them by their initial standard deviation, thus forcing all variables to have a resulting standard deviation of 1.0 is proposed (see Fig. 5.4b). Still, only for the two variables "blur" and "blocking" a structure that could be linked to visual quality could be extracted by looking at the data. All other variables have some properties that do not allow a direct interpretation. The matrix that contains the preprocessed variable measurements is now depicted as $\mathbf{X}'$.

5.2 Model y by the Use of X'

The preprocessed variable measurements in the matrix $\mathbf{X}'$ are now used to predict the visual quality y, which is stored in the column vector $\mathbf{y}$. The main tool that is used to make this prediction, is the Partial Least Squares Regression method

(PLSR). In short, the PLSR consists of a Principal Component Analysis (PCA) of $\mathbf{X}'$, where the principal components (PCs) are selected in a way to best explain the variation in $\mathbf{y}$. In the following, modeling of $\mathbf{X}'$ by the use of few PCs and some modeling error $\mathbf{E}$, and the subsequent prediction of $\mathbf{y}$ is described and illustrated using the example data set.

5.2.1 Explaining the Variance in X: PCA

The matrix $\mathbf{X}'$ consists of the preprocessed variable measurements, and the structure of the data in this matrix can be described using the single variables. However, some of those variables might not be orthogonal to each other, and $\mathbf{X}'$ can be described using a smaller number of latent variables, instead of using all single variable measurements. Those latent variables are described by the PCs of $\mathbf{X}'$, associated with the largest eigenvalues of the co-variances in the data. The PCs could be obtained by bi-linear modeling of the input data, and the first few PCs normally give quite a good representation of the data structure, removing all unnecessary redundancy between the single data entries. This method of modeling the matrix $\mathbf{X}'$ using a set of PCs is the PCA. The advantages of modeling $\mathbf{X}'$ by a set of PCs is twofold:

- The model is more compact than the individual input data, allowing easier interpretation and graphical representation.
- From a statistical point of view, the model is more stable, meaning that the main structure of the model (the first PCs) is not affected by outlier samples.

The maximum number of PCs that are needed to model $\mathbf{X}'$, is determined by the number of baseline components which can be found in the input data. Baseline components are components, which can not be constructed by combining some other components. A simple example for the PCs and the number of PCs that can be extracted from a data set, would be the ingredients for different meals as given in the different recipes: the ingredients itself are the variables, and three of the variables may be green pepper, black pepper and chili, which are used by the cook to adjust the amount of spiciness. Basically, three different pepper-PCs could be extracted for those three ingredients, but most likely those ingredients are somehow correlated, and one PC may be sufficient to represent the three types of pepper. So instead of using three different variables, one pepper-PC could be used to describe the data. For the problem of describing the visual quality, the number of underlying

components is not known, and the maximum number of PCs is given by the number of variables included in $\mathbf{X}'$. Just as for the pepper example, the optimal number of PCs needs to be determined by the modeling process itself. In general, a lower number of PCs is preferred, as the model will not only be easier to understand, but also more stable.

In any real world cases, $\mathbf{X}'$ will only be partially modeled by N PCs, and some unmodeled error will remain, because the variable measurements are possibly noisy. This error that is stored in the matrix $\mathbf{E}$, decreases with every additional PC that is extracted from $\mathbf{X}'$. $\mathbf{E}$ can be as small as required by extracting a high enough number of PCs. In equation 5.1, $\mathbf{X}'$ can be modeled as

$$\mathbf{X}' = \mathbf{T}\mathbf{P}^T + \mathbf{E}. \tag{5.1}$$

Where $\mathbf{T}$ contains the column score vectors $\mathbf{t}_n$ of the PC number n (with N PCs extracted), $\mathbf{P}^T$ contains the row vectors $\mathbf{p}_n^T$ of the loadings of the same PC. The structure for one single PC number a would be

$$\mathbf{E}_{n-1} = \mathbf{t}_n\mathbf{p}_n^T + \mathbf{E}_n. \tag{5.2}$$

The score vector $\mathbf{t}_n$ represents the amount of PC number n that is needed to describe the different samples (t_i gives the amount, to which this PC is included in the sample i). In contrast, the loading vector $\mathbf{p}_n^T$ contains the "fingerprint" of this specific PC, representing the weight each variable has for this PC. $\mathbf{E}_{n-1}$ is the error that is left after extracting PC number $n-1$ ($\mathbf{E}_0$ is identical with $\mathbf{X}'$). For the above pepper-PCs the loading vectors $\mathbf{p}_{pepper}^T$ would have the length three, weighting each of the three peppers. The score vectors $\mathbf{t}_{pepper}$ would have one entry for every recipe. Recipes that do not use any pepper at all would have the score 0 for any of the pepper-PCs, whereas recipes that contain a lot of pepper will get high score values.

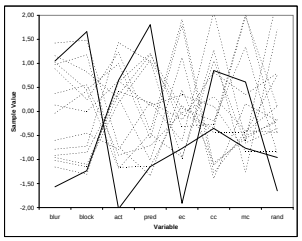
The score and loading vectors $\mathbf{t}$ and $\mathbf{p}^T$ are chosen to account for as much covariance in the matrix $\mathbf{X}'$ as possible, and minimizing the error matrix $\mathbf{E}$. Minimizing $\mathbf{E}$ means, that PCA is nothing but a least squares method. Different algorithms for the estimation of PCs were proposed, one powerful and easy to understand method is the NIPALS (Nonlinear Iterative Partial Least Squares) algorithm proposed by Wold in [137]. It is described by an iteration between preliminary loadings (5.3) and preliminary scores (5.4) for the PC number n:

$$\mathbf{p}_n^T = \left(\mathbf{t}_n^T\mathbf{t}_n\right)^{-1}\mathbf{t}_n^T\mathbf{E}_{n-1} \tag{5.3}$$

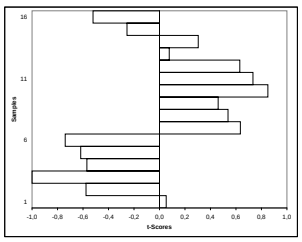
$$\mathbf{t}_n = \mathbf{E}_{n-1}\mathbf{p}_n \left(\mathbf{p}_n^T \mathbf{p}_n\right)^{-1}. \tag{5.4}$$

Mathematically speaking, PCA is an orthogonal linear transformation that transforms the data in a way, that the largest variance by any projection of the data can be found on the first PC (the second largest variance is projected on the second PC...). One difference to other transforms is, that the basis vectors (the PCs) are not fixed, but rather depend on the input data $\mathbf{X}'$.

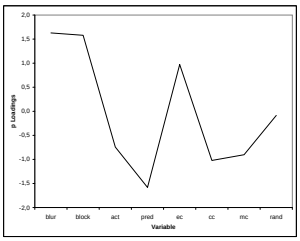
Fig. 5.5 shows, how the residual error $\mathbf{E}$ decreases with every PC that is extracted from $\mathbf{X}'$. Obviously, the first PC mainly reduces the variance for the variables "blur", "blocking" and "predictability". This is not only reflected in the plots for the error matrices $\mathbf{E}_n$, but can be also extracted from the loading plots. Variance in the variables "activity" and "motion continuity" is not significantly reduced before PC number three, and to reach a sufficient modeling of $\mathbf{X}'$, five different PCs are required. Even after extracting five PCs, the variable "random" still shows some variation that obviously can not be modeled, which is expected for a truly random variable. The first PC that contains the main information in $\mathbf{X}'$, has a very low loading value for the random variable, whereas for the higher PCs, the loading values show quite some influence of this variable on the specific structure of $\mathbf{X}'$. Inspection of the plots for $\mathbf{E}_n$ also shows, which samples contain a higher error after the extraction of n PCs, and which samples could be modeled with only one or two PCs. This valuable information can be used in the design of further variable measurements. Samples that can only be modeled with a higher number of PCs show, that maybe not the right (or not enough) variable measurements were used that possibly could lead to a lower number of PCs. Samples that need a high number of PCs to be modeled sufficiently can be deemed to be outlier samples, showing a different behavior compared to the majority of the samples. To avoid, that these outlier samples do have too big a influence on the first PCs, those can be removed temporally in the model building step. The calculation for the first PCs is then done without these outliers, and the general rule would suggest to only include these samples if the first PCs with or without these samples would be very similar. Further discussion on the model building process and the treatment of outliers is given in Section 5.2.3.



