



INSTITUTO
SUPERIOR
TÉCNICO

UNIVERSIDADE TÉCNICA DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

No-reference Image and Video Quality Assessment

Tomás Gomes da Silva Serpa Brandão
(Mestre)

Dissertação para obtenção do Grau de Doutor em
Engenharia Electrotécnica e de Computadores

Orientador: Doutora Maria Paula dos Santos Queluz Rodrigues

Júri

Presidente: Presidente do Conselho Científico do IST

Vogais: Doutor Mário Alexandre Teles de Figueiredo
Doutor Leonel Augusto Pires Seabra de Sousa
Doutor Luís António Pereira de Meneses Corte-Real
Doutor Paulo Jorge Lourenço Nunes
Doutora Maria Paula dos Santos Queluz Rodrigues

Junho de 2011

Abstract

Image and video quality assessment has become an increasingly important subject in digital video coding and transmission scenarios, such as digital television. In this context, a special interest has been put on no-reference objective quality assessment metrics, since they are suitable for real-time quality monitoring once the video delivery system is settled. This Thesis proposes new no-reference quality assessment metrics for images and video. The main goal of the proposed techniques is to estimate the quality of lossy DCT-based encoded video.

The proposed metrics share the same key idea: based on elements extracted from the bitstream of the encoded images or video arriving at the point where quality assessment has to be performed, an estimate of the quantization error associated to each DCT coefficient is obtained. Those estimates are perceptually weighted and combined in order to obtain a quality score for the image or video under analysis. The Thesis starts by proposing a technique based on watermarking, that evolves to a technique based on natural image statistics only. The results produced by the metrics are close to and well correlated with subjective quality assessment data.

Keywords:

No-reference quality assessment, image and video coding, perceptual models, parameter estimation, DCT coefficient statistics, watermarking.

Resumo

A avaliação da qualidade de imagem e vídeo é um assunto que tem tido uma importância crescente em cenários que envolvem a codificação e a transmissão de vídeo em formato digital, como é o caso da televisão digital. Dentro deste contexto, são de especial interesse as métricas objectivas que avaliam a qualidade de vídeo sem recorrer a sinais de referência, pois permitem que seja feita uma monitorização da qualidade em tempo real, tanto nos receptores, como em pontos intermédios da rede. Nesta Tese são propostos novos algoritmos para avaliação automática da qualidade de imagens fixas e de vídeo, sem utilização de sinais de referência. As técnicas propostas têm como objectivo estimar a qualidade de uma imagem ou um vídeo codificado com perdas e com base na transformada DCT.

As métricas propostas partilham a mesma ideia chave: usando o fluxo binário da imagem ou vídeo codificado que chega ao ponto da rede onde se pretende analisar a qualidade, é feita uma estimativa do erro de quantização associado a cada coeficiente DCT. Essas estimativas para o erro são pesadas, tendo em conta factores perceptuais, e combinadas entre si, de modo a ser obtido um valor de qualidade para a imagem ou para o vídeo em análise. Começa-se por propor uma técnica baseada num sistema de marcas d'água, que evolui ao longo da Tese para uma técnica que se baseia exclusivamente num modelo estatístico associado a imagens naturais. Os resultados obtidos exibem valores próximos e bem correlacionados com dados provenientes de avaliações subjectivas.

Palavras-chave:

Avaliação de qualidade sem referência, codificação de imagem e vídeo, modelos perceptuais, estimação de parâmetros, estatística dos coeficientes DCT, marcas d'água.

Acknowledgments

During the course of the Thesis, several people contributed to the work that is now being presented. In the following lines, I wish to express my gratitude to them.

First of all, I would like to thank my supervisor, professor Maria Paula Queluz, who was always available to discuss and propose new ideas, reviewed the Thesis and provided several constructive suggestions, helped and participated in the realization of the subjective quality assessment tests, motivated me for writing journal and conference papers, and allowed me to become a co-supervisor of Msc. Thesis in the image quality assessment field. For all that support, I am very grateful.

I would also like to thank the elements of the Image Group at IT, for all their support and knowledge exchange. I wish to give a special thanks to professor Fernando Pereira, for his support on the bureaucracy related with trip funding, for allowing me to participate in the organizing committee of the Picture Coding Symposium, and for giving me opportunities to present my work to expert audiences. I would also like to give a special thanks to professor Paulo Correia, for inviting me to submit and present a paper in a special session on image quality during the Eusipco 2007 conference.

To doctor Martijn Kuipers, for knowledge exchange about Latex, C++ and Matlab, as well as company on many morning coffee breaks.

A big thanks to the people from Germany that helped or discussed ideas with me during the Thesis period: doctor Tobias Oelbaum for providing me a set of video bitstreams that have been subject to quality evaluation, for testing my algorithms, as well as for the good and profitable discussions during the Thesis time; doctor Arnd Eden, for providing enough details for a correct implementation of his PSNR estimation algorithm; and to master engineer Sören Sofkle for the discussions about DCT coefficient modeling on H.264.

I also thank doctor Ahmid Sheik from USA, for providing me access to the LIVE

image quality database.

To doctor Matteo Naccari, currently working in the Image Group, for discussions about quality assessment and for providing a very nice reception on my trip to Italy after the EUSIPCO 2008 conference in Switzerland.

To professor Manuel Sequeira, for providing me access to the Matlab toolbox “PSM Tools” during the early stages of the Thesis. “PSM Tools” is a set of useful image processing related Matlab scripts.

To Ricardo Ribeiro, Fernando Batista and Marco Ribeiro, for mutual Thesis motivation. To João Ascenso and Catarina Brites, for useful discussions about software and the H.264 standard, as well as company for many lunches.

To Luís Roque, for his help on obtaining subjective quality data, as well as all the people (mostly students) who participated in the subjective quality assessment sessions.

A word of gratitude to the directors (during the my PhD’s time) of the department of information sciences and technologies from ISCTE-IUL: professors Carlos Sá da Costa and Francisco Cercas, who authorized periods of work exclusively dedicated to the investigation that led to the Thesis.

Finally, to Márcia and my daughter Catarina for constantly remembering me that there is also an important and joyful life outside the research world.

Contents

Abstract	iii
Resumo	v
Acknowledgments	vii
1 Introduction	1
1.1 Context and motivation	1
1.2 Main contributions	4
1.3 Outline of the Thesis	6
2 Image quality overview	9
2.1 What is image quality?	9
2.2 Fundamentals of human vision	12
2.2.1 The mechanics of the eye	12
2.2.2 Luminance adaptation and contrast sensitivity	13
2.2.3 Masking	16
2.2.4 Pooling	18
2.3 Image and video encoding	18
2.3.1 Compression of visual information	18
2.3.2 JPEG	20
2.3.3 JPEG2000	22
2.3.4 MPEG-2	23

2.3.5	H.264	26
2.4	Compression artifacts	28
2.4.1	Block effect	29
2.4.2	Blur	30
2.4.3	Ringing	31
2.4.4	Mosquito noise	32
2.4.5	Jitter and jerkiness	32
2.5	Transmission losses	33
2.6	Summary	35
3	Subjective quality assessment	37
3.1	Introduction	37
3.2	Subjective test preparation	38
3.2.1	Selection of test video sequences	38
3.2.2	Selection of test participants	39
3.2.3	Environment conditions	40
3.3	Standardized methodologies	40
3.3.1	<i>Double stimulus</i> methods	40
3.3.2	<i>Single stimulus</i> methods	42
3.3.3	<i>Comparison</i> methods	43
3.3.4	<i>Continuous quality evaluation</i> methods	43
3.4	Computing mean opinion scores	44
3.5	Subjective quality assessment tests	46
3.5.1	Methodology	46
3.5.2	Assessment conditions	47
3.5.3	Selection of test material	47
3.5.4	MOS computation	49
3.6	Summary	52

4	Objective quality assessment metrics	53
4.1	Introduction	53
4.2	Classifying objective quality metrics	53
4.3	Data metrics	57
4.4	Picture metrics	60
4.4.1	Psychophysical-based metrics	61
4.4.2	Artifact measurement metrics	63
4.4.3	Feature-based metrics	69
4.5	Bitstream-based metrics	70
4.5.1	Packet-oriented metrics	71
4.5.2	PSNR estimation algorithms	71
4.6	Standardization of objective metrics	72
4.7	Performance of an objective metric	74
4.8	Summary	76
5	Image quality assessment using watermarking	79
5.1	Introduction	79
5.2	Watermarking scheme	83
5.2.1	Watermark embedding	83
5.2.2	Watermark extraction	84
5.2.3	Perceptually adapted quantization functions	85
5.2.4	Choosing the DCT coefficient set for watermark embedding	88
5.3	Error estimation	93
5.3.1	Distance weighting based on watermark bit error rate	95
5.3.2	Distance weighting based on DCT coefficient statistics	97
5.3.3	Quality estimation	100
5.4	Results	101
5.4.1	PSNR estimation	101
5.4.2	Quality scores	103

5.5	Summary	105
6	Statistical image quality assessment	107
6.1	Introduction	107
6.2	Algorithm overview	109
6.3	Modeling DCT coefficient data	111
6.3.1	Parameter estimation using original coefficient data	112
6.3.2	Parameter estimation using quantized coefficient data	112
6.4	Parameter estimation using prediction	113
6.4.1	Predictor training procedure	115
6.4.2	Prediction accuracy	116
6.5	Perceptual quality estimation	119
6.6	Results	120
6.6.1	PSNR estimation	120
6.6.2	Quality scores	121
6.7	Summary	123
7	Perceptual video quality assessment	125
7.1	Introduction	125
7.2	Algorithm overview	127
7.3	Modeling DCT coefficient data	129
7.3.1	Cauchy model	129
7.3.2	Laplace model	131
7.3.3	Improving estimation using prediction	133
7.3.4	Predictor training	135
7.4	Perceptual model	137
7.4.1	Spatio-temporal CSF model	137
7.4.2	Quality scores	139
7.5	Results	139

7.5.1	Prediction accuracy	140
7.5.2	PSNR estimation	142
7.5.3	Quality assessment	144
7.5.4	Comparison and discussion	146
7.6	Summary	147
8	Conclusion	149
	Bibliography	164

List of Figures

2.1	A good quality image, or not?	10
2.2	Diagram of the human eye.	13
2.3	Sensitivity of the HVS to light intensity changes.	14
2.4	Spatial contrast sensitivity function.	15
2.5	Spatio-temporal contrast sensitivity function.	16
2.6	Masking effect example.	17
2.7	Contrast masking and facilitation.	17
2.8	JPEG encoder and decoder schemes.	21
2.9	Example JPEG quantization tables.	22
2.10	MPEG-2 generic encoder and decoder schemes.	24
2.11	Default MPEG-2 quantization tables.	25
2.12	H.264 generic encoding and decoding schemes.	26
2.13	Blocking effect.	29
2.14	Blur effect.	30
2.15	Ringing effect.	31
2.16	Packet-oriented video transmission.	33
2.17	Packet loss effect.	34
3.1	Examples of typical test sequences.	39
3.2	Subject screening tools.	40
3.3	Double stimulus quality assessment method.	41

3.4	Single stimulus quality assessment method.	42
3.5	Comparison quality assessment.	43
3.6	Continuous quality evaluation.	44
3.7	Video sequences selected for the subjective tests.	49
3.8	Spatio-temporal activity of the selected video sequences.	50
3.9	MOS values resulting from the subjective quality assessment tests. . .	50
4.1	Full reference quality assessment system.	54
4.2	No-reference quality assessment system.	55
4.3	Reduced reference quality assessment system.	56
4.4	Images with similar PSNR, but different perceptual quality impact. .	58
4.5	Ideal block signal and the corresponding difference signal.	64
4.6	<i>Flatness</i> effect caused by low bitrate JPEG encoding.	65
4.7	Blur effect on image edges.	67
4.8	Objective image quality assessment classification.	77
5.1	Watermarking-based image quality assessment system.	80
5.2	Watermark embedding scheme.	83
5.3	Watermark extraction scheme.	84
5.4	Perceptually adapted distance between quantization points.	87
5.5	Sketch of the quantization function.	88
5.6	Watermark embedding coefficient sets.	89
5.7	Test images used for DCT coefficient set selection.	90
5.8	<i>Increasing frequencies</i> test results.	91
5.9	<i>Middle frequencies</i> test results.	91
5.10	DCT coefficient set used for watermark embedding.	92
5.11	PSNR computed using the set of selected DCT coefficients.	92
5.12	Error estimation in the presence of small distortion.	93
5.13	<i>False positive</i>	94

5.14	<i>False negative.</i>	94
5.15	Different error possibilities.	95
5.16	False positive/negative rates as a function of the W_{ber} .	96
5.17	Error estimation.	98
5.18	PSNR estimation examples – error weighting based on W_{ber} .	102
5.19	PSNR estimation examples – statistical error weighting.	103
5.20	Global PSNR estimation results.	103
5.21	DMOS estimation results.	105
6.1	General scheme of the proposed quality assessment algorithm.	109
6.2	An example histogram of the DCT coefficients.	112
6.3	Original λ values for each frequency.	114
6.4	Neighborhood configuration used for prediction.	116
6.5	λ estimation results for a JPEG encoded test image (<i>lighthouse</i>).	118
6.6	Global PSNR estimation results.	121
6.7	Best and worst case PSNR estimates.	122
6.8	DMOS estimation results.	123
7.1	Architecture of the proposed algorithm for video quality assessment.	127
7.2	Typical evolution of the H.264 coefficients' distribution parameter.	134
7.3	Neighborhood configuration used in the experiments.	136
7.4	Example of parameter estimation on H.264.	142
7.5	No-reference PSNR estimation <i>vs.</i> true PSNR – training set.	143
7.6	No-reference PSNR estimation <i>vs.</i> true PSNR – test set.	144
7.7	Temporal evolution of PSNR estimates.	145
7.8	MOS estimation results.	146

List of Tables

2.1	Base quantization steps.	28
3.1	Environmental viewing conditions.	47
3.2	Resulting bitrates and MOS values for the sequences used in the tests.	51
5.1	DCT frequency sensitivity thresholds.	86
5.2	PSNR estimation accuracy.	104
5.3	Evaluation of the proposed metric.	105
6.1	Average relative prediction error [%].	117
6.2	PSNR estimation accuracy.	121
6.3	Evaluation of the proposed metric.	123
7.1	Mean PSNR estimation error (dB).	140
7.2	Parameter estimation accuracy.	141
7.3	PSNR estimation accuracy.	146
7.4	Evaluation of the proposed metric.	147

List of Acronyms

ACR *Absolute category rating*

ANSI *American National Standards Institute*

AVC *Advanced video coding*

BER *Bit error rate*

CIF *Common intermediate format*

CSF *Contrast sensitivity function*

dB *Decibel*

DCR *Degradation category rating*

DCT *Discrete cosine transform*

DMOS *Differential mean opinion score*

DSIS *Double stimulus impairment scale*

DSCQS *Double stimulus continuous quality scale*

DWT *Discrete wavelet transform*

EBCOT *Embedded block coding with optimal truncation*

FFT *Fast Fourier transform*

FMO *Flexible macroblock order*

FR *Full reference*

HD *High definition*

HDTV	<i>High definition television</i>
HVS	<i>Human visual system</i>
IDCT	<i>Inverse discrete cosine transform</i>
IPTV	<i>Internet protocol television</i>
ISO	<i>International Organization for Standardization</i>
ITU	<i>International Telecommunication Union</i>
JND	<i>Just noticeable difference</i>
JPEG	<i>Joint Photographic Experts Group</i>
JVT	<i>Joint Video Team</i>
LCD	<i>Liquid crystal display</i>
LIVE	<i>Laboratory for Image and Video Processing</i>
LSB	<i>Least significant bit</i>
ML	<i>Maximum likelihood</i>
MOS	<i>Mean opinion score</i>
MPEG	<i>Moving Picture Experts Group</i>
MPQM	<i>Moving picture quality metric</i>
MSE	<i>Mean squared error</i>
MV	<i>Motion vector</i>
NR	<i>No-reference</i>
PDF	<i>Probability density function</i>
PDM	<i>Perceptual distortion metric</i>
PEVQ	<i>Perceptual video quality metric</i>
PLR	<i>Packet loss rate</i>
PSNR	<i>Peak signal-to-noise ratio</i>

QCIF *Quarter common intermediate format*

QF *JPEG quality factor*

QoE *Quality of experience*

QoS *Quality of service*

QP *H.264 quantization parameter*

RMS *Root mean squared*

RR *Reduced reference*

SDSCE *Simultaneous double stimulus for continuous evaluation*

SPEM *Smooth pursuit eye movement*

SSCQE *Single stimulus continuous quality evaluation*

SSIM *Structural similarity index*

TCP/IP *Transmission control protocol / Internet protocol*

TV *Television*

VQEG *Video Quality Experts Group*

List of Symbols

a_k, b_k	Quantization interval limits
α_T	Luminance adaptation constant (Watson's model)
α_w	Watermark embedding strength
b	Contrast masking constant (Watson's model)
β	Zero-mean cauchy probability density function parameter
β_{ML}	ML estimate for β based on the original coefficient values
$\hat{\beta}_{ML}$	ML estimate for β based on the quantized coefficient values
\hat{D}_f	Frame-by-frame video distortion metric
\hat{D}_g	Global video sequence distortion metric
D_W	Watson's distortion metric for one image
\hat{D}_W	Estimated Watson's distortion metric for one image
ΔR_i	Statistical rank difference for the i -th sample
ε	Error
$\hat{\varepsilon}$	Error estimate
ε_p	Perceptual error
f_r	Video frame rate
f_s	Spatial frequency
γ	Control parameter for the trust given to the ML estimator
i_k	Quantization index for the k -th DCT coefficient (encoding)
$\mathbf{\Lambda}$	Neighborhood matrix (used in the predictor training)
λ	Zero-mean laplace probability density function parameter
λ_{ML}	ML estimate for λ based on the original coefficient values
$\hat{\lambda}_{ML}$	ML estimate for λ based on the quantized coefficient values
$\hat{\lambda}_p$	Estimate for λ based on linear prediction
λ_v	λ values located in the neighborhood of the value to predict
μ	Mean
O_r	Outlier's ratio
P_f	Probability of false positive / false negative

p_k	Perceptual weight for the k -th DCT coefficient
$\varphi(i, j)$	Raw opinion score given by observer i to impaired sequence j
q_k	Quantization step for the k -th DCT coefficient (encoding)
Q_l	Value at the l -th quantizer level
r_0	Rate of DCT coefficients quantized to zero during encoding
ρ_c	Pearson's correlation coefficient
ρ_s	Spearman's rank order coefficient
σ	Standard deviation
s	Slack value
τ	Control parameter in ridge regression
$T_{L_k}(i, j)$	Luminance adaptation threshold for frequency (i, j) of the k -th block (Watson's model)
$T_B(i, j)$	Sensitivity threshold for frequency (i, j) (Watson's model)
Θ	Neighborhood matrix (used in the predictor training)
θ	Zero-mean laplace probability density function parameter
θ_{ML}	ML estimate for θ based on the original coefficient values
$\hat{\theta}_{ML}$	ML estimate for θ based on the quantized coefficient values
$\hat{\theta}_p$	Estimate for θ based on linear prediction
θ_v	θ values located in the neighborhood of the value to predict
v_E	Eye movement compensation term
v_I	Angular velocity of an object on the image plane
v_R	Retinal velocity
w_r	Reference watermark signal
w_d	Distorted watermark signal
W_{ber}	Extracted watermark's bit error rate
W_{MSE}	Watermark signal mean squared error
\mathbf{w}	Linear weights vector
$\hat{\mathbf{w}}$	Linear weights vector obtained through regression
$x_k(i, j)$	DCT coefficient located at frequency (i, j) of the k -th block
x_r	Watermarked DCT coefficient
x_d	Distorted DCT coefficient
\bar{x}_{00}	Average of the DCT coefficients at frequency $(0, 0)$ (Watson's model)
X_k	k -th quantized DCT coefficient
y	Original image
\hat{y}	Distorted image
y_d	Watermarked and distorted image
y_r	Watermarked reference image

y_k	k -th pixel of an original image
\hat{y}_k	k -th pixel of a distorted image

Chapter 1

Introduction

Image quality is a characteristic of an image that measures the perceived image degradation (typically, compared to an ideal or perfect image). Imaging systems may introduce some amounts of distortion or artifacts in the signal, so quality assessment is an important problem.

Wikipedia’s definition of image quality (October/2010).

1.1 Context and motivation

Over the last years, the *image quality* topic has become an increasingly important matter, especially due to the transmission of digital video over the internet and mobile networks [1, 2]. The need of new methodologies for measuring the quality of image and video is increasing as analog systems are being replaced by digital systems. Television (TV) is perhaps the most relevant field where numerous examples of digital video systems can now be found – cable and satellite services, IPTV and terrestrial digital TV broadcast. Similarly, in photography, digital cameras are probably the most used today.

In analog TV systems, the perceived quality of video was typically associated to the correct tuning of the TV antenna, together with the associated transmission’s power. As for the reproduction of video, quality was typically associated to the storage support media (for instance, the age of the video tape). Quality assessment methodologies typically consisted of transmitting predefined signals (“pilot” signals) and measuring their power at the receiver location. Due to the characteristics of

analog systems, these measurements correlated well with the human perception of quality.

However, the above mentioned methodologies are not adequate in the context of digital video, where the perceived quality of video is mainly associated to two factors:

- **Video encoding method** – there are usually bandwidth and capacity constraints associated to the transmission and storage of video contents. These constraints imply that the visual information must be compressed prior to transmission or storage. Lossy video encoding methods are used in order to achieve the desired compression ratios, which means that the encoding process is a source of distortion that affects quality.
- **Transmission losses** – the transmission of video in packet-based networks is also subject to packet losses. For instance, if the network is congested, packets containing video data may arrive at the receiver too late for correct decoding. This situation will result in more or less visible impairments on the decoded video signal.

The importance of image and video quality assessment is also evidenced by the settling of the *Video Quality Experts Group* (VQEG)¹, created in 1997. The mission of VQEG is to perform studies and calls for the development of new video quality assessment procedures, providing input to the most relevant standardization bodies, such as *International Telecommunication Union* (ITU), in an effort that has already led to the publishing of a few recommendations in the topic [3,4].

Image quality is a subjective measurement in the sense that different viewers may rate the quality of the same image or video differently. In fact, several factors are known to have an impact on the viewer concept of quality: for instance, personal interests and expectations, viewing conditions, fidelity of the reproduction and even the quality of the sound that comes with the video. Since human viewers are the target consumers for video communications products, they are naturally the most reliable source for quality assessment. However, gathering video quality assessment data from the human viewers is not a straightforward task, since it requires the completion of *subjective quality assessment tests*. A standardization of the procedures for conducting these type of tests is described in ITU recommendations [5,6] and the quality scores that result from such experiments are usually addressed to as *subjective scores* or *mean opinion scores* (MOS). Subjective tests must be carried

¹The homepage of VQEG can be found at www.its.bldrdoc.gov/vqeg.

out in a controlled environment and they require quality judgments performed by several viewers. Thus, subjective quality scores are hard to get and they cannot be used in real-time applications.

An alternative to subjective quality assessment is to automatically score video quality using *objective metrics*. Most of the research performed in this field has been focused on the development of *full reference* (FR) metrics, which require both the original and the distorted video data to compute the quality scores. FR metrics are typically used for benchmarking image and video processing algorithms, such as lossy encoding or watermarking techniques, and media distribution networks during the testing phases. However, FR metrics are not suitable for monitoring the quality of received media once the distribution network is setup and starts working, since the original data is usually not available at the receiver.

It is thus desirable to have a quality measurement system that is able to provide quality feedback without requiring the reference signals. This has led to an increased research effort on *no-reference* (NR) quality metrics and *reduced reference* (RR) quality metrics. NR metrics rely on the received media only. RR metrics can be placed between FR and NR metrics: information about the reference is sent through a side information channel and is used at the receiver for computing the objective quality scores. RR and NR quality metrics for video may contribute to enabling new services and applications, such as:

- **Branding protection** – in order to monitor the user's *quality of experience* (QoE), content providers should be able to verify that their customers are receiving multimedia content with adequate quality.
- **Scalable billing schemes** – costumers should be billed according to the quality of the received contents. This should bring fairness in the multimedia delivering scenarios (*i.e.*, users receiving poor quality media data should pay less than those receiving the same media with higher quality).
- **Quality-oriented adaptation of streaming services** – streaming servers could dynamically adjust some transmission parameters in order to deliver content with an adequate perceived quality, while optimizing resource usage.

At the present time, there are no standardized procedures for no-reference video quality assessment. The existing standards from ITU, released in 2008 under the designations ITU-T Recommendations J.246 [3] and J.247 [4] standardize a reduced reference and a set of full reference video quality assessment metrics, respectively.

The closest standard that is related with no-reference image quality assessment is ITU-T Recommendation G.1070 [7], that presents a quality model for video-telephony applications. The model relies on features such as packet loss rate, end-to-end delay, encoding bitrate and video frame rate, putting this standardized model closer to classical network *quality of service* (QoS) measurement than to image and video quality measurement.

The research work described in this Thesis is focused on the development of new algorithms for no-reference image and video quality assessment metrics. Quality assessment metrics belonging to this class are the most adequate in image and video distribution systems, since quality scores are computed based on the distorted media only.

Classical approaches to no-reference metrics usually try to estimate artifacts that result from lossy video encoding and/or from transmission losses. In this Thesis, a different philosophy is followed: the main idea is to estimate the quality of encoded image and video data by firstly estimating local errors between the original and the distorted media, and then weighting those errors using a perceptual model. This approach resembles a typical full reference quality assessment algorithm that uses error weighting; however, the algorithms proposed in this Thesis estimate the error using the encoded image or video bitstream, without requiring the original image or video data. Note that this Thesis deals with the distortion caused by lossy encoding processes only, namely source coding and transcoding. The effect of transmission losses (*i.e.*, packet losses in IP networks) has not been considered. Nevertheless, the ideas that are described in this document can be used in a more complete system, where transmission is also taken into account.

The Thesis proposes two main approaches for the implementation of the above mentioned philosophy: a watermarking-based quality assessment algorithm and an algorithm that relies on the statistical properties of the *discrete cosine transform* (DCT) coefficients of natural images.

1.2 Main contributions

Since literature on watermarking-based image quality assessment techniques is not common, this topic is, by itself, novel. Besides the novelty associated to the topic, this Thesis makes the following contributions to this field:

- In order to improve the accuracy of *mean squared error* (MSE) estimation in the presence of large distortions, a set of procedures that compensate MSE underestimation are proposed in the Thesis.
- Non-uniform frequency adapted quantization functions, derived from the perceptual model by Watson [8], were proposed. Such functions increase the robustness of the watermark and, consequently, also increase the ability to estimate the distortion errors. These functions can also be potentially used in different watermarking applications.

The work related with the accomplishment of these contributions has been published in [9–11].

As for the statistical-based no-reference error estimation algorithm, a new method for estimating the parameters of DCT coefficients distribution was proposed during the course of this Thesis. It uses *maximum likelihood* (ML) estimates combined with linear prediction, and it has shown greater accuracy for estimating image and video *peak signal-to-noise ratio* (PSNR) than other state-of-the-art algorithms. An implementation of the algorithm was embedded into the reference H.264 software. The modified version of this software has been independently tested by Dr. Ing. Tobias Oelbaum, from the Institute for Data Processing at the Technical University of Munich, which confirmed the good performance, increasing the reliability and credibility of the algorithm. The work published in [12–14] emphasizes the error estimation module based on the DCT coefficient’s statistics.

Most of the work found in no-reference image quality assessment literature has been focused in measuring and combining a predefined set of encoding artifacts (see for instance [15–26]). The methodology for MOS estimation proposed in the Thesis follows a rather different philosophy: to estimate distortion and then to apply *human visual system* (HVS) perceptual modeling to those estimates. Since classical algorithms for full reference quality assessment usually rely in measuring the true distortion error followed by perceptual masking, the work presented in this Thesis allows to follow similar algorithms, using error estimates instead of their true values. In the context of MOS estimation, this Thesis offers the following contributions:

- When looking into the results for still images subject to JPEG encoding, the algorithms proposed in the Thesis have shown better results than those found in literature and tested using the LIVE image database.

- For video, a fair comparison of the results is not easy to perform, mostly because the reported results are obtained using different video sequences, encoded using different parameters. The algorithm proposed in this Thesis for video quality assessment seems to outperform the results reported in other works [27–29].

These contributions are an important part of the work published in [30–33].

1.3 Outline of the Thesis

The Thesis is structured according to eight chapters, whose contents are the following:

- 1. Introduction** – the current chapter, that presents the motivation, main objectives, contributions and outline of the Thesis.
- 2. Image quality overview** – an introduction to image and video quality assessment is presented in this chapter. The main characteristics of the human visual system are reviewed and the reasons for image and video distortion are discussed, with a greater focus on the standardized lossy encoding methods.
- 3. Subjective quality assessment** – a description of the standardized subjective quality assessment procedures is provided in this chapter. Besides that, the experimental subjective quality assessment tests performed for supporting the work in the Thesis are also described.
- 4. Objective quality metrics** – this chapter presents a classification for objective quality assessment and provides an overview of the research performed on this field.
- 5. Image quality assessment using watermarking** – the first technical chapter, that provides an in-depth description of the proposed watermark-based image quality assessment algorithm and the corresponding results for still images.
- 6. Statistical image quality assessment** – a detailed description of the statistical based approach, and its application to JPEG encoded images, is provided in this chapter. It presents the main ideas that will be the basis for the generalization of the work to video.

7. Perceptual video quality assessment – this chapter describes the main achievement of the Thesis. The technique described in the previous chapter is generalized in order to score video quality. The performance of the proposed no-reference metrics are evaluated and compared with other algorithms described in literature.

8. Conclusion – final remarks, the main conclusions of the work presented in the Thesis and guidelines for future research.

Chapter 2

Image quality overview

2.1 What is image quality?

Image quality is the term associated to the rate given to the inherent quality of an image. In a wider perspective, image quality also applies to the quality associated to image sequences, *i.e.*, video. It is a subjective measurement in the sense that different people may rate the quality of an image differently: for instance, when looking into the image depicted in Figure 2.1, some people may considerer that the image has high enough quality; on the other hand, an individual with an higher quality sensitivity may notice that the image is noisy and lacks sharpness, and thus his quality rating would not be that great. In fact, several factors are known to have an impact on the viewer concept of quality [34–36]:

- ***Personal interests and expectations*** – the way an individual rates image quality is influenced by his personal interest on the content that is being displayed. For instance, when watching a soccer game, a soccer fan and a non-fan will probably have different quality requirements. The expectation associated to an imaging service is also an important factor. For instance, the quality that is expected from a movie at a high definition cinema is probably higher than what is expected when the same content is displayed on the PC. This means that similar distortions would probably result in different quality scores for both situations. Even if technology and viewing conditions are the same, there are additional factors that cause expectation to vary from individual to individual. Let's suppose that two persons are both customers of the same IPTV service, and one of them thinks that the service is somewhat expensive



Figure 2.1: An image of a cat – is it a good quality image or not? (from <http://photobucket.com>).

while the other was never concerned about the price. In this case, quality requirements for the first user may be higher than those for the second.

- **Viewing conditions** – when watching video content, there are numerous factors related to the viewing conditions that contribute to the perception of quality. Among those factors are the viewing distance, which directly determines the effective size and resolution of the image built on the retina. Another important factor is the lighting conditions: the sensibility to contrast decreases with increasing ambient light; light sources may reflect on the screen. The type and resolution of the display is also important.
- **Interaction with the service** – from a wider *quality of experience* (QoE) perspective, the interface between the user and the contents, such as the existence of program guides, additional features (*e.g.*, video club) and technical support may also contribute to the user perception of quality. Additionally, quality of experience also depends on the configuration of the connection and the equipment installed at the user's home. It may have an influence on channel zapping time and it determines the number of different channels allowed to see when more than one display is present at home.
- **Sound** – there are studies [37] supporting that the quality of the audio accompanying the video has a strong influence on the perceptual quality of the video. Subjective tests have shown that videos together with high quality sound usually get better quality scores even when the subjects are asked to evaluate the quality of video only.

- ***Fidelity of reproduction*** – the fidelity of reproduction is probably the most important factor from the quality metric point of view, since almost all of the work about image quality assessment is focused on this topic. The viewer sees an encoded and transmitted version of an *original* video, and it is quite obvious that the amount of the distortion introduced by lossy encoding and transmission errors will strongly influence the overall quality of the video. On the other hand, some distortion types may actually increase content’s quality: sharpened and colorful versions of the original image data may in fact get better quality scores than the original versions.

Most of the research on image and video quality assessment is focused on the last factor listed – fidelity of reproduction. The main applications for image and video quality assessment metrics are related with the encoding and transmission of visual content. Examples of such applications are performance evaluation of a new video encoding scheme (codec) or automatic quality assessment of the received video signals in a digital television broadcasting network.

Measuring the quality of an imaging system can be carried out either by performing *subjective* quality assessment experiments, or by using *objective* quality assessment metrics. Quality scores in a subjective quality assessment experiment are those resulting from an evaluation performed by human viewers; those quality scores are usually addressed to as the *mean opinion scores* (MOS). In order to obtain consistent MOS values, quality assessment of image and video contents must consider several viewers in a controlled environment. Standardized procedures for conducting such experiments are described in ITU-R Rec. BT.500 [5] and ITU-T Rec. P.910 [6] (which will be reviewed in Chapter 3). Since the human viewer is the target consumer of image and video content, subjective quality scores are the most reliable measurements assessing image quality. However, due to the involved constraints – a controlled test environment and a representative number of viewers – they are not easy to obtain and they cannot be used for quality monitoring in real-time application scenarios.

The main goal of an objective metric is to automatically compute quality scores that match the MOS values given by the viewers. Thus, an ideal objective quality metric should produce the same quality scores as the subjective measurement. Since objective metrics produce automatic quality scores, they have greater potential, especially for real time quality monitoring of video communication systems.

In order to develop an objective quality assessment metric, knowledge and understanding about the *human visual system* (HVS) are very important factors. A brief overview of the main HVS features is provided in Section 2.2 of this chapter. Since the work developed on this Thesis is oriented to the image and video quality resulting from lossy encoding of contents, Sections 2.3 and 2.4 deal with the most significant image and video encoding standards and with the visual artifacts that result from lossy encoding according to those standards, respectively. Section 2.5 briefly discusses visual impairments caused by transmission errors. To conclude, a summary of this chapter is given in Section 2.6.

2.2 Fundamentals of human vision

In the context of image and video quality, the understanding of the HVS plays an important role. The HVS is the system by which an observer views, interprets and responds to visual stimuli. This section provides a brief overview of the main characteristics of the human visual system.

2.2.1 The mechanics of the eye

The eye is a complex biological device which can be compared to a camera, when considering its optical characteristics. A simplified diagram of the human eye ball is represented in Figure 2.2. An image is focused in the *retina* surface using the *lens*. The *ciliary muscle* controls the shape of the lens, allowing focus of an object at a given distance. The light enters the eye through the *pupil*, whose size is controlled by a set of muscles called the *iris*. The amount of light that enters the eye depends on the light levels in the exterior. The pupil can thus be considered one of the HVS mechanisms responsible for light adaptation.

Images of the outside world are projected into the retina, the neural tissue located in the back of the eye. The retina consists of an array of photoreceptors, whose function is to convert light energy into signals that can be interpreted by the brain. These photoreceptors can be classified into two types: the *cones* and the *rods*. The former are sensitive to color, under high light levels, while the latter are sensitive to luminance at low light levels. Most cones are concentrated in the *fovea*, a small area located near the center of the retina, which means that high resolution color vision is only achieved in a relatively small area of the field of view.

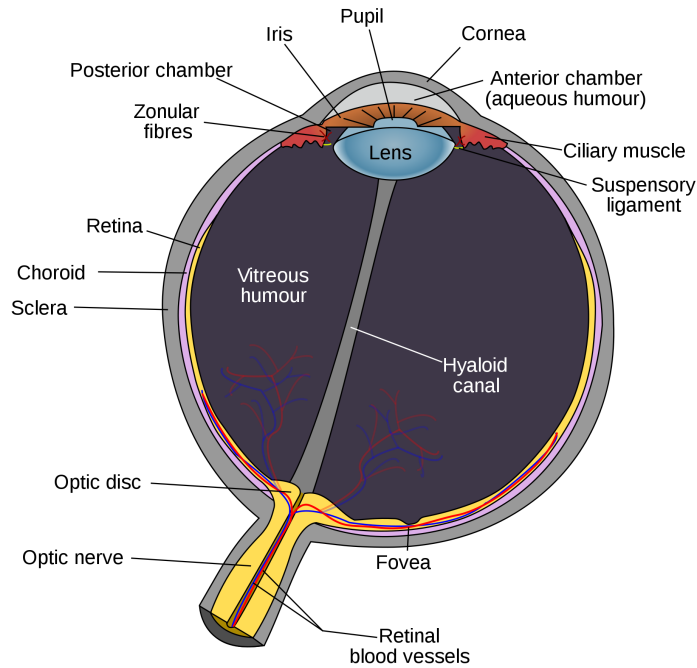


Figure 2.2: Diagram of the human eye (from <http://en.wikipedia.org>).

The nerves connected to the retina leave the eye ball through the *optic nerve*, leading the information captured by the eye to different parts of the brain. The brain processes and interprets visual information based not only in the received information, but also in prior learned responses.

2.2.2 Luminance adaptation and contrast sensitivity

The human visual system is able to adapt to a wide range of light intensities. However, once adapted to a given light intensity, the HVS can only discriminate light intensities whose values are within a range of 2–3 orders of magnitude from the adapted intensity [38]. This property is similar to the dynamic range of a camera.

Three mechanisms for luminance adaptation can be distinguished in the HVS [39]:

- Variation of the pupil's aperture – as already mentioned in Section 2.2.1, the pupil's size is controlled by the iris.
- Chemical processes in the photoreceptors – a mechanism that can be found

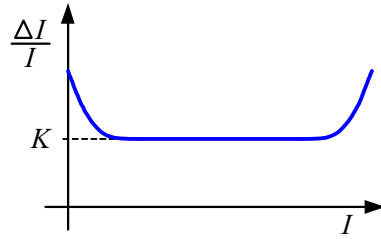


Figure 2.3: Sensitivity of the HVS to light intensity changes (adapted from [39]).

both in the cones and in the rods. If light is bright, the concentration of photochemicals in the receptors decreases, thereby reducing their sensitivity. On the other hand, when the intensity of light is small, the production of those chemicals is increased, thus sensitivity is increased.