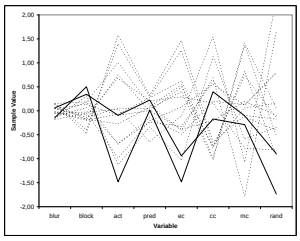
(a) Input Values



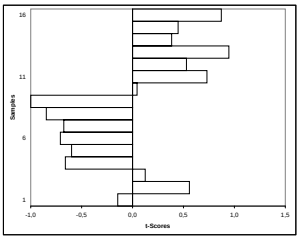
(b) Scores for PC1



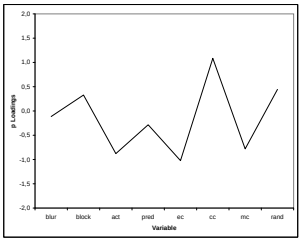
(c) Loadings for PC1



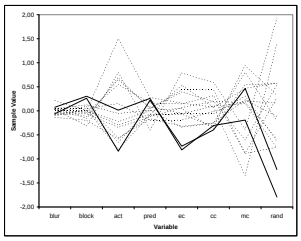
(d) Error after PC1



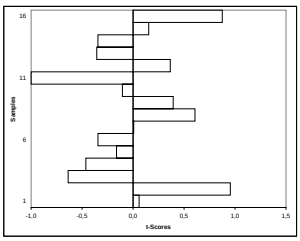
(e) Scores for PC2



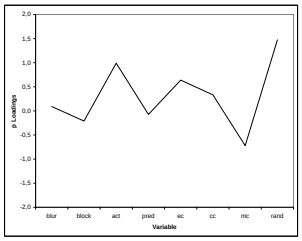
(f) Loadings for PC2



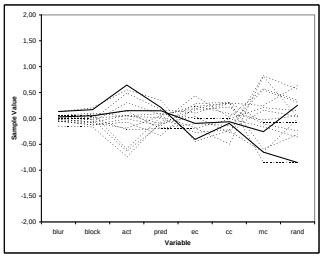
(g) Error after PC2



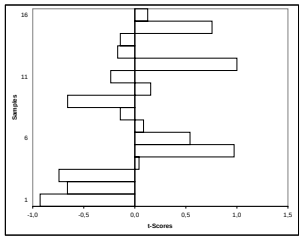
(h) Scores for PC3



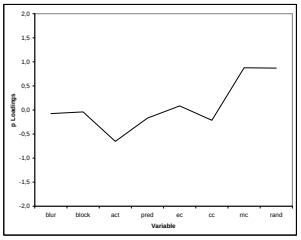
(i) Loadings for PC3



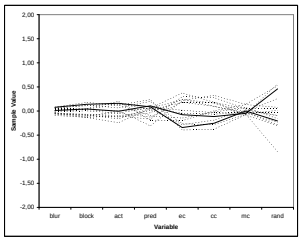
(j) Error after PC3



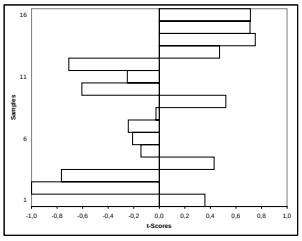
(k) Scores for PC4



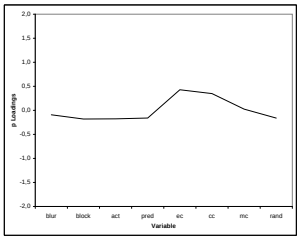
(l) Loadings for PC4



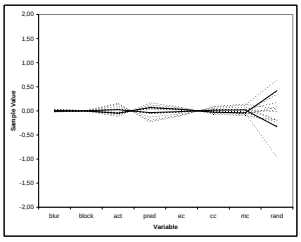
(m) Error after PC4



(n) Scores for PC5



(o) Loadings for PC5



(p) Error after PC5

Figure 5.5: PCA for the example subset

5.2.2 Predicting the Variance in y: PLSR

What has been gained by describing $\mathbf{X}'$ using a set of PCs is, that instead of dealing with a high number of different variables, one only has to cope with a lower number of principal components. In addition, these PCs are ordered according to their relevance, meaning that the first PC is more important than the subsequent PCs. Fort this reason, one can take into account a reduced number of PCs, at the cost of a higher unmodeled part $\mathbf{E}$. The complexity of finding a relationship between the visual quality vector $\mathbf{y}$ and the different variable measurements stored in $\mathbf{X}$ has been reduced significantly. Taking into account only the first (and most relevant) PC, a simple linear regression between $\mathbf{y}$ and the scores of this first PC, $\mathbf{t}_1$, would deliver a first solution for the problem of trying to predict $\mathbf{y}$ by the use of $\mathbf{X}'$.

This method of first decomposing $\mathbf{X}'$ into a set of PCs that are a function of $\mathbf{X}'$, $\mathbf{T} = w(\mathbf{X}')$, and a subsequent regression of $\mathbf{y}$ on $\mathbf{T}$ is the so called Principal Component Regression (PCR). As the best PCs for $\mathbf{X}'$ (those PCs that minimize $\mathbf{E}$) may not be the best PCs for $\mathbf{y}$, this method is slightly extended to the Partial Least Squares Regression method (PLSR). The PLSR tries to find those PCs of $\mathbf{X}'$ that are not only relevant for $\mathbf{X}'$, but also can explain the variance in $\mathbf{y}$. Whereas the PCs obtained from $\mathbf{X}'$ by PCA should model $\mathbf{X}'$, a set of PCs is needed that can also model $\mathbf{y}$. Again, $\mathbf{X}'$ can be modeled as $\mathbf{X}' = \mathbf{T}\mathbf{P}^T + \mathbf{E}$, but now also $\mathbf{y}$ needs to be modeled by

$$\mathbf{y} = \mathbf{T}\mathbf{q}^T + \mathbf{f} \tag{5.5}$$

where $\mathbf{f}$ is the error vector that should be minimal, and $\mathbf{q}$ is the loading vector for $\mathbf{y}$. Note, that for this case $\mathbf{y}$ is just a column vector having the dimension $K \times 1$, whereas $\mathbf{Y}$ with M different Y-variables is allowed.

Calculation of the PCs now has to be done slightly different, and the scores vector $\mathbf{t}_n$ for the PC number n is given by

$$\mathbf{t}_n = w_n \mathbf{X}' \tag{5.6}$$

where w_n is the so called loading weight vector. This loading weights vector ensures, that variations in both $\mathbf{X}'$ and $\mathbf{y}$, are explained by the selected PCs. To do this, the loading weight vector $\mathbf{w}_n$ is the first eigenvector of the covariance of the error that remained in $\mathbf{E}_{n-1}^T$ and $\mathbf{f}_{n-1}$ after extracting PC number $n-1$.

$$\mathbf{w}_n = \text{ first eigenvector of } \left(\mathbf{E}_{n-1}^T \mathbf{f}_{n-1} \mathbf{f}_{n-1}^T \mathbf{E}_{n-1}\right). \tag{5.7}$$

The scores $\mathbf{t}_n$ are then defined as

$$\mathbf{t}_n = \mathbf{E}_{n-1}\mathbf{w}_n \tag{5.8}$$

and the X-loadings $\mathbf{p}$ again can be calculated by solving the least squares solution 5.3. Equally, the Y-loadings $\mathbf{q}$ can be computed by solving

$$\mathbf{q}_n^T = \left(\mathbf{t}_n^T\mathbf{t}_n\right)^{-1}\mathbf{t}_n^T\mathbf{f}_{n-1}. \tag{5.9}$$

$\mathbf{y}$ may be also modeled directly from the matrix $\mathbf{X}'$ by

$$\mathbf{y} = b_0 + \mathbf{X}'\mathbf{b} + \mathbf{f} \tag{5.10}$$

where b_0 is the model offset, and $\mathbf{b}$ (with dimension $L \times 1$) is the vector of the weights for each of the L variables. The coefficients of $\mathbf{b}$ for N PCs can be obtained by

$$\mathbf{b}_N = \mathbf{W}_N\left(\mathbf{P}_N^T\mathbf{W}_N\right)^{-1}\mathbf{q}_N^T. \tag{5.11}$$

5.2.3 Model Validation

The goal for the PCA is to model the variance in $\mathbf{X}$ as precisely as possible, using as less PCs as possible. The result of the PCA, which is a set of PCs, can be validated and assessed without further problems. Most of the time, a trade off between a simpler model with only few PCs, and a reduced modeling error that is the result of more PCs can be made. But if the application limits the number of PCs, or requests a modeling error below some threshold, this request can also be met. But the scope of any model gained by PLSR (which is given by the vector of weights $\mathbf{b}$) is to predict unknown values of y. As these values are unknown, no direct validation of the model could be made. Instead, a set of validation methods (or validation measurements) has to be used to assess the possibility of the gained model to predict future values of y.

The following attributes of one PLSR model can be assessed:

- The stability of the model's parameter ($\mathbf{B}$, $\mathbf{P}$, $\mathbf{Q}$, $\mathbf{P}$, $\mathbf{W}$).
- The modeling accuracy (the unmodeled error that remains in $\mathbf{E}$ and $\mathbf{F}$) .
- The prediction accuracy (the accuracy to which new, previously unknown, values of y can be predicted).

In addition to this "internal validation", some external validation can be done, evaluating the generality of the model and it's fit to a priori information about the base components that were used to create $\mathbf{X}$. This external validation is summarized by Martens and Martens as *"No prediction without interpretability"* (on page 205 in [138]), meaning, that every model should be checked against expert knowledge to see if what the model suggests can be interpreted in a meaningful way. Whereas this external validation could be seen as soft validation, the internal validation can be done in a more formal way[3].

As the error in $\mathbf{E}$ can be made as small as required, by simply extracting a high enough number of PCs, measuring the error that remains in $\mathbf{E}_n$ after having extracted n PCs is obviously not a good measurement for the usability of the PLSR model. Assessing the quality of a PLSR model by analyzing $\mathbf{E}_N$ together with $\mathbf{F}_N$ does not lead to a more satisfying solution, as it is not known if the samples that form $\mathbf{X}$ are suited to represent all possible future samples, or if the number of (significant) different samples in $\mathbf{X}$ is high enough. It may well be, that the PLSR model can accurately predict the y-values for this known samples without having a useful prediction accuracy for any other previously unknown sample. This also shows a problem that is more important for PLSR than for PCA, and that requests some model validation during the model calibration step. So far, the question of the optimal number of PCs N_{opt} for the PLSR has not yet been addressed.

The tool that is used most often for validation during the calibration phase, as well as for the final internal validation, is the so called cross validation. Cross validation could be described as the process of temporally removing samples from the matrix $\mathbf{X}$, recalibrating the PLSR model with the remaining samples, and testing the gained model on the now unknown samples[4]. A full cross validation would mean reiterating this process for every single sample, resulting in K more or less different PLSR models. What then can be assessed is the following:

- The stability of the model, given by how much different the gained model parameters actually are. If the parameters for one model k differ significantly from those for the other $K-1$ models, the sample that was left out for model k may have a too strong influence on the other $K-1$ models. In this case,

[3] Soft validation here does not refer to the importance of that validation step, but refers to the fact, that this validation can not be done by the use of mathematical formulas only, but requires expert knowledge and experience.

[4] The samples have to be removed from $\mathbf{X}$, and not from $\mathbf{X}'$, as the values in $\mathbf{X}'$ already depend on those in $\mathbf{X}$.

removing the sample k for all models may be beneficial to avoid tailoring the model too much to this sample. As this will be appropriate for most cases, of course also the opposite may be true, and this sample may contain some valuable information, and should be included in the model in any case. At this stage, obviously external validation is necessary.

- The ability of the model to predict unknown values of y from $\mathbf{x}$ can be expressed using the RMSE (root mean squared error) between the predictions $\widehat{y}$ and the actual values y for the temporally unknown samples.

For PLSR, what is really interesting, is the prediction accuracy and therefore the number of PCs is not determined by the residuals $\mathbf{E}_N$ and/or $\mathbf{F}_N$, but by the ability to predict unknown values of y, which is analyzed in such a cross calibration process. Whereas $\mathbf{E}_N$ and/or $\mathbf{F}_N$ may be further reduced by extracting more PCs, the prediction accuracy may decrease due to fitting the model too much to the input samples, and at the same time lose generality. This is often referred as "overfitting". The optimal number of PCs N_{opt} is thus given by the highest PC n that decreases this prediction error compared to the error that was gained by extracting only $n-1$ PCs.

The cross validation process has to be adapted to the application and the behavior of the sample values. Obviously, the standard full cross validation method that always excludes only one sample would not work as intended, if for every sample at least one more sample with very similar characteristics would be included in $\mathbf{X}$. Such a similarity could be detected by comparing the score vectors $\mathbf{t}$ for the model samples, calibrated using the whole matrix $\mathbf{X}$. Similar samples would have very similar score values for the different PCs. If this is the case, instead of temporarily excluding only one sample, the whole set of similar samples should be excluded temporarily during the cross validation phase. For the application of objective video quality metrics, one should avoid predicting the visual quality y for one processed sequence using a model that contains a differently processed version of the same reference sequence. This is true, even if the score vectors would not suggest treating such samples as being similar.

For the used small demonstration dataset, the PCs that were gained by PLSR are very similar to those that were gained by PCA to explain the variation in $\mathbf{X}'$, and that are shown in Fig. 5.5. However, cross validation shows that the prediction error does not decrease from PC1 to PC2, but it increases again with PC3, which suggests the optimal number of PCs is one for this case. As already discussed in

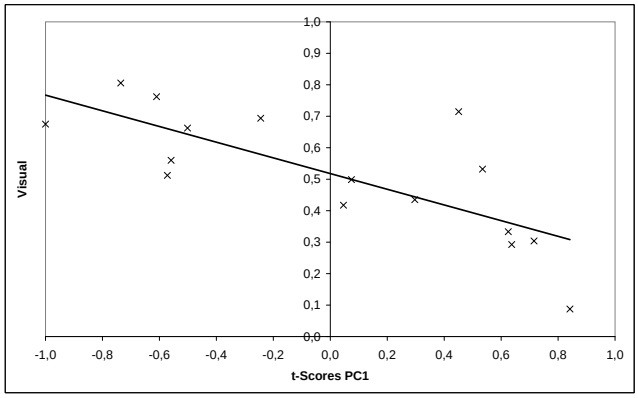


Figure 5.6: Linear regression between y and the scores of PC1

Section 5.2.1, the loading value for the random variable in PC1 is very close to zero, so as expected and wished, this variable is not needed to predict y. Fig. 5.6 shows the relationship between $\mathbf{y}$ and $\mathbf{t}_1$.

Chapter 6

Results

To demonstrate the effectiveness of the presented approach, a set of three reduced reference metrics for AVC/H.264 is developed. Those metrics are all based on the same set of features and parameters as described in Section 4.2.5, and vary only in the weights assigned to the different parameters. Each metric is calibrated to be used for one spatial resolution class, either QCIF (176×144 pixel), CIF (352×288 pixel), or 4CIF (704×576 pixel). The largest data base was available for CIF resolution sequences; therefore this class is used to explain the modeling process in more detail, whereas for the other two resolutions, only the results are given. Strict classification would rate the developed metrics to be reduced reference metrics, but one could also consider this metrics to be no reference metrics that are corrected in a final correction step, as the base metric evaluates only parameters gained by features extracted in a no reference process. Compared to "classical" reduced reference metrics, the presented approach delivers reduced reference metrics, which do not need to perform an alignment between the reference and the processed video. As shown in Section 4.5.3, not even the complete reference video needs to be evaluated.

6.1 The Database used for Calibration and Verification

The data used for calibration and verification of the proposed video quality metrics was generated within two different subjective tests that were carried out as part of the standardization activity inside MPEG. Those two tests are the verification tests for AVC/H.264 (AVC VT [42]), that were carried out in late 2003, and the tests for the call for proposals on scalable video coding (SVC CfP [43]) that lead to the

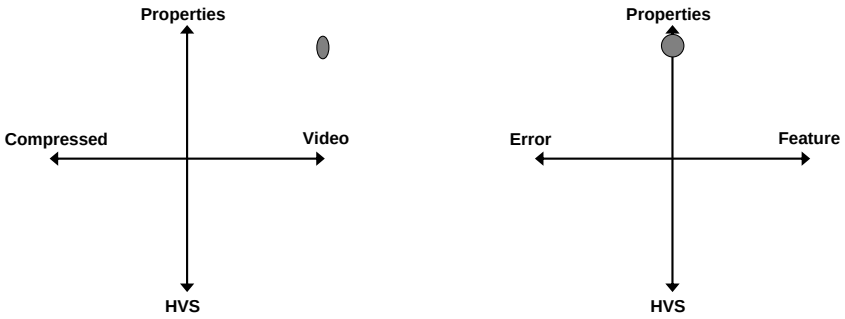


Figure 6.1: Location for proposed metrics inside the QMC

scalable extension of AVC/H.264, now known as SVC. Whereas each test did include sequences encoded by other coding technologies than AVC/H.264[1], the metrics were developed and verified on the AVC/H.264 encoded sequences only. For this reason, the resulting metrics are only suited for video encoded according to this standard.

6.1.1 Restricting the Metric to AVC/H.264

The restriction to AVC/H.264 was accepted, as general video quality metrics that are able to predict the quality of any processed video, independent of the processing steps that were applied, have not proven to work to a satisfying level so far. Furthermore almost all video quality metrics were evaluated for one codec technology, or even one special encoder. Only the metrics that were submitted to the two evaluation efforts of the VQEG ([2, 3]), were evaluated using a broader range of codecs and test conditions[2]. As the goal of this work was to develop video quality metrics that can work in a reduced reference manner with very low overhead only, and where the base metric is a no reference metric, the desirable goal of universality was dropped for the necessary goal of high prediction accuracy. AVC/H.264 has a significant different "look" than previous video codecs, such as MPEG-2 or MPEG-4 ASP, mostly due to the use of an inloop filter. This inloop filter shifts the trade off between "high details" and "low artifacts" in the direction of "low artifacts", and therefore it would be especially difficult to construct a general no reference video quality metric that could work on a "blocky" MPEG-2 version of a video sequence, and the "smooth" AVC/H.264 encoded version of the same video. In addition, the restriction to AVC/H.264 seems to be reasonable from an application or market

[1]The AVC VT anchor test points were generated using MPEG-4 ASP encoders. For the SVC CfP, a number of scalable video codecs based on different coding approaches were proposed.

[2]Here, the processing steps consisted of MPEG-2 encoding and analog encoding.

point of view. AVC/H.264 has shown to outperform their predecessors MPEG-2 and MPEG-4 ASP by roughly a factor of two (see [42]), and is widely adopted since its standardization in 2003. This is especially true for new services, such as transmission of HDTV channels or IPTV, which will be mostly based on AVC/H.264 encoded video.

Evaluation of the presented metrics at a slightly higher level also shows, that the generality is dropped only at the very last level, where the same parameters are combined using a different set of weights. It can be expected, that using a general set of features and resulting parameters, metrics for different coding technologies can be build, using different sets of weights. Some of these weights then may be zero for a certain coding technology, just as the feature "noise" is not used for the presented AVC/H.264 metrics, whereas the same feature may be relevant for other coding technologies.

6.1.2 Subjective Tests

The subjective tests done to evaluate the visual quality of the videos were performed at the video test laboratory of the Lehrstuhl für Datenverarbeitung (LDV) at the Technische Universität München (TUM) in Munich and at the Fondazione Ugo Bordoni (FUB) in Rome. The tests were done using a single stimulus method. For the single stimulus method, a discrete scale with eleven grades as shown in Fig. 3.3 was used. To minimize the contextual effect for the SS tests, each test case was shown twice during the whole test. A DLP projector was used to display the progressive scanned sequences. The room setup was as shown in Fig. 3.5, and is compliant to ITU-R BT.500 ([77]). The tests were done by at least 20 naïve test subjects, students who were not familiar with video coding or video quality assessment. After screening the subjects for visual acuity and color blindness, they were trained on the test method using a set of different sequences that were coded to represent the same variation in visual quality, as the sequences under evaluation. To maintain the same viewing distance for all the participants, not more than four test subjects took part in the test at the same time, all sitting within a viewing angle of less than 30°. To avoid fatigue during the test, no single test session lasted for more than 25 minutes, and a break of at least the same time was inserted between two test sessions. The results of the tests were then screened for outliers and inconsistent votes (see Section 3.1.3). The percentage of removed votes due to outliers and obvious errors was below 5%. The 95% confidence intervals for the single tests were in the range

of 0.025 to 0.05 on a 0 to 1 scale, having a mean confidence interval of 0.036. For calibration and verification of the visual quality metrics, all results were fitted to a 0 to 1 scale by simple linear scaling.

6.1.3 The Encoded Videos

A total of 13 different sequences were used. Some of them were well known test sequences such as “Foreman” or “Mobile&Calendar”, whereas others were comparably new, and were used the first time for tests inside MPEG (e.g. “City”, “Crew”). The sequences represent different content scenarios, such as conversational, news, or sports, and include static scenes as well as scenes with complex and fast motion, and different types of camera movement. Sample images for these sequences are shown in Fig. A.1a to Fig. A.1m

The encoding was done using different encoders that produce AVC/H.264 compliant bit streams. The bit streams that were generated for the SVC CfP were generated using the AVC/H.264 reference software (version 7.3), whereas the sequences for the AVC VT were generated using proprietary encoders. There is also some variance in the coding modes concerning the number of B-Frames inserted (0 to 2 B-Frames), or the distance between random access points. For conversational applications, no random access points were required, and therefore only the first frame needed to be an I-Frame, whereas for the other applications, a random access point was required every half a second. Table A.1 lists all the used sequences, the respective bit rates, coding modes, and the scaled subjective ratings.

6.2 Resulting Metrics

In this section, the example metrics are presented that were generated using the described data, and the feature measurements described in Section 4.2.5. As already mentioned, these metrics have been built to provide a proof of concept, and are not necessarily the optimal metrics that can be gained with the proposed method. Especially including more precise or additional feature measurements, as well as including other pooling methods for spatial and temporal collapsing, will most probably lead to more accurate metrics. As each spatial resolution has its own special attributes, and the same parameters may have a different effect for different spatial resolutions, the metrics were calibrated with the data of one resolution class each.

For comparison, results were also presented for PSNR, the Edge-PSNR according to [6], and the SSIM according to [9]. The Edge-PSNR was chosen as one representative of the metrics standardized in [31], and the SSIM was selected due to its high popularity and wide use by the video and image coding community.

6.2.1 CIF

The largest number of data points were available for CIF (352×288 pixel). Therefore, some more results are presented for this resolution. Here, the differences between the different models that are constructed in a "leave one out" process are shown, and additional results are shown for calibration of one model on a comparably small subset. For reference, the steps that are needed to build the metric are again described very briefly, and also the verification part is again addressed shortly.

Building the models

Comparing all 13 different regression models that could be built by the "leave one out" process, it turned out, that three out of the 13 sequences have a bigger impact on the gained regression models: the regression models where one of these three sequences was not included for calibration differ quite significantly from all other models. For this reason, those sequences (Crew, Husky, and Mobile&Calendar) were removed for the model calibration step, and instead of 13 different regression models, only ten different models were built. In short, the following steps were performed for the model building process:

1. Calculate the features and parameters of the sequences (results in $\mathbf{X}$).
2. Remove the sequences Crew, Husky, and Mobile&Calendar from the data set.
3. Remove one additional sequence from the data set.
4. Perform a MSC correction step on the remaining part of $\mathbf{X}$, center and weight that data (results in $\mathbf{X}'$ and a MSC model).
5. Perform a PLSR on $\mathbf{X}'$ and $\mathbf{y}$ (results in the regression models described by b_0 and $\mathbf{b}$).

Steps 3 to 5 are repeated for each of the ten remaining models, and one additional model (for reference referred as model#11) was built by omitting step 3. The resulting models were all based on only one principal component.

Table 6.1: Weights of the objective parameters for model#11

Feature	$Weight$ b
Activity	0.023
Blocking	−0.044
Blur	−0.042
Color Continuity	0.014
Edge Continuity	−0.012
Motion Continuity	0.029
Predictability	0.035
b_0	1.712

The resulting weights b_m for the model#11 are given in Table 6.1. The other ten different regression models gained are very similar to each other, and do not vary significantly from the one presented. This shows, that no single sequence has a large influence, and that model#11 could be used as a starting point for further investigation. As this regression model was calibrated using a comparably large data base, including ten different sequences and spanning a large quality range from 0.26 to 0.913, it can be expected, that the given model has a good performance on other unknown sequences.

To show, that the gained regression models deliver very similar quality predictions, the quality values for the three sequences that were left out for calibration were predicted using all eleven different regression models. In Fig. 6.2, the error bars show the prediction range for the single data points, using the eleven different regression models for the quality prediction. The maximum difference between the highest prediction and the lowest prediction for the same data point is below 0.08, and in most of the cases the variation in the prediction results is much smaller, resulting in an average difference of 0.03 between the highest and lowest prediction values.