- Neural-level adaptation – the neurons located in the retina have the ability of increasing / decreasing their signal output in order to adapt for light intensity.

Once adapted to a given light intensity, the response of the HVS to visual stimuli depends more on the relation of its local variations to the surrounding luminance rather than on the absolute luminance values. This property is partially modeled by *Weber's law*¹, which states that:

$$\frac{\Delta I}{I} = K, \quad (2.1)$$

where I is the adapted light intensity and ΔI is the minimum amount of change in the light level intensity that causes an observer to detect the change. This threshold value is also known as *just noticeable difference* (JND). However, Weber's law does not hold for the full range of visible light intensities, as illustrated in Figure 2.3. Nevertheless, it can be applied to the range of light intensities typically found in most image processing applications.

Another characteristic of the HVS, evidenced in Figure 2.3, is its lower sensibility to luminance changes under high and low light intensities. In the context of perceptual modeling for image and video applications, this characteristic is usually explored by means of *luminance masking* procedures, which assign larger sensibility thresholds in the brighter and in the darker image regions.

¹Ernst Weber (1795–1878), a German physiologist, observed that the threshold for a noticeable difference appeared to be related to the initial stimulus magnitude – this relation was known since as *Weber's Law*.

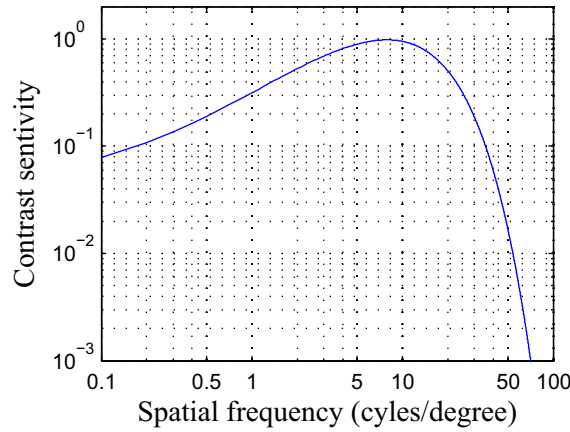


Figure 2.4: Spatial contrast sensitivity function (adapted from [40]).

Another important characteristic of the HVS is its ability to measure variations of light intensity in the visual field, with the purpose of discriminating and identifying objects of the outside world. These variations in light intensity can be measured in terms of contrast. Different definitions of contrast exist, but the most commonly used is probably *Michelson's contrast*:

$$C_{\text{Michelson}} = \frac{I_{\max} - I_{\min}}{I_{\max} + I_{\min}}, \quad (2.2)$$

where I_{\max} and I_{\min} are the maximum and minimum luminance values in a given pattern. Contrast sensitivity can be defined as the inverse of minimum contrast that is necessary for an observer to detect a stimulus. The evolution of contrast sensitivity with the frequency of the stimulus can be described by a so-called *contrast sensitivity function* (CSF). This function is usually obtained by fitting data from psychovisual experiments where visual stimuli at different frequencies are displayed.

Figure 2.4 depicts the typical evolution of a contrast sensitivity function with the spatial frequency of the visual stimulus. As can be observed from the figure, the shape of a spatial CSF resembles a low pass (or “slightly band pass”) filter response. The sensibility of the HVS to visual stimuli reaches its peak value at mid-low frequencies and, afterwards, has a fast decrease as the frequency of the stimulus increases. There is no canonical CSF, since the contrast sensitivity varies according to the adapted luminance level, the position of the retina with respect to the display and the temporal frequency of the stimulus.

For the case of video, the temporal frequency of the visual stimuli is also considered, resulting in a spatio-temporal contrast sensitivity function. This function can be

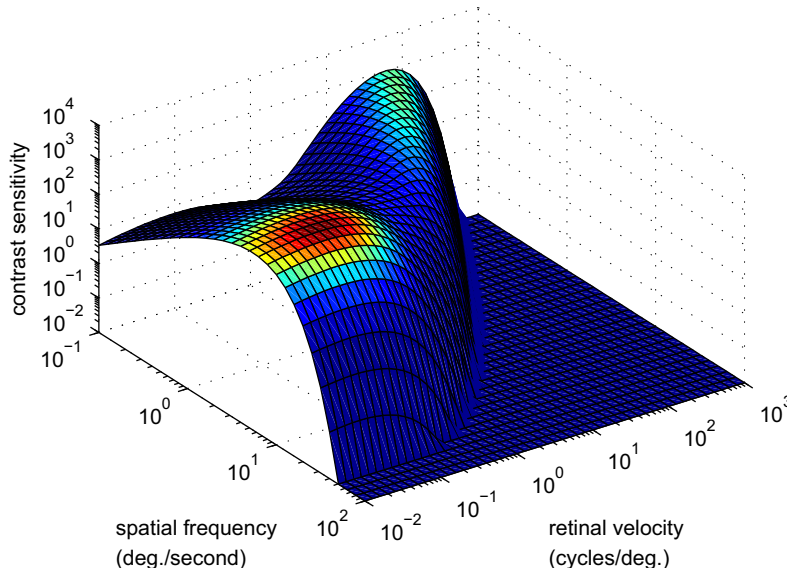


Figure 2.5: Spatio-temporal contrast sensitivity function (adapted from [41]).

graphically interpreted as a space-time surface, such as the plot depicted in Figure 2.5. In this plot, the temporal frequency of the stimulus is given in terms of the retinal velocity, *i.e.*, the object velocity in the retina plane.

The space-time separability of spatio-temporal contrast sensitivity has been subject of investigation, since this separability would simplify the models. However, no consensus on this subject has been reached yet.

2.2.3 Masking

Masking occurs when the presence of a stimulus, which could be perceptible by itself, becomes hidden due to the presence of another stimulus. For instance, consider Figure 2.6-a), depicting an original image – *house* – to which a regular noise patch (shaped as a sine wave) has been added, resulting in the image depicted in Figure 2.6-b). It can be observed that the noise is clearly perceptible in the homogeneous regions of the image, such as the sky and the water. On the other hand, due to spatial masking effects, noise is not very perceptible (or even imperceptible) in textured regions such as the trees, or the barn’s roof.

The effect of masking is usually quantified by measuring the detection threshold for a target stimulus embedded on a masker with varying contrast. Figure 2.7 shows the possible outcome of such an experiment. T_0 represents the contrast sensitivity threshold in the absence of masking. In Figure 2.7-a), a typical masking effect is



Figure 2.6: Masking effect example.

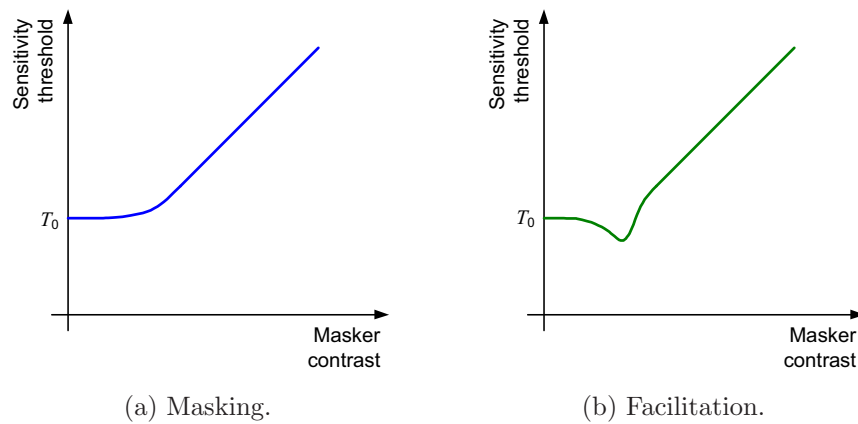


Figure 2.7: Contrast masking and facilitation.

illustrated: the sensitivity threshold increases as the masker contrast increases – this effect occurs if the masker and target stimulus have different characteristics. The second effect, illustrated in Figure 2.7-b), corresponds to a situation where the contrast sensitivity threshold starts by decreasing with increasing masker contrast, which means that the masker contrast actually increases the perceptibility of the stimulus. In this case, there is a *facilitation* phenomenon, which occurs when the stimulus and the masker contrast have similar characteristics.

In the case of video, the concept of masking applies both spatially and temporally. For the latter case, if there are temporal discontinuities in image intensity values (*e.g.*, scene changes) an observer may not detect a visual stimulus, which would be otherwise detected, in a subsequent video frame [42]. Additional studies on this subject [43–45] suggest that temporal masking effects occur in the temporal vicinity before and after a scene change. These are usually addressed to as *backward* and *forwarding* masking, respectively.

2.2.4 Pooling

As discussed in the previous sections, sensitivity and masking models can be used to provide an estimate for the perceptibility of distortions associated to a given spatial or temporal frequency. However, since distortion is usually spread across multiple frequencies, the sensitivities associated to each frequency component must be combined – this process is called *pooling*.

The pooling process is usually performed using rules of probability or vector summation, which are typically expressed in the form of a *Minkowski summation* (or L_p – norm). Therefore, a global distortion measurement, $D(y, \hat{y})$, between a reference image, y , and its degraded version, \hat{y} , can be expressed as:

$$D(y, \hat{y}) = \sqrt[p]{\sum_k |d_k(y, \hat{y})|^p}, \quad (2.3)$$

where $d_k(y, \hat{y})$ represents the perceptibility of the distortion associated to the k -th individual image component that is been accounted for. Since the pooling process is usually performed along different dimensions, these individual components can be, for instance, spatial frequencies, pixel locations or temporal samples. The exponent p is usually set to 4 [46, 47], a value that emphasizes stronger distortions that may capture the viewers attention.

2.3 Image and video encoding

Since this Thesis is focused on the quality of images and video subject to lossy encoding, this section provides a brief coverage of today's main image and video encoding standards.

2.3.1 Compression of visual information

Images and video, together with the way HVS perceives visual information, have specific characteristics that led to the development of lossy compression methods suitable for this kind of information. Thus, besides exploring statistical redundancies as in typical data compression applications, the compression of image and video data achieves higher compression ratios by also exploring two different types of visual information redundancy:

- Spatio-temporal information redundancy – the value of a pixel at a given image location is often well correlated with the values in its surroundings (spatial redundancy); similarly, two consecutive video frames are usually well correlated (temporal redundancy).
- Psychovisual redundancy or irrelevancy – as seen in the previous section, the human visual system is not equally sensible to all types of image content; therefore, the compression algorithm tries to discard information that is not perceptible by the HVS. The exploitation of this type of redundancy is the cause of lossy compression.

Before encoding, color information is usually converted to the YCbCr color space². This conversion is performed in order to explore the lower acuity of the HVS with respect to color. In order to achieve higher data compression in video broadcasting applications, the chrominance signals are usually subsampled by a factor of two in the horizontal and vertical directions (this type of subsampling is denoted by 4:2:0).

Compression methods for images and video can be roughly divided into two categories: model-based methods (*e.g.*, fractal compression) and waveform-based methods (*e.g.*, DCT-based or wavelet-based compression). Today's image and video encoding standards belong to the latter class – waveform-based compression methods. Generally, today's standards achieve data compression using the following steps:

- Transformation – pixel values in the spatial image or video domain are transformed into coefficients' values in the frequency domain. The main purpose of the transformation stage is to compact the signal's energy in the lower frequency coefficients. It decorrelates pixel values, thus exploring spatial information redundancies. On the other hand, the representation of the visual information in the frequency domain is suitable for exploiting psychovisual redundancies in posterior encoding stages. The most popular transform in nowadays image and video encoding standards is the DCT. It is also worth to mention that this part of the encoding process is reversible, unless numerical computation errors are introduced by the transform process.
- Quantization – the coefficients resulting from the previous stage are quantized, in order to reduce the number of bits used for their representation. The

²Y denotes luminance, Cb and Cr are the chrominance components of the image / video signal. Cb is the difference between the blue primary channel and the luminance; Cr is the difference between the red primary channel and the luminance.

quantization stage explores psychovisual redundancies, by considering basic characteristics of the HVS: higher frequency coefficients are quantized with coarser quantization steps. Note that quantization is the encoding step where irreversible losses occur, and the reason behind the term “lossy encoding”.

- Entropy coding – the final stage is the lossless encoding of the values resulting from the quantization stage and of the remaining information required for correctly handling the bitstream during decoding (*e.g.*, quantization step sizes, prediction modes, *etc.*).

For video, an additional, and probably the most important, encoding step to achieve high data compression rates is *motion compensation*. The main purpose of this step is to explore the temporal redundancy in successive video frames. An estimate of the motion vectors between a reference frame (or a set of reference frames) and the frame under encoding is computed. These motion vectors, together with the associated transformed and quantized prediction errors, are entropy coded and written into the bitstream. Higher compression rates are achieved because the amount of information necessary to describe the differences between two successive frames after motion compensation is usually much less than to encode them independently.

The following subsections give a brief overview of the main image and video encoding standards used during the course of this Thesis. Note that, for the case of video standards, only the syntax and semantics of the output bitstream and the decoding process are specified. Nevertheless, there are implicit dependencies that lead to a common encoder architecture.

2.3.2 JPEG

Despite its age, the JPEG standard is probably the most commonly used method for lossy encoding of still images. The designation JPEG stands for the name of the group responsible for the publication of the standard – *Joint Photographic Experts Group*. The first version of standard was issued in 1992 as ITU-T Recommendation T.81 [48] and in 1994 as ISO-IEC 10918-1.

A typical architecture of a JPEG encoder can be observed in Figure 2.8-a). The input image which is to be encoded is firstly split according to 8×8 blocks which are independently encoded. Pixel values in each block are then subtracted an offset value which basically shifts the range of pixel values from $[0; 255]$ to $[-128; 127]$. The pixel values at position (m, n) of the k -th block, $y_k(m, n)$, are then transformed into

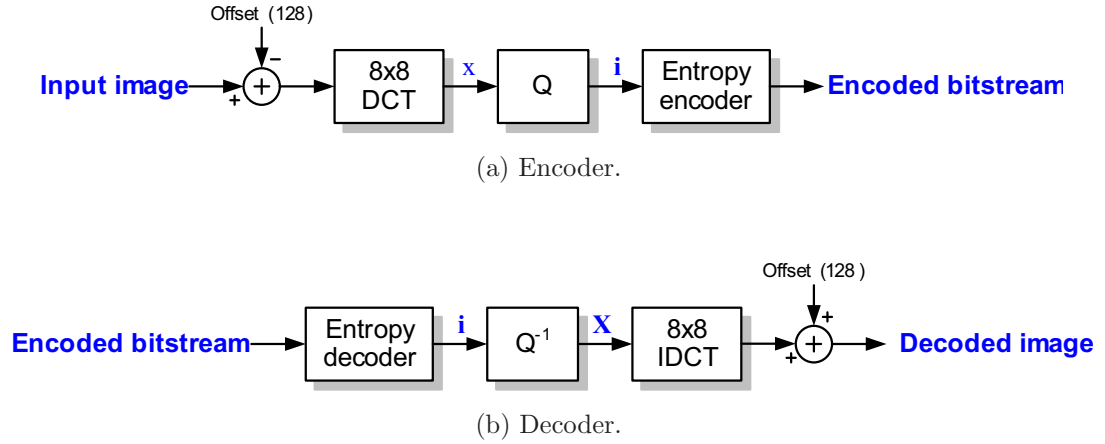


Figure 2.8: JPEG encoder and decoder schemes.

the frequency domain using the *discrete cosine transform* (DCT). This transform is applied in a blockwise fashion according to:

$$x_k(i, j) = \frac{c(i)c(j)}{4} \sum_{m=0}^7 \sum_{n=0}^7 y_k(m, n) \cos\left(\frac{i\pi(2m+1)}{16}\right) \cos\left(\frac{j\pi(2n+1)}{16}\right), \quad (2.4)$$

where $x_k(i, j)$ are the resulting transformed coefficient values in the k -th block, (m, n) and (i, j) are the indexes within each block in the spatial and frequency domain, respectively. As for $c(i)$ and $c(j)$, they are defined as:

$$c(i), c(j) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } i = 0 \text{ or } j = 0; \\ 1, & \text{otherwise.} \end{cases}$$

The values of $x_k(i, j)$ are then quantized using uniform quantization. In the JPEG standard, different quantization steps are assigned to coefficients located at different spatial frequencies (*i.e.*, the quantization step depends on the (i, j) position), but they do not vary from block to block. Figure 2.9 depicts the quantization tables, for the luminance and for the chrominance image components, provided in the standard as examples. In the reference JPEG encoding/decoding software [49], the sizes of the quantization steps are scaled versions of those matrices, where the scaling is controlled by the *quality factor* (QF) parameter.

For simplicity, the indexes (i, j) will be dropped in the remaining of the text. Each

Diagram (a) shows an 8x8 matrix of Luminance values. The horizontal axis is labeled j and the vertical axis is labeled i . The matrix values are:

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

(a) Luminance.

Diagram (b) shows an 8x8 matrix of Chrominance values. The horizontal axis is labeled j and the vertical axis is labeled i . The matrix values are:

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

(b) Chrominance.

Figure 2.9: Example JPEG quantization tables (from [48]).

DCT coefficient, x_k , is quantized according to:

$$i_k = \text{round} \left(\frac{x_k}{q_k} \right), \quad (2.5)$$

where q_k is the quantization step and i_k is the resulting quantization index. To complete the process, the quantization indexes are entropy coded, together with additional data that will be required at the decoder side for correctly handling the encoded bitstream.

The JPEG decoder, represented in Figure 2.8-b), performs the inverse operations of the encoding process. The quantization indexes are entropy decoded and then dequantized according to:

$$X_k = i_k \times q_k. \quad (2.6)$$

The result is a reconstructed DCT coefficient, X_k , that generally differs from its original value. This difference is mainly due to the quantizing process of the DCT coefficients. The coefficients X_k are inverse transformed to the spatial domain and the offset value is added, resulting in the decoded image.

2.3.3 JPEG2000

The JPEG2000 standard was developed by the JPEG committee and was published in year 2000 under the name ISO/IEC 15444 [50], with the purpose of replacing the older JPEG standard. However, JPEG2000 was not been implemented in the most influential web browsers, and thus its use has not been globally generalized up to now. Although not considered in the work presented in the Thesis, a short overview of the JPEG2000 standard is provided on this section.

JPEG2000 encoding is based on the *discrete wavelet transform* (DWT). In the lossy encoding mode, the DWT is computed using the *Cohen-Daubechies-Feauveau* (CDF) 9/7 wavelet, resulting in an m -level wavelet sub-band decomposition of the input image.

The quantization of the resulting wavelet coefficients is performed according to:

$$i_k = \text{sign}(x_k) \left\lfloor \frac{|x_k|}{q_k} \right\rfloor, \quad (2.7)$$

where the quantization step size, q_k , depends on the sub-band where the DWT coefficient to be quantized is located. The quantization indexes are further organized into blocks which are encoded in a process called *embedded block coding with optimal truncation* (EBCOT). This process starts by encoding the most significant bits of each block and progressively goes to the least significant bits. The least significant bit planes can be dropped in order to save bits, and thus, in addition to coefficient quantization, the EBCOT process can also introduce losses during image encoding. Deeper overviews of this encoding standard can be found in [51, 52].

2.3.4 MPEG-2

MPEG-2 is the second audiovisual encoding standard from the *Moving Picture Experts Group* (MPEG) and it was developed in order to cover a wider range of applications than the previous MPEG-1 standard. The specification of the video codec used in MPEG-2 was originally published in 1996 under the name of ISO/IEC 13818 [53] part 2 (or ITU-T H.262). Nowadays, the main application of MPEG-2 is, probably, DVD-video. It is also widely used in digital television broadcasting although newer systems are adopting the more recent H.264 standard.

General architecture

Figure 2.10-a) depicts a partial scheme of a typical MPEG-2 video encoder (entropy encoding is not represented). Let's admit that an input frame, \mathbf{F}_{in} , is to be encoded. \mathbf{F}_{in} is split in 16×16 block-wise units called *macroblocks* (MBs)³.

Each MB can be encoded either in *Intra* or *Inter* mode. In Intra mode, pixel values

³Note that the 16×16 macroblock size is applied to the luminance component of the input frame. The correspondent macroblock size for the chrominance components depends on the chroma subsampling scheme used in the input video sequence.

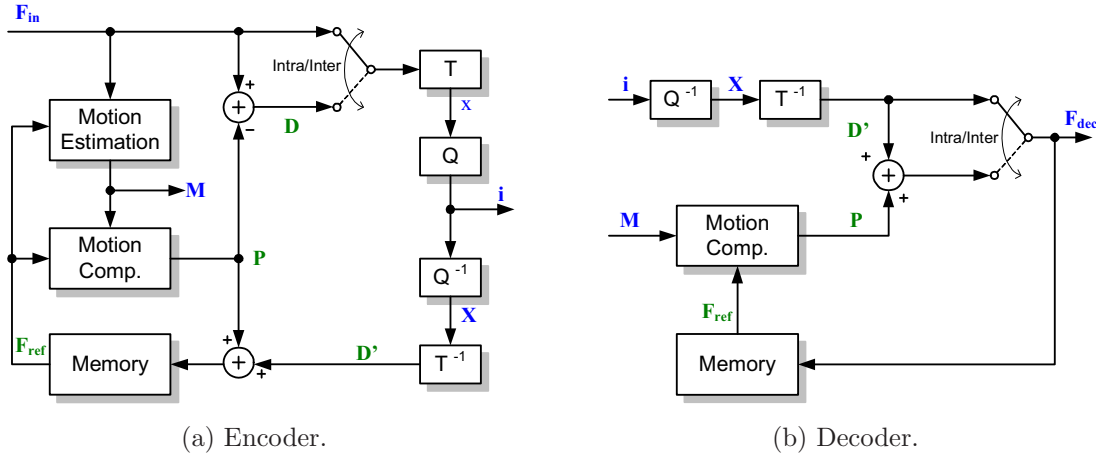


Figure 2.10: MPEG-2 generic encoder and decoder schemes.

of each MB are DCT transformed, similarly to what is done in JPEG, resulting in a set of coefficients \mathbf{x} . These coefficients are then quantized, resulting in the corresponding quantization indexes \mathbf{i} .

As for the Inter mode, a prediction block \mathbf{P} is computed by motion-compensated prediction from a set of previously encoded reference frames \mathbf{F}_{ref} . The difference, \mathbf{D} , between \mathbf{P} and the original MB pixel values is DCT transformed and quantized, resulting in the quantization indexes \mathbf{i} .

The indexes \mathbf{i} and the motion vectors \mathbf{M} associated to the encoding process are entropy coded and written into the bitstream.

The decoder, represented in Figure 2.10-b), starts by performing entropy decoding of the input bitstream. The decoded elements are reordered in order to produce a set of quantized coefficient data, \mathbf{X} , and, if in Inter mode, the associated motion vectors. In Intra mode, the reconstructed MB results from applying the inverse DCT to the coefficients. In the case of an Inter MB, the reconstructed MB is the result of the inverse transformed coefficients added to the prediction signal \mathbf{P} that results from motion compensation.

Transform and quantization

The transform operation used in MPEG-2 is an 8×8 block-wise DCT similar to what is used in JPEG. Using matrix notation, this operation can be written as:

$$\mathbf{x} = \mathbf{T}\mathbf{D}\mathbf{T}^T, \quad (2.8)$$

	$j \rightarrow$								
$i \downarrow$	8	16	19	22	26	27	29	34	
	16	16	22	24	27	29	34	37	
	19	22	26	27	29	34	34	38	
	22	22	26	27	29	34	37	40	
	22	26	27	29	32	35	40	48	
	26	27	29	32	35	40	48	58	
	26	27	29	34	38	46	56	69	
	27	29	35	38	46	56	69	83	

(a) Intra.

	$j \rightarrow$								
$i \downarrow$	16	16	16	16	16	16	16	16	
	16	16	16	16	16	16	16	16	
	16	16	16	16	16	16	16	16	
	16	16	16	16	16	16	16	16	
	16	16	16	16	16	16	16	16	
	16	16	16	16	16	16	16	16	
	16	16	16	16	16	16	16	16	
	16	16	16	16	16	16	16	16	

(b) Inter.

Figure 2.11: Default MPEG-2 quantization tables.

where \mathbf{T} is the transform matrix, \mathbf{D} are the values to transform and \mathbf{x} are the resulting DCT coefficients. The elements of \mathbf{T} at row i and column j can be defined as:

$$T_{ij} = \begin{cases} \frac{1}{\sqrt{N}}, & \text{if } i = 0; \\ \frac{2}{\sqrt{N}} \cos\left(\frac{\pi(2j+1)i}{2N}\right), & \text{otherwise,} \end{cases} \quad (2.9)$$

with $N = 8$ and $i, j = 0, \dots, 7$.

The quantization scheme used in MPEG-2 is slightly different from what was described for JPEG, because it includes a “dead zone” around 0. In practice, this means that the quantization interval around 0 is larger than the remaining ones. During decoding, the quantization steps q_k are derived from the bitstream according to:

$$q_k = \begin{cases} 2^{(3-DC_{precision})}, & \text{for DC Intra coefficients;} \\ [2 \times Q_{scale} \times Q_{Intra}(i, j)]/32, & \text{for AC Intra coefficients;} \\ [2 \times Q_{scale} \times Q_{Inter}(i, j)]/32, & \text{for Inter coefficients.} \end{cases} \quad (2.10)$$

The parameters $DC_{precision}$, Q_{scale} , Q_{Intra} and Q_{Inter} are also derived from the bitstream. Q_{Intra} and Q_{Inter} are given by the quantization tables for Intra and Inter blocks, respectively (Figure 2.11 presents the default tables). DC coefficients are those located at spatial frequency $(i, j) = (0, 0)$ while AC coefficients are those where $(i, j) \neq (0, 0)$.

The decoded DCT coefficients X are reconstructed according to:

$$X_k = \begin{cases} i_k \times q_k, & \text{for Intra coefficients;} \\ i_k \times q_k + \text{sign}(i_k) \frac{q_k}{2} & \text{for Inter coefficients,} \end{cases} \quad (2.11)$$

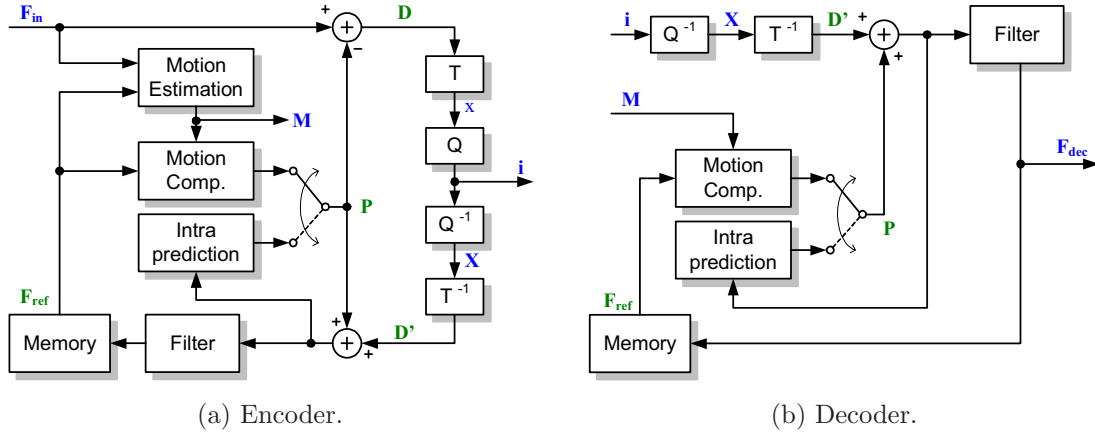


Figure 2.12: H.264 generic encoding and decoding schemes.

where i_k and q_k are, respectively, the quantization index and step size associated to the k -th DCT coefficient.

2.3.5 H.264

H.264 [54] is the latest encoding standard developed by ITU-T together with MPEG, in a partnership effort known as the *Joint Video Team* (JVT). It is also known as MPEG-4 Part 10: Advanced Video Coding or simply by MPEG-4 AVC. The first version of the standard was published in May 2003. In the following, a short description of a typical H.264 encoder/decoder architecture is presented. This description is strongly oriented to the H.264 features that are related to the work reported on this Thesis, namely the transform and quantization schemes. Deeper overviews of the H.264 standard can be found in [55–57].

General architecture

A typical H.264 encoder is partially represented in Figure 2.12-a). Similarly to what was described for the MPEG-2 standard, an input frame, \mathbf{F}_{in} , subject to encoding, is split in 16×16 *macroblocks* (MBs). Each MB can be encoded in Intra or Inter mode. In Intra mode, a prediction block, \mathbf{P} , is computed from samples taken from the current frame, that have been previously encoded, decoded and reconstructed. In Inter mode, \mathbf{P} is computed by motion-compensated prediction from reference frame(s), \mathbf{F}_{ref} . The difference between \mathbf{P} and the original MB pixel values, \mathbf{D} , is transformed (using a block-wise transform) and quantized, resulting in the set of quantized transform coefficients \mathbf{X} , as well as the corresponding quantization

indexes. These indexes, as well as the motion vectors, \mathbf{M} , that result from the motion estimation block in Inter mode, are then re-ordered and entropy encoded for transmission.

As for the H.264 decoder, partially represented in Figure 2.12-b), it receives a compressed bitstream, whose elements are entropy decoded and reordered in order to produce a set of quantized coefficient data, \mathbf{X} . The quantized coefficients are then rescaled and inverse transformed, resulting in a residual signal, \mathbf{D}' , which is added to the current prediction signal, \mathbf{P} . If the macroblock is Inter encoded, the prediction \mathbf{P} is computed by motion compensation, using the motion vectors, \mathbf{M} , and macroblocks belonging to previously decoded frames. The decoded frame results from the sum of \mathbf{P} and \mathbf{D}' , for all MBs.

Transform and quantization

The transform operation used in H.264 is an integer approximation of the classical block-wise DCT used in previous standards, such as JPEG and MPEG-2. The main transform block size in H.264 is 4×4 , although the use of an 8×8 transform is also possible. Let \mathbf{D} represent the differences between the original and the predicted image values in a 4×4 block. The transformed coefficient values, \mathbf{x} , can be computed as:

$$\mathbf{x} = \mathbf{T} \mathbf{D} \mathbf{T}^t \odot \mathbf{S}, \quad (2.12)$$

where \odot represents point-by-point multiplication, \mathbf{T} is the transform matrix and \mathbf{S} is a post-scaling matrix, which are defined as [57]:

$$\mathbf{T} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}; \quad \mathbf{S} = \begin{bmatrix} \frac{1}{4} & \frac{1}{\sqrt{5}} & \frac{1}{4} & \frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{1}{10} & \frac{1}{\sqrt{5}} & \frac{1}{10} \\ \frac{1}{4} & \frac{1}{\sqrt{5}} & \frac{1}{4} & \frac{1}{\sqrt{5}} \\ \frac{1}{\sqrt{5}} & \frac{1}{10} & \frac{1}{\sqrt{5}} & \frac{1}{10} \end{bmatrix}. \quad (2.13)$$

The transform operation can be implemented using integer arithmetic only (add and shift operations). As for the post-scaling operation, the reference software [58] implements it together with the quantization, using integer operations only.

The value of the quantized coefficient, X_k , is given by:

$$X_k = \underbrace{\text{sign}(x_k) \times \left\lfloor \frac{|x_k|}{q_k} + 1 - \alpha \right\rfloor}_{i_k} \times q_k, \quad (2.14)$$

$\text{mod}(QP, 6)$	q_B
0	0.6250
1	0.6875
2	0.8125
3	0.8750
4	1.0000
5	1.1250

Table 2.1: Base quantization steps.

where q_k is the quantization step, α is a parameter that controls the width of the dead zone around 0 and i_k represents the quantization index that is actually transmitted. In the reference software [58], $\alpha \simeq 2/3$ for Intra blocks and $\alpha \simeq 5/6$ for Inter blocks. The quantization step, q_k , can be derived from a H.264 parameter called QP , which may differ from macroblock to macroblock. The general rule to compute q_k from QP is:

$$q_k = q_B(\text{mod}(QP, 6))2^{\lfloor QP/6 \rfloor}, \quad (2.15)$$

where q_B is a base quantization step (see table 2.1) and $\text{mod}(m, n)$ is the remainder of integer division of m by n .

2.4 Compression artifacts

As seen in the previous section, unless a lossless encoding method is used, the procedure of encoding images or video using today's standards always produces a result that is different from the original. Due to bandwidth or storage space constraints, the bitrate of the encoded stream must be reduced in order to represent the visual information using less bits. However, in such cases, the amount of information that is lost due to encoding can be significant and thus distortions may become visible. These distortion effects due to lossy encoding of images or video are generally called *coding artifacts*.

The types of artifacts introduced by a given encoder are strongly dependent on the lossy encoding method that is used. This means that different image encoding standards, for instance JPEG and JPEG2000, may cause the visibility of different

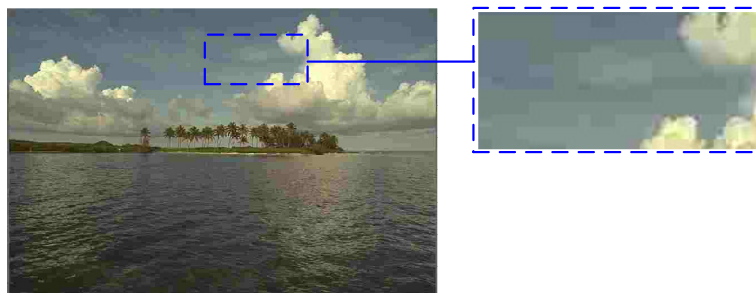


Figure 2.13: Blocking effect – *Ocean* image encoded using JPEG. Blocking patterns are clearly visible in the sky.

types of artifacts. However, all of the artifacts share one point in common: the noise produced on the encoded images or video is structured. Generally, although several artifacts may be found in the same image or video, the most perceptible artifact tends to hide or mask the effect of the others.

In the following, the most common artifacts generated during lossy encoding of visual media are described. Their causes are discussed and their effects are illustrated using examples.

2.4.1 Block effect

The *block effect* is characterized by the visibility of structured noise organized according to small blocks and is due to a blockwise encoding method associated to coarse quantization.

Each block is encoded without considering the correlation between pixel values belonging to adjacent blocks. When quantization is coarse, the representation of the pixels belonging to a block may be significantly different from their neighbors and, in that case, discontinuities become quite perceptible at the blocks' boundaries. This effect is typically visible at high compression rates (coarser quantization steps).

Due to the generalized use of blockwise image and video encoding standards, such as JPEG, MPEG-2 or H.264, the block effect is probably the most studied encoding artifact. An example of this effect can be observed in Figure 2.13, for a JPEG encoded image.

A particularly annoying form of the block effect is known as the *mosaic pattern*, and it applies to situations where a block does not blend with the surrounding blocks.

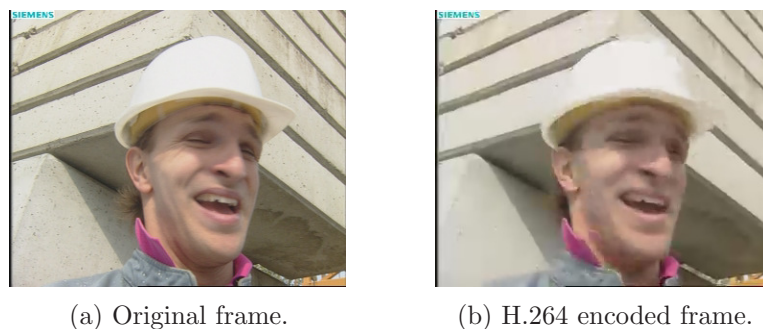


Figure 2.14: Blur effect – a frame taken from an H.264 encoded version of the *Foreman* sequence.

This phenomena typically occurs due to coarse quantization in blocks located inside textured objects.

2.4.2 Blur

Blur is characterized by loss of spatial detail in textured areas and edges of an image; it can have different sources; amongst them are: a badly focused camera, object motion at the moment of the image capture, low-pass filtering or lossy encoding.

Since the HVS is less sensible to changes in the higher spatial frequency image components, lossy encoding methods that operate in the frequency domain usually assign lower quantization steps to the low frequency coefficients, and larger quantization steps to the high frequency coefficients. In such cases, as quantization becomes coarser, high frequency coefficients tend to be quantized to 0. Consequently, high frequency spatial details, such as edges or textures, are subject to perceptible losses and blur manifests itself.

In older image and video lossy encoding standards, such as JPEG or MPEG-2, blur is usually masked by the presence of the blocking effect. However, in recent standards, such as JPEG2000 or H.264, blur is clearly perceptible at the lower bitrates. In JPEG2000, as the bitrate decreases, the number of DWT coefficients located in the detail sub-bands that are quantized to null values also increases. This situation means that spatial image details will be increasingly smoothed as the bitrate decreases, causing blur.

The main reason for blur in H.264 is the presence of the deblocking filter at the encoding/decoding processes. It applies low-pass filtering at the block's boundaries,

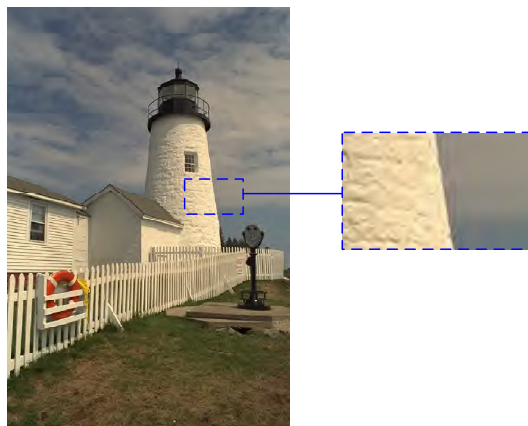


Figure 2.15: Ringing effect – *Lighthouse* image encoded using JPEG2000. Ringing is visible around the lighthouse edges.

reducing the block effect at the expense of adding blur to the decoded video frames. At low bitrates, the overall blur artifacts are less annoying than the block effect which would be present if the filter was not used. An example of blur in H.264 is illustrated in Figure 2.14.