The gained weights are used to build a no reference quality prediction. This quality prediction is then corrected using the correction method as described in Section 4.5. The resulting metric is a reduced reference metric in the sense, that some additional data has to be sent from the reference video to the processed video. However, the metric is not a classical reduced reference metric, as those methods normally directly compare the gained parameters from the reference video and the processed video.

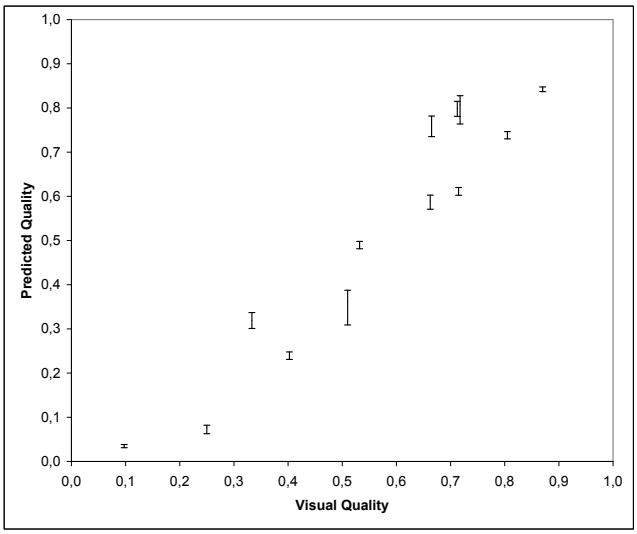


Figure 6.2: Prediction differences for the three selected sequences using all eleven different regression models

Gaining the quality prediction

The low quality video needed for the correction step was constructed by encoding the video using the AVC/H.264 standard with high (fixed) quantization parameter (resulting in low quality), using the very simple settings as described in 4.5.2. As already mentioned, not only the coding parameters for producing this low quality video differ significantly from those used to encode the videos under test, but also a different encoder has been used for this task. The assumed visual quality for the low quality video was set to 0.25. To take into account the nonlinear relationship of subjective ratings, a final sigmoid correction as specified in Section 3.3.2 and shown in Fig. 3.12 was applied.

The visual quality of each sequence was predicted using the regression model that was gained without the use of this special sequence. For the three sequences Crew, Husky and Mobile&Calendar, the regression model#11 was used (here, any of the models could be used). In total, 54 data points were available for verification, spanning a quality range from 0.098 to 0.913[3].

[3]The low quality data point - Husky at 96 kbps - was not included in any of the calibration data sets, as it was found that it would spoil the calibration process. Still, it was verified using the model#11 resulting in a predicted quality of 0.036.

Table 6.2: Prediction result for the CIF dataset

Metric	Pearson Correlation	Spearman Correlation	MSE	Outlier Ratio
Proposed	0.835	0.808	0.018	0.630
PSNR	0.665	0.657	0.039	0.907
Edge-PSNR	0.763	0.770	0.025	0.778
SSIM	0.763	0.730	0.024	0.741

Table 6.3: Regression lines before data fitting CIF

Model	$Slope$	$Offset$
Proposed	1.004	−0.033
PSNR	17.488	20.728
Edge-PSNR	0.334	0.357
SSIM	0.171	0.818

Verification and comparison

The comparison results for PSNR, Edge-PSNR and SSIM were first fitted to the actual data by the use of first order fitting (a step, that was not performed for the proposed metric), and the same sigmoid correction as for the proposed metric was then applied to take into account the nonlinear relationship of the subjective ratings. Note, that the applied fitting step is not possible in a real world scenario, as the parameters for slope and offset are not known in advance. The parameters used for this fitting step are given in Table 6.3. As reference, the linear regression parameters for the gained combined model are also given, which show, that with this method no fitting is needed, and a 1:1 relationship between the predicted quality and the visual quality is obtained. For comparing the proposed method with the comparison metrics, the Pearson and Spearman correlation values, the mean squared error (MSE), and the outlier ratio is calculated. Performing a test on statistical significance using Fisher's z-Transform shows, that the proposed method clearly outperforms PSNR, which is not the case for the two other metrics. The metric also comes close to outperform the Edge-PSNR and the SSIM at a 95% confidence level. Detailed results for the proposed method can be found in Fig. A.6, whereas detailed results for the comparison metrics are given in Fig. A.2 to Fig. A.5.

Table 6.4: Weights of the objective parameters for model#11 and the reduced calibration set

Feature	*Weight model#11*	*Weight reduced set*
Activity	0.023	0.035
Blocking	−0.044	−0.037
Blur	−0.042	−0.038
Color Continuity	0.014	0.023
Edge Continuity	−0.012	−0.026
Motion Continuity	0.029	0.019
Predictability	0.035	0.029
b_0	1.712	1.239

Calibration using a reduced data set

To check, if a prediction model can also be built using a small set of selected calibration sequences, a model was built on the basis of using only four sequences for calibration. Those sequences were Bus, Football, Harbour and Mobile&Calendar. The sequences were selected, as they span a wide range of image features and vary significantly concerning motion, the amount of details or the color properties. For this case, the verification data set consists only of the nine other sequences, resulting in only 36 data points for verification. The restriction on only four selected sequences did not have a significant influence on the prediction ability: the Pearson correlation for the smaller verification set is 0.845 compared to 0.850 what can be gained with a "leave one out" calibration, that uses much more data points for calibration of the model. Table 6.4 shows also, that the weights vary in a comparably small range.

Effect of the correction step

The proposed correction step introduces a very small amount of reduced reference data: only two values per sequence or measurement period are needed. This small amount of data can be easily embedded in the data stream e.g. in the SEI (Supplemental Enhancement Information) messages provided by the AVC/H.264 video coding standard. However, the system looses the ability to predict the quality without access to the reference video, and introduces a comparably high amount of extra

Table 6.5: Prediction performance for the no reference system

Metric	Pearson Correlation	Spearman Correlation	MSE	Outlier Ratio
Proposed	0.835	0.808	0.018	0.630
Proposed NR	0.560	0.519	0.038	0.722

computational complexity. This burden needs to be justified by a significant increase in prediction accuracy. As already shown in Section 4.5.2, the increase in prediction accuracy for PSNR is significant. However, for the Edge-PSNR and the SSIM the effect can be demonstrated, but is not significant from a statistical point of view. For the presented metric, Table 6.5 shows that the Pearson correlation is significantly lower than what can be reached with the correction step included, whereas the outlier ratio for the no reference metric is comparably low, and also the MSE is at least comparable to what PSNR can deliver. The prediction results for the pure no reference metric are shown in Fig. A.7.

Interpreting the model weights

Taking a look at the weights gained by the multivariate calibration, it can be seen that in fact the input parameters “Blur” and “Blocking” do have the biggest influence on the measured visual quality, though the image quality is not as much dominated by these two parameters as one could have expected. As expected, the weights for these two parameters are negative: increasing blur and blockiness will decrease the visual quality. “Edge Continuity” is the third parameter with a negative weight. Here one has to remember, that high similarity in the edges between two successive images results in lower values, whereas an increased difference results in higher values. So if the images do have a stronger similarity in the edge regions, this results in an increased image quality. The same is true for the other continuity measurements: the more similar the two successive images (or motion vector fields) are, the higher is the measured visual quality. Also the weight for the “Activity” measurement backs the a priori assumptions: an increased amount of details will contribute positively to visual quality.

Table 6.6: Prediction result for the QCIF dataset

Metric	Pearson Correlation	Spearman Correlation	MSE	Outlier Ratio
Proposed	0.741	0.732	0.049	0.800
Proposed fitted	0.741	0.732	0.029	0.771
PSNR	0.717	0.726	0.038	0.875
Edge-PSNR	0.789	0.792	0.026	0.743
SSIM	0.732	0.735	0.034	0.743

6.2.2 QCIF, 4CIF

For the 4CIF and QCIF dataset, the same procedure was used to generate the metric and compare it's results to the comparison metrics PSNR, Edge-PSNR, and SSIM. The weights given for the parameters were gained using the whole dataset for calibration. Of course, slightly different weights were used to produce the prediction results according to the "leave one out" procedure. All results were subjected to the fixed sigmoid correction to force the predicted visual quality to be in the range of 0 to 1, and to take into account the nonlinearities in the subjective votes. The data for the QCIF and the 4CIF test cases show results that differ only slightly compared to the results for CIF. For the QCIF dataset, that contains 25 data points from the same 13 sequences as used for CIF, the proposed metric, and the three comparison metrics are not distinguishable from a statistical perspective: Fishers's z-Transform as well as ANOVA on the absolute difference between predicted and actual quality show, that the prediction metrics (including PSNR) deliver the same prediction accuracy. It can be observed, that the proposed method clearly overestimates the visual quality which is also reflected in the offset of roughly 0.1 for the linear regression line. As a comparison, results are also presented for the proposed method with linear fitting, as it is applied to the three comparison metrics. Whereas the Pearson and Spearman correlation values remain unchanged, when only first order fitting was applied, the outlier ratio decreases slightly and the MSE is reduced. Similar to the CIF dataset, the QCIF dataset spans a visual quality range ranging from very low quality (0.06) to high quality (0.81). Detailed results for the QCIF data are shown in Fig. A.8 to Fig. A.12.

The comparison of the weights for the CIF and the QCIF model, that are based on exactly the same sequences, shows that the variations are still comparably small.

Table 6.7: Regression lines before data fitting QCIF

Model	$Slope$	$Offset$
Proposed	1.021	0.103
PSNR	16.482	21.985
Edge-PSNR	0.309	0.388
SSIM	0.206	0.805

Table 6.8: Weights of the objective parameters for QCIF to 4CIF using all sequences for calibration

	$Weight$		
Feature	$QCIF$	CIF	$4CIF$
Activity	0.022	0.034	0.016
Blocking	−0.059	−0.046	−0.028
Blur	−0.046	−0.041	−0.011
Color Continuity	0.020	0.016	0.036
Edge Continuity	−0.017	−0.006	0.033
Motion Continuity	0.032	0.028	−0.006
Predictability	0.035	0.034	0.006
b_0	0.683	1.573	0.671

The weights for the QCIF metric are in the range of the weights that were gained for CIF using slightly different calibration sets. The used features are relative features, that are not bound to the size of the frames. The values for the weights show, that the general relationship of the contributions of one feature to visual quality is preserved for these two different resolutions. The offset parameter b_0 that is significantly lower for QCIF compared to CIF shows, that if the same amount of distortion and the same amount of "positive" features can be found in the video, the image quality will be significantly lower compared to CIF videos.

For 4CIF, the highest resolution that was evaluated, only a very small number of twelve data points is available, which limits the possibility to draw general conclusions from the presented results. In addition, these data points span a comparably small quality range from 0.54 to 0.83, with most of the data points grouped around

Table 6.9: Prediction result for the 4CIF dataset

Metric	Pearson Correlation	Spearman Correlation	MSE
Proposed	0.845	0.820	0.025
PSNR	0.629	0.505	0.052
Edge-PSNR	0.845	0.827	0.026
SSIM	0.513	0.554	0.044

0.8. However, results show, that the proposed method also does provide good results for higher resolutions and higher bit rates. Calculating an outlier ratio for this data set would show, that none of the data points could be predicted with high enough accuracy to be classified as inlier for any of the four metrics. The data set that was used for generating the sample weights for 4CIF differs significantly from the set that was used for CIF and QCIF. For this reason, the significant differences between the weights for 4CIF and the weights for CIF/QCIF are most likely not only a result of differences in resolution and bit rates, but also a result of the different calibration sequences.

Chapter 7

Conclusion

This work showed, that it is possible to build visual quality metrics utilizing available no reference feature measurement methods, and using advanced methods for combining the parameters gained from such a set of features. Multivariate data analysis has shown to be a very powerful tool in this context. In the past, more attention was given to the development of vision models, whereas relatively simple methods have been used to combine the features derived from those models into one overall quality measurement. Most of the time, combination of features and parameters was based on vector summation or simple regression models. Multivariate data analysis has shown to be able to efficiently remove features and parameters that do not have an influence on the perceived quality, and provides stable metrics, even if trained with a comparably small dataset. Increasing the calibration dataset by the use of a cross calibration and cross verification procedure, leads to relatively small improvements compared to the use of a carefully selected set of calibration sequences. Furthermore, this method decreases the danger of tailoring one metric to a comparably small set of sequences, and allows one to gain an indication about the prediction ability even for very small data sets.

The decision not to build any HVS model, but to treat the HVS as a "black box", described only by some inputs and the output "visual quality", has eased the process of building a metric, with no disadvantages detected so far. However, including masking models, or models of the primary visual system may still improve the prediction ability of visual quality metrics. Features gained by modeling parts of the HVS may be evaluated using multivariate data analysis, and their effects on the perceived visual quality can be determined. In the presented approach, spatial and temporal pooling was performed by simple averaging, and it can be expected, that

additional pooling methods, resulting in additional parameters that are based on the same features, can lead to better prediction results. Especially pooling methods that emphasize strong local artifacts most likely affect the prediction process. This is caused by the observation, that strong artifacts have a bigger influence on the perceived visual quality as it would be expected by simple averaging. Also, including more different features to measure properties or distortions of the video that have not been addressed for the presented example metrics (e.g. color, contrast, ringing), or more accurate feature measurements may lead to more accurate prediction results.

The correction step described in Section 4.5 has been shown to significantly improve the prediction results, not only for the proposed method, but to be beneficial to all four evaluated metrics, delivering a significant increase in the prediction ability for PSNR. Applying the correction step for PSNR results in a metric that is among the best full reference metrics for video, introducing a comparably high burden for new full reference metrics. This correction step can be used for no reference metrics, as well as for full reference or reduced reference metrics. It has also been shown, that the additional complexity that is added by this step can be reduced by evaluating additional instances that were generated from parts of the reference video only. This also shows, that no exact alignment between the reference video and the processed video is necessary for the correction step, which is something that is normally required for reduced reference metrics. As the alignment process would need some computational resources, the complexity increase for the overall system is further reduced.

Combining an extended set of feature measurements, that consider distortions as well as features of the video, together with spatial and temporal pooling methods that gather the distribution of this features in space and time, an effective method for combining these parameters into one quality measurement, and a final correction step that uses an estimation of the sequence dependent regression, seems to be a promising way toward new objective video quality metrics. The presented methods are very generic, and can be used to design a wide variety of quality metrics, including full reference, reduced reference and no reference metrics, and including metrics that can work on a broader range of codecs and applications, and those that are tailored to one codec family or one limited quality range.

Objective video quality metrics are still a field of research, with only very few and limited results that found its way to the market. Every new objective video quality

metric, no matter if it was designed using the proposed method or any other approach, will have to present reliable results not only to the market but also to the academic world (even if the latter one is more open to "interesting" results that may leave some open questions). The proposed framework for the verification of video quality metrics, that consists of carefully designed subjective tests, disallowing the use of data fitting methods, and applying a proper statistical evaluation using only some basic statistical tools, shows one way to transparently communicate the results that can be gained with such objective quality metrics.

However, to compare coding approaches that are based on different basis techniques, for cases where a very precise quality comparison is needed, or for the design and verification of objective quality metrics, there is still no better, and sometimes no other, method than conducting a thoroughly designed and well executed subjective test, having a number of people sitting in front of a display, equipped with nothing more than a sheet of paper and a pencil.

Appendix A

Additional Tables and Figures

Table A.1: Test sequences used for CIF resolution

Sequence	Bit Rate [kbit/s]	Resolution	Frame Rate [fps]	Coding Structure	Visual Quality [a]
Bus	128	CIF	15.0	IPPPP	0.42
	256	CIF	15.0	IPPPP	0.56
	512	CIF	15.0	IPPPP	0.68
City	192	CIF	15.0	IBBP	0.51
	384	CIF	30.0	IBBP	0.76
	750	CIF	30.0	IBBP	0.81
Crew	192	CIF	15.0	IBBP	0.33
	384	CIF	30.0	IBBP	0.53
	750	CIF	30.0	IBBP	0.71
Football	96	CIF	15.0	IBBP	0.09
	192	CIF	15.0	IBBP	0.29
	256	CIF	15.0	IPPPP	0.30
	384	CIF	15.0	IBBP	0.44
	512	CIF	15.0	IPPPP	0.50
	768	CIF	15.0	IBBP	0.66
	1024	CIF	30.0	IPPPP	0.69
Foreman	96	CIF	15.0	IPPPP	0.26
	128	CIF	15.0	IPPPP	0.56
	192	CIF	15.0	IPPPP	0.53
	256	CIF	15.0	IPPPP	0.68
	384	CIF	15.0	IPPPP	0.79
	512	CIF	15.0	IPPPP	0.91
	768	CIF	15.0	IPPPP	0.91

[a] scaled to a 0 to 1 range

Table A.2: Test sequences used for CIF resolution (Part 2)

Sequence	Bit Rate [kbit/s]	Resolution	Frame Rate [fps]	Coding Structure	Visual Quality [a]
Harbour	192	CIF	15.0	IBBP	0.44
	384	CIF	30.0	IBBP	0.47
	750	CIF	30.0	IBBP	0.73
Head	96	CIF	15.0	IPPPP	0.54
	192	CIF	15.0	IPPPP	0.79
	384	CIF	15.0	IPPPP	0.84
	768	CIF	15.0	IPPPP	0.86
Husky	96	CIF	15.0	IBBP	0.10
	192	CIF	15.0	IBBP	0.25
	384	CIF	15.0	IBBP	0.51
	768	CIF	15.0	IBBP	0.72
Ice	192	CIF	15.0	IBBP	0.44
	384	CIF	30.0	IBBP	0.70
	750	CIF	30.0	IBBP	0.76
Mobile	96	CIF	15.0	IPPPP	0.40
	192	CIF	15.0	IBBP	0.66
	256	CIF	15.0	IPPPP	0.67
	384	CIF	15.0	IBBP	0.81
	512	CIF	15.0	IPPPP	0.71
	768	CIF	15.0	IBBP	0.87
	1024	CIF	30.0	IPPPP	0.81
Paris	96	CIF	15.0	IPPPP	0.38
	192	CIF	15.0	IPPPP	0.69
	384	CIF	15.0	IPPPP	0.86
	768	CIF	15.0	IPPPP	0.89
Tempete	96	CIF	15.0	IBBP	0.44
	192	CIF	15.0	IBBP	0.72
	384	CIF	15.0	IBBP	0.85
	768	CIF	15.0	IBBP	0.91
Zoom	96	CIF	15.0	IPPPP	0.51
	192	CIF	15.0	IPPPP	0.74
	384	CIF	15.0	IPPPP	0.78
	768	CIF	15.0	IPPPP	0.81

[a] scaled to a 0 to 1 range

Table A.3: Test sequences used for QCIF resolution

Sequence	Bit Rate [kbit/s]	Resolution	Frame Rate [fps]	Coding Structure	Visual Quality [a]
Bus	64	QCIF	15.0	IPPPP	0.60
City	64	QCIF	15.0	IBBP	0.63
	128	QCIF	30.0	IBBP	0.81
Crew	64	QCIF	15.0	IBBP	0.30
	128	QCIF	30.0	IBBP	0.56
Football	24	QCIF	10.0	IBBP	0.06
	48	QCIF	10.0	IBBP	0.25
	96	QCIF	10.0	IPPPP	0.40
	128	QCIF	15.0	IBBP	0.50
Foreman	24	QCIF	10.0	IPPPP	0.13
	48	QCIF	10.0	IPPPP	0.45
	64	QCIF	15.0	IPPPP	0.63
	96	QCIF	10.0	IPPPP	0.70
Harbour	64	QCIF	15.0	IBBP	0.37
	128	QCIF	30.0	IBBP	0.64
Head	24	QCIF	15.0	IPPPP	0.32
	48	QCIF	15.0	IPPPP	0.64
	96	QCIF	15.0	IPPPP	0.75
Husky	24	QCIF	7.5	IBBP	0.06
	48	QCIF	7.5	IBBP	0.22
	96	QCIF	7.5	IBBP	0.35
Ice	64	QCIF	15.0	IBBP	0.54
	128	QCIF	30.0	IBBP	0.71
Mobile	24	QCIF	7.5	IPPPP	0.22
	48	QCIF	7.5	IBBP	0.50
	96	QCIF	7.5	IPPPP	0.66
Paris	24	QCIF	10.0	IPPPP	0.11
	48	QCIF	10.0	IPPPP	0.40
	96	QCIF	10.0	IPPPP	0.64
Tempete	24	QCIF	10.0	IBBP	0.34
	48	QCIF	10.0	IBBP	0.54
	96	QCIF	10.0	IBBP	0.79
Zoom	24	QCIF	10.0	IPPPP	0.21
	48	QCIF	10.0	IPPPP	0.46
	96	QCIF	10.0	IPPPP	0.71

[a] scaled to a 0 to 1 range

Table A.4: Test sequences used for 4CIF resolution

Sequence	Bit Rate [kbit/s]	Resolution	Frame Rate [fps]	Coding Structure	Visual Quality [a]
City	1500	4CIF	30.0	IBBP	0.80
	3000	4CIF	60.0	IBBP	0.80
	6000	4CIF	60.0	IBBP	0.80
Crew	1500	4CIF	30.0	IBBP	0.68
	3000	4CIF	60.0	IBBP	0.68
	6000	4CIF	60.0	IBBP	0.76
Harbour	1500	4CIF	30.0	IBBP	0.54
	3000	4CIF	60.0	IBBP	0.60
	6000	4CIF	60.0	IBBP	0.70
Ice	1500	4CIF	30.0	IBBP	0.66
	3000	4CIF	60.0	IBBP	0.78
	6000	4CIF	60.0	IBBP	0.83