2.4.3 Ringing

The *ringing* artifact is characterized by the appearance of spurious oscillations around edges or other image areas with an high local contrast. Most image and video encoding standards work in the frequency transform domain by representing the image as a sum of periodic signals, which are bandwidth limited. Those signals are suitable for representing smooth image transitions, but they are not adequate to represent fast transitions in image locations such as edges. Since lossy encoding in the frequency domain tends to concentrate signal energy in the low frequency components, high frequency components can be cut-off, and thus edges will in practice be represented as a sum of lower frequency signal components, causing small oscillations to be visible around stronger image signal transitions (*Gibbs* phenomenon).

Figure 2.15 illustrates the presence of ringing in a JPEG2000 encoded image. In block based encoding standards, ringing is usually masked by the blocking artifact (which is more annoying), but sometimes it is also noticeable, especially in the boundaries between smooth regions with high contrast between them.

2.4.4 Mosquito noise

The *mosquito noise* is a video coding artifact that Rec. ITU-T P.930 [59] defines as *a form of edge busyness distortion sometimes associated with movement, characterized by moving artifacts and/or blotchy noise patterns superimposed over the objects (resembling a mosquito flying around a person's head and shoulders).*

In short, there are two possible causes for the mosquito noise. The first cause is ringing effect associated to motion. In the presence of a small amount of motion, the oscillations caused by ringing can be seen as a flickering activity around the edges separating two different smooth regions.

The other cause for mosquito noise can be explained by considering that most video encoding standards allow different macroblock encoding modes. If they differ from frame to frame (for instance, in the previous frame is Intra mode while in the current is Inter mode), quantization errors introduced by a coarse quantizer may lead to significantly different pixel values. In these conditions, the corresponding reconstructed frames may exhibit flickering around the edges separating smooth image regions.

2.4.5 Jitter and jerkiness

Jitter and *jerkiness* are temporal video artifacts that are not directly caused by the specific encoding standard which is being used. Rather than that, they are due to temporal delays or the presence of transmission losses.

Jitter is characterized by an inconsistent frame freezing. This freezing may be due to different reasons. For instance, in the presence of transmission errors, instead of trying to decode corrupted video frames, the decoder may decide to repeat previously decoded frames until it receives error free content. In error free communications, *jitter* may also occur if the decoder is not efficient enough to decode video at the desired rate, skipping some frames in order to recover time. If the target application requires a very low bitrate, the encoder may also decide to drop frames in order to save bits, thus “embedding” jitter in the video stream.

Jerkiness is usually caused by the encoder, as a result of regularly skipping video frames to reduce the amount of information that is transmitted. This procedure may resulting in a more or less “jumpy” video sequence whose flow may resemble a sequence of snapshots instead of reproducing a natural and continuous scene flow.

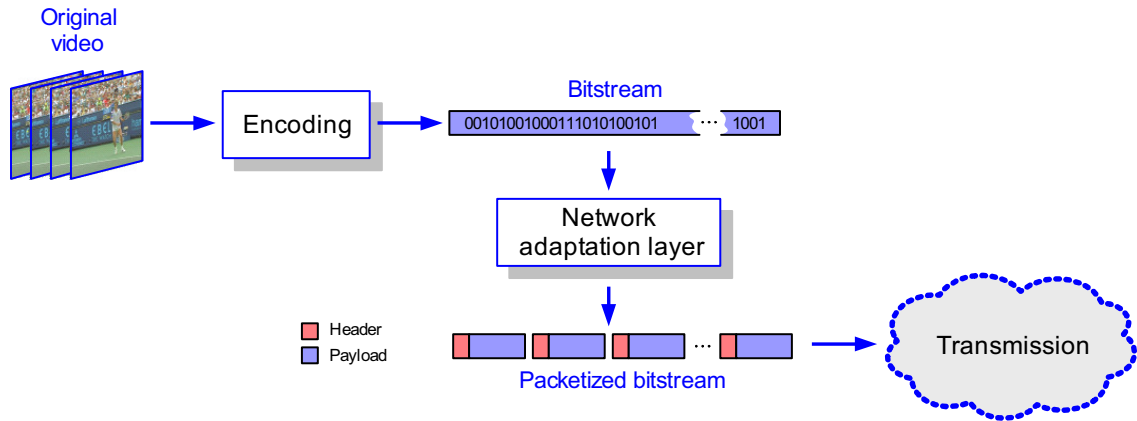


Figure 2.16: Packet-oriented video transmission.

2.5 Transmission losses

Encoded video data is usually transmitted over a packet-switched network, using a transport protocol such as *transmission control protocol / internet protocol* (TCP/IP). The network adaptation layer splits the bitstream into packets, adding an header to each packet. The header contains sequencing, timing and information about the payload, which contains the actual encoded video data. This process can be observed in Figure 2.16.

In packetized video transmission, losses typically occur due to delays caused by routing and queuing in the network elements (for instance, if the network is congested), or caused by the detection of corrupted packets and subsequent retransmission delay. There is also the possibility of receiving a corrupted packet without detecting the error, but this is a relatively rare situation nowadays. If the delay of a packet arriving at the decoder is too high, it is discarded, and thus it will produce the same effect as a missing packet.

The visual effects that will appear in the decoded video vary according to the type of information that was transmitted in a missing packet. For instance, and since the major video encoding standards are based on the concept of prediction, the loss of information related to a given macroblock will affect all macroblocks that depend on the corrupted macroblock. Thus, errors may propagate spatially, since data of a given block may depend on its surroundings, or temporally, since block values may depend on blocks belonging to previously decoded frames. In order to deal with error propagation issues, resynchronization points are usually included in the bitstream. These ensure that subsequent data located after a resynchronization



Figure 2.17: Packet loss effect – a frame taken from an H.264 encoded version of the *Foreman* sequence.

marker is decoded without using reference data located before the marker, and therefore error propagation is stopped.

The visual effect of transmission errors also depends on the ability of the decoder to deal with bitstream syntax errors: some decoders may never recover from certain errors while other decoders may cause jitter until error free content is received. It is also possible to use error concealment techniques in order to minimize the effects of transmission errors. The way the bitstream is encoded is also important: for instance, in the H.264 standard, an encoding procedure known as *flexible macroblock order* (FMO) tries to minimize the susceptibility to transmission errors by putting information of neighboring macroblocks in different packets of the bitstream. The benefit is that transmission errors may be spread more evenly across the video frame. However, the cost is coding efficiency, since dependent macroblocks are not predicted based on their neighbors, but from blocks that are probably less correlated.

Figure 2.17 depicts an example of the packet loss effect on the transmission of an H.264 encoded version of the *Foreman* sequence. In this example, two different effects of transmission losses can be observed: errors due to missing slices below the center of the frame (the correspondent packets were lost during transmission); propagation errors due to coding dependencies are noticeable in the upper region of the background. It can also be observed that the background of the region corresponding to the missing slices has been quite effectively recovered by the error concealment algorithm.

2.6 Summary

This chapter presented the basic aspects of human visual system. Its understanding is very important for image quality assessment algorithms. Besides that, the chapter also presented the basics of image and video coding, relating some of the existing methodologies with the characteristics of the human visual system. It ended with a description and discussion about the impact on image and video quality caused by the artifacts associated to the use of lossy encoding methods.

Chapter 3

Subjective quality assessment

3.1 Introduction

The subjective quality assessment procedures are those where the quality is evaluated by human viewers. In a subjective test, a set of images or videos is shown to the observer and then he is asked to judge the quality of the contents being displayed. Different methodologies for performing subjective quality assessment tests of video are described in Recommendations ITU-R BT.500 [5] and ITU-T P.910 [6]. The former describes the procedures for subjective quality evaluation for digital television, while the latter describes the procedures for evaluating video quality in multimedia applications. This chapter provides an insight on the subjective quality assessment procedures.

Subjective quality assessment tests are also used for supporting the development of objective quality metrics. The data resulting from the subjective assessment can be used to train and validate objective quality assessment algorithms. In the context of this Thesis, subjective quality assessment tests were organized in order to obtain MOS data for H.264 encoded video sequences. This data was used for validation of the no-reference video quality assessment algorithm that will be presented in Chapter 7.

This chapter is organized as follows: Section 3.2 discusses the most important details related to the preparation of a subjective quality assessment test. A brief overview of the current standards for subjective video quality assessment is given in Section 3.3. Section 3.4 depicts the details for computing MOS values based on the raw opinion scores resulting from the subjective assessment. The subjective

experiments performed in the scope of this Thesis are described in Section 3.5. The chapter concludes with a brief summary given in Section 3.6.

3.2 Subjective test preparation

In order to obtain reliable and repeatable MOS values, it is advisable to carefully plan the subjective test session. First of all, it is necessary to clearly define the objectives of the test session. Once the objective is defined, a proper selection of the material to be used in the tests is performed; the test room must be prepared and the test participants must be screened for vision problems and limitations (*e.g.*, color blindness and lack of visual acuity).

3.2.1 Selection of test video sequences

Since video contents vary significantly from sequence to sequence, the selection of the video sequences used in the tests is an important matter. To illustrate this issue, consider that the video sequences represented in Figure 3.1 are encoded at the same bitrate and using the same parameters. Sequence *Akyo* is a static video sequence and the background is very smooth, while sequence *Football* contains intense (and chaotic) motion and a significant amount of texture. If both were encoded at an average bitrate of 512 kbit/s (as an example), and using an H.264 encoder, the encoding would most likely produce annoying artifacts in the sequence *Football*, but sequence *Akyo* would still look very similar to the original (uncoded) sequence. Thus, different content may lead to different quality scores when subject to the same encoding procedure.

Ideally, all types of content should be considered in a subjective quality test. However, this variety is difficult to put in practice. An approximation can be performed using criteria similar to those defined in Rec. ITU-T P.910 [6], where the selection of video sequences is performed based on their spatial and temporal activities. The goal is to get a relatively small set of sequences that covers a wide range of content possibilities, avoiding long tests that could bother the participants. Additionally, the choice of the test material should be adequate for the target application. For instance, if the target application is video-conferencing it would be very unwise to choose sport or landscape video clips.

(a) *Akyo*.(b) *Football*.

Figure 3.1: Examples of typical test sequences. Sequence *Akyo* has low spatio-temporal activity, in contrast with sequence *Football*, where spatio-temporal activity is high.

3.2.2 Selection of test participants

Candidates for participating in subjective quality assessment tests can be classified as *experts* and *non-experts* observers. The experts are those familiar with image processing algorithms while the non-experts are those who represent the general consumer of video products.

The use of expert observers may lead to faster and easier test procedures, as pointed out in [1]. However, both video quality standards [5,6] recommend that at least 15 non-expert observers should be used for performing the assessment. Since the non-expert observer is not familiar with image processing algorithms, it is believed that his judgment on quality will not be biased. On the contrary, an expert observer will probably look for distortions in specific image locations, which would cause content awareness.

Regardless of the observers that are selected as potential test participants, it must be ensured that each observer has a normal visual acuity (or corrected to normal) and that he is not a color blind person. Visual acuity can be tested using a *Snellen* eye chart, such as the one represented in Figure 3.2-a). To detect most color blindness diseases, a set of *Ishihara* plates similar to the one depicted in Figure 3.2-b) can be used.

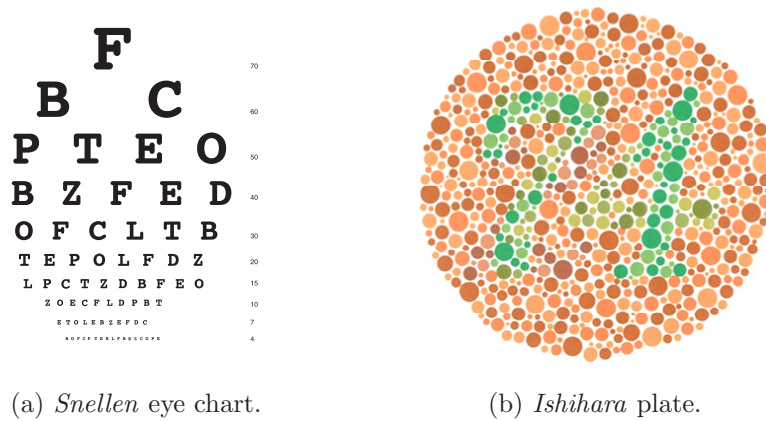


Figure 3.2: Subject screening tools.

3.2.3 Environment conditions

The test room environment can be defined as the set consisting of the display type and settings, the ambient light and the layout of the room's equipment and furniture.

Several parameters can be set according to the ITU standards P.910 and BT.500. In brief, the following guidelines should be followed: the ambient light should be set to low values (*e.g.*, ≤ 20 lux); no reflexes should be seen on the screen(s); the distance from the subject to the display should depend on the target application and on the type of display used (typical values in the range from $4H$ to $8H$, where H represents the image's height in the display).

3.3 Standardized methodologies

Different methodologies for subjective quality assessment are described in [5,6]. The choice of the methodology to adopt depends on the objectives defined for the test. In the following, an overview of the different methodologies proposed in the standards is provided.

3.3.1 *Double stimulus* methods

In a *double stimulus* quality assessment test, video sequences are organized and displayed in pairs. One of those sequences is called the *reference*, an high quality sequence which usually corresponds to the original video clip captured by the camera. The other sequence is the called the *test* or *impaired* video sequence, which

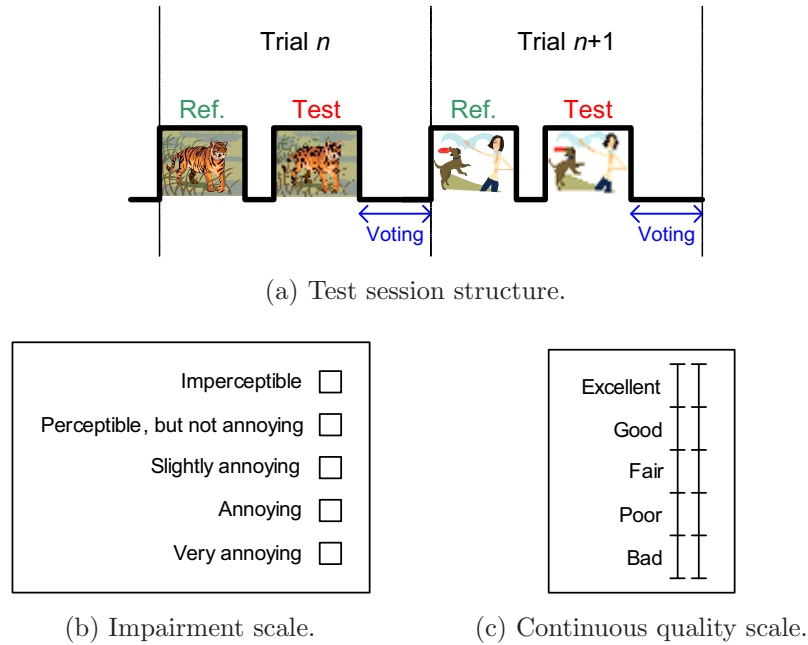


Figure 3.3: Double stimulus quality assessment method.

usually is a distorted version of the reference. During a session trial, the reference is displayed in the first place, with the purpose of acting as a benchmark. It follows the display of the test sequence and finally the observer is asked to judge the fidelity of the test with respect to the reference. An illustration of this procedure is depicted in Figure 3.3-a).

The standard ITU-R BT.500 [5] defines two classes of double stimulus subjective tests: the *double stimulus impairment scale* (DSIS) and the *double stimulus continuous quality scale* (DSCQS). The main difference between these methods is the quality scale used by the viewers for voting. Possible scales are represented in Figures 3.3-b) and c).

A methodology that is very similar to the DSIS method is described in the ITU-T P.910 standard [6] under the name *degradation category rating* (DCR). This methodology uses the categorical rating scale represented in Figure 3.3-b) and allows simultaneous display of both the reference and the test sequence, for lower resolution video formats (QCIF, CIF).

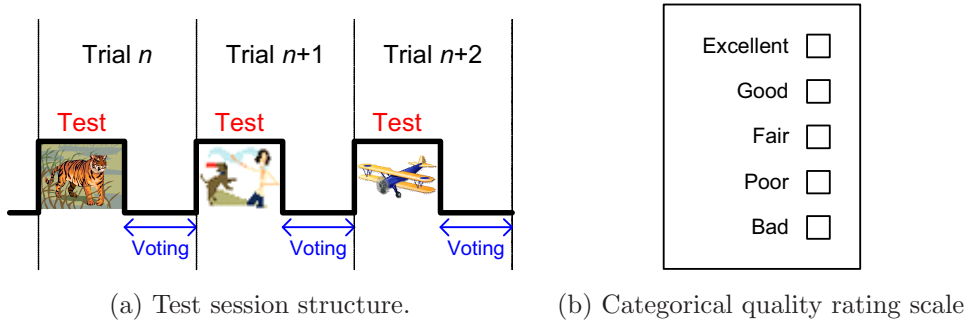


Figure 3.4: Single stimulus quality assessment method.

3.3.2 *Single stimulus methods*

In *single stimulus* methods, a single video sequence is presented on each trial. In this class of methods, there is no concept of reference sequences and thus only test sequences are presented to the observers. The structure of a single stimulus test session is represented in Figure 3.4-a). These methods are generally used for quantifying the quality of a system when no reference signals are available. The type of presentation on these methods is also very similar to what happens in a practical system implementation, *i.e.*, users will judge the quality of video without explicitly using a reference.

The standard ITU-R BT.500 [5] defines two variants for single stimulus test sessions: one variant where the set of test sequences is presented once, and another variant where the entire set is presented three times (but test sequences are displayed in different order on each presentation). The three-presentation variant is obviously more time consuming, but it's goal is to get more stabilized opinion scores: the first run is used for observer judgment calibration, and the opinion scores are computed based on the results of the second and third runs.

A method that corresponds basically to the first variant is also described in the ITU-T P.910 standard [6] using the designation *absolute category rating* (ACR).

In single stimulus tests, the observers express their opinions using quality scales that can be categorical, such as the one depicted in Figure 3.4-b), or continuous, such as the one used for DSCQS quality assessment, represented in Figure 3.3-c).

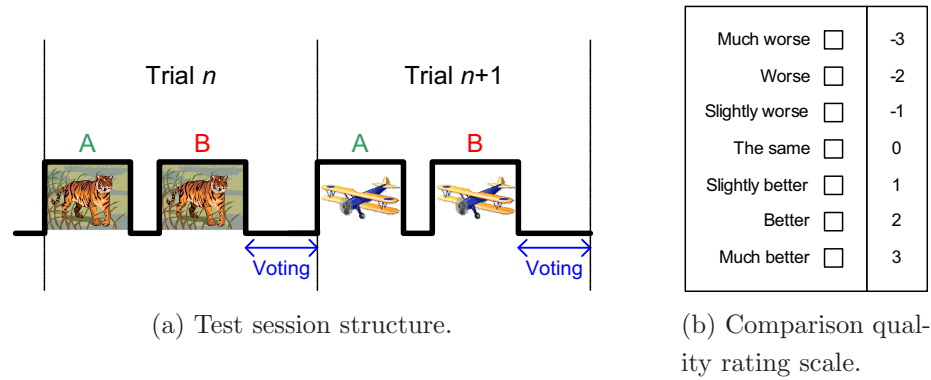


Figure 3.5: Comparison quality assessment.

3.3.3 Comparison methods

At some point, *comparison* methods are similar to double stimulus methods, in the sense that video sequences are also presented in pairs; however, there is no explicit use of a reference. Instead, two test sequences are directly compared. This class of methods can be used, for instance, to compare different video encoders (or different encoder parameters) with respect to their output's quality.

The structure of a comparison test session is depicted in Figure 3.5-a). At the end of each pair presentation, the participant is asked to judge the quality of sequence A with respect to sequence B. In order to express their opinions, the participants use a relative quality scale such as the one depicted in Figure 3.5-b).

3.3.4 Continuous quality evaluation methods

Unlike the methodologies described so far, where the participants are asked to express their opinions at the end of each trial, in *continuous quality evaluation* methods the viewers continuously express their opinions along the presentation of the test video sequences.

In order to feed the measurement system with their judgments, the subjects continuously adjust the position of an hand held slider device, such as the one represented in Figure 3.6-a). An example layout for the equipment used in this type of tests is represented in Figure 3.6-b). The slider devices can be connected to a PC which is responsible for synchronizing the participants' quality scores with the video sequence under display. Before obtaining the corresponding MOS values, judgment delays due to the reaction time of the test participants must be compensated.

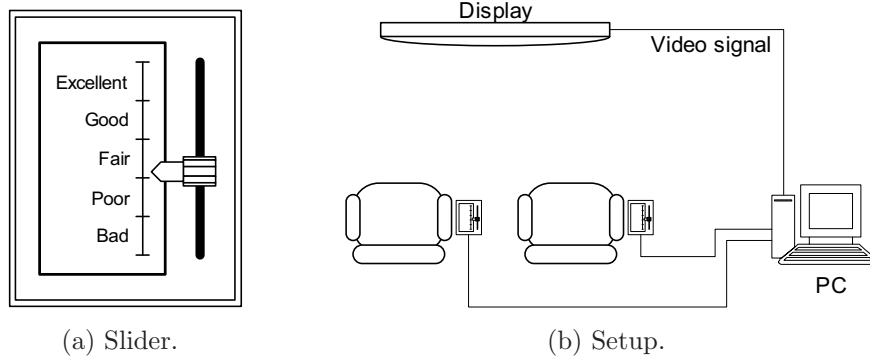


Figure 3.6: Continuous quality evaluation.

The main motivation for these methods is the fact that the impairments found in digitally encoded video are time-varying and scene-dependent. Even in a short video clip, the perceived quality of the encoded video may vary significantly.

Continuous quality evaluation methods can be further divided into two classes: *single stimulus continuous quality evaluation* (SSCQE) and *simultaneous double stimulus for continuous evaluation* (SDSCE). As the designations suggest, in the first only test sequences are presented, while in the second a pair reference-test is presented to the viewer at the same time (in the same display, or in aligned displays).

3.4 Computing mean opinion scores

In order to detect and reject inconsistent opinion scores, raw quality scores collected during the subjective experiment should be subject to an additional post-processing procedure, such as the one defined in ITU-R Rec. BT.500 [5].

This standardized procedure starts by computing the average and the standard deviation of observers' scores for each test condition. Assuming that L different test conditions were presented for judgment during the subjective quality assessment test, and that each of them was evaluated by N observers, the average score for test condition i , $\mu(i)$, and the corresponding standard deviation, $\sigma(i)$, are defined as:

$$\mu(i) = \frac{1}{N} \sum_{j=1}^N \varphi(i, j) \quad \text{and} \quad \sigma(i) = \sqrt{\frac{1}{N-1} \sum_{j=1}^N [\varphi(i, j) - \mu(i)]^2}, \quad (3.1)$$

where $\varphi(i, j)$ is the score given by the observer j to the test condition i . Note that the resulting values for $\mu(i)$ can be seen as the raw MOS values. In order to check

for the consistency of the opinion scores given by observer j , it is verified if its voting score, $\varphi(i, j)$, is in the interval:

$$[\mu(i) - \delta(i), \mu(i) + \delta(i)], \quad (3.2)$$

where $\delta(i)$ is a margin that depends on the distribution of the score values and the number of observers. In ITU-R Rec. BT.500, two possibilities for $\delta(i)$ are defined, depending on the statistical distribution of the individual scores given to each test condition. The procedure analyses the shape of this distribution by computing the “kurtosis” proper coefficient, β_2 , which is the ratio between the fourth central moment and the square of the second central moment:

$$\beta_2(i) = \frac{m_4(i)}{m_2(i)^2}, \quad \text{with} \quad m_n(i) = \frac{1}{N} \sum_{j=1}^N [s(i, j) - \mu(i)]^n. \quad (3.3)$$

Based on the outcome of the β_2 test, $\delta(i)$ is defined as:

$$\delta(i) = \begin{cases} \frac{2\sigma(i)}{\sqrt{N}}, & \text{if } 2 \leq \beta_2(i) \leq 4, \\ \frac{2\sqrt{5}\sigma(i)}{\sqrt{N}}, & \text{otherwise.} \end{cases} \quad (3.4)$$

It is worth to mention that, for the normal distribution case, the theoretical outcome of the β_2 test would be equal to 3. Therefore, the first possibility for $\delta(i)$ corresponds to the case where the score’s distribution for test condition i is considered to be normal. In such case, the value of $\delta(i)$ defines an interval that is very close to the 95% confidence interval.

Next, for each observer j , the number of times a quality score exceeds the upper limit of the confidence interval, $P(j)$, is accounted for. Formally, it can be written:

$$P(j) = \sum_{i=1}^L p(i, j) \quad \text{with} \quad p(i, j) = \begin{cases} 1, & \text{if } \varphi(i, j) \geq \mu(i) + \delta(i), \\ 0, & \text{otherwise.} \end{cases} \quad (3.5)$$

Similarly, $Q(j)$ is the number of times that a quality score given by observer j is below the lower limit of the confidence interval:

$$Q(j) = \sum_{i=1}^L q(i, j) \quad \text{with} \quad q(i, j) = \begin{cases} 1, & \text{if } \varphi(i, j) \leq \mu(i) - \delta(i), \\ 0, & \text{otherwise.} \end{cases} \quad (3.6)$$

After computing $P(j)$ and $Q(j)$ for each observer, rejection of observer j is carried

out if the following conditions are both met:

$$\frac{P(j) + Q(j)}{L} > 0.05 \quad \text{and} \quad \left| \frac{P(j) - Q(j)}{P(j) + Q(j)} \right| < 0.3. \quad (3.7)$$

The first condition is related to the number of times that the quality scores given by an observer were considered as outliers. If the rate of 5% is exceeded, this condition is met. The second condition is related to the outlier's behaviour: it tends to 0 if the scores are dispersed in both tails of the quality scores distribution. If the outliers are concentrated only in one of the distribution's tails, it means that the participant's quality judgements are biased, but not inconsistent. In practice this procedure detects and rejects test participants that voted randomly, or didn't correctly use the quality scale.

Finally, after rejecting participants with inconsistent quality scores, MOS values for test condition i are computed according to:

$$\text{MOS}(i) = \frac{1}{N_v} \sum_{j=1}^{N_v} \varphi(i, j), \quad (3.8)$$

where N_v is the number of valid test participants.

3.5 Subjective quality assessment tests

3.5.1 Methodology

The subjective quality assessment tests were performed in accordance with Recommendation ITU-T P.910 [6]. The method followed in this Thesis was the *degradation category rating* (DCR), which corresponds to the *double stimulus impairment scale* (DSIS) method described in [5] – see Section 3.3.1 for additional details. In short, the observer is presented with video sequences organized in pairs, as illustrated in Figure 3.3-a): the first to be displayed is the reference sequence (the original sequence) while the second is the test or impaired sequence (in this case, the result of lossy encoding); the five grade impairment scale depicted in Figure 3.3-b) has been used for collecting the observers' votes.

Height of the picture shown in the screen	8 cm
Viewing distance	64 cm
Background room illumination	13.45 lux
Peak luminance of the LCD screen	95.8 lux
Luminance of inactive screen	2.23 lux
Luminance of background behind the display	10.15 lux
Ratio of luminance of inactive screen to peak luminance	0.023
Ratio of luminance of background to peak luminance	0.14

Table 3.1: Environmental viewing conditions.

3.5.2 Assessment conditions

According to [6], at least 15 observers are needed in order to produce reliable results. In our case, 42 observers (mostly students) participated in the subjective experiments, and it was ensured that each impaired sequence was judge by at least 20 observers. The observers were screened for visual acuity and color blindness, using a Snellen Eye Chart and Ishihara’s plates, respectively. The duration of each session was about 20 minutes with the room setup allowing two observers to simultaneously participate in each session.

As for the environmental viewing conditions, three factors must be considered: the lighting, the ambiance noise and the quality and calibration of the display. Two high quality LCD displays of the same model (ASUS VW193S 19” Wide) with native resolutions of 1440×900 pixels have been used and they were calibrated in order to achieve the same test parameters. The display and room characteristics used in the subjective tests, listed in Table 3.1, are within the values recommended in [6].

3.5.3 Selection of test material

In order to avoid boring the observers and get meaningful results, a small, but representative, set of video sequences should be used during the tests. In particular, the spatial and temporal activities are important parameters which should be considered when choosing the test sequences – a set of sequences that span a wide range of values for those activities should be chosen. The literature provides several meth-

ods of measuring a video spatial and temporal activity. In this work, the methods recommended in [6] have been used:

- **Spatial activity:** the horizontal and vertical picture gradient are computed using the well known Sobel filters. The gradient norm (the square root of the sum of the vertical and horizontal gradient squares) is then computed for each pixel. The standard deviation of the gradient norm is calculated for each frame, resulting in a time series of frame-by-frame spatial activities. In order to achieve a global value for the spatial activity, the maximum value in the time series is selected.
- **Temporal activity:** the temporal activity measure is obtained by computing the difference, pixel-by-pixel, between each pair of successive frames. After this procedure has been carried out, the standard deviation of the frames differences is computed. Similarly to what happens in the spatial activity case, the global temporal activity value is computed as the maximum of these standard deviations.

Due to changes of the camera perspective during video acquisition or scene changes, the global activity measurements could have a high value even if the sequence has a low temporal and/or spatial activity. In order to minimize this effect, the global activity values result from applying the 95% percentile to the temporal and spatial activities series, instead of using its maximum.

Figure 3.7 represents the video sequences used in the subjective tests. They have been selected based on their spatio-temporal activities, whose values are depicted in Figure 3.8. These sequences are in CIF format (352×288 pixels), with a frame rate of 30 Hz.

The sequences were encoded using the reference H.264 [58] software tools. Each sequence has been encoded at different bit rates, in the range of 64 to 2048 kbit/s, using the *Main Profile*¹. The resulting bitrates at the encoder's output are summarized in Table 3.2 (the upper row associated to each video sequence). A GOP-15 structure with two *B* frames inserted between *I/P* frames (*IBBPBBP...*) has been used in all encoding runs. The result is a set of 50 encoded sequences (impaired sequences), whose qualities were judged by the test participants. This set allows

¹An H.264 *profile* is basically a set of coding tools that are used for generating a conforming bitstream. The most relevant characteristics of the *Main Profile*, in the scope of this Thesis, are: the possibility of using B frames and the use of the 4×4 sized transform only.

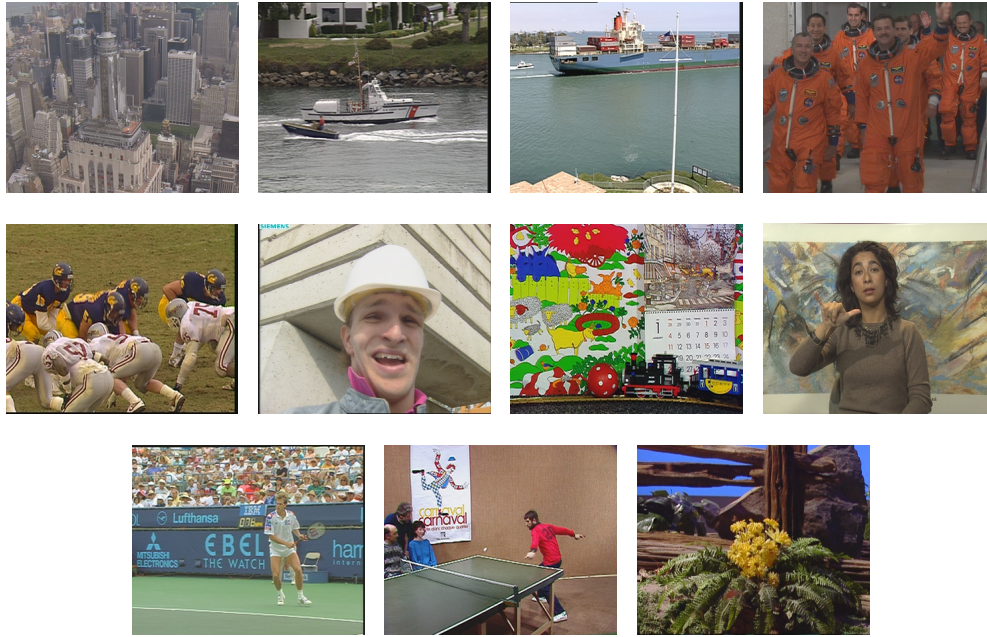


Figure 3.7: Video sequences selected for the subjective tests. From left to right: *City*; *Coastguard*; *Container*; *Crew*; *Football*; *Foreman*; *Mobile & Calendar*; *Silent*; *Stephan*; *Table-tennis*; *Tempete*.

to evaluate the human visual system perception to different kinds of video qualities and to indirectly force the observers to use all grades of the rating scale.

3.5.4 MOS computation

The mean opinion scores were computed at the end of the test sessions, based on the image quality assessment results given by all observers. In order to guarantee the coherence and the consistency of the results provided by the subjective tests, the statistical analysis described in Section 3.4 was applied to the assessment results. For each test condition, MOS values were computed by averaging the quality scores of the coherent observers, only.

The resulting MOS values, together with their associated standard deviation values, are depicted in Figure 3.9 (for better display, MOS values have been sorted). The correspondence between the MOS values and the sequences used in the subjective tests, for the considered bitrates, is given in table 3.2 (the lower row associated to each video sequence).

The resulting MOS values and the video sequences used in the subjective quality

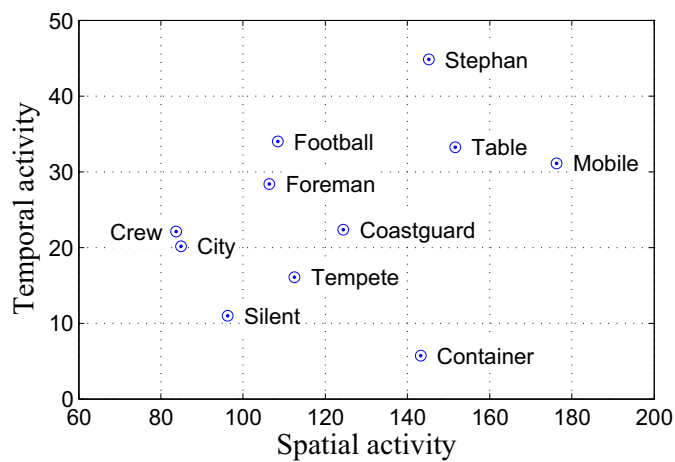


Figure 3.8: Spatio-temporal activity of the selected video sequences.

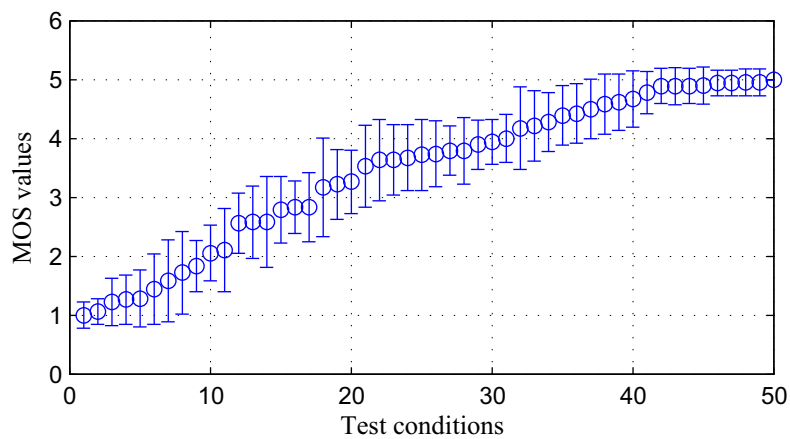


Figure 3.9: MOS values resulting from the subjective quality assessment tests.

Sequence	Bitrates (kbit/s) / MOS					
City	128.2	200.2	256.3	512.4		
	3.26	3.63	4.21	4.90		
Coastguard	65.7	100.2	131.3	200.4	262.6	524.7
	1.83	2.79	3.17	3.90	4.39	4.61
Container	65.7	131.2	262.1	524.4		
	3.67	3.72	4.78	4.94		
Crew	127.9	200.0	400.2	1024.1		
	1.26	2.05	3.74	4.95		
Football	263.8	401.7	526.3	756.3	1050.5	2104.5
	1.72	2.58	3.22	3.79	4.00	4.94
Foreman	131.3	262.5	525.4	1051.0		
	1.06	2.83	4.17	4.89		
Mobile	131.3	262.3	524.5	1048.9		
	1.28	1.44	3.94	4.67		
Silent	64.2	200.5	400.6	1025.2		
	1.58	3.79	4.58	5.00		
Stephan	140.1	200.4	263.0	401.2	525.0	1049.9
	1.00	2.11	2.56	3.63	4.28	4.89
Table	65.6	131.5	263.3	525.0		
	1.22	2.83	4.50	4.89		
Tempete	130.2	202.1	405.4	756.2		
	2.58	3.53	4.42	4.95		

Table 3.2: Resulting bitrates and MOS values for the sequences used in the tests.

assessment tests (both the reference sequences and the encoded bitstreams) are available online at http://amalia.img.lx.it.pt/~tgsb/H264_test/.

3.6 Summary

This chapter presented the main concepts associated with subjective quality assessment. It started by discussing several aspects related to the preparation of subjective quality assessment tests. Afterwards, a brief overview of the standardized subjective test methodologies was provided. It also presented the statistical computations (suggested in the standards) required for obtaining MOS data from the opinion scores collected in the test sessions.

Finally, the subjective quality assessment tests performed in the context of the Thesis were presented. The aim of those tests was to obtain MOS data for video sequences subject to H.264 encoding, which were used for validating the no-reference video quality assessment method that will be presented in Chapter 7.

Chapter 4

Objective quality assessment metrics

4.1 Introduction

In the previous chapter, standardized subjective quality assessment procedures have been described. An alternative to subjective quality measurements is the use of objective quality metrics. As already mentioned, objective metrics aim to automatically predict the viewers' MOS that would result from a subjective assessment.

This chapter categorizes objective quality assessment metrics and provides an overview of the state-of-the-art on this topic. In Section 4.2, objective quality assessment metrics are organized into classes. An overview of the algorithms proposed in the literature is then presented in Sections 4.3 to 4.5. The existing standardized objective quality assessment metrics are reviewed in Section 4.6. Section 4.7 presents the indicators that are usually used for performance evaluation of an objective quality assessment metric. To conclude, a summary is provided in Section 4.8.

4.2 Classifying objective quality metrics

Objective quality assessment metrics can be classified according to the amount of information that is required for computing the quality scores. Using this criterion, three classes of objective metrics can be specified [60]:

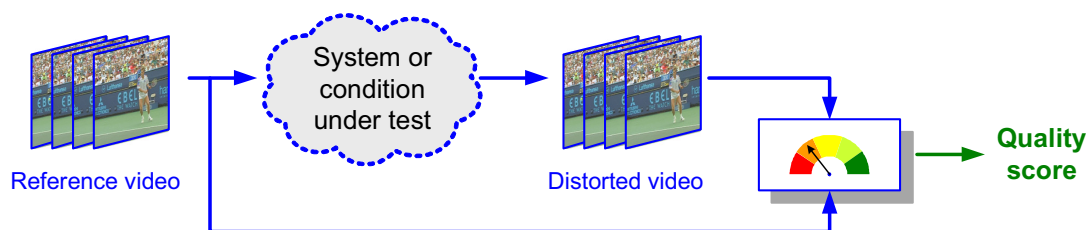


Figure 4.1: Full reference quality assessment system.

- **Full reference (FR)** – the original media data is required at the quality evaluation system;
- **No-reference (NR)** – quality scores are computed using the distorted media, only;
- **Reduced reference (RR)** – the quality evaluation system uses the distorted media and additional information about the original media data.

Figure 4.1 depicts the general structure of a full reference quality evaluation system. Since the reference (original) media data is required, the applicability of FR quality metrics in video communication scenarios is very limited. Besides that, in order to use a full reference metric, the reference and the distorted video sequences must be correctly aligned, in such a way that all pixels in a given frame of the reference match their corresponding locations in the distorted frame. This alignment operation is not easy to implement if spatial and temporal video signal scalability is also considered in the scenario.

Nevertheless, full reference metrics are playing an increasingly important role for benchmarking image processing algorithms. For instance, to compare the quality of video encoded with different codecs, to evaluate the performance of artifact reduction algorithms or to confirm the imperceptibility of watermarks. In the context of video broadcasting, they can also be applied for quality-oriented testing and planning. However, they cannot be used for quality monitoring at the user end, since the reference data is usually not available.

The classical *peak signal-to-noise ratio* (PSNR) metric, which is based on the squared differences between the reference and distorted images or video sequences, is probably the most known example of a full reference quality metric.

No-reference image quality assessment systems are those that compute quality scores without using any knowledge about the original signals. This class of image qual-

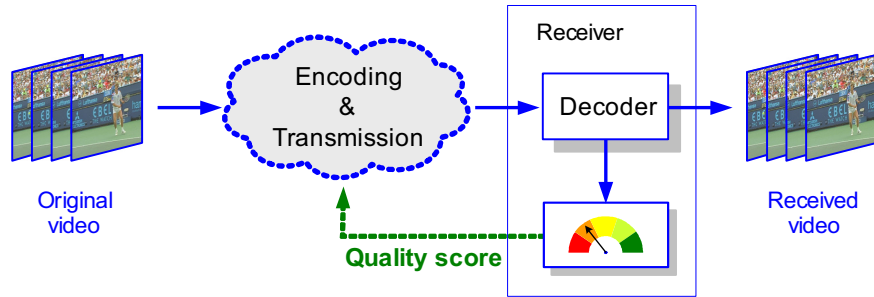


Figure 4.2: No-reference quality assessment system.

ity assessment metrics are the most adequate for image and video communication systems, due to their ability to compute quality scores based on the distorted media only, as represented in Figure 4.2. The use of no-reference quality metrics enables network-oriented applications such as automatic QoE monitoring, real-time adjustment of video streaming parameters as a function of the perceived quality and scalable billing schemes (*i.e.*, users paying in proportion to the quality they get).

However, since they use less information, the development of quality metrics in the NR class is more difficult than the development of FR metrics. In order to workaround this increased difficulty, quality assessment algorithms within the NR class typically make some assumptions about the sources of distortion. Since the typical scenario for using NR metrics is a video communication network, it is reasonable to consider the lossy encoding methods that are used, as well as the properties of the transmission channel. Due to those assumptions, classical approaches to this class of metrics usually try to estimate artifacts that result from lossy video encoding (*e.g.*, block effect) and/or transmission errors (*e.g.*, jitter).

Reduced reference metrics can be placed somewhere between FR and NR metrics. In a reduced reference metric scenario (see Figure 4.3), the content provider also sends additional information that depends on the original data under transmission.

When compared with NR metrics, RR metrics require an additional channel (or additional bandwidth) to transmit the side information, and the presence of additional algorithms for generating the side information data at the server side. Generally, side information data consists of video features, such as edge maps or spatio-temporal activity measurements. These features are extracted from the original video, transmitted through the side channel and compared with the same features extracted from the degraded media, at the receiver. Quality scores provided by the algorithms in this class of metrics depend on such comparison.

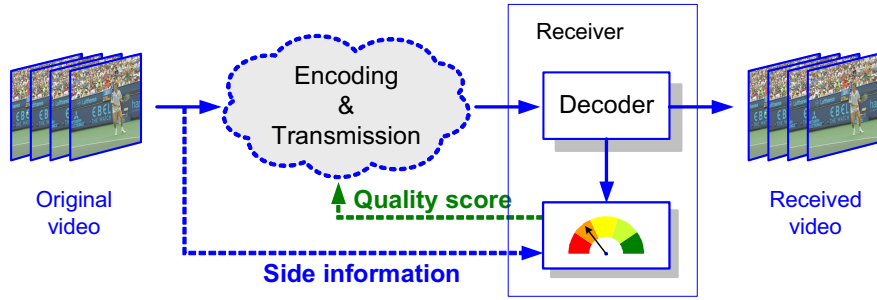


Figure 4.3: Reduced reference quality assessment system.

The type and amount of information that is transmitted through the side channel is strongly dependent on the algorithm's design. Similarly to what happens in the FR class, reduced reference metrics must also deal with alignment issues, since side information and received media data must be synchronized. In transmission scenarios, the possibility of losses in the side information required for computing the metric's score should also be considered.

Another possibility for classifying objective quality assessment algorithms is to consider the type of data that is used for the quality measurement. Using this criterion, objective metrics can be classified into *data metrics*, *picture metrics* and *bitstream-based metrics* [61]:

- ***Data metrics*** – quality scores are based on the pixel values of the image or video, without explicitly considering its content;
- ***Picture metrics*** – the visual information in the image or video is explicitly considered for quality assessment;
- ***Bitstream-based metrics*** – the quality assessment system uses information extracted from the encoded bitstream, without fully decoding the image or video.

Data metrics usually belong to the FR metrics class, since they are typically based on the comparison of pixel values (thus requiring the reference). Depending on the approach that is followed, existing picture metrics can be within the three metric classes – FR, RR and NR. As for bitstream-based metrics, since they are designed for transmission scenarios, they usually fall into the NR or RR metrics category.

4.3 Data metrics

Quality scores resulting from data metrics are those computed without explicitly considering the content of the image or video. Data metrics usually take pixel values from the reference and the distorted signals, compare them (*e.g.*, by computing their differences) and then combine all pixel-by-pixel comparisons into a single image or video quality index.

The most commonly used data metric is the PSNR. In the context of image processing, PSNR is a popular full reference quality metric, especially among the image and video coding communities. Formally, it is defined as:

$$\text{PSNR (dB)} = 10 \log_{10} \frac{L^2}{\text{MSE}}, \text{ with } \text{MSE} = \frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2, \quad (4.1)$$

where y_k and \hat{y}_k represent the values of the k -th pixel on the reference and distorted images, respectively, L is the maximum pixel value per color component (usually 255), N is the number of pixels, and MSE stands for *mean squared error*.

The simplicity of equation (4.1) is one of the main reasons for the generalized use of the PSNR; another reason for the popularity of the PSNR is its clear mathematical meaning. Since it is based on the MSE, it can be easily applied in minimization or optimization problems (*e.g.*, rate control). Besides that, another reason for using PSNR is the fact that the existing standards for objective video quality metrics are very recent, and thus their use is not yet spread among the video community; since this community is familiar with the PSNR, it tends to continue to use it.

However, the PSNR is known not to correlate well with the human perception of quality [62, 63]. When looking into (4.1), it is easily understood that PSNR treats all errors in the same way, regardless their image context. Thus, the PSNR is completely blind to the way the human visual system perceives image errors.

An example that illustrates the above statement is given in Figure 4.4. Two versions of the *House* image have been generated in such a way that similar PSNR is obtained in both cases. Figure 4.4-b) depicts the result of JPEG encoding (with the quality factor set to 15) while Figure 4.4-c) is the result of corrupting the original image with uniform random noise. When comparing both corrupted images, the perceptual quality in Figure 4.4-c) is better than the one of Figure 4.4-b), although the PSNR of the latter is in fact higher.

Other quality metrics based on the difference between pixel values have been pro-

(a) Original *House* image.(b) JPEG encoded with QF=15
(PSNR=24.33 dB).(c) Corrupted with additive noise
(PSNR=24.28 dB).

Figure 4.4: Images with similar PSNR, but different perceptual quality impact.

posed and tested [64]. In general, metrics based on pixel differences may be effective for a specific distortion source, but they usually fail across different distortion sources. Since they also require the reference image or video, the range of applications is also restricted.

In [65], it is pointed out that PSNR and the generality of data metrics are *distortion-agnostic* and *content-agnostic*. Being distortion-agnostic, it means that data metrics are blind to the distortion sources. For instance, they deal with noise resulting from low-pass filtering the same way they deal with structured noise such as the block effect. Being content-agnostic means that data metrics do not consider image content such as textures and edges, or video content such as motion.

A data metric that can be viewed as a possible exception is the increasingly popular *structural similarity index* (SSIM). SSIM is a full reference metric that was originally proposed by Wang *et al.* in [66, 67], for assessing the quality of still images subject to different distortion sources. Although not considering image content explicitly, image quality scores resulting from the SSIM metric have shown a high correlation

with MOS values. Besides that, SSIM works well across different image distortion sources.

MOS predictions computed by SSIM are based on three different measurements, computed across image blocks (or using a sliding window across the image). Basically, these measurements are indicators for luminance, contrast and structure comparison between the two image regions under analysis. Representing by $l_j(y, \hat{y})$, $c_j(y, \hat{y})$ and $s_j(y, \hat{y})$ the luminance, contrast and structural terms computed in the j -th image block, they are defined as:

$$l_j(y, \hat{y}) = \frac{2\mu_y\mu_{\hat{y}} + K_1}{\mu_y^2 + \mu_{\hat{y}}^2 + K_1}; \quad c_j(y, \hat{y}) = \frac{2\sigma_y\sigma_{\hat{y}} + K_2}{\sigma_y^2 + \sigma_{\hat{y}}^2 + K_2}; \quad s_j(y, \hat{y}) = \frac{\sigma_{y\hat{y}} + K_3}{\sigma_y\sigma_{\hat{y}} + K_3}, \quad (4.2)$$

where μ_y and $\mu_{\hat{y}}$ represent the mean of the pixel values at the j -th block of the reference image y and of the distorted image \hat{y} , respectively; σ_y and $\sigma_{\hat{y}}$ are the corresponding standard deviations; $\sigma_{y\hat{y}}$ is the standard deviation of the joint term $y\hat{y}$. K_1 , K_2 and K_3 are small valued constants add for stability (they avoid that a term goes to infinity).

For each image block, these three measurements are combined into a single local measurement, according to:

$$\text{SSIM}_j(y, \hat{y}) = l_j(y, \hat{y}) \times c_j(y, \hat{y}) \times s_j(y, \hat{y}). \quad (4.3)$$

To complete, an overall image quality index, the $\text{SSIM}_{\text{global}}$, is computed by averaging all local SSIM values:

$$\text{SSIM}_{\text{global}} = \frac{1}{M} \sum_{j=1}^M \text{SSIM}_j(y, \hat{y}), \quad (4.4)$$

where M is the number of individual block-wise SSIM measurements.

An extension of the SSIM metric to video was proposed in [68]. Basically, SSIM indexes are computed on a frame-by-frame basis and then weighted according to the motion properties of the video sequence. The result of this weighting is a quality index for the whole video sequence. Since the video SSIM metric also uses motion features, it is closer to a picture metric than to a data metric.

4.4 Picture metrics

Picture metrics are those which explicitly consider the contents of an image or video. According to the approach that is followed for deriving the metric, picture metrics may be classified according to three classes:

- ***Psychophysical-based metrics*** – these metrics model the characteristics of the human visual system, using data collected from psychophysical experiments. Typical models include contrast sensitivity functions, temporal masking and color perception. It is also worth to mention that most (if not all) objective quality metrics that follow this philosophy fall into the full reference class.
- ***Artifact measurement metrics*** – metrics within this class are usually much simpler than those following the psychophysical approach. However, they are designed for specific applications and must make assumptions about the sources of distortion that will affect the images or video. Instead of modeling the human visual system, the approach is to select a set of artifacts that result from a specific encoding method or specific transmission errors, measure and combine them in order to fit data collected from subjective quality assessment experiments. Most of the no-reference objective quality metrics found in literature follow this approach.
- ***Feature-based metrics*** – similarly to artifact measurement metrics, feature-based metrics are also simpler than those following the psychophysical approach and designed for a specific distortion source. The main difference is that feature-based metrics use image or video content characteristics, such as spatial activity measurements, edge maps, temporal activity measurements or motion vectors, and lower level data, such as video bitrate, and combine them in order to fit subjective quality data. Feature-based metrics can be found within the FR, RR and NR classes, but most of the work found on literature is oriented to the RR and NR classes.

4.4.1 Psychophysical-based metrics

Contrast sensitivity functions

As already discussed in Chapter 2, the response of the human visual system to a visual stimulus, located at a given image pixel, strongly depends on the relation between the luminance at that pixel and at its surrounding pixels. A *contrast sensitivity function* (CSF) quantifies the HVS response to local luminance changes, both in space and in time. Contrast sensitivity can be defined as the inverse of the minimum contrast that is necessary for an observer to detect a stimulus. The most significant research on spatio-temporal CSFs is due to Kelly [69] and Daly [41]. In their work, the spatio-temporal sensitivity is computed as a function of the spatial frequency, f_s , and the retinal velocity, v_R :

$$\text{CSF}(v_R, f_s) = S c_0 c_2 v_R (2\pi c_1 f_s)^2 \exp\left(-\frac{4\pi c_1 f_s}{f_{max}}\right), \quad (4.5)$$

with the terms S and f_{max} defined as:

$$S = \left(s_1 + s_2 \left|\log\left(\frac{c_2 v_R}{3}\right)\right|^3\right) \text{ and } f_{max} = \frac{p_1}{c_2 v_R + 2},$$

where s_1 , s_2 and p_1 are constants; c_0 , c_1 and c_2 are parameters that allow model tuning. An illustration of the contrast sensitivity function by Kelly and Daly was already depicted in Figure 2.5.

Spatial frequency depends on the observation angle of a pixel, which is given by parameters external to the video, such as the distance of the observer to the screen, the resolution of the display or the size of picture on the screen. The object velocity in the retina plane is related to the object velocity in the image plane. In [41], the object velocity in the retina plane is given by the angular velocity of the object on the image plane, compensated with a term associated to eye movements.

Contrast sensitivity functions can be used for weighting image errors. For instance, since the CSF by Kelly and Daly operates on the frequency domain, it may be used for weighting the differences between reference and distorted DCT coefficients.

Another important work that follows a psychophysical approach to image quality assessment is the perceptual model proposed by Watson in [8]. This model lead to the development of the *DCTune* [70] algorithm, which is a perceptually adapted distortion metric for deriving optimized quantization matrices in the context of JPEG encoding.

The model by Watson computes the perceptibility of modifications in 8×8 DCT coefficients in terms of *just noticeable differences* (JNDs), whose threshold values are called *slacks*. Each slack value is computed by considering luminance adaptation and spatial contrast masking.

After obtaining the slack values, the local errors between the reference and distorted coefficient values are computed. The perceptual error associated to each quantized DCT coefficient, $\varepsilon_{p_k}(i, j)$, is computed as the ratio between the error, $\varepsilon_k(i, j)$, and its corresponding slack, $s_k(i, j)$:

$$\varepsilon_{p_k}(i, j) = \frac{\varepsilon_k(i, j)}{s_k(i, j)}. \quad (4.6)$$

Similarly to what was discussed in Section 2.2.4, local perceptual errors are combined using a Minkowski summation (or L_p -norm) in the form $(\sum \varepsilon^p)^{\frac{1}{p}}$, resulting in a global distortion measurement for the whole image. Studies from Watson [8, 46] and Lambrecht [47] suggest that the exponent p may be set to 4 in order to emphasize the fact that higher distortions may draw the viewer's attention, having a stronger impact on the global perception of quality. Thus, a global distortion metric, D_W , is computed by combining all perceptual errors using L_4 error pooling:

$$D_W = \sqrt[4]{\frac{1}{M} \sum_{k=1}^N \sum_{i=0}^7 \sum_{j=0}^7 \varepsilon_{p_k}(i, j)^4}, \quad (4.7)$$

where M is the total number of coefficients under analysis and N is the number of coefficients per frequency (which is same as the number of 8×8 blocks in the image).

Multi-channel metrics

The human visual system exhibits a large number of neurons in the primary visual cortex that work as oriented band-pass filters [71]. This fact motivated the development of several *multi-channel* approaches to the image quality assessment problem. Examples of multi-channel perceptual models are the *Sarnoff just noticeable difference vision model* [72, 73] by Lubin, the *moving picture quality metric* (MPQM) [74] by Lambrecht and Verscheure, and the *perceptual distortion metric* (PDM) by Winkler [75, 76].

Generally, multi-channel models operate on both the luminance and chrominance components of an image or video. After a color space conversion step (usually a

conversion to the YCbCr color space), each color component is subject to a “perceptual” decomposition. Depending on the model, this decomposition can either be performed using Gabor filters [74], steerable pyramid decomposition [75, 76] or Gaussian and Laplacian pyramid decompositions [72, 73].

The result of the decomposition is a set of sub-band signals (or channels) with different resolutions and orientations that try to mimic the mechanisms of the human visual system. These sub-band signals are subject to spatial filtering and contrast measurements in order to obtain a contrast masking map for each sub-band. If the metrics are to be applied to video, temporal filtering processes are also included in the model. The resulting masking maps are used for error weighting at each sub-band. Weighted errors are combined in a pooling process that computes a global quality measurement.

As can be concluded from this short description, quality metrics based on multi-channel models implement most known features of the HVS, thus they have the great potential in providing quality scores that correlate well with the subjective assessment scores. However, their complexity (both for implementation and for computing) is very high, which may be a drawback for their generalized use.

4.4.2 Artifact measurement metrics

Quality metrics based on artifact measurements are typically designed for no-reference quality assessment of images and video. These metrics are designed keeping in mind that images or videos are subject to an encoding method that may cause the visibility of specific compression artifacts (such as those described in Section 2.4). The main motivation for their use is the fact that most artifact measurements correlate well with the perception of quality.

Block effect measurement

Since most image and video encoding standards are block-based (*e.g.*, JPEG, MPEG-2, H.26x), it is not a surprise that the majority of quality metrics based on artifact measurements are oriented to the block effect. Examples of such metrics are published in [15, 18, 22, 23].

Probably the first quality metric that quantifies the block effect is the work by Wu and Yuen in [15]. It starts by computing the horizontal and vertical differences of pixel values located at the boundaries of 8×8 blocks. Then, it weights those

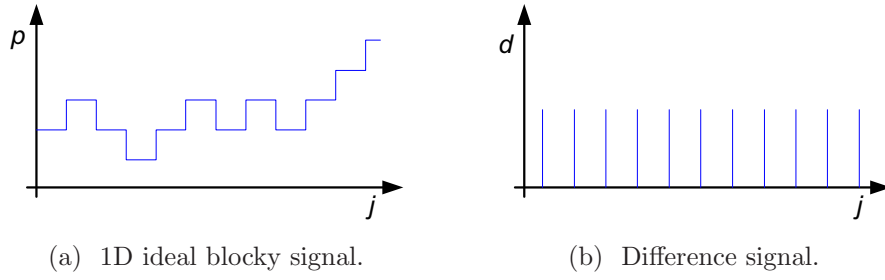


Figure 4.5: Ideal block signal and the corresponding difference signal (adapted from [18]).

differences considering the luminance and contrast masking models given in [77]. Luminance masking is performed by assigning lower weights to the differences located in the extreme dark or extreme bright image areas; contrast masking is performed by computing a spatial activity measurement in the blocks adjacent to each boundary, assigning lower weights to the differences located in busy regions of the image. A similar algorithm can be found in [22], where the author also proposes to measure block effect using block boundaries differences. When compared with [15], the main difference is that inter-block pixel differences are weighted using a different luminance masking model.

In [18], Wang *et al.* proposed to model a blocky image as the result of a non-blocky image interfered with a blocky signal. Measurement of the block effect can thus be seen as an estimation of the blocky signal's power. In order to do so, the algorithm starts by computing the difference of consecutive pixel values along the vertical and horizontal directions. Then, a one dimensional *fast Fourier transform* (FFT) is applied to those differences (separately for both directions). In the frequency domain, a blocky image is characterized by the existence of equally spaced peaks in the power spectrum. The concept is illustrated in Figure 4.5. The power of the blocky signal is computed as the difference between the power of the blocky image and an estimate of the power of the original, non-blocky, image. This estimate is computed by smoothing (removing the peaks) the power spectrum of the blocky image, using median filtering – the result is an estimate for the power spectrum of the blocky signal, which is used as a block effect measurement.

All of the above mentioned methods are based in the premise that a blocky image always contains discontinuities across the block's boundaries. However, that is not the case if coarse quantization is applied to an homogeneous image region (*e.g.*, part of an image representing a sky without clouds). In such cases, pixel differences

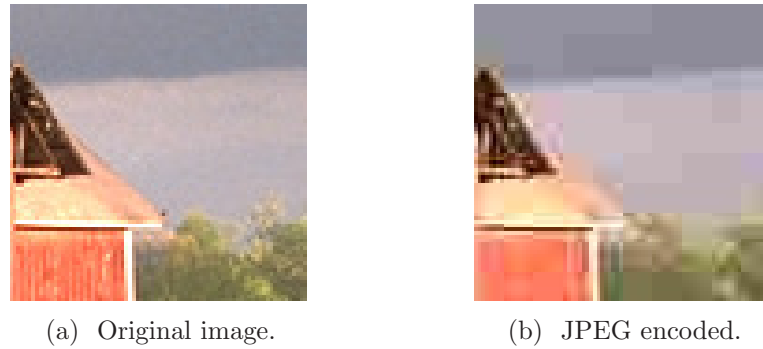


Figure 4.6: *Flatness* effect caused by low bitrate JPEG encoding, the concept which is explored by Pan *et al.* in [23].

across block’s boundaries may in fact be null, leading to an inaccurate block artifact measurement if only those differences are considered. In order to overcome this issue, Pan *et al.* proposed in [23] a block artifact measurement technique that is the result of two terms: *blockiness* and *flatness*. The “blockiness” term is based on inter-block differences, considering the values of four pixels on either side of the block’s boundaries. As for the “flatness” term, it is basically the number of null differences between consecutive pixel values measured in the vicinity of each blocks’s boundary (differences are measured along directions orthogonal to the boundary direction).

Figure 4.6 illustrates the concept: the original image in Figure 4.6-a) has been JPEG encoded with a small quality factor in order to emphasize the block effect, resulting in the image depicted in Figure 4.6-b). When looking into the sky of the encoded image, it can be observed that several adjacent blocks exhibit the same pixel values. If artifact measurement was based on block differences only, those blocks would not contribute for the measurement – that is the reason behind the “flatness” term, which accounts for those situations.

Blur measurement

Blur is another typical artifact addressed by artifact measurement metrics. Although different sources for blur exist, in the context of image quality assessment these metrics typically deal with blur caused by lossy encoding of the media.

In [17], Marichal *et al.* propose a blur measurement that works in the DCT domain. Each DCT frequency position is assigned a “blur importance” weight, with larger weights located in the diagonal from the upper-left corner (DC coefficient) to the

lower-right corner. This weighting grid can be justified because it does not privilege any blur direction. For each DCT frequency, if there are enough small valued coefficients, it accumulates the weight at the corresponding position. After all frequencies have been dealt with, the accumulated weight values are divided by the sum of all values in the weighting grid.

Marziliano *et al.* proposed to quantify blur by measuring the thickness of image edges [19]. In their work, the process of measuring edge thickness is performed for vertical edges only, which are found using a Sobel filter. For each image row, the algorithm determines the width of each edge that is found and, after processing all image rows, the average edge width is computed and used as a blur measurement. An illustration of the algorithm's main idea is depicted in Figure 4.7: Figures 4.7-a) and b) represent the original *Caps* image and the corresponding gradient in the horizontal direction, respectively; similarly, Figures 4.7-c) and d) depict a blurred version of the *Caps* image and the corresponding gradient in the horizontal direction. When comparing the gradient images, it can be observed that the edges found in the blurred image are wider than those of the original image. A measurement of the average edge width can thus be used for measuring the amount of blur. The algorithm has been tested in images corrupted by gaussian blur and in images subject to JPEG2000 encoding. Results were good for the former case, but modest for the latter case.

Wang *et al.* proposed in [21] an artifact oriented metric that measures blur and block effect artifacts, combining them into a single quality index. The block effect is measured in the traditional way, by averaging the differences at block's boundaries, while blur is measured by accounting for the signal activity in the image. The signal activity is given by the average inter-pixel absolute differences inside each block and by accounting for zero-crossings in pixel differences along the horizontal and vertical directions (which in practice works as a count for local maximums and minimums along those directions). The three measurements (block effect and the two blur terms) are combined in the form of a product of powers, plus an offset, using the exponents of those powers as tuning parameters.

Related work that is close to blur estimation is due to Caviedes and Gurbuz in [78]. Here, it is proposed to measure image and video sharpness (which can be seen as the inverse of blur). The proposed method starts by computing the edges of the image under evaluation. Then, it applies an 8×8 2D DCT centered at each pixel belonging to an edge, and computes the kurtosis of the resulting AC DCT coefficients. This analysis is performed for all edge pixels and the average of the kurtosis values is used

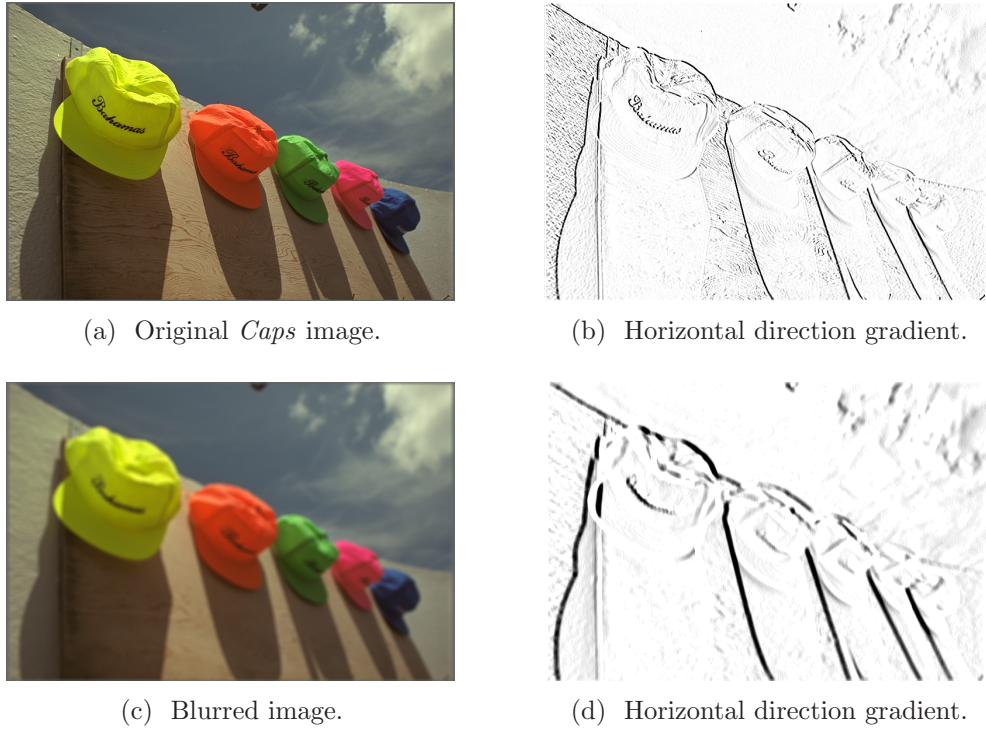


Figure 4.7: The effect of blur on image edges, the concept explored in the algorithm proposed by Marziliano *et al.* in [19].

as a sharpness metric. This work is further improved in [79], by also considering spatial information of the edges and directional image energy in the DCT frequency spectrum.

Ringings measurement

The main motivation for the measurement of the ringing artifact is image quality assessment of JPEG2000 encoded images. As already mentioned in Section 2.4.3, ringing is due to Gibbs phenomenon, and it is perceived by the HVS as spurious oscillations in pixel values around image edges separating smooth regions. Thus, most approaches for the measurement of ringing are based on the detection of image regions around those edges [16, 24–26].

One of the first techniques that quantifies the ringing artifact is due to the work of Oğus *et al.* in [16]. Their work emphasizes a ringing artifact reduction technique based on morphological operations, but they also propose a no-reference objective quality metric based on the ringing artifact measurement. In order to compute this

measurement, the algorithm starts by locating the image areas where the ringing artifact is likely to be visible. Since these locations usually correspond to transitions between smooth image regions, the algorithm tries to find them using edge detection techniques and morphological operations. Since ringing is characterized by spurious oscillations in those areas, it is quantified by computing the variance of pixel values in 4×4 blocks located inside the regions detected in the previous step. Although interesting ideas are presented in [16], the proposed method was tested using cartoon images, without texture, and thus not representative of natural images. Similar ideas for ringing measurement in JPEG2000 encoded images are proposed by Barland and Saadane in [25].

In [24], Marziliano *et al.* proposed a full reference ringing metric that is combined with their previous blur measurement given in [19] (already mentioned in this section), for assessing the quality of images subject to JPEG2000 encoding. In order to quantify the ringing artifact, the algorithm starts by finding the most significant vertical edges in the image. This operation is performed by Sobel filtering followed by thresholding. Then, for each image row, a distortion analysis is performed at both sides of each edge. The number of pixels that are analyzed at each side of the edge depend on the size of the wavelet decomposition filters used in JPEG2000 and on the effects of blur in the edges (previously measured). Distortion analysis in the vicinity of each edge is carried out by computing the errors between the corresponding reference and distorted pixel values. The difference between the maximum and the minimum error found on each edge side is then multiplied by the number of analyzed pixels, resulting in a local ringing measurement. A global ringing measurement is computed by averaging all local measurements.

In all the works mentioned above, natural textures present in the images could negatively influence the detection of regions where ringing could be perceived, the “ringing regions”. Thus, those algorithms also measure the ringing artifact in regions where it is not perceived by the human observer (such as in textured regions). More recently, Liu *et al.* proposed a no-reference ringing metric in [26], where special attention is given to the detection of the “ringing regions”. This algorithm performs a pre-filtering step in order to smooth textured regions of the image. After this pre-filtering process, it performs edge detection and, due to the previous step, only the stronger edges (*e.g.*, object contours) are detected. The “ringing regions” are extracted by inspection of the image regions in the vicinity of the stronger edges. Once those regions are detected, ringing measurements are computed similarly to [16], by computing local pixel variances in 3×3 windows.

4.4.3 Feature-based metrics

Feature-based quality assessment metrics are those that collect a set of features from the reference and distorted videos (or from the distorted video only, in the case of a no-reference metric), and combine them in order to obtain a quality prediction. These type of metrics can be seen as an *engineering approach* [1] to the HVS modeling. The main motivation for the development of feature-based quality metrics is that it does not require a deep knowledge about the mechanisms of the human vision. The HVS is implicitly modeled by the way the features are combined, which generally follows methodologies that are more familiar to engineers, such as neural networks or regression models.

An example of a feature-based no-reference quality assessment metric for still images is the work by Gastaldo *et al.* in [80, 81], where several features based on pixel values, correlation measurements, and DCT frequency domain coefficients are extracted from the distorted image. Those features are then sent to a circular back-propagation neural network that combines them and computes a quality score. The use of neural networks was also proposed by Babu and Perkiş in [82], who developed a no-reference metric for quality assessment of JPEG encoded images. In their work, the inputs of the neural network are block-based features which are measurements of the background luminance, background activity, edge amplitude and length.

Neural networks have also been used for video quality assessment. In [28], Ries *et al.* propose the use of motion-based features that are combined using neural networks. The system has been trained for working in a mobile video streaming context. The same authors proposed in [29] a linear model for combining those motion-based features, as well as the video bitrate, in order to perform no-reference quality assessment of H.264 encoded video sequences (without transmission errors).

In [27], Oelbaum and Diepold propose a reduced reference quality metric for H.264 video that is based on the extraction of a set of features. It uses temporal features such as motion continuity and frame predictability, spatial features such as edge continuity, a color continuity feature, and combines them linearly with artifact measurements, namely blur and block effect measurements.

Pinson and Wolf proposed a feature-based full reference quality assessment metric in [83]. In their model, quality scores result from combining seven different features, which are computed from both the reference and distorted videos. Five of those features are in fact measurements that provide an indication of artifacts present in the video (blur, block effect and color impairments). There is also one feature that

addresses spatio-temporal masking effects and a feature that accounts for actual improvements of quality in the distorted video (edge and contrast enhancements). The combination of these feature values follows a simple linear model, where the linear weights work as tuning parameters.

The *edge-PSNR* (E-PSNR) [84] by Lee *et al.* can be seen as a feature-based metric for video that can either work in full reference mode or reduced reference mode. E-PSNR uses an edge map of the original image, where each pixel location is a low-level feature. In full reference operation, the edge map is extracted from the reference and the PSNR is measured by accounting the pixel differences at edge locations, only. In reduced reference operation mode, the edge map is computed before transmission and a selection of edge locations and their luminance values is sent as side information. At the receiver, the PSNR is computed by considering the received edge locations only. In order to account for the possibility of jitter, the reduced reference operation mode also compensates the E-PSNR measurement by considering the frame freezing time.

4.5 Bitstream-based metrics

For long, the network quality of service community has been using metrics such as the *bit error rate* (BER) or the *packet loss rate* (PLR) in order to quantify transmission errors. Similarly to what was discussed in Section 4.3 for the PSNR, those simple measurements may be sufficient to characterize data transmissions where all bits are equally important, but they are not suitable for fully characterizing transmission losses in multimedia data. For instance, packet losses in video transmissions may lead to different subjective quality impacts depending on which parts of the video bitstream have been affected.

Thus, bitstream-based quality assessment metrics may consider simple measurements such as the packet loss rate, but that is clearly not sufficient. Generally, additional information is extracted from the bitstream, using partial decoding of the bitstream elements. Other features extracted from the received packets can also be used (for instance, packet numbers and timestamps).

4.5.1 Packet-oriented metrics

In [85], Kanamuri *et al.* proposed bitstream-based no-reference and reduced reference metrics that use several features extracted from the video bitstream. Some of those features are packet-oriented, such as the number of frames affected by packet loss, type of frame associated to each lost packet and number of lost slices. Besides those features, it also uses other features that are not packet-oriented, based on the motion vectors, which are also extracted from the bitstream. The proposed algorithms do not provide a quality assessment score. Instead, they predict whether or not a given packet loss will be perceived by users. Following the same line of work, in [86] the authors proposed a reduced-reference metric that checks the visibility of packet losses in H.264 encoded video. The same kind of packet-oriented and motion-oriented features is used.

Winkler and Mohandas proposed in [65] a no-reference metric – the *V-factor* – oriented to packetized transmission of MPEG-2 and H.264 video. The metric uses information collected from the packet headers, from the bitstream and from the decoded video, and combines the collected data in order to obtain a quality score. However, since the metric was developed for commercial purposes, there are not many details on its implementation.

4.5.2 PSNR estimation algorithms

A possible approach to no-reference quality assessment of encoded video, that uses information collected from the bitstream, is to estimate the PSNR of the received encoded media with respect to its reference. Although PSNR is a rough quality metric when considering all possible distortion sources, as discussed in Section 4.3, it may actually be a reasonable quality indicator when the distortion is caused by lossy image or video encoding. The motivation for using PSNR estimates in such cases may also be justified by the absence of other effective no-reference quality assessment methods.

In [87], Turaga *et al.* were probably the first authors to propose an algorithm that estimates the PSNR of encoded video sequences based on the statistical properties of DCT coefficients. The distribution of the original (reference) DCT coefficient data is estimated at the receiver. Using those statistical distributions, together with knowledge about the quantization steps used for video encoding, allows an estimation of the quantization error produced during video encoding, and thus it

is possible to estimate the PSNR of encoded video by looking into its bitstream. However, the method described in [87] does not include a robust estimation of the original DCT coefficients distribution, and thus PSNR estimation accuracy decreases as coding rate decreases. This is due to the increasing number of DCT coefficients quantized to zero values, which lead to inaccurate statistical parameter estimation.

Aware of the above mentioned problem, Ichigaya *et al.* have proposed a method for PSNR estimation [88] that models DCT coefficient distributions as a weighted mixture of Laplace *probability density functions* (PDFs): one is computed by considering all quantized coefficient values and the other is computed by considering the non-zero quantized values only. Although a considerable improvement is reported in [88] when compared with [87], the proposed method still fails when all (or almost all) DCT coefficients at the same frequency are quantized to zero.