[a] scaled to a 0 to 1 range

(a) Bus

(b) City

(c) Crew

(d) Football

(e) Foreman

(f) Harbour

(g) Head

(h) Husky

Figure A.1: Used video sequences

(i) Ice

(j) Mobile

(k) Paris

(l) Tempete

(m) Zoom

Figure A.1: Used video sequences

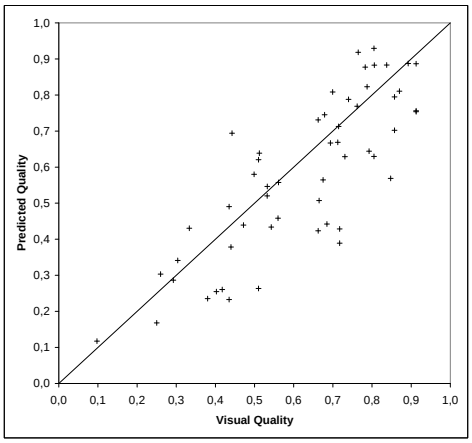


Figure A.2: CIF dataset: Results for PSNR+

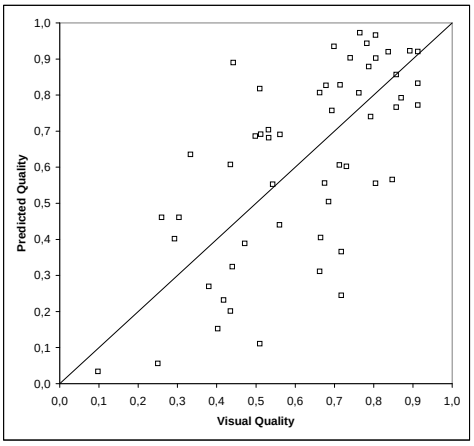


Figure A.3: CIF dataset: Results for PSNR

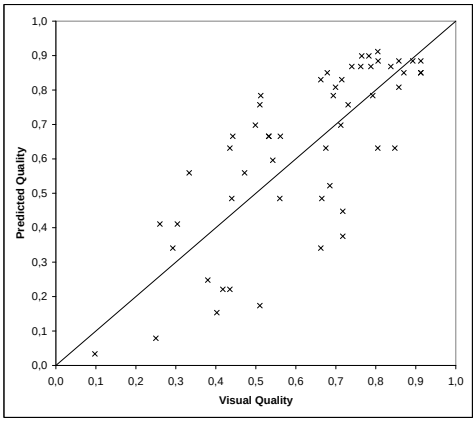


Figure A.4: CIF dataset: Results for Edge PSNR

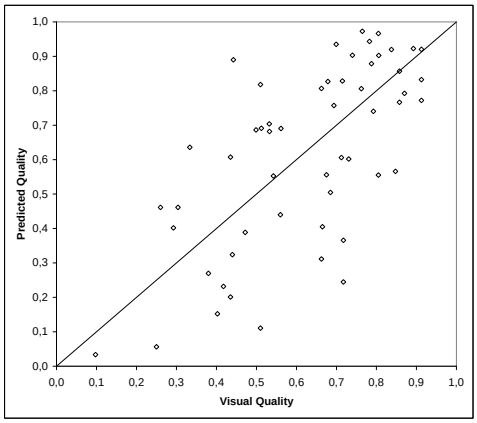


Figure A.5: CIF dataset: Results for SSIM

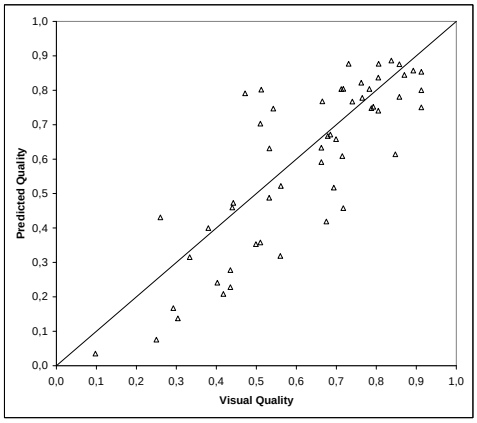


Figure A.6: CIF dataset: Results for the proposed method

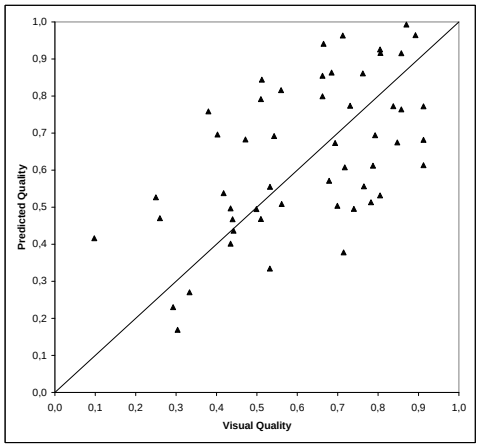


Figure A.7: CIF dataset: Results for the pure no reference method

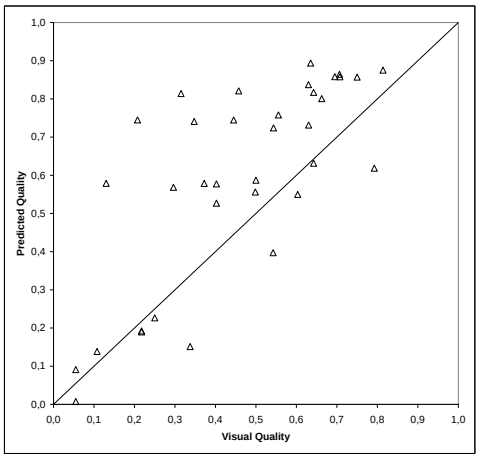


Figure A.8: QCIF dataset: Results for the proposed method

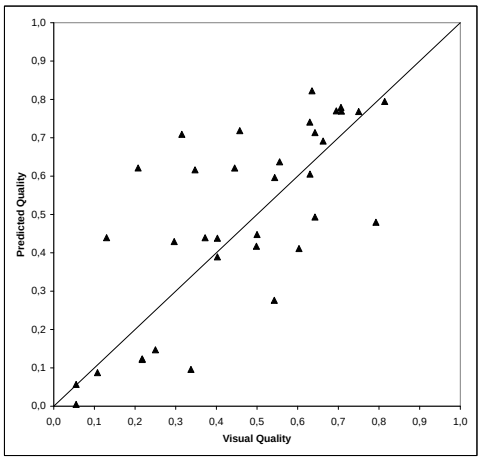


Figure A.9: QCIF dataset: Results for the proposed method (first order fitting)

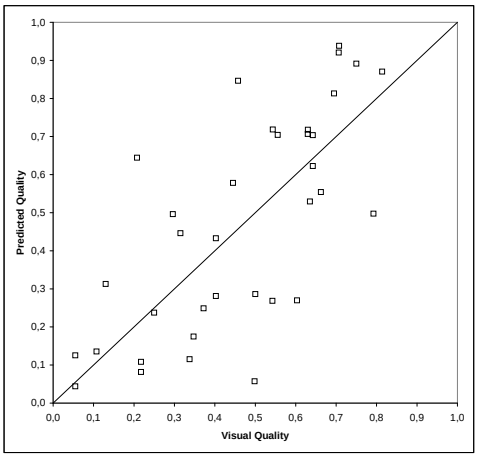


Figure A.10: QCIF dataset: Results for PSNR

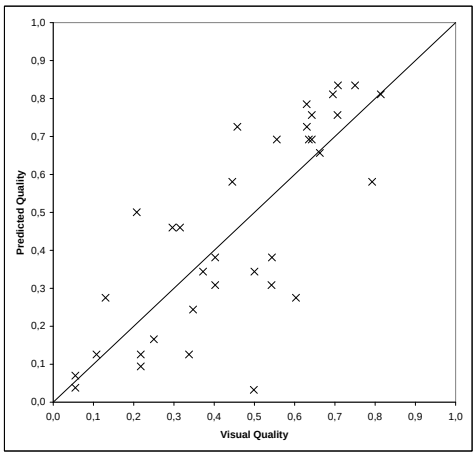


Figure A.11: QCIF dataset: Results for Edge-PSNR

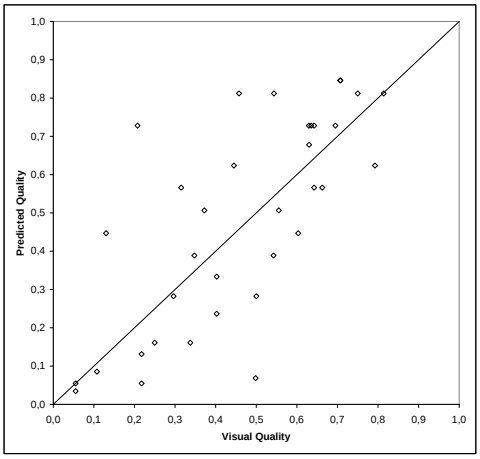


Figure A.12: QCIF dataset: Results for SSIM

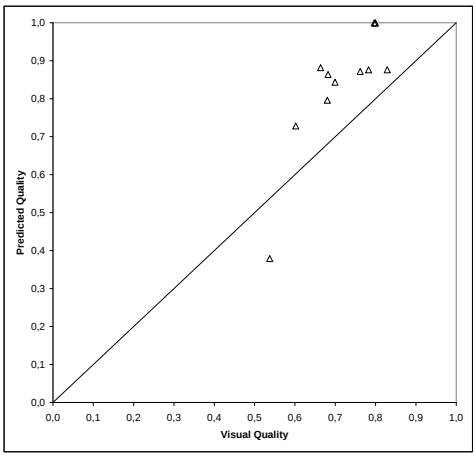


Figure A.13: 4CIF dataset: Results for the proposed method

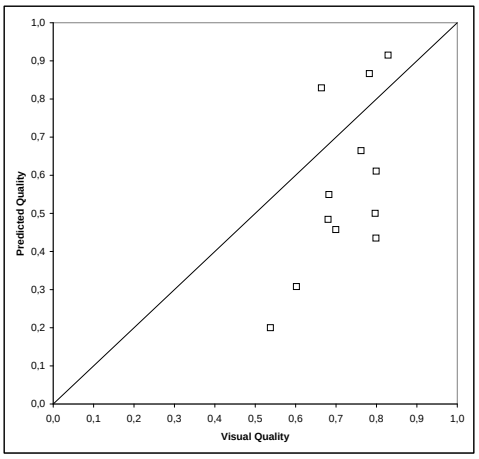


Figure A.14: 4CIF dataset: Results for PSNR

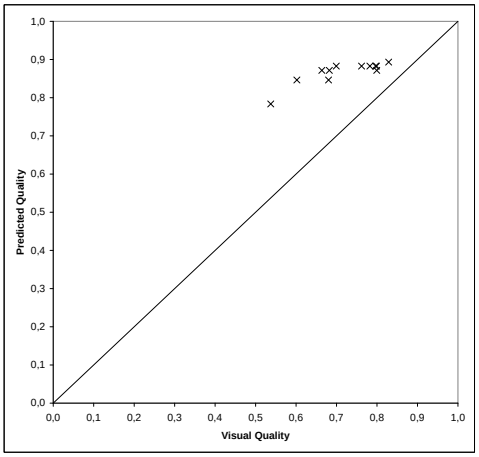


Figure A.15: 4CIF dataset: Results for Edge-PSNR

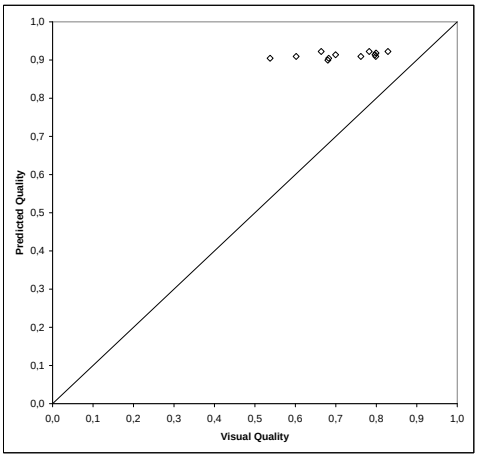


Figure A.16: 4CIF dataset: Results for SSIM

Bibliography

[1] B.Girod, *Digital images and human vision.* Cambridge, MA, USA: MIT Press, 1993, ch. What's wrong with mean-squared error?, pp. 207–220.

[2] VQEG, "Final report from the video quality experts group on the validation of objective models of video quality assessment," VQEG, Tech. Rep., Mar. 2000.

[3] VQEG, "Final report from the video quality experts group on the validation of objective models of video quality assessment, phase 2," VQEG, Tech. Rep., Aug. 2003.

[4] U.Benzler and M.Wien, "Results of SVC CE3 (quality evaluation)," ISO/IEC JTC1/SC29/WG11, Tech. Rep. M10931, Jul. 2004.

[5] O.Nemethova, M.Ries, E.Siffel, and M.Rupp, "Subjective evaluation of video quality for H.264 encoded sequences," in *Proc. IEEE Workshop on Mobile Future, 2004 and the Symposium on Trends in Communications (SympoTIC)*, Oct. 2004, pp. 191–194.

[6] 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.

[7] J.Hu, S.Choudhury, and J.D.Gibson, "$PSNR_{r,f}$: Assessment of delivered AVC/H.264 video quality over 802.11a WLANs with multipath fading," in *MultiComm 2006*, Jun. 2006.

[8] O.Kwon and C.Lee, "Objective method for assessment of video quality using wavelets," in *Proc. International Symposium on Industrial Electronics*, vol. 1, Jun. 2001, pp. 292–295.

[9] Z.Wang, A. C.Bovik, and E. P.Simoncelli, *Handbook of Image and Video Processing.* Academic Press, 2005, ch. Structural Approaches to Image Quality Assessment.

[10] Z.Zhe and H. R.Wu, "A ratio sensitive image quality metric," in *Proc. International Symposium on Intelligent Signal Processing and Communication Systems*, Nov. 2004, pp. 362–364.

[11] T. M.Kusuma, H.-J.Zepernick, and M.Caldera, "On the development of a reduced-reference perceptual image quality metric," in *Proc. IEEE Conference on Systems Communications*, vol. 1, Aug. 2005, pp. 178–184.

[12] Z.Wang, G.Wu, H. R.Sheikh, E.P.Simoncelli, E.Yang, and A. C.Bovik, "Quality-aware images," *IEEE Trans. Image Process.*, vol. 15, no. 6, pp. 1680–1689, Jun. 2006.

[13] H. R.Sheikh, A. C.Bovik, and G.deVeciana, "An information fidelity criterion for image quality assessment using natural scene statistics," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2117–2128, Dec. 2005.

[14] Z.Wang and E.P.Simoncelli, "Reduced-reference image quality assessment using a wavelet-domain natural image statistic model," in *Proc. SPIE Human Vision and Electronic Imaging X*, vol. 566, Jan. 2005.

[15] 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.

[16] 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.

[17] J.Lubin, *Visual Models for Target Detection and Recognition.* World Scientific Publishers, 1995, ch. A visual discrimination mode for imaging system design and evaluation.

[18] S.Daly, *Digital Images and Human Vision.* MIT Press, 1993, ch. The visible differences predictor: An algorithm for the assessment of image fidelity.

[19] A. B.Watson, J.Hu, and J. F. I.McGowan, "Digital video quality metric based on human vision," *Journal of Electronic Imaging*, vol. 10, no. 1, pp. 20–29, 2001.

[20] A. A.Webster, C. T.Jones, M. H.Pinson, S. D.Voran, and S.Wolf, "An objective video quality assessment system based on human perception," in *SPIE Proc. of Human Vision, Visual Processing and Digital Display*, vol. 1913, 1993, pp. 15–26.

[21] M.Carnec, P. L.Callet, and D.Barba, "An image quality assessment method based on perception of structural information," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 3, Sep. 2003, pp. 185–188.

[22] 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.

[23] 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.

[24] M.Masry, S.S.Hemami, and Y.Sermadevi, "A scalable wavelet-based video distortion metric and applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 2, pp. 260–273, Feb. 2006.

[25] G.Zhai, W.Zhang, Y.Xu, and X.Yang, "Image quality assessment metrics based on multi-scale edge presentation," in *Proc. IEEE Workshop on Signal Processing Systems Design and Implementation*, Nov. 2005, pp. 331–336.

[26] G.Zhai, W.Zhang, X.Yang, and Y.Xu, "GES: a new image quality assessment metric based on energy features in gabor transform domain," in *Proc. IEEE International Symposium on Circuits and Systems,(ISCAS)*, May 2006, p. 4 pp.

[27] H.Zhu and H.Wu, "New paradigm for compressed image quality metric: exploring band similarity with CSF and mutual information," in *Proc IEEE Conference on Geoscience and Remote Sensing Symposium 2005*, vol. 2, Jul. 2005, p. 4pp.

[28] A.Beghdadi and B.Pesquet-Popescu, "A new image distortion measure based on wavelet decomposition," in *Proc. Seventh International Symposium on Signal Processing and Its Applications*, vol. 1, Jul. 2003, pp. 485–488.

[29] D.Gayle, H.Mahlab, Y.Ucar, and A. M.Eskicioglu, "A full-reference color image quality measure in the DWT domain," in *Proc. European Signal Processing Conference*, Sep. 2005.

[30] Y.Horita, M.Sato, Y.Kawayoke, P.Sazzad, and K.Shibata, "Image quality evaluation model based on local features and segmentation," in *Proc. IEEE International Conference on Image Processing 2006, (ICIP)*, 2006, pp. 405–408.

[31] *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.

[32] E.Ong, W.Lin, Z.Lu, and S.Yao, "Colour perceptual video quality metric," in *Proc. IEEE International Conference on Image Processing (ICIP)*, Sep. 2005, pp. III: 1172–1175.

[33] P.Le Callet, C.Viard-Gaudin, and D.Barba, "A convolutional neural network approach for objective video quality assessment," *IEEE Trans. Neural Netw.*, vol. 17, no. 5, pp. 1316–1327, Sep. 2006.

[34] P.Gastaldo, S.Rovetta, R.Zunino, and I.Heynderickx, "CBP neural network for objective assessment of image quality," in *Proc. IEEE International Conference on Neural Networks*, vol. 1, Jul. 2003, pp. 194–199.

[35] T.Morris, K.Angus, R.Butt, A.Chilton, P.Dettman, and S.McCoy, "CQA - subjective video codec quality analyser," in *BMVC99*, 1999.

[36] Y.-B.Tong, Q.Chang, and Q.-S.Zhang, "Image quality assessing by using NN and SVM," in *Proc. Fifth International Conference on Machine Learning and Cybernetics*, Aug. 2006, pp. 3987–3990.

[37] *T1.TR.74-2001. Objective video quality measurement using a peak-signal-to-noise-ratio (PSNR) full reference technique*, ANSI Std., 2001.

[38] *ITU-R BT.1683. Objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference*, ITU-R Std., Jun. 2004.

[39] H. R.Sheikh, M.Sabir, and A. C.Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3440–3451, Nov. 2006.

[40] J.Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, 1986.

[41] K.Egiazarian, J.Astola, N.Ponomarenko, V.Lukin, F.Battisti, and M.Carli, "Two new full-reference quality metrics based on HVS," in *Second Interna-*

tional Workshop on Video Processing and Quality Metrics for Consumer Electronics, Jan. 2006.

[42] 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

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

[44] Z.Wang and A. C.Bovik, "A universal image quality index," *IEEE Signal Process. Lett.*, vol. 9, no. 3, pp. 81–84, Mar. 2002.

[45] Z.Wang, A. C.Bovik, H. R.Sheikh, and E. P.Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[46] L.Lu, Z.Wang, and A. C.Bovik, "Full-reference video quality assessment considering structural distortion and no-reference quality evaluation of MPEG video," in *Proc. IEEE International Conference on Multimedia and Expo*, vol. 1, Aug. 2002, pp. 61–64.

[47] 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.

[48] Z.Wang, A. C.Bovik, and E. P.Simoncelli, "Multi-scale structural similarity for image quality assessment," in *37th IEEE Asilomar Conference on Signals, Systems and Computers*, vol. 2, Nov. 2003, pp. 1398–1402.

[49] Z.Wang and E. P.Simoncelli, "An adaptive linear system framework for image distortion analysis," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 3, Sep. 2005, pp. 1160–3.

[50] G.-H.Chen, C.-L.Yang, L.-M.Po, and S.-L.Xie, "Edge-based structural similarity for image quality assessment," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, May 2006, pp. 933–936.

[51] A.Medda and V.DeBrunner, "Color image quality index based on the UIQI," in *Proc. IEEE Southwest Symposium on Image Analysis and Interpretation*, Mar. 2006, pp. 213–217.

[52] *T1.801.03-1996, 2003. Digital Transport of One-Way Video Signals - Parameters for Objective Performance Assessment*, ANSI Std., 2003.

[53] H. R.Sheikh and A. C.Bovik, "A visual information fidelity approach to video quality assessment," in *First International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2005.

[54] K.Seshadrinathan and A. C.Bovik, "Statistical video models and their application to quality assessment," in *Second International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2006.

[55] K.Seshadrinathan and A. C.Bovik, "An information theoretic video quality metric based on motion models," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[56] M. H.Pinson and S.Wolf, "Low bandwidth reduced reference video quality monitoring system," in *First International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2005.

[57] F.Meng, X.Jiang, H.Sun, and S.Yang, "Objective perceptual video quality measurement using a foveation-based reduced reference algorithm," in *IEEE International Conference on Multimedia and Expo*, Jul. 2007, pp. 308–311.

[58] M.Carnec, P.Le Callet, and D.Barba, "Visual features for image quality assessment with reduced reference," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 1, Sep. 2005, pp. 421–424.

[59] 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.

[60] 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.

[61] M.Ries, O.Nemethova, and M.Rupp, "Reference-free video quality metric for mobile streaming applications," in *Proceedings of the DSPCS 05 & WITSP 05*, Sunshine Coast, Australia, Dec. 2005, pp. 1–5.

[62] M.Ries, O.Nemethova, and M.Rupp, "Motion based reference-free quality estimation for H.264/AVC video streaming," in *Proceedings of International Symposium on Wireless Pervasive Computing 2007*, Feb. 2007.

[63] P.Gastaldo, S.Rovetta, and R.Zunino, "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.