In a more recent work [89], Eden proposes a PSNR estimation method for H.264 encoded video sequences. Similarly to what was done in the previous works, the coefficients' distributions are modeled according to Laplace densities. The main novelty is the use of a low complexity algorithm for the estimation of the probability density's parameter and the ability to deal with the possibility of all DCT coefficients at a given frequency being quantized to zero. The results depicted in [89] show that this strategy provides good PSNR estimates for I-frames but the results for P and B-frames still need to be improved.

Another work by Shim *et. al* [90], follows a line that is close to Eden's work: the main difference is that instead of using Laplace densities, it is suggested to use Cauchy densities for modeling the DCT coefficients' distributions.

4.6 Standardization of objective metrics

The first standardized full reference objective quality assessment metrics were published in 2001 under ITU-T Recommendation J.144 [91]. However, the performance of those metrics was not satisfying enough for ITU: in the conclusion of standard document it reads "*Since no method of measurement can be recommended at this time, this clause will list some general advice on the models for video quality assessment utilizing full reference methodology*". Nevertheless, it is worth to mention that the full reference algorithm from Pinson and Wolf [83], one of the metrics present in ITU-T Rec. J.144, was published in 2003 as part of the ANSI T1.801.03-2003 [92] standard.

Afterwards, another call for standardization of objective quality assessment metrics resulted in the publication, in 2008, of ITU-T Recommendations J.246 [3] and J.247 [4]. Recommendation J.246 standardizes a reduced reference metric while Recommendation J.247 provides the necessary details for the implementation of four possible full reference metrics.

The full reference models described in ITU-T Rec. J.247 are the following:

- *NTT full reference model* – a full reference metric developed by *Nippon Telegraph and Telephone* (NTT) Corporation, Japan. It consists in three main modules: video alignment, temporal and spatial feature derivation and quality estimation. This metric uses an edge related feature and two motion related features. Besides those features, it also uses the PSNR between reference and distorted signal and a frame freeze measurement.
- *Opticom PEVQ* – the *perceptual evaluation video quality* (PEVQ) metric, from Opticom, Germany, is a metric based on the work of Hekstra *et al.* published in [93]. Quality prediction result from combining five features (four spatial features and one temporal feature) extracted from both the original and distorted videos. All features are combined using a weighted sum of logistic functions (one logistic function per feature).
- *Psytechnics model* – in the FR model from *Psytechnics Ltd.*, U.K., spatio-temporal features are extracted either directly from the distorted video signal or result from a comparison between the original and the distorted videos. This metric also uses artifact measurement algorithms focused on the measurement of blur, block effect and distortion around edges (which implicitly measures ringing / mosquito noise artifacts). All features and artifact measurements are then linearly combined in order to obtain a MOS prediction.
- *Yonsei model* – this metric, from *Yonsei University*, Korea, is probably the most simple standardized FR model. It combines the full reference operation of the E-PSNR metric [84] (already mentioned in Section 4.4.3) with block effect and blur measurement metrics.

It is interesting to notice that the FR models standardized in [4] use a wide mixture of the ideas presented in the previous section. It is also worth to mention that all of the above standardized metrics include sophisticated procedures for spatio-temporal alignment of the reference and distorted video sequences.

As for ITU-T Recommendation J.246, it describes a reduced reference model which is basically the E-PSNR algorithm from Lee *et al.* proposed in [84], already mentioned at the end of Section 4.4.3. It defines different operation modes on the side information channel that depend on the video resolution and on the available bandwidth.

Concerning the no-reference approach for objective quality assessment metrics of digital television content, there are no standardized procedures yet. The only standard that is related with no-reference image quality assessment is ITU-T Recommendation G.1070 [7], that standardizes a quality model for video-telephony applications. This standard describes separate models for audio and video, as well as a multimedia model, that joins both audio and video. The algorithm for video quality assessment uses packet and bitstream-oriented features such as packet loss rate, end-to-end delay, encoding bitrate and frame rate. The standard also states that the applicability of the model should be restricted for QoS/QoE planning purposes only.

4.7 Performance of an objective metric

The performance of an objective quality assessment metric is evaluated by comparing the MOS predictions computed by the metric with those resulting from subjective quality assessment tests. In order to quantify such comparison, VQEG suggests the use of the performance indicators described in the following text.

Root mean squared (RMS) error

The *root mean squared* error evaluates how close MOS predictions are to the ground data. In other words, it measures the predictions' accuracy. It is defined as:

$$RMS = \sqrt{\frac{1}{N_s} \sum_{i=1}^{N_s} (M\hat{O}S_i - MOS_i)^2}, \quad (4.8)$$

where N_s is the number of assessed images or video sequences, $M\hat{O}S_i$ and MOS_i are the predicted and true MOS values for the i -th image or video sequence, respectively. An high value for the RMS error indicates a poor metric's performance.

Pearson's correlation coefficient

When applied to the evaluation of an objective metric's performance, *Pearson's correlation coefficient*, ρ_c , measures the degree of linear relation between MOS predictions and their true values. The ideal value of $\rho_c = 1$ means that such relationship could be described by an affine function. ρ_c can be computed as:

$$\rho_c = \frac{\sum_{i=1}^{N_s} (M\hat{O}S_i - \mu_{M\hat{O}S})(MOS_i - \mu_{MOS})}{\sqrt{\sum_{i=1}^{N_s} (M\hat{O}S_i - \mu_{M\hat{O}S})^2} \sqrt{\sum_{i=1}^{N_s} (MOS_i - \mu_{MOS})^2}}, \quad (4.9)$$

where $\mu_{M\hat{O}S}$ and μ_{MOS} represent the average values for MOS predictions and true MOS values, respectively.

Spearman's rank correlation coefficient

The *Spearman's rank order coefficient*, ρ_s , is a form of correlation that measures how well the relation between two variables can be described by a monotonic function. Thus it measures the monotonicity of the MOS predictions without making assumptions about the functional form of the relationship between those predictions and their corresponding true values. It can be computed as:

$$\rho_s = 1 - \frac{6}{N(N^2 - 1)} \sum_{i=1}^{N_s} (\Delta R_i)^2, \quad (4.10)$$

with

$$\Delta R_i = \text{rank}(M\hat{O}S_i) - \text{rank}(MOS_i),$$

where $\text{rank}(x_i)$ represents the statistical rank of value x_i (the relative position of x_i in a sorted list of all sample values). Similarly to Pearson's correlation coefficient, the ideal value of ρ_s is 1, a condition that would be verified if the estimated values grow with the true values.

Outlier ratio

The *outlier ratio* evaluates how consistent is the accuracy of MOS predictions performed by an objective metric. A small value for the outlier ratio means that the metric performs equally well for all assessed images or video sequences. An higher value may indicate that the metric's predictions are inconsistent, *i.e.*, it does not perform well in a subset of the assessed images or sequences.

The outlier ratio, O_r , is given by the ratio between predictions considered as outliers and the total number of predictions:

$$O_r = \frac{N_{outliers}}{N_s}. \quad (4.11)$$

A MOS prediction is considered to be an outlier if it verifies the condition:

$$|\hat{MOS}_i - MOS_i| > 2\sigma_{MOS_i}, \quad (4.12)$$

where σ_{MOS_i} is the standard deviation of the valid scores given to the impaired image or video sequence i (during the subjective assessment).

4.8 Summary

This chapter presented an overview of the research work on objective quality assessment metrics. It started by performing a classification of these metrics. They can be classified according to two possibilities: 1) considering the amount of information about the reference signal that is required for computing the quality score; 2) considering the type of image / video data used for computing the metric's result. These possibilities are depicted in the scheme of Figure 4.8.

After presenting this classification, a state-of-the-art of the literature on the topic has been presented. It included relevant examples of proposed quality metrics among all their classes, with a greater emphasis on those belonging to the no-reference quality assessment. It also presented the recent standardized procedures for objective assessment.

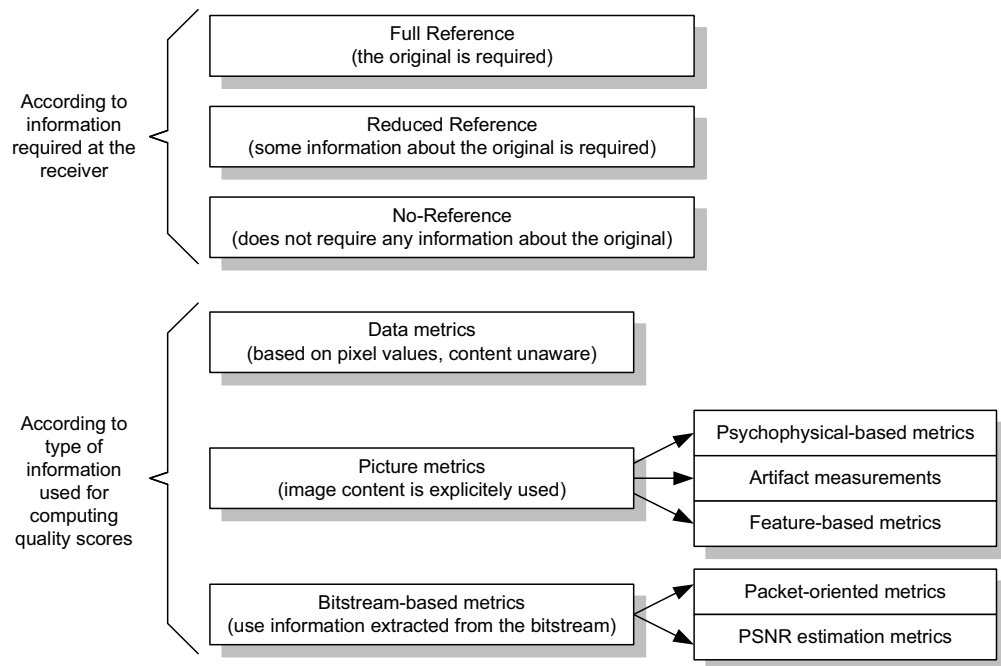


Figure 4.8: Objective image quality assessment classification.

Chapter 5

Image quality assessment using watermarking

5.1 Introduction

In the previous chapter, a brief overview of objective image and video quality assessment metrics has been presented. Objective quality metrics have been grouped into classes and the most significant algorithms have been briefly described. As already mentioned, most of the no-reference quality assessment algorithms described in literature are oriented for measuring specific artifacts originated by the encoding of visual contents. In this chapter, the possibility of using watermarking techniques for image quality assessment is investigated. More specifically, this chapter proposes a no-reference image quality assessment algorithm based on watermarking techniques.

Most of the research on watermarking techniques was performed during the second half of the 90's. Watermarking techniques were initially proposed as possible solutions for audio, image and video copyright protection issues. Later, new watermarking algorithms enabled applications such as identification and authentication, steganography and fingerprinting.

The main motivation for using watermarking on NR image and video quality assessment is as follows: an imperceptible reference signal, the *watermark*, is embedded into the original image data, the *host* (which can either be an image or a video). During encoding and transmission, both the watermark and the host will be subject to the same distortion sources. At the receiver, it is expectable that the quality of the host signal is related with the distortion of the watermark signal. Thus, a

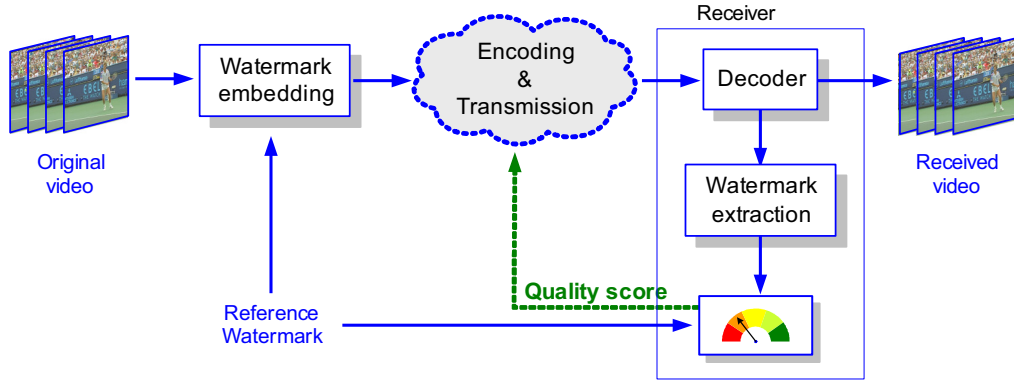


Figure 5.1: Watermarking-based image quality assessment system.

quality score could be derived from a comparison between the reference watermark (*i.e.*, the watermark embedded at the source) and the extracted watermark at the receiver. This concept can be observed in Figure 5.1.

The characteristics of a watermarking-based quality assessment system place it between a reduced reference and a no-reference system. It resembles a RR metric system in the sense that the watermark signal can be viewed as the additional information transmitted. However, this information is usually independent of the host signal, which is not the case in a RR metric system. On the other hand, and since the transmission of the watermark information does not require the use of additional bandwidth, such system is closer to a NR metric system.

In the context of this Thesis, the information carried by the watermark does not depend on the reference image. It is also assumed that the reference watermark signal is known at the receiver side. Since no additional bandwidth is required by the quality assessment system, the proposed watermark-based quality assessment techniques will be considered as no-reference metrics.

In general, watermarking for quality assessment of video transmission can be performed according to the following steps:

1. *Watermark embedding* – an imperceptible watermark signal is embedded into the host image or video. The resulting watermarked media will be considered as the reference (thus the invisibility of the watermark must be assured). The majority of watermarking algorithms follow a spread-spectrum [94] or a quantization-based approach [95].
2. *Encoding and transmission of the watermarked media* – this is the part of the

communication process where the watermarked media is subject to distortion. As already discussed, distortion is mainly due to lossy encoding / transcoding of media, or transmission losses.

3. *Watermark extraction* – the watermark signal is extracted from the received (possibly distorted) media.
4. *Comparison between reference and extracted watermark signals* – it is assumed that the reference watermark signal is known or can be replicated at the receiving side. The extracted watermark is compared with the reference watermark. Since it is expectable that distortion in the extracted watermarked signal increases with the host signal's distortion, image quality scores are based on the result of such comparison.

Computing quality scores based on the comparison between reference and extracted watermarking signals can be performed using different methodologies. The following methods have been proposed in literature:

- *Watermark bit error rate* – this is probably the most simple but also the most inaccurate metric, since it correlates poorly with the human perception of quality. This is the metric proposed by Wang *et al.* in [96] and also by Zheng *et al.* in [97]. As expected, the watermark bit error rate increases with increasing distortion, but its direct use is not adequate for image quality scoring.
- *Watermark signal mean squared error (W_{MSE})* – the MSE between extracted and reference watermark signals can provide a greater accuracy than the extracted mark error rate, since it exhibits a better correlation with the MSE of host image or video. This fact suggests that it is possible to relate an image or video PSNR with W_{MSE} , using a function in the form $PSNR_{image} = f(W_{MSE})$, where $f(x)$ is usually a linear function approximation valid for a given range of PSNR values. This measurement (or similar) is proposed by the majority of the authors: Campisi *et al.* in [98–100], Farias *et al.* in [101, 102], Saviotti *et al.* in [103] and Sugimoto *et al.* in [104].
- *Correlation between extracted and reference watermark signals* – another proposed alternative is to compute the correlation between extracted watermark signal and the reference watermark signal. This is the metric used in the work of Bossi *et al.* [105] and it is also one of the metrics analyzed by Holliman *et al.* in [106].

All of the previously mentioned methods explicitly use the watermark signal, *i.e.*, the quality rating of the received media is estimated directly from the watermark degradation. However, the nature of the problem also suggests an implicit use of the watermark signal – the watermark could carry information to be used as side information for the quality evaluation system following an even closer approach to reduced reference metrics, with the advantage of not requiring additional bandwidth or extra channels. This possibility is addressed on the work by Holliman *et al.* in [106]. In this paper, it is suggested the use of a watermark that comprises information regarding the maximum distortion allowed at each image point. This is accomplished by using a set of quantizers, from whose output the watermark depends (the authors suggest that the quantizer set can be designed according to the characteristics of the human visual system). During the extraction phase, out-bounded local distortions will originate watermark extraction errors, and thus allowing to localize those distortions.

In the watermarking-based quality assessment system described in this chapter, image distortion is measured by estimating the error on the extracted watermark signal. From this estimate, it is possible to measure the watermark's MSE similarly to what is done in [98–104], with a greater advantage: the proposed watermarking scheme was designed in such a way that the differences between extracted and reference watermark signals correspond to the difference between the distorted and the original images.

It is also proposed to weight these differences as a function of the extraction bit error rate or using image statistics in the frequency domain. These weighting strategies aim to compensate watermark's MSE underestimation as distortion increases. In order to control watermark imperceptibility, while maximizing the watermark's robustness to distortion, non-uniform and frequency adapted quantization functions that consider the characteristics of the human visual system have also been derived. Error estimates resulting from this step are also perceptually weighted, in order to predict MOS values, which is also a novel and advantageous proposed feature, when compared with other watermarking-based quality assessment algorithms found in the literature.

This chapter is organized as follows: after the introduction, Section 5.2 describes the proposed watermarking system. Section 5.3 explains how local error estimates are computed based on the watermark's extraction. Section 5.4 depicts the results achieved by the algorithms and, finally, Section 5.5 draws the main conclusions from the work presented along this chapter.

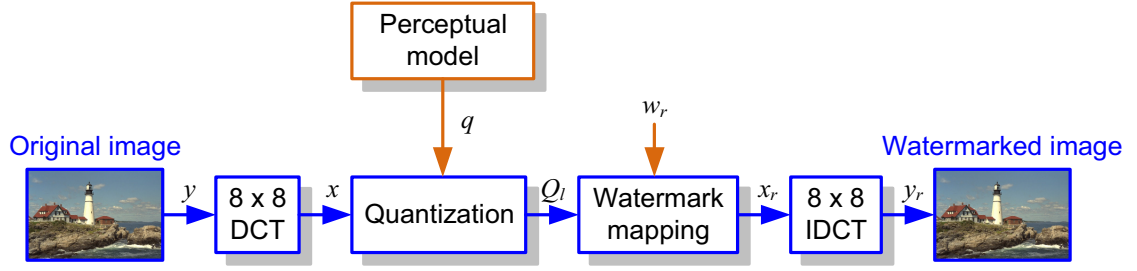


Figure 5.2: Watermark embedding scheme.

5.2 Watermarking scheme

5.2.1 Watermark embedding

Consider that a reference binary watermark message, w_r , is to be embedded into the luminance component of a host image y (for the case of video, the watermark is embedded in a frame-by-frame basis). Watermark embedding and extraction are both performed in the 8×8 blockwise DCT domain of y , using a quantization-based approach [95]. When used in conjunction with blockwise DCT-based encoding standards (JPEG, MPEG, H.264), this scheme allows the watermarking processes to work directly in the frequency domain, without the need for fully decoding the images.

Figure 5.2 depicts the embedding scheme. Let $x_k(i, j)$ represent the resulting DCT coefficient located at frequency position (i, j) of the k -th block. Each DCT coefficient is then quantized according to function $q(\cdot)$, resulting in value Q_l , which can assume one out of $2L$ possible values, with $l \in \{-L, -1, 1, \dots, L\}$. Each coefficient used for embedding is modified to the nearest quantization value whose least significant bit (*LSB*) is equal to the watermark bit to be embedded. Note that this quantization process is for watermark embedding purposes only and should not be confused with the quantization process associated to lossy image encoding. Formally, assuming that Q_l is the quantization value nearest to $x_k(i, j)$, the watermarked coefficient, $x_{r_k}(i, j)$, is obtained by:

$$x_{r_k}(i, j) = \begin{cases} Q_l, & \text{if } \text{mod}(l, 2) = w_{r_k}(i, j); \\ Q_{l+u}, & \text{otherwise,} \end{cases} \quad (5.1)$$

where $w_{r_k}(i, j)$ is the reference watermark bit to embed, $\text{mod}(n, m)$ is the remainder

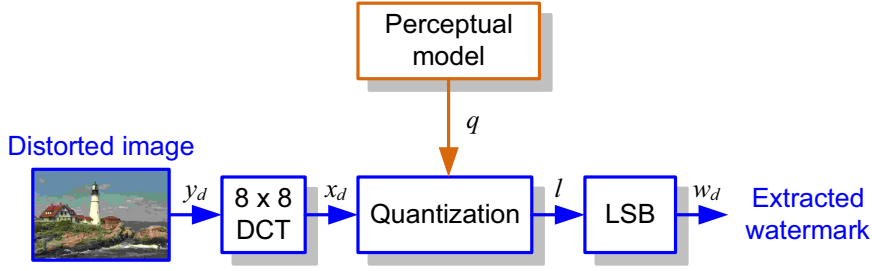


Figure 5.3: Watermark extraction scheme.

of the integer division of n by m , and u is defined as:

$$u = \text{sgn}(x_k(i, j) - Q_l). \quad (5.2)$$

To complete the process, the inverse DCT transform is computed, resulting in the watermarked image y_r . From this point, the watermarked image will be considered as the reference image.

5.2.2 Watermark extraction

The input for the watermark extraction module, represented in Figure 5.3, is a watermarked and possibly distorted image y_d . This image is subject to 8×8 block-wise DCT and the resulting DCT coefficients, $x_{d_k}(i, j)$, are quantized with the same function used for embedding. The extracted watermark, w_d , is computed by inspecting the LSB of the resulting quantization levels.

Formally, the extracted watermark bit at position (i, j) of the k -th block, can be defined as:

$$w_{d_k}(i, j) = \text{mod}(l, 2), \quad (5.3)$$

where l is the corresponding quantization level.

It is important to guarantee that the same quantization function is used both at the embedding and at the extraction modules. A simple solution is to use uniform quantization with a pre-defined value for the quantization step, as proposed in [103] and in [96]. However, there is a major drawback when using this approach: the pre-defined value for the quantization step must be too small in order to keep the imperceptibility of the watermark signal. Since the watermark signal is lost if distortion is greater than the quantization step, this approach will only succeed in the presence of relatively small distortions.

In order to minimize this problem, a non-uniform quantization scheme was developed. The main objective is to assign larger quantization steps to locations of the transform domain where greater modifications on the values of DCT coefficients are allowed, without compromising the imperceptibility of the watermark signal.

In this Thesis, it is proposed to derive those non-uniform quantization functions by considering the perceptual model developed by Watson in [8], already mentioned in Section 4.4 of the previous chapter. Additional model details, as well as its applicability in the derivation of the non-uniform quantization functions, are provided in the following section.

5.2.3 Perceptually adapted quantization functions

The perceptual model proposed by Watson in [8], reviewed in Section 4.4.1, estimates the perceptibility of modifications in individual DCT coefficients in terms of *just noticeable differences* (JNDs) whose threshold values are called *slacks*. Each slack measures the maximum allowed modification to a DCT coefficient value that is possible to perform before resulting in one JND.

The model comprises two components, which account for luminance adaptation and contrast masking. The luminance adaptation threshold for a given coefficient, $T_{L_k}(i, j)$, is given by [8]:

$$T_{L_k}(i, j) = T_B(i, j) \left(\frac{x_k(0, 0)}{\bar{x}_{00}} \right)^{\alpha_T}, \quad (5.4)$$

where $T_B(i, j)$ is the frequency sensitivity of block coefficient at position (i, j) (typical values are depicted in Table 5.1), $x_k(0, 0)$ is the value of the DC coefficient of block k , \bar{x}_{00} is the average of the DC coefficients' values in the image, and α_T is a constant whose suggested value is 0.649.

The slack values are computed by also considering the effect of contrast masking, according to [8]:

$$s_k(i, j) = \max \left\{ T_{L_k}(i, j), |x_k(i, j)|^{b(i, j)} T_{L_k}(i, j)^{1-b(i, j)} \right\}, \quad (5.5)$$

where $b(i, j)$ is a tuning parameter between 0 and 1. Watson suggests $b(i, j) = 0.7$ for all $(i, j) \neq (0, 0)$ and $b(i, j) = 0$ for $(i, j) = (0, 0)$, ensuring that DC coefficients

	\xrightarrow{j}							
$i \downarrow$	1.40	1.01	1.16	1.66	2.40	3.43	4.79	6.56
	1.01	1.45	1.32	1.52	2.00	2.71	3.67	4.93
	1.16	1.32	2.24	2.59	2.98	3.64	4.60	5.88
	1.66	1.52	2.59	3.77	4.55	5.30	6.28	7.60
	2.40	2.00	2.98	4.55	6.15	7.46	8.71	10.17
	3.43	2.71	3.64	5.30	7.46	9.62	11.58	13.51
	4.79	3.67	4.60	6.28	8.71	11.58	14.50	17.29
	6.56	4.93	5.88	7.60	10.17	13.51	17.29	21.15

Table 5.1: DCT frequency sensitivity thresholds (from [94]).

are not subject to contrast masking. Equation (5.5) can be also rewritten as:

$$s_k(i, j) = \begin{cases} T_{L_k}(i, j), & \text{if } |x_k(i, j)| \leq T_{L_k}(i, j); \\ |x_k(i, j)|^{b(i, j)} T_{L_k}(i, j)^{1-b(i, j)}, & \text{otherwise,} \end{cases} \quad (5.6)$$

An insight into the second case of equation (5.6) allows to conclude that slack values increase with increasing magnitudes of the DCT coefficient's values (which resembles Weber's law, mentioned in Section 2.2.2). Thus, for quantization-based watermarking in the DCT domain, this conclusion suggests the use of non-uniform quantization, where the quantization steps increase with the absolute values of DCT coefficients.

One possible criteria for controlling the perceptibility of the error caused by watermark embedding associated to a given coefficient is to make it proportional to the corresponding slack value. Using this approach, the error due to watermark embedding, $\varepsilon_{w_k}(i, j)$, at coefficient (i, j) of the k -th block, can be expressed as:

$$\varepsilon_{w_k}(i, j) = \alpha_w \times s_k(i, j), \quad (5.7)$$

where α_w is a parameter that controls the watermark's embedding strength.

For the purpose of deriving the quantization functions based on Watson's model, consider Figure 5.4, which represents three consecutive quantization points. Suppose that the original coefficient value, $x_k(i, j)$, is the value marked with a cross, and that the corresponding watermark bit value to embed is 0. During the watermark embedding procedure, this coefficient will be modified to match the value of Q_{l+1} . In these conditions, the watermark embedding error, $\varepsilon_{w_k}(i, j)$, associated to coefficient

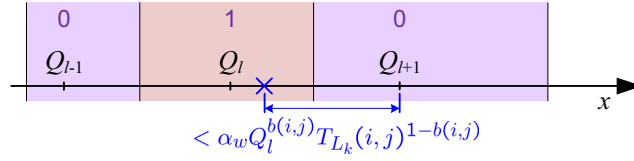


Figure 5.4: Perceptually adapted distance between quantization points.

$x_k(i, j)$ is:

$$\varepsilon_{w_k}(i, j) = Q_{l+1} - x_k(i, j). \quad (5.8)$$

For simplicity purposes, let's assume that $x_k(i, j) > T_{L_k}(i, j)$, a condition that corresponds to the second case of equation (5.6). Under this assumption, and considering (5.7), the embedding error can be written as:

$$\varepsilon_{w_k}(i, j) = \alpha_w |x_k(i, j)|^{b(i,j)} T_{L_k}(i, j)^{1-b(i,j)}, \quad (5.9)$$

or, using (5.8):

$$Q_{l+1} - x_k(i, j) = \alpha_w |x_k(i, j)|^{b(i,j)} T_{L_k}(i, j)^{1-b(i,j)}. \quad (5.10)$$

Attending to the above equation, a solution for relating Q_l with Q_{l+1} is to find the worst case of embedding distortion for any coefficient located in the interval $[Q_l, Q_{l+1}]$. A simple solution, although not optimal¹, is to consider the case where $x_k(i, j) = Q_l$. Substituting in (5.10), leads to:

$$Q_{l+1} = \alpha_w Q_l^{b(i,j)} T_{L_k}(i, j)^{1-b(i,j)} + Q_l, \quad (5.11)$$

which relates Q_{l+1} with Q_l , and thus defines the quantization functions recursively. The initial point for the recurrence can be chosen by considering the first case in (5.6). In this work, the value of $\alpha_w T_{L_k}(i, j)/2$ was selected to begin the recursive definition of the quantization points.

The positive quantization values can then be defined as:

$$Q_l = \begin{cases} \frac{\alpha_w T_{L_k}(i, j)}{2}, & \text{if } l = 1; \\ Q_{l-1} + \alpha_w Q_{l-1}^{b(i,j)} T_{L_k}(i, j)^{1-b(i,j)}, & \text{otherwise} \end{cases} \quad (5.12)$$

The quantization function can be completely defined by also considering the negative

¹The proposed solution is not optimal, because there is a sub-set of points in the interval $[Q_l, Q_{l+1}]$ whose distance to Q_{l-1} is less than to Q_{l+1}

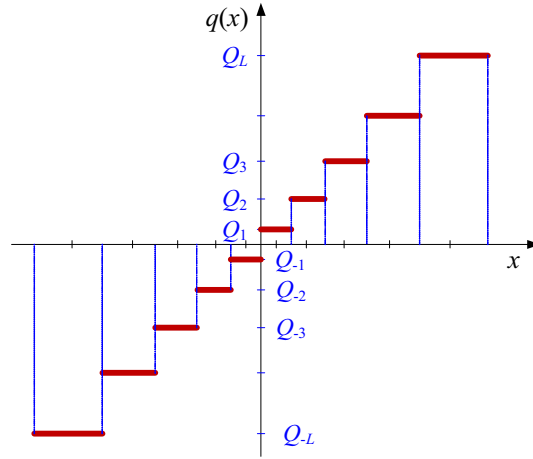


Figure 5.5: Sketch of the quantization function.

quantization points, *i.e.*, the symmetric points that result from (5.12). A sketch of the resulting function is depicted in Figure 5.5.

A brief analysis of equation (5.12) shows that the quantization functions depend on the coefficient position (i, j) within the block, as expected. It can also be concluded that quantization functions depend on the values of the DC coefficients, $x_k(0, 0)$, due to the luminance masking term, T_L , defined in (5.4). Due to distortion, the DC coefficient values at the receiver may become different from their original values. Thus, at watermark extraction, this situation may lead to quantization functions different from the ones used during embedding. Due to this issue, the luminance masking component of Watson's model was not considered, setting $T_{L_k}(i, j) = T_B(i, j)$, for all k . Nevertheless, the robustness of the watermark when considering contrast masking only is substantially greater than when not considering any perceptual characteristic at all.

5.2.4 Choosing the DCT coefficient set for watermark embedding

The perceptibility of the watermark signal is strongly related to the watermark embedding strength and with the number of DCT coefficients that carry the watermark signal (the coefficients whose values have been modified for watermark embedding). It is expected that the watermark imperceptibility increases if less DCT coefficients are used for watermark embedding.

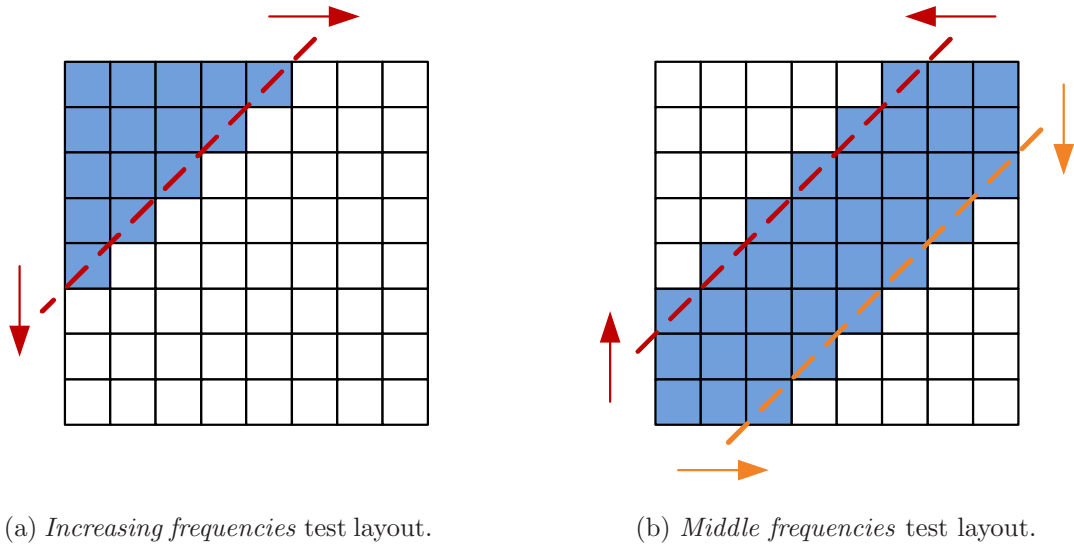


Figure 5.6: Watermark embedding coefficient sets.

Thus, it should be possible to further improve the watermark's imperceptibility, while maximizing its embedding strength, if a smaller set of the 8×8 DCT coefficients per block is selected for carrying the watermark's information. The criteria for choosing this set were as follows:

1. The error that could be measured using the selected coefficient set should be representative of global image error.
2. Changing the values of the DC coefficients, as well as those of the AC coefficients at low spatial frequencies, should be avoided, because it may lead to the visibility of artifacts in homogeneous regions of the image.
3. Watermarking high frequency components is not very useful if the images are to be subject to lossy encoding. Since the quantization steps associated to the high frequencies have higher values, most of those coefficients will be quantized to null values. Therefore, the recovery of the corresponding watermark bits will likely result in errors.

In order to comply with these considerations, a few experiments were performed. The evaluation of the first consideration (*i.e.*, representativity of the coefficient's set for error prediction purposes) has been performed by JPEG encoding different images at several compression rates. The true PSNR of each encoded image was then measured and compared with the PSNR measured using only the coefficients located

(a) *Lenna*.(b) *House crop*.

Figure 5.7: Test images used for DCT coefficient set selection.

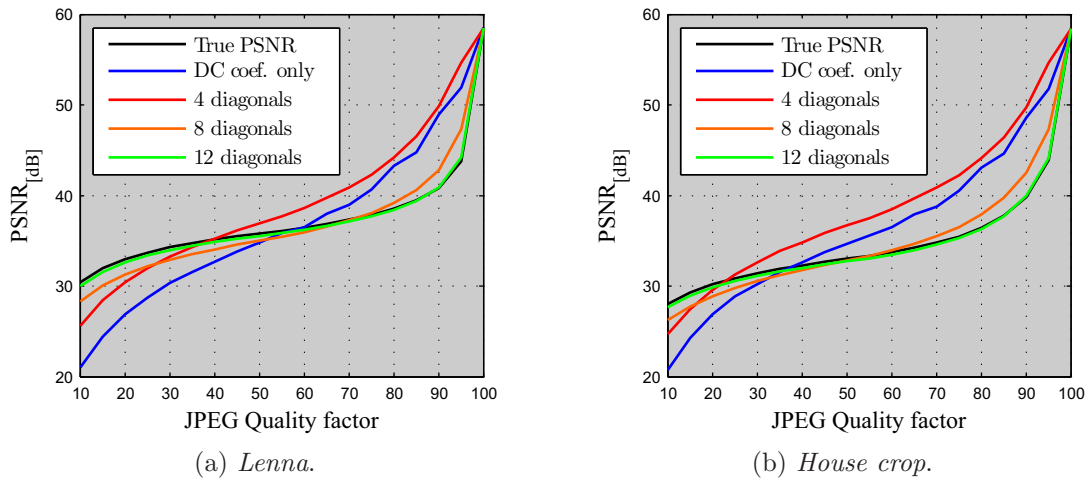
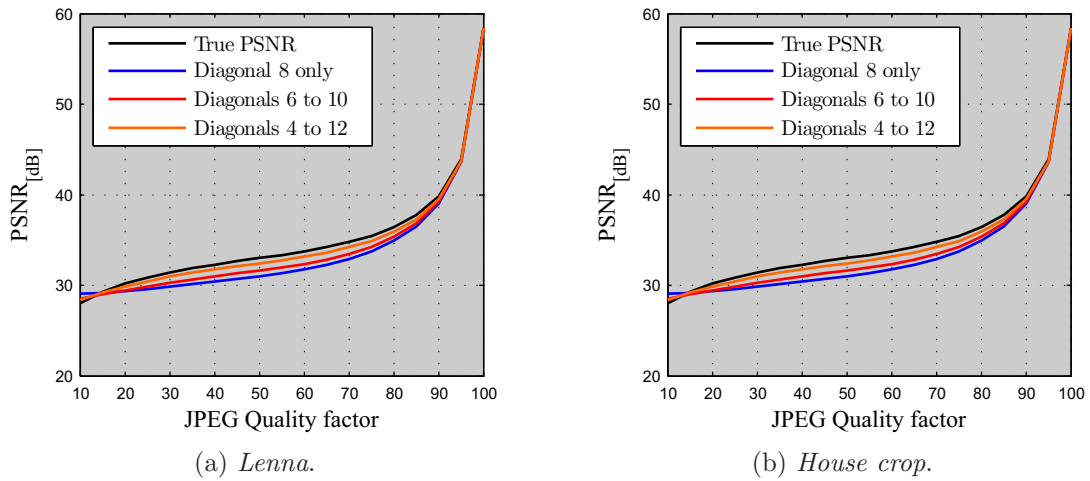
within a given selection set. Note that, in accordance with Parseval's theorem², it is indifferent to measure the PSNR in the pixel or in the frequency domain. In order to deal with the above considerations 2 and 3, two different tests were performed:

- *Increasing frequencies* test – different coefficient sets were generated by using the DCT coefficients located in first diagonals of an 8×8 block, as illustrated in the example of Figure 5.6-a). The objective of this test is to check the importance (or irrelevance) of the high frequency coefficients for PSNR computing.
- *Middle frequencies* test – the goal of this experiment was to evaluate the relevance of the middle frequencies when computing the PSNR. Different coefficient sets were generated by increasing the number of diagonals in both the directions of low and high frequency, starting at the main diagonal of the 8×8 coefficient block – an example of this procedure is illustrated in Figure 5.6-b).

In the following, the images depicted in Figure 5.7 will be used to illustrate the results of these tests.

Figure 5.8 depicts the results of the *increasing frequencies* experiment for the two test images. As it can be observed from the plots, computing the PSNR based on the first 12 diagonals (6 high frequency coefficients are discarded) result in values quite close to the true PSNR values.

²If the transform is unitary (which is the case of the DCT) Parseval's theorem states that the sum (or integral) of the square of a signal is equal to the sum (or integral) of the square of its transform.

Figure 5.8: *Increasing frequencies* test results.Figure 5.9: *Middle frequencies* test results.

As for the second experiment, results are shown in Figure 5.9. It can be observed, for instance, that the values attained for the PSNR using the coefficients located in the diagonals 4 to 12 are close to the true PSNR values. This means that some low frequency coefficients (including the DC coefficient) can be eventually discarded when computing the PSNR.

The set of DCT coefficients' positions used for watermark embedding was then chosen by combining the results from these experiments. A potential DCT coefficient's set candidate for watermark embedding is the one depicted in Figure 5.10. In order

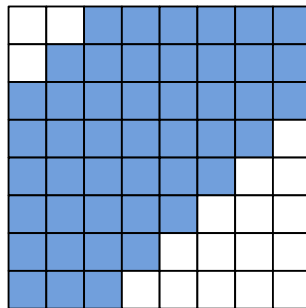
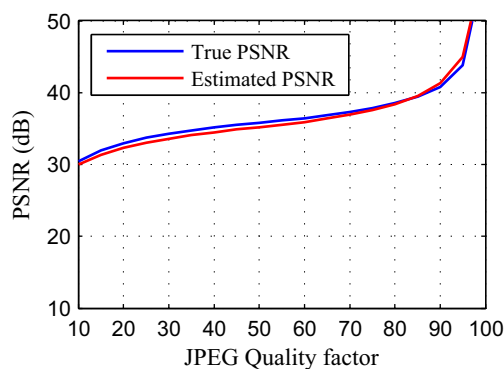
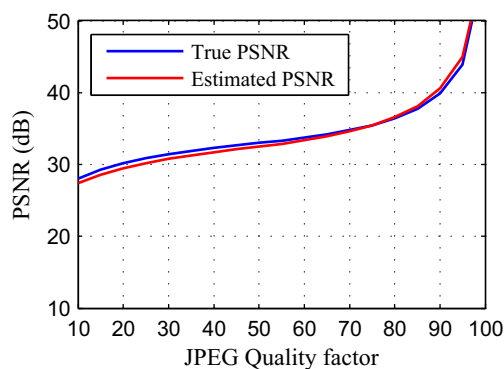


Figure 5.10: DCT coefficient set used for watermark embedding.



(a) *Lenna*.



(b) *House crop*.

Figure 5.11: PSNR computed using the set of DCT coefficients represented in figure 5.10.

to test the behavior of the system when using this coefficient set, several test images have been subject to JPEG encoding with different quality factors. The error associated with the PSNR values computed using this set of coefficients has been compared with the true PSNR values. The results for PSNR measured using the coefficients located in the set represented in Figure 5.10 can be observed in Figure 5.11, for two test images subject to JPEG encoding.

It has been verified that the average difference between the true and the estimated PSNR values based on a coefficient's subset was near 0.5 dB and the maximum value for that difference was 1.1 dB.

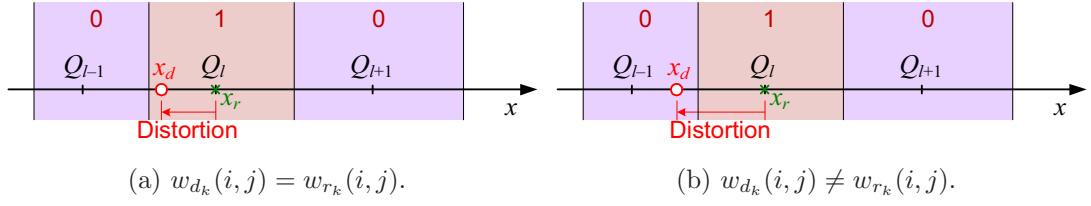


Figure 5.12: Error estimation in the presence of small distortion.

5.3 Error estimation

Using the watermark embedding and extraction algorithms derived in the previous section, the goal is now to estimate the error between the reference and distorted images, using the watermark signal.

The local error at position (i, j) of the k -th block, $\varepsilon_k(i, j)$, is the difference between the reference (watermarked) and distorted coefficient values, *i.e.*:

$$\varepsilon_k(i, j) = |x_{r_k}(i, j) - x_{d_k}(i, j)|. \quad (5.13)$$

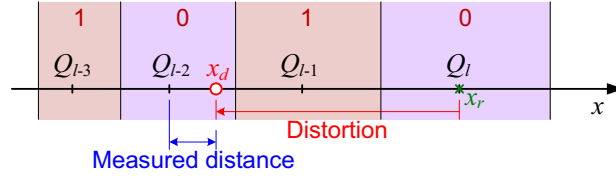
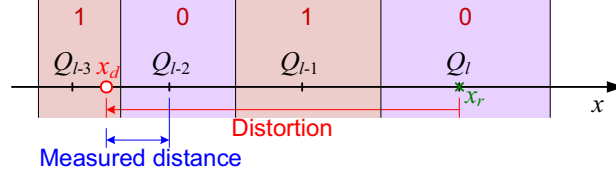
Thus, for small distortions, where the error between the originally watermarked and the distorted coefficients is smaller than the difference between two consecutive watermark quantization levels, the following relation holds:

$$\varepsilon_k(i, j) = \begin{cases} |x_{d_k}(i, j) - Q_l|, & \text{if } w_{d_k} = w_{r_k}; \\ \min \{|x_{d_k}(i, j) - Q_{l-1}|, |x_{d_k}(i, j) - Q_{l+1}|\}, & \text{otherwise,} \end{cases} \quad (5.14)$$

meaning that the error estimated using the watermark and the distorted image is the same as the error computed using the reference and the distorted images. Figure 5.12 illustrates both cases of equation (5.14).

However, with increasing distortion, DCT coefficients may become distorted in such a way that the extracted watermark bit is the same as the embedded one, but with a quantization level (l) different from the one assigned during embedding. In such cases, the error computed using (5.14) is underestimated.

Figure 5.13 shows an example of such an event: suppose that a watermark bit with value '0' was embedded by quantizing the DCT coefficient to the value Q_l (represented with a green cross). The image was then subject to distortion, which caused the value of the DCT coefficient to be changed to the value represented with a red circle, whose nearest quantization value assigned to bit '0' is point Q_{l-2} .

Figure 5.13: *False positive*.Figure 5.14: *False negative*.

In this situation, the correct bit is extracted, but an incorrect distortion distance, $|Q_{l-2} - x_d|$, is computed instead of the correct one, $|Q_l - x_d|$. This situation will be addressed to as *false positive*.

Similarly, a situation such as the one represented in Figure 5.13 will be addressed to as a *false negative* occurrence. In this situation, despite the detection of a watermark bit extraction error – bit ‘1’ was extracted instead of bit ‘0’ – the distance computed using (5.14) is lower than the true distortion distance.

False positive and false negative situations are also applicable for distorted coefficients values separated from the reference values by a larger number of quantization levels. Let D represent the difference, measured in number of levels, between quantization values assigned during embedding and the ones retrieved during extraction. It can be easily concluded that false positives occur when D is even and different from ‘0’, while false negatives occur when D is odd and different from ‘1’. Figure 5.15 illustrates this generalization: as distortion increases, the number of possibilities for the reference (watermarked) coefficient values also increases. The error between reference and distorted coefficient values can thus assume different values from the one resulting from equation (5.14).

One possible solution to tackle the false positives/negatives problem would be to increase the value of the watermark embedding strength (parameter α_w). False positives and false negatives would become less frequent, since quantization steps

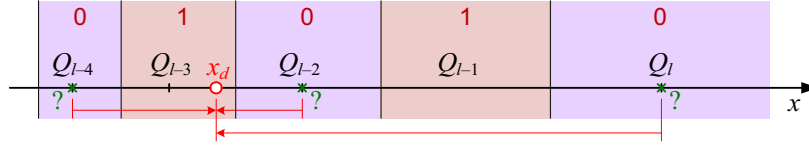


Figure 5.15: Different error possibilities.

used for watermark embedding would be larger. However, as the embedding strength increases, the watermark may eventually become perceptible to a human observer. This perceptibility is not acceptable in a quality assessment application. Due to this restriction, two different error compensation procedures have been tested:

- Empirical distance weighting based on the watermark bit error rates – distances are weighted at the extraction according to estimated values for the false negative/positive rates, P_f . These estimates are based on watermark extraction bit error rate, W_{ber} .
- Distance weighting based on the distribution of the DCT coefficient values – distance weights are computed using an estimate for the distribution of the DCT coefficients and assuming that the distortion is bounded. For instance, if the source of distortion is lossy encoding on the DCT domain, then the error is due to quantization. The bounds for distortion can be derived since quantization steps are available at the decoder.

In the following sections, both strategies are described.

5.3.1 Distance weighting based on watermark bit error rate

In order to improve the accuracy of the local error estimation, distances are weighted at the extraction, by accounting for false negative/positive rates, P_f . Since these rates are not known *a priori*, statistics were obtained from several images, by computing P_f as a function of the watermark extraction bit error rate, W_{ber} , and the value of D .

As an example, results for two test images are depicted in Figure 5.16, showing the evolution of two false positives rate (for $D = 2$ and $D = 4$) and a false negative rate (for $D = 3$). It can be observed that, as expected, P_f increases with W_{ber} and that the maximal amplitude of P_f decreases with increasing D . It was also empirically

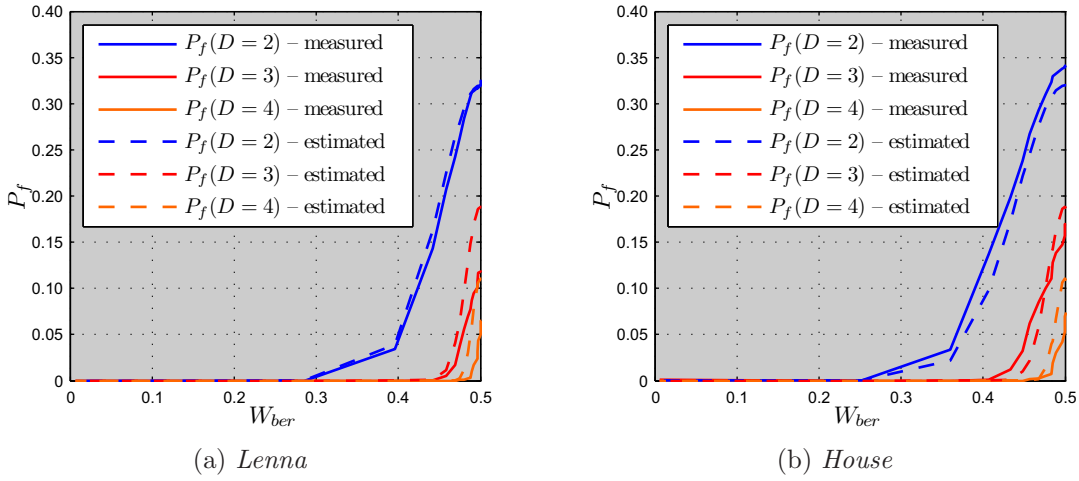


Figure 5.16: False positive/negative rates as a function of the watermark extraction bit error rate.

verified that the exponential decay of these rates is generally less accentuated if the variance of DCT coefficients is larger, thus P_f also depends on image content.

Observing the evolution of the P_f curves, and accounting for the previous conclusions, one possible function type to approximate the rate, $P_f(D)$, of false positives (or negatives) for different values of D is in the form:

$$P_f(D) = C_0 \exp(-f_1(W_{ber}, D, \sigma)) f_2(D), \quad (5.15)$$

where σ represents the standard deviation of the DCT coefficients. C_0 is a constant, function f_1 regulates the slope of the exponential decay and f_2 regulates the maximum amplitude of P_f . Through curve fitting, it was found that the following expressions for f_1 and f_2 conduct to estimations of $P_f(D)$ which are close to the measured ones:

$$f_1(W_{ber}, D, \sigma) = \frac{(W_{ber} - 0.5)^2 (D - 1)^2}{2\sigma^2}; \quad f_2(D) = 0.558^D; \quad C_0 = 0.925. \quad (5.16)$$

Figure 5.16 also depicts a comparison between the estimated functions, using (5.15), and the data measured from statistical tests. As can be observed from the plots, reasonable approximations for $P_f(D)$ can be obtained.

The estimated values for false positive/negative rates can then be used for computing

the weighted error, $\hat{\varepsilon}_k(i, j)$, associated to the distortion, according to:

$$\hat{\varepsilon}_k(i, j) = \begin{cases} P_0 \varepsilon_k(i, j) + \sum_{t=1}^{\lfloor D_{max}/2 \rfloor} P_f(2t) d_f(2t), & \text{if } w_{d_k}(i, j) = w_{r_k}(i, j); \\ P_1 \varepsilon_k(i, j) + \sum_{t=1}^{\lfloor D_{max}/2 \rfloor} P_f(2t+1) d_f(2t+1), & \text{otherwise,} \end{cases} \quad (5.17)$$

with

$$P_0 = 1 - \sum_{t=1}^{\lfloor \frac{D_{max}}{2} \rfloor} P_f(2t) \quad \text{and} \quad P_1 = 1 - \sum_{t=1}^{\lfloor \frac{D_{max}}{2} \rfloor} P_f(2t+1).$$

The summation at the upper term accounts for the probabilities of false positives (even values of D) while the summation at the bottom term accounts for the probabilities of false negatives. P_0 and P_1 are normalizing constants for even and odd values of D , respectively. D_{max} is the maximum value for the error, measured in number of quantization levels, that is to be accounted for. The distance $d_f(\cdot)$ is given by:

$$d_f(D) = \min \{ |x_{d_k}(i, j) - Q_{l+D}|, |x_{d_k}(i, j) - Q_{l-D}| \}, \quad (5.18)$$

representing the distance between the distorted DCT coefficients to the nearest possible quantization value distancing D levels.

5.3.2 Distance weighting based on DCT coefficient statistics

In this section, it will be assumed that distortion on the reference (watermarked) image is due to linear quantization of the DCT coefficients, which is a realistic assumption when considering the distortion due to lossy compression in the context of the main DCT-based encoding standards. For simplicity purposes, the notation that is being used for DCT block position and indexing will be dropped (*e.g.*, $x_k(i, j)$ will be simply addressed to as x_k).

The expected value of the local absolute error, $\hat{\varepsilon}_k$, between reference and distorted coefficients in a given position, can be estimated by:

$$\hat{\varepsilon}_k = \frac{\sum_l P(Q_l) |x_{d_k} - Q_l|}{\sum_l P(Q_l)}, \text{ for } \begin{cases} Q_l \in [x_{d_k} - \frac{q}{2}, x_{d_k} + \frac{q}{2}] \\ LSB(l) = w_{r_k}, \end{cases} \quad (5.19)$$

where $P(Q_l)$ is the probability of the reference coefficient value (after watermark embedding) to be Q_l and q is the quantization step used for image encoding at the corresponding coefficient's position. To illustrate (5.19), consider Figure 5.17, which

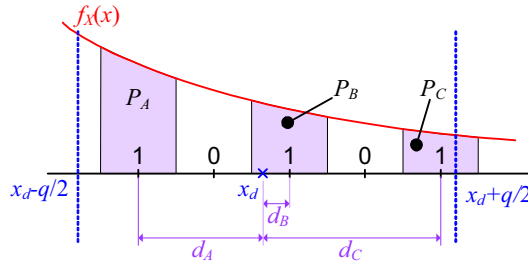


Figure 5.17: Error estimation.

represents a set of reference coefficient values Q_l (small ticks labeled with '1's and '0's). Let's admit that the embedded reference watermark bit at a given coefficient's position is '1'. $\hat{\varepsilon}_k$ is estimated by first computing the distances from the received coefficient, x_{d_k} , to the reference points laying inside the interval $[x_{d_k} - \frac{q}{2}, x_{d_k} + \frac{q}{2}]$ and assigned to watermark bit '1'. In the figure, those distances are represented by d_A , d_B and d_C . Each distance is weighted by P_A , P_B and P_C , respectively, which are the probabilities of the reference coefficient value to be in each of the points corresponding to $w_{r_k} = 1$.

However, knowledge about the probability $P(Q_l)$ is not available at the receiver, thus it must be estimated based on the received coefficient data. In order to do so, the original coefficient data is modeled using a Laplace probability density function [107] with parameter λ , which represents a reasonable trade-off between model accuracy and simplicity. According to this model, the PDF for the original coefficient values, $f_X(x)$, is given by:

$$f_X(x) = \frac{\lambda_{(i,j)}}{2} \exp(-\lambda_{(i,j)}|x|), \quad (5.20)$$

where λ is the distribution's parameter at the corresponding DCT frequency.

For algebraic simplicity purposes, it will be considered that the statistics of the watermarked DCT coefficients (the reference coefficients) are similar to those of the original DCT values. This is a reasonable approach, since the distortion due to watermark embedding is small. Thus, assuming that lossy encoding results from linear quantization with step q , the probability for the original coefficient, x_k , to be quantized to value X_k is:

$$P(X_k) = \int_{X_k - \frac{q}{2}}^{X_k + \frac{q}{2}} \frac{\lambda}{2} e^{-\lambda|x|} dx. \quad (5.21)$$

If the quantization function is symmetric and includes the zero value, which is the

case for JPEG and MPEG-2 encoding, (5.21) can be rewritten as:

$$P(X_k) = \begin{cases} 1 - e^{-\frac{\lambda q}{2}}, & \text{if } X_k = 0; \\ \frac{1}{2}e^{-\lambda|X_k| + \frac{\lambda q}{2}}(1 - e^{-\lambda q}), & \text{otherwise.} \end{cases} \quad (5.22)$$

In order to estimate the parameter λ of the original PDF using quantized coefficient values, the *maximum likelihood* (ML) method is used [108]:

$$\hat{\lambda}_{ML} = \arg \max_{\lambda} \left\{ \log \prod_{k=1}^N P(X_k) \right\}. \quad (5.23)$$

Substituting (5.22) in (5.23) leads to:

$$\hat{\lambda}_{ML} = \arg \max_{\lambda} \left\{ \sum_{k_0=1}^{N_0} \log(1 - e^{-\frac{\lambda q}{2}}) + \sum_{k_1=1}^{N_1} \log \left[\frac{1}{2} (e^{-\lambda|X_{k_1}| + \frac{\lambda q}{2}}) (1 - e^{-\lambda q}) \right] \right\}, \quad (5.24)$$

where N_0 and N_1 represent the number of coefficients at a given frequency, quantized to zero and non-zero values, respectively (with $N = N_0 + N_1$). The two summation terms in (5.24) correspond to the two possible cases in (5.22) and the quantized coefficient set, $\{X_k\}$, has been split into sets $\{X_{k_0}\}$ and $\{X_{k_1}\}$, according to those cases. Using the substitution

$$S = \sum_{k_1=1}^{N_1} |X_{k_1}|, \quad (5.25)$$

and after simple algebraic manipulations, (5.24) can be rewritten as:

$$\hat{\lambda}_{ML} = \arg \max_{\lambda} \left\{ N_0 \log(1 - e^{-\frac{\lambda q}{2}}) - \lambda S + \frac{N_1 \lambda q}{2} + N_1 \log(1 - e^{-\lambda q}) \right\}. \quad (5.26)$$

The solution can be found by looking for the zeros of the derivative of (5.26) with respect to λ , leading to:

$$(Nq + 2S)e^{-\lambda q} + N_0 q e^{-\frac{\lambda q}{2}} + N_1 q - 2S = 0. \quad (5.27)$$

Equation (5.27) can be viewed as a second order polynomial in $e^{-\frac{\lambda q}{2}}$, whose solution is:

$$\hat{\lambda}_{ML} = -\frac{2}{q} \log \frac{-N_0 q + \sqrt{N_0^2 q^2 - 4(Nq + 2S)(N_1 q - 2S)}}{2Nq + 4S}. \quad (5.28)$$

The parameter $\hat{\lambda}_{ML}$ retrieved by (5.28) can then be used to compute the values of

$P(Q_l)$:

$$P(Q_l) = \int_{\frac{Q_{l-1}+Q_l}{2}}^{\frac{Q_l+Q_{l+1}}{2}} \frac{\hat{\lambda}_{ML}}{2} \exp(-\hat{\lambda}_{ML}|x|) dx. \quad (5.29)$$

The absolute value for the local error, $\hat{\varepsilon}_k$, can then be estimated by substituting $P(Q_l)$ in (5.19), using the result from (5.29).

5.3.3 Quality estimation

Local error estimates resulting from the methods described in the previous section can be used for quality estimation purposes. For instance, an estimate for the distorted image PSNR can be computed according to:

$$\text{PSNR}_{\text{est}}(dB) = 10 \log_{10} \frac{255^2}{\frac{1}{M} \sum_{k=1}^M \hat{\varepsilon}_k^2}, \quad (5.30)$$

where M is the number of DCT coefficients and $\hat{\varepsilon}_k^2$ is the error estimate associated to the k -th coefficient, computed using equation (5.19).

However, it is far more interesting to use the estimated error with the purpose of scoring the perceptual quality of the received images. A “no-reference” approach for Watson’s model can be performed by computing estimates for slack values, \hat{s} , based on the received coefficient values. Attending to (5.6), these estimates can be written as:

$$\hat{s}_k = \begin{cases} \hat{T}_{L_k}, & \text{if } |\hat{x}_k| \leq \hat{T}_{L_k}; \\ |\hat{x}_k|^b \hat{T}_{L_k}^{1-b}, & \text{otherwise,} \end{cases} \quad (5.31)$$

where \hat{T}_{L_k} is an estimate for Watson’s luminance threshold and \hat{x}_k is an estimate for the original coefficient value, which are given by:

$$\hat{T}_{L_k} = T_B \left(\frac{\hat{x}_k}{x_{00}} \right)^{\alpha_T}; \quad \hat{x}_k = x_{d_k} + \hat{\varepsilon}'_k. \quad (5.32)$$

$\hat{\varepsilon}'_k$ is a signed estimate for the local error, computed similarly to (5.19), but using the difference $(x_{d_k} - Q_l)$ inside the summation, instead of the absolute value. This signed error estimate is added to the received coefficient value, resulting in an estimate of the original coefficient value, which is used in (5.31) for computing the corresponding slack value. The perceptual error is then computed using the slack value and local

perceptual error estimates:

$$\hat{\varepsilon}_{p_k} = \frac{\hat{\varepsilon}_k}{\hat{s}_k}, \quad (5.33)$$

To conclude, a global distortion measure for the whole image, \hat{D}_W , is computed by combining all local perceptual errors that result from (5.33) according to:

$$\hat{D}_W = \sqrt[4]{\frac{1}{M} \sum_{k=1}^N \sum_{i=0}^7 \sum_{j=0}^7 \hat{\varepsilon}_{p_k}(i, j)^4}, \quad (5.34)$$

which is the L_4 error pooling procedure suggested in Watson's model. Note that the value resulting from (5.34) is an estimate for the Watson's perceptual distortion measurement, that is computed without using the reference image.

5.4 Results

5.4.1 PSNR estimation

In order to evaluate the performance of the algorithms for PSNR estimation purposes, the reference images in LIVE database [109] have been watermarked and JPEG encoded using quality factors from 10 to 100, using steps of 10. The result is a set of 290 encoded images that span a wide range of content, resolutions and quality.

Both error estimation strategies described in 5.3.1 and 5.3.2 have been used in the experiments. For the first strategy, the watermark's embedding strength, α_w , was set to 1.0 and the watermark has been embedded only in the set of DCT coefficients represented in Figure 5.10. In the second strategy, all DCT coefficients (except the DC coefficients) have been watermarked and α_w was set to 0.5. Note that the strategy of error estimation based on the watermarking bit error rate requires a larger embedding strength than the error estimation strategy based on DCT coefficient statistics. The larger embedding strength is required for monotonically scale W_{ber} values as distortion increases (otherwise the value of W_{ber} would saturate at 0.5 for relatively small distortions).

After lossy encoding of the reference (watermarked) images, no-reference PSNR estimates given by (5.30) have been compared with the true PSNR values.

Figure 5.18 depicts the results attained for two test images (from the LIVE data-

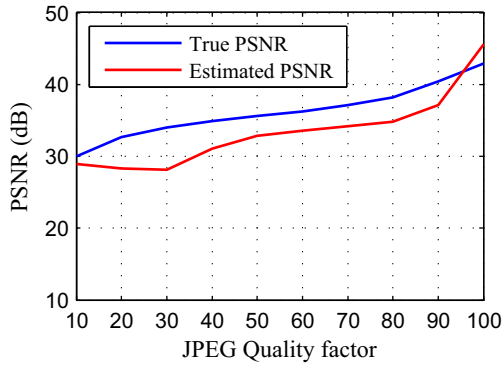
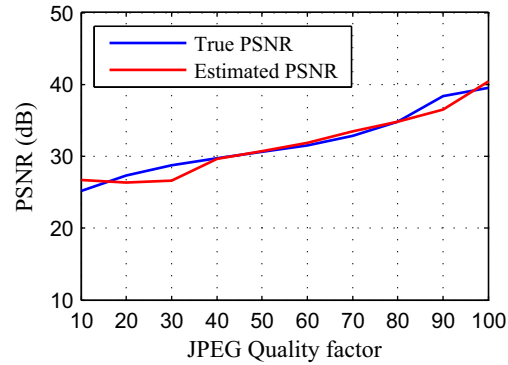
(a) Image *Monarch*.(b) Image *Dancers*.(c) PSNR estimation – *Monarch*.(d) PSNR estimation – *Dancers*.

Figure 5.18: Examples of PSNR estimation under JPEG encoding, using distance weighting based on the watermark extraction bit error rate.

base), using the error weighting strategy based on the watermark bit error rates. These examples illustrate the worst (image *Monarch*) and the best (image *Dancers*) results that were obtained using this strategy. As Figure 5.18-c) shows, PSNR estimates may not be accurate enough in some cases.

On the other hand, error weighting based on DCT coefficient statistics performs much better, as can be observed in the plots depicted in Figure 5.19.

These considerations are confirmed in the plots shown in Figure 5.20, which represent the true versus estimated PSNR values, for all JPEG encoded images used in the experiments (290 images). As can be observed, PSNR estimates using the error weighting strategy based on coefficient statistics clearly outperform those resulting from the weighting based on the watermark's bit error rate.

Table 5.2 synthesizes global statistics for the error between the true and the estimated PSNR values resulting from these experiments. Again, it can be observed that the results corresponding to error weighting based on DCT coefficient statistics

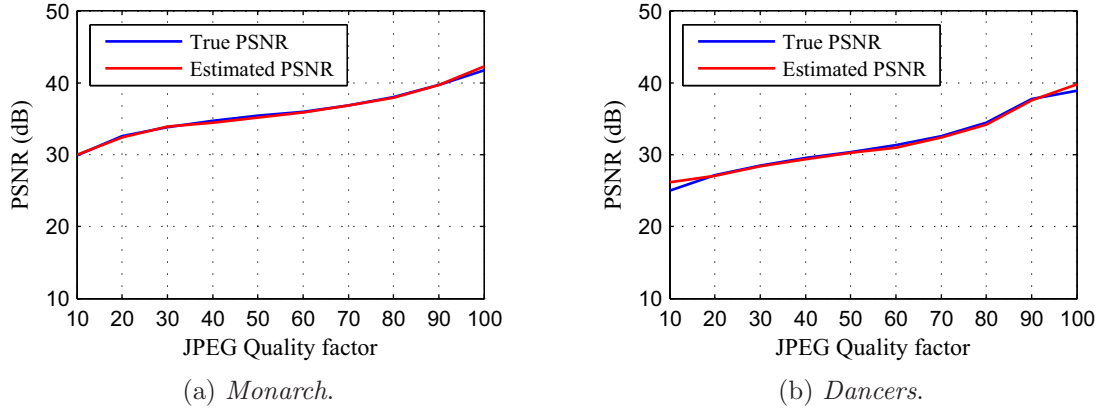


Figure 5.19: Examples of PSNR estimation under JPEG encoding, using distance weighting based on DCT coefficients' statistics.

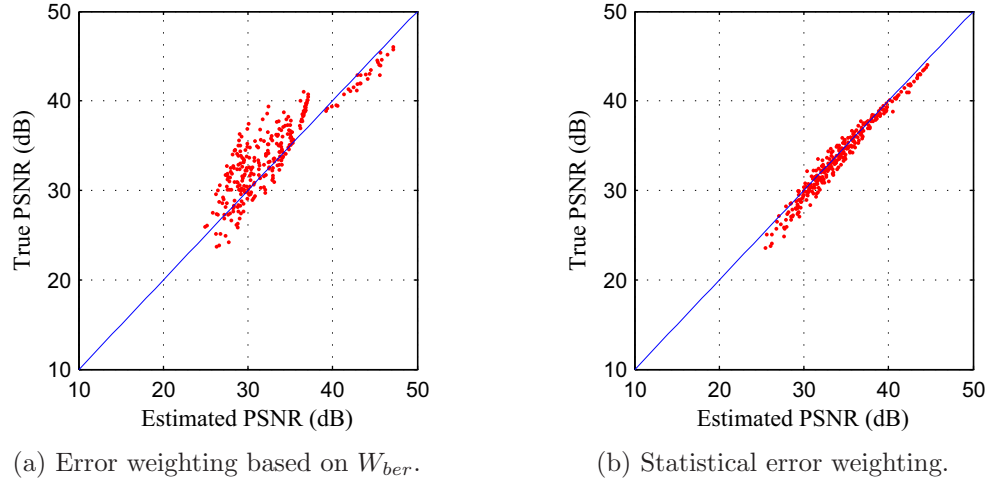


Figure 5.20: Global PSNR estimation results for the two distance weighting methods.

are significantly better than those achieved with error weighting based on W_{ber} .

5.4.2 Quality scores

In the previous section, two different strategies for distortion estimation have been tested. While both strategies have similar implementation costs, the results corresponding to the distance weighting strategy outperformed those corresponding to distance weighting based on the watermark's bit error rate. Therefore, MOS

	Based on W_{ber}	Based on coef. statistics
Average estimation error	2.123 dB	0.703 dB
Root mean squared error	2.686 dB	0.888 dB
Linear correlation (estimated <i>vs.</i> true PSNR)	0.883	0.984

Table 5.2: PSNR estimation accuracy.

predictions using the proposed watermark-based algorithm have been computed considering the best error estimation strategy only.

The results for quality assessment have been evaluated by comparing the quality scores computed by the algorithm with the ones that result from a subjective test. LIVE database contains subjective scores for images subject to JPEG encoding using different quality factors. The corresponding subjective scores are expressed by their *differential mean opinion scores* (DMOS), which is the quality score difference between the reference and the distorted image (*i.e.*, image quality decreases with increasing values of DMOS).

Figure 5.21-a) depicts the estimated Watson's distance, using (5.34) and the perceptual error estimates given by (5.33), versus the corresponding DMOS values. Following a procedure similar to what is suggest by VQEG in [110], a logistic function was used in order to normalize the values that result from (5.34) into a linear quality scale from 0 – 100. The logistic function has the form:

$$\text{Estimated DMOS} = a_0 + \frac{a_1}{1 + \exp(a_2 \hat{D}_W + a_3)}, \quad (5.35)$$

where a_0 to a_3 are parameters that can be tuned for fitting. These parameters have been computed in order to minimize the square differences between the estimated DMOS scores given by (5.35) and the true DMOS values in a given training set. The training set for finding the parameter values consists of DMOS scores given to the JPEG encoded versions of 15 reference images randomly chosen from the LIVE database. The a_i parameters have been computed using the *Levenberg-Marquardt* method for non-linear least squares minimization problems. The resulting logistic function can be observed in Figure 5.21-a).

Figure 5.21-b) depicts the normalized no-reference quality scores versus their DMOS values. As can be observed, objective quality scores resulting from the proposed

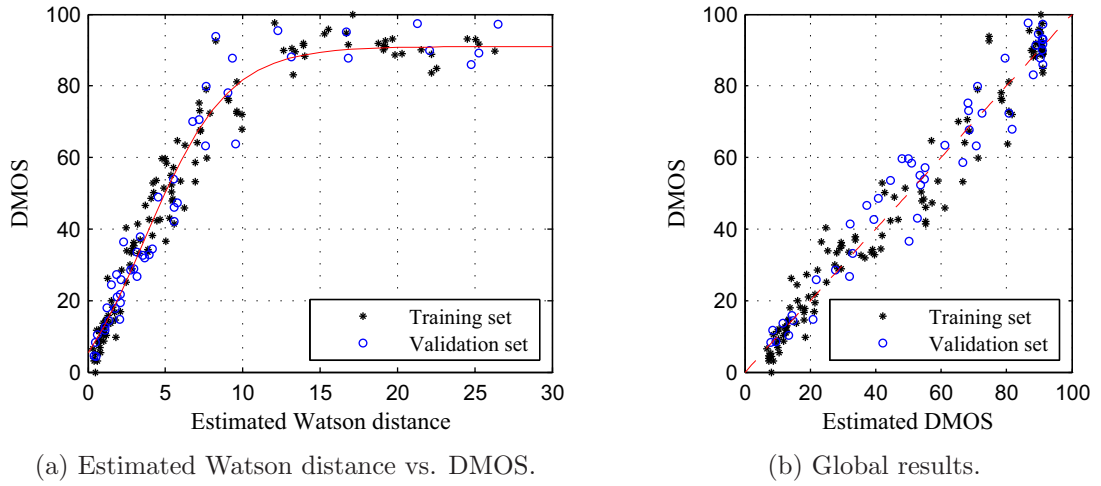


Figure 5.21: DMOS estimation results.

Root mean square (RMS)	6.430
Pearson's correlation coefficient (ρ_c)	0.975
Spearman's rank order coefficient (ρ_s)	0.955

Table 5.3: Evaluation of the proposed metric.

algorithm are well correlated with the subjective quality ranks, both in training and validation sets (the latter consists of the JPEG encoded image versions of 14 reference images in LIVE database). The metric's performance indicators suggested by VQEG [110], described in Section 4.7 of the previous chapter, have been measured and synthesized in Table 5.3, for the validation set, only. The results confirm the good performance of the proposed algorithm.

5.5 Summary

The main achievement of this chapter was the development of a watermark-based no-reference image quality assessment algorithm that resembles the JND-based quality metric derived by A. B. Watson in [8].

The chapter started by describing the watermark embedding and extraction schemes. For controlling the strength of watermark embedding, frequency adapted quantization functions have been derived, aiming to maximize the watermark's embedding

strength while assuring the imperceptibility of the watermark. This derivation was performed by considering the characteristics of the human visual system, using the perceptual model by Watson.

Two different strategies that aim to compensate the distortion errors obtained directly by the watermark signal have been proposed: an empirical weighting strategy based on the watermark's extraction bit error rate and a strategy based on the statistical distribution of the DCT coefficients. The former was not able to provide accurate PSNR estimates while the latter showed good results for blindly computing local error estimations in the presence of JPEG lossy encoding.

Using the most accurate error prediction strategy, error estimates resulting from the watermark-based algorithm were then used for computing image quality scores, following a close approach to the model derived by Watson. Using those error estimates, slack values have been estimated and consequently, a prediction for local perceived errors has been computed. By pooling those perceptual error predictions, no-reference quality scores have been computed and compared with DMOS data resulting from subjective quality assessments. The results have shown that there is a strong relation between the quality scores given by human viewers and those resulting from the proposed algorithm.

Chapter 6

Statistical image quality assessment

6.1 Introduction

In the previous chapter, a no-reference image quality assessment algorithm based on watermarking techniques was presented. It has been verified that the use of DCT coefficients' statistics, combined with the watermark signal, lead to a significant error estimation improvement when compared with the use of the watermark signal only. Furthermore, this fact suggests that it may be possible to design an error estimation algorithm that relies only on the observed coefficient values at the decoder side. However, estimates for the coefficient distribution parameters using the distorted data, were not always accurate, especially at lower encoding bit rates – an inaccuracy that was partially compensated due to the use of the watermark signal. In this chapter, a no-reference quality assessment algorithm that overcomes this problem, without using watermarks, is presented. It was designed to work with images subject to quantization noise in the blockwise DCT domain.

In the work by Turaga *et al.* in [87], reviewed in Section 4.5.2, the PSNR of MPEG-2 encoded video is estimated based on the statistical properties of DCT coefficients. The statistical distribution of the DCT coefficients is modeled according to a zero-mean Laplace PDF, whose parameter is estimated at the decoder side. However, the parameter estimation procedure proposed in [87] becomes inaccurate as the encoding bitrate decreases. The reason for such inaccuracy is the increasing number of DCT coefficients that are quantized to zero values during encoding, which is quite high for low bitrates.

This problem has been handled by Ichigaya *et al.* in [88], where the DCT coefficients distribution is modeled using a weighted mixture of two Laplace PDFs. One of those PDFs is computed by considering all quantized coefficient values while the other is computed by considering the non-zero quantized values only. A considerable improvement is reported in [88], when compared with [87]. However, the method is inaccurate if all (or almost all) DCT coefficients at the same frequency are quantized to zero, a situation which is quite common for high-frequency coefficients subject to DCT-based encoding (even at average compression rates). The method in [88] was further improved in [111] in order to deal with uncoded macroblocks that may occur in P and B frames, increasing the accuracy of PSNR estimates for those frame types, but the limitation regarding the “all coefficients quantized to zero” is still present.

One of the main goals of the work described in this chapter is to improve the estimation of the DCT coefficients data distribution when compared with the above mentioned works [87, 88, 111], using the quantized values available at the receiver side. In order to accomplish this task, the following ideas have been considered:

- to explore the correlation between distribution parameters at adjacent DCT frequencies;
- to combine the above prediction results with maximum likelihood (ML) parameter estimates.

In order to provide image quality scores, the proposed method is extended by considering perceptual characteristics of the human eye, in terms of *just noticeable differences*. The objective is to estimate perceptual weights and distortion errors at the receiver, in such a way that quality scores given to the distorted image resemble the perceptual metric proposed by Watson in [8], already used in the previous chapter. The results of the proposed algorithm are compared with those resulting from other state-of-the-art algorithms presented in [21] and [80]. In [21], quality scores are computed by measuring and combining specific JPEG compression artifacts; in [80], quality scores result from combining the outputs of neural networks, whose input data consists of block-based features taken from the image under evaluation.