[64] 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.

[65] 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.

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

[67] 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.

[68] 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.

[69] 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.

[70] G.Luo, "Image noise analysis with a fast lifting wavelet algorithm for objective image quality evaluation," in *Proc. IEEE Third International Conference on Image and Graphics*, Dec. 2004, pp. 39–42.

[71] H. R.Sheikh, A. C.Bovik, and L.Cormack, "No-reference quality assessment using natural scene statistics: JPEG2000," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1918–1927, Nov. 2005.

[72] H. R.Sheikh, Z.Wang, A. C.Bovik, and L.Cormack, "Blind quality assessment of JPEG2000 compressed images," in *Conference Record of the Thirty-Sixth*

Asilomar Conference on Signals, Systems and Computers, vol. 2, Nov. 2002, pp. 1403–1407.

[73] H. R.Sheikh, A. C.Bovik, and L.Cormack, "Blind quality assessment of JPEG2000 compressed images using natural scene statistics," in *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 2, Nov. 2003, pp. 1403–1407.

[74] K.-C.Yang, C. C.Guest, P. K.Das, and K.El-Maleh, "Perceptual temporal quality metric for compressed video," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[75] A.Younkin, R.Fernald, R.Doherty, E.Salskov, and P.Corriveau, "Predicting an average end-users experience of video playback," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[76] R. R.Pastrana-Vidal and J.-C.Gicquel, "Automatic quality assessment of video fluidity impairments using a no-reference metric," in *Second International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2006.

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

[78] *ITU-R BT.710 Subjective assessment methods for image quality in high-definition television*, ITU-R Std., Rev. 4, Nov. 1998.

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

[80] J.Beerends and F.deCaluwe, "The influence of video quality on perceived audio quality and vice versa," *Journal of the Audio Engineering Society*, vol. 47, no. 5, pp. 355–362, 1999.

[81] S.Winkler and C.Faller, "Audiovisual Quality Evaluation of Low-Bitrate Video," in *IS&T/SPIE International Symposium Electronic Imaging*, vol. 5666, 2005, pp. 139–148.

[82] D. S.Hands, "A basic multimedia quality model," *IEEE Trans. Multimedia*, vol. 6, no. 6, pp. 806–816, Dec. 2004.

[83] S.Jumisko-Pyykkö and J.Häkkinen, "Evaluation of subjective video quality of mobile devices," in *Proc. of the 13th annual ACM international conference on Multimedia*, Nov. 2005, pp. 535–538.

[84] ITU-R, "Question ITU-R 67/6 methodologies for subjective assessment of audio and video quality," 1999.

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

[86] T.Alpert and J.-P.Evain, "Subjective quality evaluation the SSCQE and DSCQE methodologies," EBU, Tech. Rep., 1997.

[87] S.Winkler and R.Campos, "Video quality evaluation for Internet streaming applications," in *Proc. IS&T/SPIE Electronic Imaging 2003: Human Vision and Electronic Imaging VIII*, vol. 5007, 2003, pp. 104–115.

[88] S.Winkler and F.Dufaux, "Video quality evaluation for mobile applications," in *SPIE/IS&T Visual Communications and Image Processing Conference*, vol. 5150, 2003, pp. 593–603.

[89] T.Alpert and L.Contin, "DSCQE (double stimulus using a continuous quality evaluation) experiment for the evaluation of the MPEG-4 VM on error robustness functionality," ISO/IEC JTC1/SC29/WG11, Tech. Rep. M1604, Feb. 1997.

[90] J. L.Blin, "New quality evaluation method suited to multimedia context SAMVIQ," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[91] Q.Huynh-Thua, M.Brotherton, D.Hands, K.Brunnström, and M.Ghanbari, "Examination of the SAMVIQ methodology for the subjective assessment of multimedia quality," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[92] M. D.Brotherton, Q.Huynh-Thu, D. S.Hands, and K.Brunnström, "Subjective multimedia quality assessment," *IEICE Transaction Fundamentals*, vol. E89A, no. 11, pp. 2920–2932, Nov. 2006.

[93] Q.Huynh-Thu and M.Ghanbari, "A comparison of subjective video quality assessment methods for low-bit rate and low-resolution video," in *Proc. IASTED Intl Conf. on Signal and Image Processing*, Aug. 2005, pp. 70–76.

[94] Q.Huynh-Thu, M.Ghanbari, D.Hands, and M.Brotherton, "Subjective video quality evaluation for multimedia applications," in *SPIE Proc. of Human Vision and Electronic Imaging XI*, vol. 6057, Feb. 2006, pp. 15–26.

[95] C.Poynton, "The rehabilitation of gamma," in *Proceedings of SPIE Human Vision and Electronic Imaging*, Jan. 1998, pp. 232–249.

[96] C.Poynton. (2000) Brightness and contrast controls. [Online]. Available: http://www.poynton.com/notes/brightness_and_contrast/index.html

[97] M. H.Pinson and S.Wolf, "The impact of monitor resolution and type on subjective video quality testing," NTIA, Tech. Rep., 2004.

[98] S.Tourancheau, P.Le Callet, K.Brunnström, and D.Barba, "Display awareness in subjective and objective video quality evaluation," in *Proc. European Signal Processing Conference EUSIPCO*, Sep. 2007, pp. 164–168.

[99] D. S.Hands, M.Brotherton, A.Bourret, and D.Bayart, "Subjective quality assessment for objective quality model development," *IEEE Electronic Letters*, vol. 41, no. 7, pp. 408–409, 2005.

[100] H. R.Lindman, *Analysis of variance in complex experimental designs.* W. H. Freeman & Co, 1974.

[101] W.Xu and G.Hauske, "Perceptually relevant error classification in the context of picture coding," in *Proc. IEEE International Conference on Image Processing and its Applications*, Jul. 1995, pp. 589–593.

[102] D.Costantini, C. J.van den Branden Lambrecht, G.Sicuranza, and M.Kunt, "Motion rendition quality metric for MPEG coded video," in *Proc. IEEE International Conference on Image Processing*, Sep. 1996, pp. 889–892.

[103] W.Xu and G.Hauske, "Picture quality evaluation based on error segmentation," in *Proc. SPIE Visual Communications and Image Processing*, ser. Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, A. K.Katsaggelos, Ed., vol. 2308, Sep. 1994, pp. 1454–1465.

[104] S. J. P.Westen, R. L.Lagendijk, and J.Biemond, "Perceptual image quality based on a multiple channel HVS model," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 4, 1995, pp. 2351–2354. [Online]. Available: citeseer.ist.psu.edu/westen95perceptual.html

[105] N.Suresh and N.Jayant, "Mean time between failures: A subjectively meaningful video quality metric," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2006.

[106] N.Jayant and N.Suresh, "Objective measurement of video quality: Prediction of mean time between failures," in *National Association of Broadcasters*, Apr. 2006.

[107] A.Shnayderman, A.Gusev, and A.Eskicioglu, "An SVD-Based grayscale image quality measure for local and global assessment," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 422–429, Feb. 2006.

[108] 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.

[109] 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.

[110] K.Hermiston and D.Booth, "Image quality measurement using integer wavelet transformations," in *Proc. IEEE International Conference on Image Processing*, vol. 2, Oct. 1999, pp. 293–297.

[111] F.Chin and C.Xydeas, "Dual-mode image quality assessment metric," in *EURASIP-IEEE International Symposium on Video/Image Processing and Multimedia Communications*, Jun. 2002, pp. 137–140.

[112] Y.Jia, W.Lin, and A. A.Kassim, "Estimating just-noticeable distortion for video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 7, pp. 820–829, Jul. 2006.

[113] C.Perra, F.Massidda, and D.Giusto, "Image blockiness evaluation based on sobel operator," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 1, Sep. 2005, pp. 389–392.

[114] A. R.Reibman and A.Leontaris, "Comparison of blocking and blurring metrics for video compression," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2006.

[115] A.Leontaris and A. R.Reibman, "Measuring the added high frequency energy in compressed video," in *Proc. IEEE International Conference on Image Processing (ICIP)*, vol. 2, Sep. 2005, pp. 498–501.

[116] H.-H.Ho, T.Wolff, M.Salatino, J. M.Foley, S. K.Mitra, T.Yamada, and H.Harasaki, "An investigation on the subjective quality of H.264 compressed/decompressed videos," in *Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2007.

[117] 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.

[118] 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.

[119] 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.

[120] H. R.Sheikh and A. C.Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[121] A. B.Watson, "Toward a perceptual video-quality metric," B. E.Rogowitz and T. N.Pappas, Eds., vol. 3299, no. 1. SPIE, 1998, pp. 139–147.

[122] M.Carnec and D.Barba, "Simulating the human visual system: towards objective measurement of visual annoyance," in *Proc. of the 2002 IEEE International Conference on Systems, Man and Cybernetics*, vol. 6, Oct. 2002, p. 6 pp.

[123] D. J.Simons and C.Chabris, "Gorillas in our midst: Sustained inattentional blindness for dynamic events," *Perception*, vol. 28, pp. 1059–1074, 1999.

[124] D. T.Levin and D. J.Simons, "Failure to detect changes to attended objects in motion pictures," *Psychonomic Bulletin and Review*, vol. 4, pp. 501–506, 1997.

[125] S.Yao, W.Lin, S.Rahardja, E.Ong, and Z.Lu, "Video quality evaluation based on wavelet visible difference measure," in *Second International Workshop on Video Processing and Quality Metrics for Consumer Electronics*, Jan. 2006.

[126] W.Zia, T.Oelbaum, and K.Diepold, "Subjective evaluation of error control strategies for mobile video communication," in *Proc. Picture Coding Symposium*, Nov. 2007.

[127] W.Gao, C.Mermer, and Y.Kim, "A de-blocking algorithm and a blockiness metric for highly compressed images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, pp. 1150–1159, Dec. 2002.

[128] K.Tan and M.Ghanbari, "Measuring blocking artefacts using harmonic analysis," *IEEE Electronic Lett.*, vol. 35, pp. 1322–1323, Aug. 1999.

[129] Z.Zhe, H. R.Wu, Z.Yu, T.Ferguson, and D.Tan, "Performance evaluation of a perceptual ringing distortion metric for digital video," in *Proc. International Conference on Multimedia and Expo (ICME)*, vol. 1, Jul. 2003, pp. 825–828.

[130] M. C.Farias, S. K.Mitra, and J. M.Foley, "Perceptual contributions of blocky, blurry and noisy artifacts to overall annoyance," *IEEE Signal Process. Lett.*, vol. 12, no. 10, pp. 685–688, Oct. 2005.

[131] *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.

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

[133] 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/

[134] 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.

[135] T.Oelbaum and K.Diepold, "Building a reduced reference video quality metric with very low overhead using multivariate data analysis," in *Proc. Int. Conf. on Cybernetics and Information Technologies, Systems and Applications CITSA*, Jul. 2007.

[136] 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.

[137] H.Wold, *Multivariate Analysis.* Academic Press, 1996, ch. Estimation of principal components and related models by iterative least squares, pp. 391–420.

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

All internet links were last checked in November 2007.

List of Figures

List of Tables