The work described in this chapter has been evaluated using JPEG encoded images. Nevertheless, it presents the main ideas that support the video quality assessment algorithm described in the next chapter.

This chapter is organized as follows: after the introduction, Section 6.2 describes the general architecture of the proposed algorithm. Section 6.3 provides the details

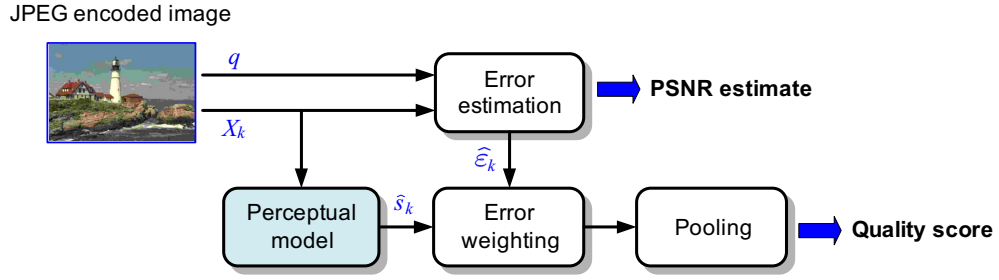


Figure 6.1: General scheme of the proposed quality assessment algorithm.

related to DCT coefficient distribution modeling, using a Laplace PDF. Maximum-likelihood estimation of the Laplace distribution parameter from the original and from the quantized coefficient values is also explained. Section 6.4 presents the proposed method for estimating the distribution’s parameter, exploring the correlation between adjacent coefficients distributions. The procedures to obtain quality scores from the estimated DCT distributions are given in Section 6.5, and Section 6.6 depicts the results, which are compared with the methods proposed in [21, 80, 88]. Finally, the main conclusions of this chapter are synthesized in Section 6.7.

6.2 Algorithm overview

A general scheme of the proposed algorithm for assessing the quality of JPEG encoded images is represented in Figure 6.1. It consists of an error estimation block, whose function is to compute local error estimates, and an error weighting block, which weights those error estimates according to a perceptual model, and combines them in order to compute the image’s quality score.

It is admitted that the image under evaluation is corrupted by quantization noise in the DCT domain. More specifically, this assumption applies to images encoded (or transcoded) using the JPEG standard. The inputs for the proposed quality assessment algorithm are the quantized DCT coefficients’ values and their corresponding quantization steps.

Let us assume that the distribution of the original DCT coefficient data (*i.e.*, before lossy encoding) is known. In this case, an estimate for the local mean square error,

$\hat{\varepsilon}_k^2$, at the k -th coefficient, can be computed by observing its quantized value, X_k :

$$\hat{\varepsilon}_k^2 = \int_{-\infty}^{+\infty} f_X(x|X_k)(X_k - x)^2 dx, \quad (6.1)$$

where $f_X(x|X_k)$ represents the probability density function of the original DCT coefficient values, conditioned to the observed value of X_k . According to *Bayes rule*, it can also be written as:

$$f_X(x|X_k) = \frac{P(X_k|x)f_X(x)}{P(X_k)}, \quad (6.2)$$

where $P(X_k|x)$ is the probability of having quantizer's output X_k given the coefficient's value x . This probability is 1 if x lies in the quantization interval centered in X_k , and 0 otherwise. Assuming uniform quantization with a constant step size q (which is true for JPEG encoding), it can be formally written as:

$$P(X_k|x) = \begin{cases} 1, & \text{if } x \in [X_k - \frac{q}{2}; X_k + \frac{q}{2}]; \\ 0, & \text{otherwise.} \end{cases} \quad (6.3)$$

Substituting (6.2) in (6.1), and considering (6.3), leads to:

$$\hat{\varepsilon}_k^2 = \frac{1}{P(X_k)} \int_{X_k - \frac{q}{2}}^{X_k + \frac{q}{2}} f_X(x)(X_k - x)^2 dx, \quad (6.4)$$

where $f_X(x)$ is the density of the original coefficient data and $P(X_k)$ is the probability of a random coefficient to be quantized to value X_k :

$$P(X_k) = \int_{X_k - \frac{q}{2}}^{X_k + \frac{q}{2}} f_X(x) dx. \quad (6.5)$$

From equations (6.4) and (6.5), it can be concluded that the squared error estimate depends on the value of the quantized coefficient X_k , on the quantization step, q , and on the probability density function of the coefficient values, $f_X(x)$. In the context of JPEG encoded images, both X_k and q can be extracted directly from the image bitstream. A no-reference PSNR estimate can therefore be computed after finding $f_X(x)$.

Based on the estimated values for $\hat{\varepsilon}_k^2$, it is possible to estimate the PSNR of the encoded image, using square error estimates, instead of their true values:

$$\text{PSNR}_{\text{est}}(\text{dB}) = 10 \log_{10} \frac{255^2}{\text{MSE}_{\text{est}}}; \text{ with } \text{MSE}_{\text{est}} = \frac{1}{N} \sum_{k=1}^N \hat{\varepsilon}_k^2, \quad (6.6)$$

where N is the number of DCT coefficients. Remember that, as already stated in Section 5.2.4, it is indifferent to measure the PSNR in the pixel or in the DCT domain (Parseval's theorem).

Similarly to what was done in the previous chapter, the DCT coefficient error estimates that result from the local error estimation module are then weighted according to a perceptual model based on Watson's work [8], adapted for no-reference quality assessment. Remember that the function of this model is to compute the local perceptual weights, s_k , which reflect the sensibility of the HVS to the corresponding local errors. For a given image location, the more sensible the HVS is, the lower will be the value of s_k . The only inputs for the perceptual model are the DCT coefficient values, which can be extracted from the encoded bitstream. Local errors are weighted accordingly and pooled together, resulting in a global distortion metric that acts as a non-normalized image quality score.

6.3 Modeling DCT coefficient data

Block-wise DCT coefficient data distribution of natural images can be well modeled using a Laplace probability density function [107]. In this case, for a $K \times K$ block-wise DCT transform ($K = 8$ in the case of JPEG encoding), the coefficient's PDF at each horizontal/vertical frequency pair, $(i, j) \in \{0, \dots, K-1\} \times \{0, \dots, K-1\}$ and $(i, j) \neq (0, 0)$, is given by:

$$f_X(x) = \frac{\lambda_{(i,j)}}{2} \exp(-\lambda_{(i,j)}|x|), \quad (6.7)$$

where $\lambda_{(i,j)}$ is the distribution's parameter for frequency pair (i, j) (for simplicity, these indexes will be dropped along the text) and x is the coefficient value.

Other PDF models have been suggest in the literature: generalized Gaussian [112], Gaussian mixtures [113] and generalized gamma [114] are probably the most relevant examples. Nevertheless, the Laplacian model represents a good trade-off between model accuracy and mathematical simplicity. As an example, Figure 6.2 depicts a comparison between the histogram of coefficient values and the corresponding Laplace PDF (at a given frequency).

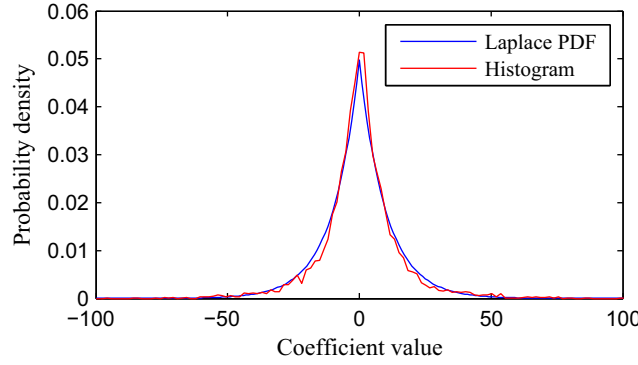


Figure 6.2: An example histogram of the DCT coefficients located at frequency (3,3) of image *Rapids*.

6.3.1 Parameter estimation using original coefficient data

An estimate for λ , using the original coefficient data, is generally computed using the maximum likelihood method [115]. Representing by x_k the k -th original coefficient value at a given frequency, an ML estimate for λ is given by:

$$\lambda_{ML} = \arg \max_{\lambda} \left\{ \log \prod_{k=1}^N f_X(x_k) \right\}, \quad (6.8)$$

where N is the number of DCT coefficients at that frequency (which is the same as the number of image blocks). The value of λ_{ML} is computed by finding the zeros of the derivative with respect to λ , which leads to:

$$\lambda_{ML} = \frac{N}{\sum_{k=1}^N |x_k|} = \frac{1}{E[|x|]}, \quad (6.9)$$

where $E[\cdot]$ represents the expected value. In the context of this work, λ_{ML} was used as the benchmark for parameter estimation based on quantized data. Therefore, it will be addressed to as the *original* λ .

6.3.2 Parameter estimation using quantized coefficient data

Using the same procedure described in Section 5.3.2 of the previous chapter, an ML estimate for the parameter λ based on the quantized coefficient values, $\hat{\lambda}_{ML}$, is:

$$\hat{\lambda}_{ML} = -\frac{2}{q} \log \frac{-N_0 q + \sqrt{N_0^2 q^2 - 4(Nq + 2S)(N_1 q - 2S)}}{2Nq + 4S}, \quad (6.10)$$

where q is the quantization step, N_0 and N_1 are the the number of coefficients quantized to zero and nonzero values, respectively, and S is defined as:

$$S = \sum_{k_1=1}^{N_1} |X_{k_1}|, \quad (6.11)$$

where X_{k_1} are the nonzero quantized coefficient values.

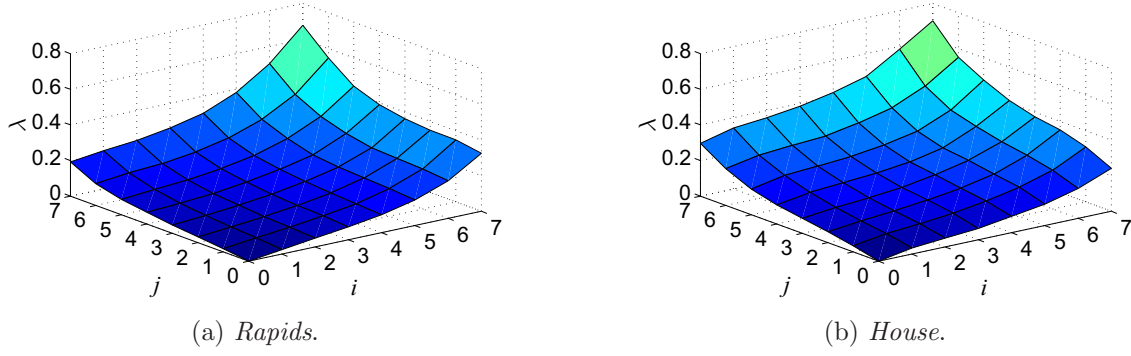
Let us now take a deeper insight into equation (6.10), for the case where most of the DCT coefficients have been quantized to value zero. As the number of coefficients quantized to zero increases, $N_0 \rightarrow N$, and it is easy to conclude that $N_1 \rightarrow 0$ and $S \rightarrow 0$. Under these conditions, the argument of the logarithm in (6.10) will tend to zero and therefore $\hat{\lambda}_{ML}$ will tend to infinity, meaning that the estimated distribution will approach a *Dirac's delta* function. This situation will be common in the presence of DCT-based compression, since high frequency DCT coefficients are typically quantized to zero, even at average compression rates. As a consequence, ML estimates for λ based on the quantized data will be inaccurate for these cases.

This phenomenon has already been noticed in [88], where the authors propose to compute the original DCT distribution as a mixture of two Laplace PDFs: one of them is estimated by considering all the quantized coefficient values, while the other is estimated by considering quantized non-zero values only. However, the proposed algorithm does not deal with the case where all DCT coefficients at a given frequency are quantized to zero. Due to its characteristics, the algorithm proposed in [88] will be addressed to as the *Laplacian compensation* method, for the remainder of the Thesis.

6.4 Parameter estimation using prediction

In order to tackle the problem described at the end of the previous section, a new approach to improve the estimation of the Laplacian parameter is proposed: to explore the correlation between λ values at neighboring DCT frequencies.

Consider Figures 6.3-a) and b), which represent the original λ values, computed using equation (6.9), for all 8×8 DCT frequencies in two test images. The figures show a strong correlation between λ values at adjacent frequencies. Besides the evidence shown in the figures, the average correlation between parameter values was also measured using the set of reference (original) images in LIVE database [109]. Using a 4-connected neighborhood, the resulting value for the correlation between

Figure 6.3: Original λ values for each frequency.

the value of λ at a given frequency and the values of λ in adjacent frequencies was 0.971.

One possible way to explore this correlation, is to use a linear predictor. Representing the prediction result by $\hat{\lambda}_p$, it can be written:

$$\hat{\lambda}_p = w_0 + \sum_{k=1}^K \lambda_{v_k} w_k, \quad (6.12)$$

where λ_{v_k} is the k -th neighbor of the λ value to predict, K is the number of neighbors and w_k is the corresponding linear weight (which is found using a training procedure). Equation (6.12) can also be written in matrix form as:

$$\hat{\lambda}_p = \boldsymbol{\lambda}_v^T \mathbf{w}, \quad \text{with } \boldsymbol{\lambda}_v = \begin{bmatrix} 1 \\ \lambda_{v_1} \\ \vdots \\ \lambda_{v_K} \end{bmatrix} \quad \text{and } \mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_K \end{bmatrix}. \quad (6.13)$$

The value of $\hat{\lambda}_p$ that results from (6.13) can then be combined with $\hat{\lambda}_{ML}$ in order to improve the estimation accuracy for the original DCT distribution parameter.

Since ML estimates become more inaccurate as the rate of coefficients quantized to zero increases, more trust should be given to the predictor in these situations. On the other hand, if the number of coefficients quantized to zero is low, the ML estimator will most likely get accurate results, so there is no real need for the predicted value. Based on these premises, a simple criterion for combining $\hat{\lambda}_p$ with $\hat{\lambda}_{ML}$ is to weight them proportionally to the rate of DCT coefficients quantized to zero:

$$\hat{\lambda}_f = r_0 \hat{\lambda}_p + (1 - r_0) \hat{\lambda}_{ML}, \quad (6.14)$$

where $r_0 = \frac{N_0}{N}$ represents the rate of coefficients quantized to zero and $\hat{\lambda}_f$ is the final estimate for the distribution's parameter.

6.4.1 Predictor training procedure

The objective of the training procedure is to find a weight vector \mathbf{w} that is suitable for the proposed linear prediction scheme, given by equation (6.12). One possible way to compute \mathbf{w} is by minimizing the square error between the original λ and $\hat{\lambda}_p$, for all images in the training set. Admitting that L images are available for training, then there will be L vectors $\boldsymbol{\lambda}_v$ and L values of λ per DCT coefficient frequency. Therefore, it can be written:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} \left\{ \sum_{l=1}^L (\lambda^{(l)} - \hat{\lambda}_p^{(l)})^2 \right\} = \arg \min_{\mathbf{w}} \left\{ \sum_{l=1}^L (\lambda^{(l)} - \boldsymbol{\lambda}_v^{T(l)} \mathbf{w})^2 \right\}. \quad (6.15)$$

Using matrix notation, the equation above can be rewritten as:

$$\hat{\mathbf{w}} = \arg \min_{\mathbf{w}} \{ [\boldsymbol{\lambda} - \boldsymbol{\Lambda} \mathbf{w}]^T [\boldsymbol{\lambda} - \boldsymbol{\Lambda} \mathbf{w}] \}, \quad (6.16)$$

where $\boldsymbol{\Lambda}$ is an $L \times K$ matrix, with the neighborhood samples for image l , $\lambda_{v_k}^{(l)}$, organized in rows and $\boldsymbol{\lambda}$ is a vector with the original λ values at the position to predict, *i.e.*:

$$\boldsymbol{\Lambda} = \begin{bmatrix} 1 & \lambda_{v_1}^{(1)} & \dots & \lambda_{v_K}^{(1)} \\ 1 & \lambda_{v_1}^{(2)} & \dots & \lambda_{v_K}^{(2)} \\ \vdots & \vdots & & \vdots \\ 1 & \lambda_{v_1}^{(L)} & \dots & \lambda_{v_K}^{(L)} \end{bmatrix} \quad \text{and} \quad \boldsymbol{\lambda} = \begin{bmatrix} \lambda^{(1)} \\ \lambda^{(2)} \\ \vdots \\ \lambda^{(L)} \end{bmatrix}.$$

The solution of equation (6.16) can be found by differentiating with respect to \mathbf{w} :

$$\nabla_{\mathbf{w}} = 0 \Leftrightarrow \boldsymbol{\Lambda}^T (\boldsymbol{\lambda} - \boldsymbol{\Lambda} \mathbf{w}) = 0. \quad (6.17)$$

Finally, if $\boldsymbol{\Lambda}^T \boldsymbol{\Lambda}$ is non-singular, the unique solution for $\hat{\mathbf{w}}$ is given by:

$$\hat{\mathbf{w}} = (\boldsymbol{\Lambda}^T \boldsymbol{\Lambda})^{-1} \boldsymbol{\Lambda}^T \boldsymbol{\lambda}. \quad (6.18)$$

The original (reference) images in LIVE database [109] have been randomly split into two groups, one for training (15 images) and another for validation (14 images). The neighborhood configuration used in the experiments is illustrated in Figure 6.4. The criteria for choosing this structure were the minimization of neighborhood ele-

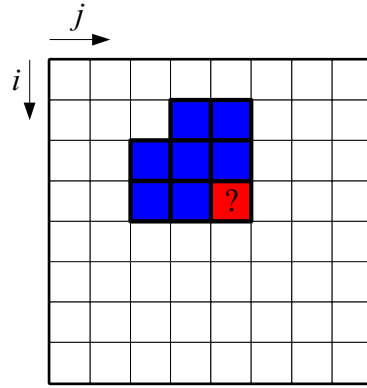


Figure 6.4: Neighborhood configuration used for prediction.

ments required for an accurate parameter estimation and to allow the possibility of recursively predict values for λ , starting from the low frequency positions (since the corresponding coefficients are less vulnerable to the effect of lossy encoding). The procedure to obtain vector \mathbf{w} can thus be described in the following steps:

1. for each image in the training set, the original values of λ are computed for all frequencies, using (6.9) and the original coefficient data;
2. for each frequency, the neighborhood matrix $\mathbf{\Lambda}$ and vector $\mathbf{\lambda}$ are constructed using the values that result from the previous step;
3. predictor weights are computed for each frequency, using equation (6.18).

6.4.2 Prediction accuracy

To evaluate the effectiveness of the prediction scheme, two experiments have been performed using the validation image set. In a first experiment, $\hat{\lambda}_p$ was computed using the original λ values in the neighborhood. The main purposes of this experiment were to evaluate the lowest prediction error that could be expected and to validate the training procedure. The average relative prediction error (in percentage) is depicted in Table 6.1-a). From the table, it can be observed that the relative error is generally low, which confirms that, due to high correlation of λ values in adjacent frequencies, prediction results are quite accurate. It can also be observed that the error is higher for the cases where less neighborhood values are available for the prediction (for instance, in the first row and the first column).

	\xrightarrow{j}							
$i \downarrow$	-	-	11.7	9.9	4.9	4.5	4.7	7.7
	-	11.9	6.2	5.1	2.7	2.3	3.3	4.1
	10.9	5.2	2.7	2.5	2.8	2.4	4.0	2.0
	7.5	4.0	2.8	4.4	2.6	3.1	2.7	4.6
	6.1	3.7	4.3	5.4	2.1	4.3	4.2	2.7
	6.5	3.0	2.6	3.3	3.4	2.2	1.6	2.8
	5.8	2.7	1.8	2.6	2.4	1.8	3.4	2.7
	3.3	5.0	2.9	3.0	2.5	2.5	2.3	1.9

(a) Normal prediction.

	\xrightarrow{j}							
$i \downarrow$	-	-	-	-	-	4.5	6.0	8.8
	-	-	-	-	2.7	3.6	7.4	7.3
	-	-	-	2.5	4.2	4.2	8.5	8.2
	-	-	2.8	4.5	5.5	6.5	11.0	18.3
	-	3.7	4.7	6.1	5.7	6.6	8.5	13.9
	6.5	6.0	7.3	7.5	7.8	8.7	10.2	13.9
	12.2	10.0	10.6	8.5	11.4	10.0	12.0	14.8
	19.1	12.8	10.9	11.2	13.0	14.0	16.5	18.3

(b) Recursive prediction.

Table 6.1: Average relative prediction error [%].

A second experiment tried to make an approximation to what happens when images are subject to lossy compression. Remember that, in the presence of lossy compression, most high frequency coefficients are quantized to zero, resulting in unreliable ML parameter estimates. On the other hand, low frequency coefficients are less affected by distortion, and thus more reliable parameter estimates can be performed. To approximate this situation, the original values of λ have been computed only for the low frequency range (frequency pairs (i, j) with $i + j < 5$ and $(i, j) \neq (0, 0)$, which correspond to the 14 frequencies marked with symbol ‘-’ in Table 6.1-b)). The remaining values of λ have been computed recursively from the values at those frequencies, using equation (6.12). The average relative prediction error that has result from this test is shown in Table 6.1-b). It can be observed that prediction error increases with increasing frequency. This is due to error propagation in the recursive algorithm: for lower frequencies, predictions are performed using the orig-

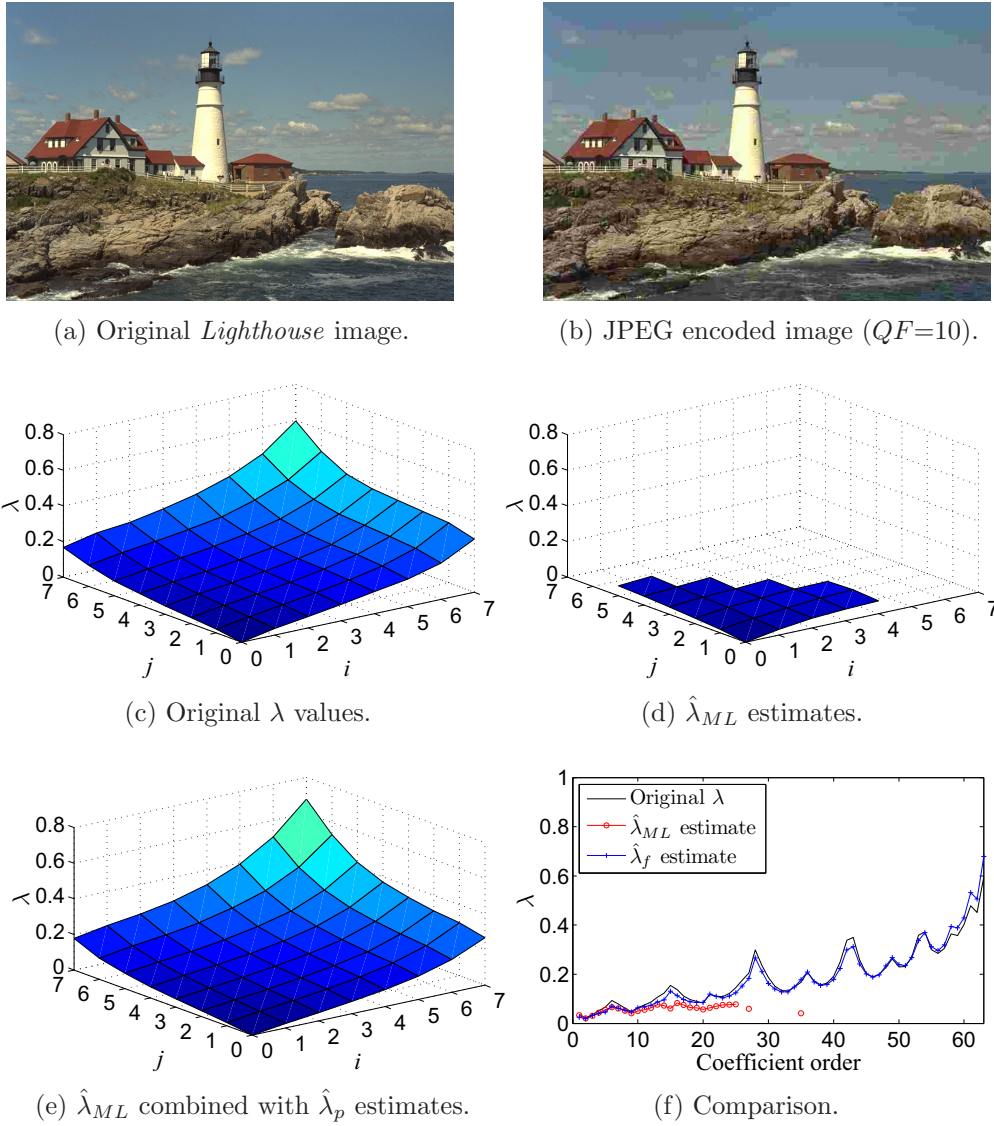


Figure 6.5: λ estimation results for a JPEG encoded test image (*lighthouse*).

inal values of λ , but with increasing frequency, predictions are based on previously predicted values. Nevertheless, the results are quite satisfactory.

Additionally, Figure 6.5 illustrates λ estimation results in the presence of lossy DCT-based encoding. For this example, the image *Lighthouse*, shown in Figure 6.5-a), has been subject to JPEG encoding, with the quality factor set to 10, resulting in the image depicted in 6.5-b). Figure 6.5-c) shows the original values of λ , computed using equation (6.9). These values have been considered as an estimation benchmark. Figure 6.5-d) depicts the estimated values for λ using the quantized coefficient data, computed through (6.10). Due to all data quantized to zero at the medium-high DCT frequencies, the resulting estimates at those frequencies are

infinite (for better visualization, λ values in these conditions were not represented). By combining ML estimates with the proposed prediction scheme, using equation (6.14), it can be observed in Figure 6.5-e) that the resulting λ estimates are stable (no infinite values are computed) and quite close to the estimates based on the original coefficient data. For a better comparison, Figure 6.5-f) joins together on a 2-D plot the results presented in Figures 6.5-c), d) and e).

6.5 Perceptual quality estimation

In order to score the perceptual quality of the received images, Watson's model [8] was used, similarly to what was done in Section 5.3.3 of the previous chapter. In short, a global perceptual distortion metric, \hat{D}_W , is computed by combining all the ratios between estimated errors and the corresponding slack estimates, using L_4 error pooling, *i.e.*:

$$\hat{D}_W = \sqrt[4]{\frac{1}{M} \sum_{k=1}^M \left(\frac{\hat{\varepsilon}_k}{\hat{s}_k} \right)^4}, \quad (6.19)$$

where M is the number of coefficients under analysis.

An estimate for the local error, $\hat{\varepsilon}_k$, can be computed similarly to the squared error estimate given by equation (6.4), replacing the squared term inside the integral with the absolute value of the error, *i.e.*:

$$\hat{\varepsilon}_k = \frac{1}{P(X_k)} \int_{X_k - \frac{q}{2}}^{X_k + \frac{q}{2}} \frac{\hat{\lambda}_f}{2} \exp(-\hat{\lambda}_f |x|) |X_k - x| dx, \quad (6.20)$$

with

$$P(X_k) = \int_{X_k - \frac{q}{2}}^{X_k + \frac{q}{2}} \frac{\hat{\lambda}_f}{2} \exp(-\hat{\lambda}_f |x|) dx. \quad (6.21)$$

The estimate for the slack value can be obtained similarly to what was described in Section 5.3.3:

$$\hat{s}_k = \begin{cases} \hat{T}_{L_k}, & \text{if } |\hat{x}_k| \leq \hat{T}_{L_k}; \\ |\hat{x}_k|^b \hat{T}_{L_k}^{1-b}, & \text{otherwise,} \end{cases} \quad (6.22)$$

with

$$\hat{T}_{L_k} = T_B \left(\frac{\hat{x}_k}{\bar{x}_{00}} \right)^{\alpha_T} \quad \text{and} \quad \hat{x}_k = X_k + \hat{\varepsilon}'_k. \quad (6.23)$$

\hat{T}_{L_k} is an estimate for Watson's luminance threshold and \hat{x}_k is an estimate for the original coefficient value. The latter results from adding the received quantized coefficient value to an estimate for the error, $\hat{\varepsilon}'_k$ (which is signed). This error can be computed similarly to (6.20), using the difference $(X_k - x)$ inside the summation, instead of its absolute value. After estimating slack values, the global distortion metric is computed using (6.19).

6.6 Results

6.6.1 PSNR estimation

In order to evaluate the accuracy of the PSNR estimation algorithm, all reference images in LIVE database have been subject to JPEG compression, with quality factors in the range from 5 to 90 with increments of 5. The PSNR of each encoded image has been computed using squared error estimates that result from (6.4) and (6.5). Both equations use the value of $\hat{\lambda}_f$ obtained from (6.14). These values have been computed recursively starting from the lower frequencies (zig-zag scan order [48]), according to the following procedure:

1. $\hat{\lambda}_{ML}$ and r_0 are computed for each frequency;
2. $\hat{\lambda}_p$ is computed by using previously estimated values of $\hat{\lambda}_f$ in the neighborhood (or by using the values of $\hat{\lambda}_{ML}$ at the start of the recurrence).

The resulting no-reference PSNR values have been plotted in Figure 6.6 and confronted with their true values. Figure 6.6-a) shows the results from the algorithm described in this chapter, while Figure 6.6-b) depicts the results achieved by an implementation of the Laplacian compensation algorithm proposed in [88]. Table 6.2 depicts global statistics of the error between true and estimated PSNR, that result from this experiment, for both algorithms.

As can be observed from both the figures and the table, PSNR estimates based on the described algorithm for λ estimation are quite accurate, and better than the ones resulting from the implementation of Laplacian compensation method.

Additionally, Figure 6.7 depicts the PSNR estimation results attained for two images, subject to JPEG compression, with quality factors in the range 10 – 90. These examples represent the best and the worst PSNR estimates for the images used in the experiments.

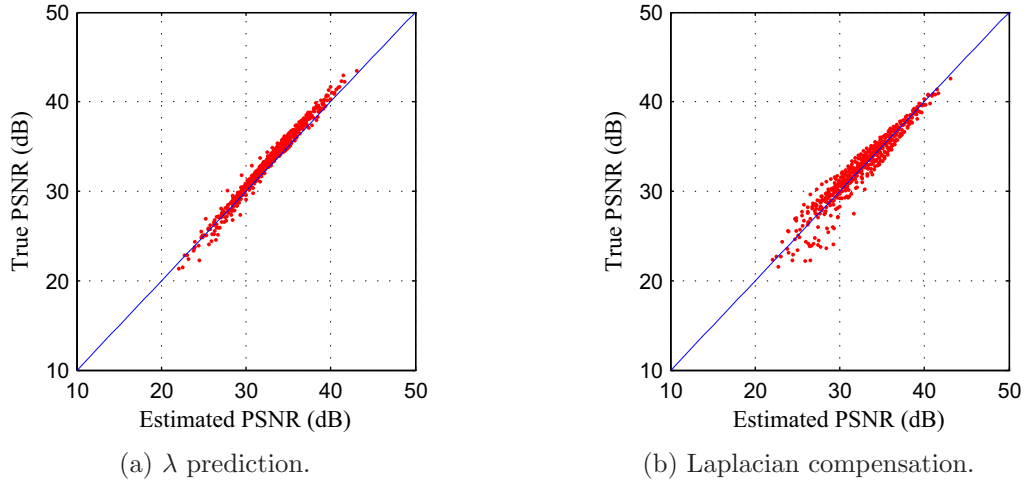


Figure 6.6: Global PSNR estimation results.

	λ prediction	Laplacian compensation
Average absolute error	0.660 dB	1.047 dB
Root mean square error	0.789 dB	1.324 dB
Linear correlation (estimated <i>vs.</i> true PSNR)	0.992	0.957

Table 6.2: PSNR estimation accuracy.

6.6.2 Quality scores

Similarly to what has been done in the previous chapter, the results for quality assessment have been evaluated by comparing the quality scores retrieved by the algorithm with those in LIVE database. In order to perform this evaluation, the estimated Watson's distance values, \hat{D}_W , have been computed for all images in the database, using equation (6.19). The resulting values are depicted in Figure 6.8-a).

Afterwards, those values have been mapped into the interval $[0; 100]$, using the same process described in Section 5.4.2, namely using the logistic function depicted in equation (5.35). The training set used for this procedure consists of DMOS values associated to the JPEG encoded images in LIVE database (125 out of 175). The remaining 50 DMOS values have been assigned to the validation set. The resulting logistic function can be observed in Figure 6.8-a).

Figure 6.8-b) depicts the normalized no-reference quality scores versus their DMOS

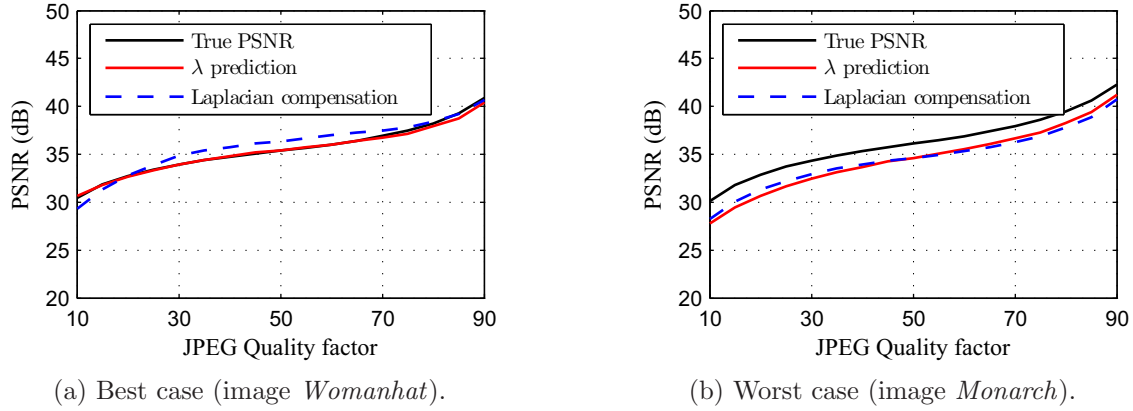


Figure 6.7: Best and worst case PSNR estimates.

values. As can be concluded, the resulting objective quality scores using the proposed algorithm are well correlated with the subjective quality scores.

The performance measurements described in Section 4.7 have been synthesized in Table 6.3, where a comparison with the algorithms proposed in [21] and in [80] is also depicted (only the results for the validation image set have been presented in the table).

Remember that the algorithm proposed in [21] estimates quality scores based on artifact measurements (more specifically, blurriness and blockiness effects). In order to perform a fair comparison with the algorithm proposed in this paper, it has been implemented following the description given in [21]. As for the algorithm proposed in [80], quality scores result from combining the outputs of neural networks, whose input values are block-based features extracted from the image under evaluation. The comparison with this algorithm has been performed using the results given in [80], which were also obtained through training and validation image sets taken from LIVE database (with the same sizes as the mentioned in this work). The results depicted in [80] have been scaled from the range $[-1; 1]$ to the range $[0; 100]$. The measurements affected by this scaling have been signaled with ‘*’ in Table 6.3.

These results confirm the good performance of the algorithm proposed in this paper. When compared with the performance of [21], the proposed scheme shows better results for all the measurements, with more emphasis on the ρ_s measurement. When compared with the results depicted in [80], it shows slightly worse results for the RMS and the average absolute error measurements. On the other hand, and considering that the ideal target value for ρ_c is 1, the results for the ρ_c measurement

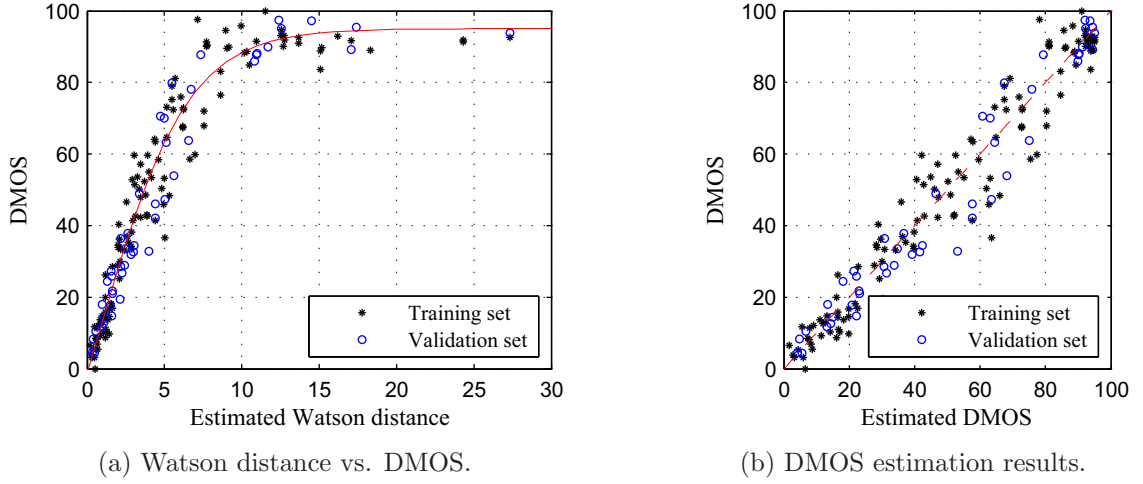


Figure 6.8: DMOS estimation results.

	Wang [21]	Gastaldo [80]	Proposed
Average absolute error	7.558	5.5*	5.640
Root mean square error (RMS)	8.910	7.0*	7.393
Pearson correlation coefficient (ρ_c)	0.960	0.95	0.974
Spearman rank order coefficient (ρ_s)	0.967	N/A	0.978

Table 6.3: Evaluation of the proposed metric.

are noticeable better.

6.7 Summary

A new approach for estimating original DCT coefficient distribution parameters from their quantized values has been proposed in this chapter. It explores the correlation between coefficients distribution at adjacent DCT frequencies. The resulting distribution estimates are then used for computing a no-reference quality score of images subject to quantization noise. Two different approaches for quality scoring have been considered: PSNR estimation and perceptual quality estimation, based on a JND model by Watson.

For the PSNR values, results have shown that the proposed algorithm provides

estimates that are more accurate than the ones provided by a state-of-the-art algorithm [88]. The results concerning perceptual quality scores have also been quite satisfactory, showing a high correlation with the human perception of quality. They have also been compared with other no-reference metrics for evaluating the quality of JPEG encoded images [21, 80], generally exhibiting better results.

Chapter 7

Perceptual video quality assessment

7.1 Introduction

In the previous chapter, an image quality assessment algorithm based on statistical models of the DCT coefficients' distribution has been presented. Since the proposed metric belongs to the no-reference class, it was necessary to accurately estimate the distribution of the original DCT coefficients using the received (corrupted by quantization) coefficient data.

The work by Turaga *et al.* [87] and Ichigaya *et al.* [88, 111], already mentioned in the previous chapter, are both statistical based PSNR estimation algorithms that are not accurate at low bitrate image or video encoding. This lack of accuracy is mainly due to the increasing number of DCT coefficients that are quantized to zero values as the encoding bitrate decreases.

In a more recent work [89], Eden proposed a PSNR estimation method for H.264 encoded video sequences. The coefficients' distributions are modeled according to Laplace densities, using a low complexity algorithm for the estimation of the density's parameter. It tackles the “all coefficients quantized to zero” problem by imposing bounds in the parameter's value at the corresponding frequencies. The results depicted in [89] show that this strategy provides good PSNR estimates for I-frames but the results for P and B-frames still need to be improved.

All the above mentioned works estimate PSNR values, which is a rough quality metric that does not correlate well with MOS values [62]. The algorithm presented in

the previous chapter is a no-reference quality assessment method for images subject to JPEG encoding that, besides producing PSNR estimates, also outputs MOS estimates that were shown to be well correlated with the corresponding subjective assessment data. The goal of this chapter, and also the main achievement of the Thesis, is to extend the work presented in the previous chapter for assessing the quality of encoded video sequences.

Since there are important differences between JPEG encoding and standard video encoding methods, the effectiveness of the algorithm proposed in the previous chapter is not automatically proven for encoded video sequences. At this point of the Thesis, there is no strong guarantee that the prediction compensated PDF parameter estimation method, derived for the JPEG case, also works effectively in the case of encoded video. In the JPEG case, quantization step sizes are the same for all DCT coefficients located at the same spatial frequency; in the case of standard video codecs, the quantization step sizes at a given spatial frequency may vary from macroblock to macroblock. Accordingly, maximum likelihood parameter estimates for the DCT coefficients' distributions, based on the observed values of quantized data, must consider this quantization step variation. Another difference from JPEG to encoded video is as follows: in the JPEG case, the inputs for the DCT transform are pixel values; in the encoded video case, the inputs of the DCT transform are (or can be) prediction errors – the residuals – obtained during encoding. For instance, in H.264 encoded video those residuals may result from spatial or temporal predictions; in MPEG-2 the residuals result from temporal predictions only (spatial prediction is not used in this case).

Considering those differences, this chapter generalizes the method described in the previous chapter to the more challenging case of encoded video sequences. Although the H.264 standard has been considered, the proposed method can be straightly applied to other DCT-based video encoding schemes, such as MPEG-2. It starts by estimating the DCT coefficient's error, assuming that these are corrupted by quantization noise only. Error estimates that result from this procedure are then perceptually weighted, by considering characteristics of the human eye, namely its sensitivity to spatio-temporal contrast. The spatio-temporal contrast sensitivity function based on the work of Kelly and Daly, described in Section 4.4.1, is used. In [69], Kelly devised an analytic model for the spatio-temporal CSF, based on data collected from his experiments. His work was further extended by Daly in [41] by considering movements of the eye, namely *smooth pursuit*, *natural drift* and *saccadic* eye movements.

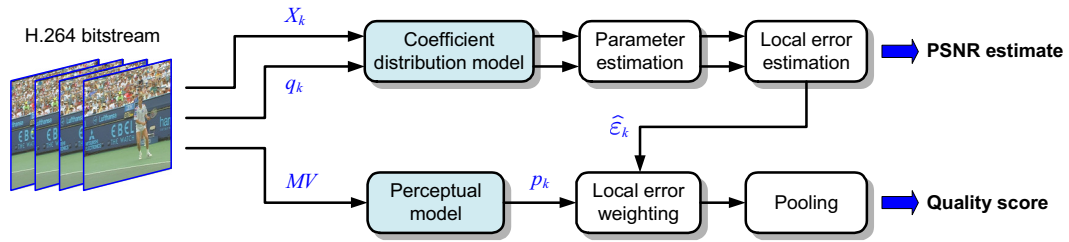


Figure 7.1: Architecture of the proposed algorithm for video quality assessment.

The performance of the metric proposed in this chapter was evaluated using the results of the subjective quality assessment tests described in Section 3.5. Remember that these tests consisted on the evaluation of a set of video sequences encoded at different bitrates, according to the *degradation category rating* (DCR) methodology suggested in ITU-T Rec. P.910 [6].

This chapter is organized as follows: in Section 7.2, the no-reference quality estimation framework is introduced and its modules are detailed in Sections 7.3 and 7.4. Results are depicted in Section 7.5 and a summary of this chapter is provided in Section 7.6.

7.2 Algorithm overview

The proposed algorithm for assessing the quality of an H.264 encoded video sequence is represented in Figure 7.1. Its architecture resembles the one described in the previous chapter. However, instead of dealing with JPEG encoded images, the algorithm now deals with DCT-based encoded video, more specifically H.264 [54] encoded video. The main new contributions that can be found with respect to the algorithm presented in the previous chapter are as follows:

- Since H.264 allows variable quantization steps across macroblocks belonging to the same frame, the parameter estimation method presented in this chapter deals with the possibility that DCT coefficients at the same frequency are quantized with different quantization steps.
- The local error weighting module accounts for a key perceptual factor in video: motion. Therefore, the perceptual model used for error weighting is substantially different from the one used in the previous chapter.

Similarly to what has been performed in the previous chapter, let us start by considering that the probability density function of the original DCT coefficient data is known. In this condition, an estimate for the local mean square error, $\hat{\varepsilon}_k^2$, at the k -th DCT coefficient, can be obtained by observing the value of its quantized value, X_k :

$$\hat{\varepsilon}_k^2 = \int_{-\infty}^{+\infty} f_X(x|X_k)(X_k - x)^2 dx. \quad (7.1)$$

Again, $f_X(x|X_k)$ represents the PDF of the original DCT coefficients values conditioned to the observed value of X_k . Using *Bayes rule* and considering that $P(X_k|x) = 1$ if x is in the quantization interval around X_k , and $P(X_k|x) = 0$, otherwise, (7.1) can be rewritten as:

$$\hat{\varepsilon}_k^2 = \frac{\int_{a_k}^{b_k} f_X(x)(X_k - x)^2 dx}{\int_{a_k}^{b_k} f_X(x) dx}, \quad (7.2)$$

where $f_X(x)$ is the original coefficient data distribution and a_k and b_k are the limits of the quantization interval around X_k . For the H.264 encoding standard case (see Section 2.3.5), they can be defined as:

$$\begin{cases} a_k = -\alpha q_k \\ b_k = \alpha q_k, \end{cases} \quad \text{if } X_k = 0; \quad \text{and} \quad \begin{cases} a_k = |X_k| - (1 - \alpha)q_k \\ b_k = |X_k| + \alpha q_k, \end{cases} \quad \text{if } X_k \neq 0. \quad (7.3)$$

Note that the quantization interval limits derived in the previous chapter, used in equations (6.4) and (6.5), correspond to the case where $\alpha = 0.5$ and q_k is constant for all DCT coefficients located at the same frequency position.

From (7.2), it can be concluded that the squared error estimate depends on the value of the quantized coefficient X_k , on the quantization step q_k (which determines a_k and b_k) and on the coefficient distribution $f_X(x)$. X_k and q_k can be derived from the encoded video bitstream. As for $f_X(x)$, it is estimated from the available quantized data, through a procedure that will be described in Section 7.3.

At this point, it is possible to estimate the PSNR of the received sequence, using square error estimates, instead of their true values:

$$\text{PSNR}_{\text{est}}(\text{dB}) = 10 \log_{10} \frac{255^2}{\text{MSE}_{\text{est}}}; \quad \text{MSE}_{\text{est}} = \frac{1}{N} \sum_{k=1}^N \hat{\varepsilon}_k^2, \quad (7.4)$$

where N is the number of DCT coefficients. The DCT coefficient error estimates are then perceptually weighted using a spatio-temporal perceptual model based on [41, 69]. The function of this model is to compute local perceptual weights p_k , which reflect the sensibility of the HVS to the corresponding local errors. The inputs for the model are the motion vectors, MV , and the video frame rate, f_r , both extracted from the encoded bitstream. From the weighted local errors, $p_k \hat{\varepsilon}_k$, a global perceptual distortion metric is obtained using error pooling.

7.3 Modeling DCT coefficient data

The distribution of block-based DCT coefficient data in H.264 is typically modeled by zero-mean Laplace [107] or Cauchy [116, 117] PDFs. In this chapter, both models have been considered. They require the estimation of a single parameter and represent a reasonable trade-off between accuracy and simplicity. In the following, the methodology for estimating the distribution's parameter is described, using the original and the quantized (corrupted) DCT coefficient data.

7.3.1 Cauchy model

Using $K \times K$ DCT blocks, for each horizontal/vertical frequency pair, $(i, j) \in \{0, \dots, K-1\} \times \{0, \dots, K-1\}$, the statistical distribution of the DCT coefficient's value, x , at spatial frequency (i, j) can be described as:

$$f_X(x)_{(i,j)} = \frac{1}{\pi} \frac{\beta_{(i,j)}}{\beta_{(i,j)}^2 + x^2}, \quad (7.5)$$

where $\beta_{(i,j)}$ is the parameter of the zero-mean Cauchy PDF. For simplicity, the indexes (i, j) will be dropped along the text; however, it must be kept in mind that there is a distinct parameter value at each spatial frequency (i, j) .

Estimating β using the original coefficient values

If the original coefficient data is known, an estimate for parameter β can be computed using the maximum likelihood method [115]:

$$\beta_{ML} = \arg \max_{\beta} \left\{ \log \prod_{k=1}^N f_X(x_k) \right\}, \quad (7.6)$$

where x_k is the k -th coefficient value and N is the number of coefficients at the frequency under analysis. Using (7.5) in (7.6) leads to:

$$\beta_{ML} = \arg \max_{\beta} \left\{ \sum_{k=1}^N (\log \beta - \log(\beta^2 + x_k^2)) \right\}. \quad (7.7)$$

The value of β that maximizes (7.7) can be computed by finding the zeros of the derivative with respect to β . Therefore, it is solution of:

$$\frac{N}{\beta} - 2 \sum_{k=1}^N \frac{\beta}{\beta^2 + x_k^2} = 0. \quad (7.8)$$

To solve (7.8), an iterative root finding algorithm can be used. On this work, the *Newton-Raphson*'s method was used, starting with a small value (0.1) as the initial guess for β . Since β_{ML} is obtained using knowledge of the original (unquantized) DCT coefficient data, it can be seen as a reference value. Thus, it will be addressed to as the “original” β parameter value along this chapter.

Estimating β using quantized coefficient values

Let us now suppose that the only data available for the estimation of β is the quantized (corrupted) DCT coefficient data extracted from the encoded video bitstream. In this case, the ML estimation method can still be used, similarly to what was done before:

$$\hat{\beta}_{ML} = \arg \max_{\beta} \left\{ \log \prod_{k=1}^N P(X_k) \right\}, \quad (7.9)$$

where $P(X_k)$ represents the probability of having value X_k at the quantizer's output,

$$P(X_k) = \int_{a_k}^{b_k} \frac{1}{\pi} \frac{\beta}{\beta^2 + x^2} dx. \quad (7.10)$$

The quantization interval limits $-a_k$ and b_k are given in (7.3). Since the algorithm is designed for H.264 encoded video, it is assumed that the quantizer is linear with a step size q_k , whose value may be different from block to block. It also includes a dead zone around 0, whose size is controlled by parameter α . Solving the integral in (7.10) leads to:

$$P(X_k) = \begin{cases} \frac{2}{\pi} \tan^{-1} \left(\frac{\alpha q_k}{\beta} \right), & \text{if } X_k = 0; \\ \frac{1}{\pi} \left(\tan^{-1} \left(\frac{b_k}{\beta} \right) - \tan^{-1} \left(\frac{a_k}{\beta} \right) \right), & \text{otherwise.} \end{cases} \quad (7.11)$$

Using (7.11) in (7.9) we get:

$$\begin{aligned} \hat{\beta}_{ML} = \arg \max_{\beta} & \left\{ \sum_{k_0=1}^{N_0} \log \left(\frac{2}{\pi} \tan^{-1} \left(\frac{\alpha q_{k_0}}{\beta} \right) \right) + \right. \\ & \left. + \sum_{k_1=1}^{N_1} \log \frac{1}{\pi} \left(\tan^{-1} \left(\frac{b_{k_1}}{\beta} \right) - \tan^{-1} \left(\frac{a_{k_1}}{\beta} \right) \right) \right\} \end{aligned} \quad (7.12)$$

The two summation terms in (7.12) correspond to the two possible cases in (7.11): quantized coefficients with zero and non-zero values, respectively. Accordingly, N_0 and N_1 represent the number of coefficients (at a given frequency), that fall in those cases. The value of β that maximizes (7.12) can be obtained by finding the zero of the derivative with respect to β , which corresponds to:

$$\sum_{k_1=1}^{N_1} \frac{\frac{a_{k_1}}{\beta^2 + a_{k_1}^2} - \frac{b_{k_1}}{\beta^2 + b_{k_1}^2}}{\tan^{-1} \left(\frac{b_{k_1}}{\beta} \right) - \tan^{-1} \left(\frac{a_{k_1}}{\beta} \right)} - \sum_{k_0=1}^{N_0} \frac{\alpha q_{k_0}}{\tan^{-1} \left(\frac{\alpha q_{k_0}}{\beta} \right) ((\alpha q_{k_0})^2 + \beta^2)} = 0. \quad (7.13)$$

If $N_0 < N$, a solution for (7.13) can be found numerically, similarly to what was done for solving (7.8). If $N_0 = N$, then $\beta \rightarrow 0$, meaning that the estimated coefficient distribution is a *Dirac's delta* function centered in 0. Similarly to what was described in the previous chapter for the case of Laplace PDF applied to JPEG encode images, the ML method fails if all coefficients at a given frequency are quantized to zero.

7.3.2 Laplace model

As already mentioned, the zero-mean Laplace PDF model for the block-wise DCT coefficients' distribution located at a given spatial frequency is:

$$f_X(x) = \frac{\lambda}{2} \exp(-\lambda|x|), \quad (7.14)$$

where λ is the distribution's parameter and x is the coefficient value.

Estimating λ using the original coefficient values

Following a procedure similar to what has been done in Section 7.3.1, an ML estimation for λ , using the original coefficient data, is given by:

$$\lambda_{ML} = \frac{N}{\sum_{k=1}^N |x_k|}, \quad (7.15)$$

a result that was already seen in the previous chapter.

Estimating λ using quantized coefficient values

Assuming that only quantized data is available for parameter estimation, λ can be computed using the ML method in the same way as in (7.9). For the Laplace PDF case, the probability $P(X_k)$ can be written as:

$$P(X_k) = \int_{a_k}^{b_k} \frac{\lambda}{2} \exp(-\lambda|x|) dx = \begin{cases} 1 - e^{-\lambda b_k}, & \text{if } X_k = 0; \\ \frac{1}{2} e^{-\lambda b_k} (e^{\lambda q_k} - 1), & \text{otherwise.} \end{cases} \quad (7.16)$$

Using (7.9) for the Laplacian case, and substituting $P(X_k)$ by the result in (7.16) leads to:

$$\hat{\lambda}_{ML} = \arg \max_{\lambda} \left\{ \sum_{k_0=1}^{N_0} \log(1 - e^{-\lambda b_{k_0}}) + \sum_{k_1=1}^{N_1} \log(e^{\lambda q_{k_1}} - 1) - \lambda b_{k_1} \right\} \quad (7.17)$$

Again, the value that maximizes (7.17) can be found by looking for the zeros of the derivative with respect to λ ,

$$\sum_{k_0=1}^{N_0} \frac{b_{k_0}}{e^{\lambda b_{k_0}} - 1} + \sum_{k_1=1}^{N_1} \left(\frac{q_{k_1} e^{\lambda q_{k_1}}}{e^{\lambda q_{k_1}} - 1} - b_{k_1} \right) = 0. \quad (7.18)$$

When compared with the results obtained in the previous chapters (equations (5.27) and (5.28)), where the zeros of the derivative are given by closed form solutions, finding the value of λ_{ML} is now a more complex task. This increase in complexity is due to the possibility of multiple quantization step sizes in each frame. A solution for equation 7.18 is now found by using an iterative root finding algorithm.

However, if all coefficients have been quantized to zero, *i.e.*, $N = N_0$, only the first

sum term of (7.18) stands, leading to:

$$\sum_{k=1}^N \frac{b_k}{e^{\lambda b_k} - 1} = 0 \quad (7.19)$$

whose solution is $\lambda \rightarrow +\infty$. Thus, the estimated distribution is a *Dirac's delta* function, which is the same phenomena as previously described for the Cauchy case.

7.3.3 Improving estimation using prediction

In order to enable PDF parameter estimation at the frequencies where all DCT coefficients were quantized to zero, the method described in the previous chapter will be adapted for the H.264 case.

Figure 7.2 depicts the “original” β and λ values, computed using equations (7.10) and (7.15), of a test I-frame subject to H.264 encoding. The figures show that there is a strong correlation between parameter values at adjacent frequencies. Although these plots are related to a particular example, a similar evolution is verified on other I frames, and also in P and B frames. The plots also show that a similar evolution is verified in both possible H.264 transform sizes (4×4 and 8×8). In order to support these statements, the correlation between neighboring parameter values in a 4-connected neighborhood has been measured for all the frames used in the experiments (described on Section 7.5). For instance, using the 4×4 sized transform, those correlation measurements were of 0.92, 0.91 and 0.93 for I, P and B frames, respectively.

Similarly to what has been presented in the previous chapter, a linear predictor can be used in order to explore this correlation. Representing the predicted parameter value by $\hat{\theta}_p$, where θ can either be the Cauchy's β or the Laplace's λ , it can be written:

$$\hat{\theta}_p = w_0 + \sum_{k=1}^{K_v} \theta_k w_k, \quad (7.20)$$

where K_v is the number of neighbors, θ_k is the parameter value at the k -th neighbor and w_k is the associated linear weight. Using matrix notation, equation (7.20) can also be written as:

$$\hat{\theta}_p = \boldsymbol{\theta}^T \mathbf{w}, \quad (7.21)$$

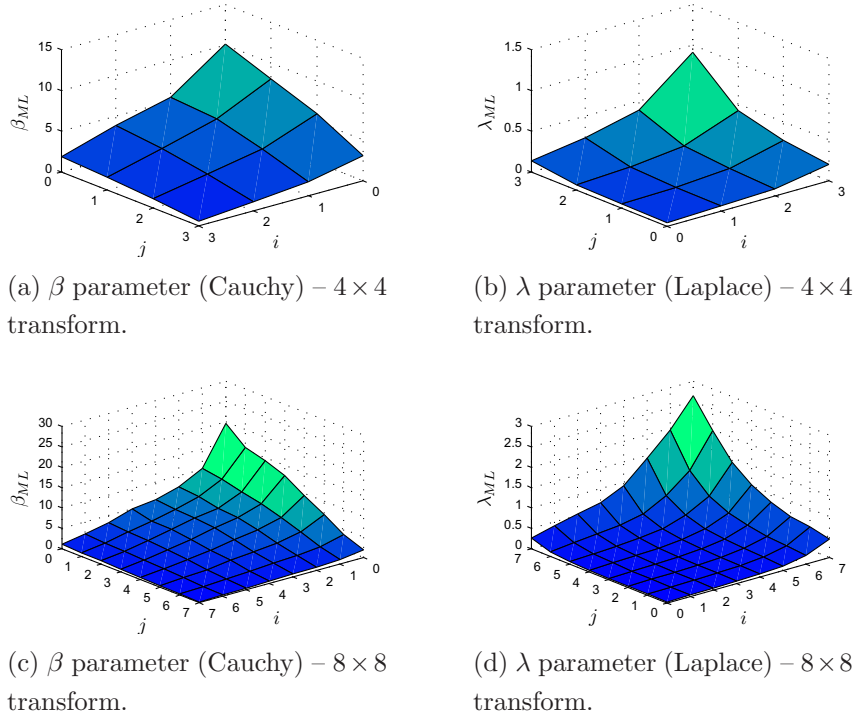


Figure 7.2: Typical evolution of the H.264 coefficients' distribution parameter as a function of the spatial frequency (original coefficient values taken from an I-frame of sequence *Stephan*).

with

$$\boldsymbol{\theta} = \begin{bmatrix} 1 \\ \theta_1 \\ \vdots \\ \theta_{K_v} \end{bmatrix} \quad \text{and} \quad \mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{K_v} \end{bmatrix}.$$

Similarly to what was performed in Section 6.4, the prediction value, $\hat{\theta}_p$, that results from (7.21) is combined with the parameter's ML estimate, $\hat{\theta}_{ML}$. However, it has been verified that combining $\hat{\theta}_{ML}$ with $\hat{\theta}_p$ proportionally to the rate of DCT coefficients quantized to zero, as performed in the previous chapter, lead to an excessive penalty on the ML estimates, when the value of r_0 is low (*i.e.*, $r_0 < 0.3$). In order to account for this effect, the criterion for combining $\hat{\theta}_p$ with $\hat{\theta}_{ML}$ was slightly modified to:

$$\hat{\theta}_f = r_0^\gamma \hat{\theta}_p + (1 - r_0^\gamma) \hat{\theta}_{ML}, \quad (7.22)$$

where $\hat{\theta}_f$ is the final estimation for the distribution's parameter and γ regulates how fast the confidence on the ML estimates decreases with increasing r_0 . The best

results for video were obtained using $\gamma = 2$. Using this value, as r_0 increases, the trust on ML estimates decreases slowly for low values of r_0 and decreases faster as r_0 approaches 1.

7.3.4 Predictor training

The goal of the training procedure is to find a weight vector \mathbf{w} suitable for the linear prediction scheme given in (7.21). One possible way is to compute \mathbf{w} by minimizing the square error between the “original” and predicted parameter values in a given training set, as described in Section 6.4.1. However, and since the number of video sequences available for the experiments is short, the variety of video content is rather limited, increasing the variability of the training results. In order to consider this effect, training has now been performed according to a procedure known as *Ridge regression* [118]. Ridge regression is a shrinkage method that, while minimizing the square error between the “original” and predicted values, also imposes a penalty on the value’s size of the linear weights. By limiting the weight values, it prevents unstable and variable results due to an exaggerated value assigned to a particular weight. According to this method, the linear weights can be found by solving:

$$\hat{\mathbf{w}}_{ridge} = \arg \min_{\mathbf{w}} \left\{ \sum_{i=1}^N (\theta_i - \hat{\theta}_i)^2 + \tau \sum_{k=1}^{K_v} w_k^2 \right\}, \quad (7.23)$$

where N is the number of video frames available for training, K_v is the neighborhood size and τ is a positive value that controls the penalty applied to the value of the weights (note that, for $\tau = 0$, this method falls in the pure least squares solution). Since there are N video frames, there will also be N “original” parameter values of θ and their corresponding neighborhood vectors $\boldsymbol{\theta}$, per frequency. Using matrix notation, (7.23) can be rewritten as:

$$\hat{\mathbf{w}}_{ridge} = \arg \min_{\mathbf{w}} \{ (\boldsymbol{\theta} - \boldsymbol{\Theta}\mathbf{w})^T (\boldsymbol{\theta} - \boldsymbol{\Theta}\mathbf{w}) + \tau \mathbf{w}^T \mathbf{w} \}, \quad (7.24)$$

where $\boldsymbol{\Theta}$ is an $N \times K_v$ matrix, where each element, θ_{ik} , is the k^{th} neighbor of the value to predict in video frame i . $\boldsymbol{\theta}$ is a vector with the “original” parameter values

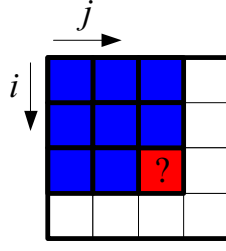


Figure 7.3: Neighborhood configuration used in the experiments.

at the position to predict, *i.e.*:

$$\Theta = \begin{bmatrix} \theta_{11} & \dots & \theta_{1K_v} \\ \theta_{21} & \dots & \theta_{2K_v} \\ \vdots & & \vdots \\ \theta_{N1} & \dots & \theta_{NK_v} \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_N \end{bmatrix}. \quad (7.25)$$

The solution that minimizes (7.24) can be found by differentiating with respect to \mathbf{w} :

$$\nabla_{\mathbf{w}} = 0 \Leftrightarrow -2\Theta^T(\boldsymbol{\theta} - \Theta\mathbf{w}) + 2\tau\mathbf{w} = 0, \quad (7.26)$$

leading to

$$\hat{\mathbf{w}}_{ridge} = (\Theta^T\Theta + \tau\mathbf{I})^{-1}\Theta^T\boldsymbol{\theta}. \quad (7.27)$$

The neighborhood configuration used by the error estimation module is illustrated in Figure 7.3. It is similar to the configuration used in the previous chapter (see Section 6.4.1), with the inclusion of an additional element. This additional element was used because it lead to better PDF parameter estimates in the case of H.264 encoded video.

The training procedure can be synthesized in the following steps:

1. for each original video frame in the training set, compute the “original” parameter values, using (7.8), if using Cauchy model, or (7.15), if using Laplace model;
2. for each encoded video frame in the training set, compute r_0 and $\hat{\theta}_{ML}$ using (7.13) or (7.18) for all spatial frequencies;
3. for each DCT frequency, in zig-zag scan order:
 - (a) build the neighborhood matrix Θ . The values of $\hat{\theta}_f$ are computed using

the values of r_0 and $\hat{\theta}_{ML}$ that result from step 2, as well as previously computed predictions (if not computed yet, assume that $\hat{\theta}_f = \hat{\theta}_{ML}$)

- (b) build $\boldsymbol{\theta}$ using the values that result from step 1;
- (c) compute the weight vector \mathbf{w} for the current frequency position, using (7.27);
- (d) use the resulting values of \mathbf{w} to perform predictions at that frequency (which will be used in step (a) in posterior iterations).

7.4 Perceptual model

The function of the perceptual model is to weight and combine the local error estimates that result from the module described in the previous section. It is based on the CSF derived in [69] and extended in [41], accounting for the mechanics of the human eye. Since the goal of the metrics proposed in this Thesis is to perform no-reference video quality assessment, only video elements available at the decoder are used, namely: the motion vectors, MV , and the video frame rate, f_r .

In the following, a brief description of the model is provided, detailing the necessary steps for computing the estimated video quality scores.

7.4.1 Spatio-temporal CSF model

A spatio-temporal CSF describes the evolution of the HVS sensitivity to luminance changes and it depends on the spatial and temporal frequencies of the stimulus. In the model by Kelly [69] and Daly [41], already presented in Section 4.4.1, the spatio-temporal sensitivity is computed as a function of the spatial frequency, f_s , and the retinal velocity, v_R , as:

$$CSF(v_R, f_s) = S c_0 c_2 v_R (2\pi c_1 f_s)^2 \exp\left(-\frac{4\pi c_1 f_s}{f_{max}}\right), \quad (7.28)$$

with the terms S and f_{max} defined by:

$$S = \left(s_1 + s_2 \left|\log\left(\frac{c_2 v_R}{3}\right)\right|^3\right) \text{ and } f_{max} = \frac{p_1}{c_2 v_R + 2}.$$

The constants s_1 , s_2 and p_1 have been set to 6.1, 7.3 and 45.9, respectively [69]. The parameters c_0 , c_1 and c_2 allow model tuning and have been set to the same values as in [41]: $c_0 = 1.14$, $c_1 = 0.67$ and $c_2 = 1.7$.

The spatial frequency f_s can be computed as the euclidean norm of the spatial frequency components:

$$f_s = \sqrt{f_x^2 + f_y^2}. \quad (7.29)$$

In the $K \times K$ block-wise DCT domain, the spatial frequency components f_y and f_x (in cycles per degree), at location (i, j) of a DCT block are given by:

$$f_y = \frac{i}{2K\alpha_y} \text{ and } f_x = \frac{j}{2K\alpha_x}, \quad (7.30)$$

where α_x and α_y are the observation angle of a pixel along the horizontal and vertical directions, respectively. The observation angle of a pixel along a generic direction ϕ can be computed as:

$$\alpha_\phi = \arctan \frac{l_\phi}{2dN_\phi} \simeq \frac{l_\phi}{2dN_\phi}. \quad (7.31)$$

where l_ϕ is the height/width of the images displayed on the screen, d is the distance from the observer to the screen and N_ϕ is the vertical/horizontal resolution of the displayed video sequence.

The object velocity on the retina plane is strongly related with the object velocity in the image plane. However, the human eye has the ability to track objects, slowing down the velocity of the object in the retina plane. This characteristic is called the *smooth pursuit eye movement* (SPEM). Additionally, there are other movements of the eye, namely the *natural drift* and *saccadic* eye movements. The former is a slow eye movement that causes a little amount of motion in the retina plane, while the latter are fast eye movements caused by changing the eye gaze to new image plane locations.

According to [41], the retinal image velocity can be computed as:

$$v_R = v_I - v_E, \quad (7.32)$$

where v_I is the angular velocity of the object on the image plane and v_E is a compensation term associated to the eye movements, computed as:

$$v_E = \min\{g_S \times v_I + v_{MIN}; v_{MAX}\}, \quad (7.33)$$

where g_S is the SPEM gain, set to 0.92; v_{MIN} and v_{MAX} are the minimum and maximum velocities associated to the eye natural drift and saccadic eye movements, set to 0.15 and 80 deg/s, respectively.

The angular velocity on the image plane, v_I , is given by:

$$v_I = f_r \sqrt{(MV_x \alpha_x)^2 + (MV_y \alpha_y)^2}, \quad (7.34)$$

where f_r is the frame rate of the video sequence and (MV_x, MV_y) are the components of the motion vector along the horizontal and vertical directions, respectively. The components of the observation angle of a pixel, α_x and α_y , are those resulting from (7.30).

7.4.2 Quality scores

Based on the result of the CSF computed at each location of the block-wise DCT domain, a global distortion value for each video frame, \hat{D}_f , is computed using L_4 error pooling, as suggest in [46, 47], according to:

$$\hat{D}_f = \sqrt[4]{\sum_k (\hat{\varepsilon}_k \hat{p}_k)^4}, \quad (7.35)$$

where $\hat{p}_k = \text{CSF}(v_{r_k}, f_{s_k})$ is the result of the contrast sensitivity function at the k -th DCT coefficient's location and $\hat{\varepsilon}_k$ is the error estimate that results from the error estimation module. As already discussed, the use of L_4 error pooling emphasizes higher distortions perceived by the viewer, which may drawn his visual attention from smaller distortions. To conclude, the same pooling process is applied along the time axis in order to get a global distortion metric for the encoded video sequence:

$$\hat{D}_g = \sqrt[4]{\sum_i \hat{D}_{f_i}^4}. \quad (7.36)$$

Note that, for longer video sequences, a granularity period for computing \hat{D}_g could be defined (*e.g.*, \hat{D}_g could be computed every 10 seconds of video).

7.5 Results

The input video sequences used in the following experiments were the same used in the subjective quality assessment tests, described in Section 3.5.

Frame type	Model	
	<i>Laplace</i>	<i>Cauchy</i>
I	0.647	0.402
P	0.450	0.618
B	0.507	0.522

Table 7.1: Mean PSNR estimation error (dB).

7.5.1 Prediction accuracy

Training has been performed separately for each frame type (I, P or B) and, since there are two possible coefficient distribution models which can be used, the model that has been selected for each frame type was the one leading to the highest PSNR estimation accuracy. To evaluate this criterion, the PSNR was first estimated using equations (7.1) and (7.4) and the distribution parameters have been estimated using the original coefficient data (*i.e.*, using the benchmarking θ_{ML} estimates). The resulting mean PSNR estimation error is depicted in Table 7.1. Based on these results, the Cauchy model was selected for the I frames, while the Laplace model was selected for P and B frames.

The training of the parameter prediction module was performed using one half of the available video sequences, following the procedure described in Section 7.3.4. The effectiveness of the proposed prediction scheme has been evaluated using the remaining sequences.

To illustrate the results, Tables 7.2-a) and b) depict the performance of the ML estimation method, when used alone. Table 7.2-a) depicts the percentage of video frames where the method fails, at each DCT coefficient frequency. As for Table 7.2-b), it represents the relative error between “original” and ML estimated parameter values for the I-frames, for the cases where a value is successfully estimated. It can be observed that the ML method fails and becomes increasingly inaccurate as the DCT frequency increases.

Table 7.2-c) represents the relative error between “original” and prediction estimates, when the trained predictor is used alone. Similarly, Table 7.2-d) represents the relative estimation error that results after combining ML with prediction estimates. It can be observed that the effectiveness of the prediction scheme increases with increasing frequency. This is due to the higher rates of DCT coefficients quan-

	\vec{j}					\vec{j}			
$i \downarrow$	0	0	1.8	14.6	$i \downarrow$	11.7	27.6	37.1	53.7
	0	0.7	6.2	35.4		17.4	34.5	45.1	53.7
	0.4	3.7	16.4	50.0		25.6	43.8	48.8	58.4
	10.0	28.8	46.4	75.9		45.6	51.8	58.1	71.5
(a) ML estimation failure rate [%].					(b) ML only (estimation error [%]).				
	\vec{j}					\vec{j}			
$i \downarrow$	–	23.6	25.7	21.3	$i \downarrow$	11.7	20.0	24.5	20.8
	24.7	20.9	20.3	17.2		17.0	20.3	20.2	17.2
	24.4	21.2	16.8	14.5		22.6	21.3	17.0	14.5
	19.0	20.5	19.1	12.7		19.3	19.8	18.8	13.0
(c) Prediction only (estimation error [%]).					(d) ML and prediction (estimation error [%]).				

Table 7.2: Parameter estimation accuracy.

tized to zero that are associated to the higher frequencies (causing failure and inaccuracy of the ML method). For low frequency coefficients, parameter estimates that result from combining ML with prediction are substantially better than those resulting individually from the ML method or from the prediction scheme.

In addition, Figure 7.4 depicts an example that illustrates the estimation of the Cauchy parameter, β , in the presence of H.264 encoding. Figure 7.4-a) shows the “original” values of β that result from solving (7.6), which can be seen as the no-reference estimation benchmark. Figure 7.4-b) shows the results of ML parameter estimation based on the quantized data. As can be observed from this plot, the parameter could not be estimated at seven spatial frequencies, due to all DCT coefficients quantized to zero at those frequencies. After using the predictor, the missing parameter values are computed and the estimates are improved, as shown in Figure 7.4-c). For a better comparison, Figure 7.4-d) depicts in a 2D plot the information of the previous plots.

Note that, since all video sequences were encoded using the H.264’s Main Profile, the results and corresponding plots were obtained for the 4×4 sized transform only. Nevertheless, and considering the plots depicted in Figures 7.2-c) and d), the same process is expected to work in higher H.264 profiles, where the 8×8 transform is allowed. In such cases, distribution parameter predictors should be

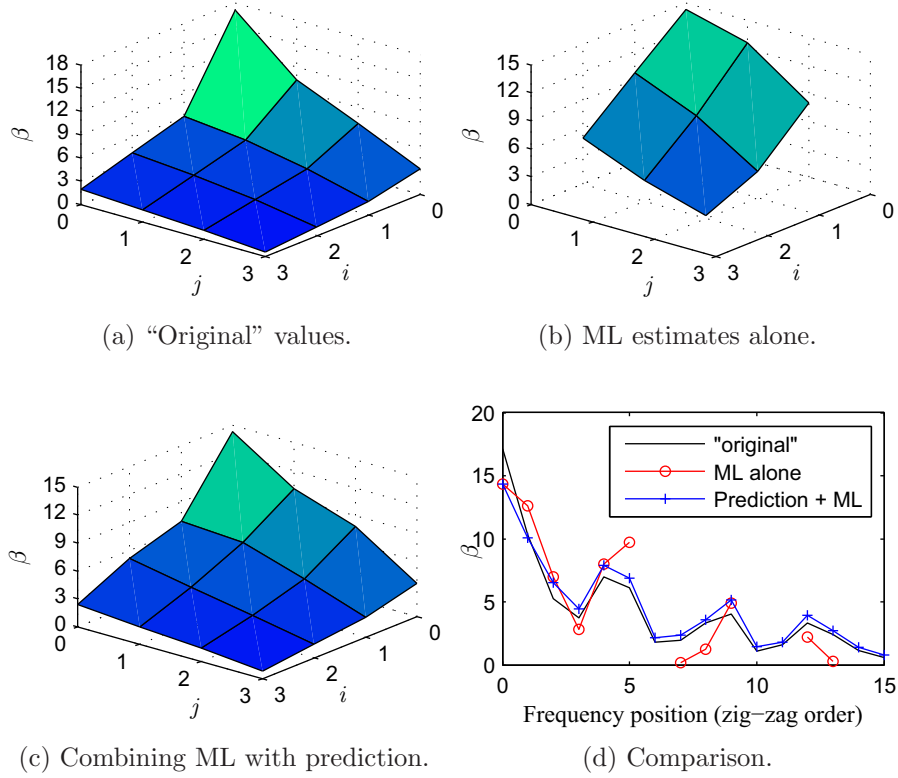


Figure 7.4: Example of parameter estimation (on an H.264 encoded I-Frame, using Cauchy model).

trained separately for each transform size.

7.5.2 PSNR estimation

Using the full set of encoded video sequences, the PSNR has been estimated and compared with its true value. Results are presented in Figures 7.5 and 7.6, for the training and test sets, respectively, and separated according to the frame type. As can be observed from the plots, the proposed method is quite accurate. Note that a compensation procedure has been performed in order to consider the possibility of *skipped* macroblocks, which become quite common in P and B frames as the encoding bit rate decreases. This compensation procedure is given by:

$$\text{MSE}_{\text{est}} = r_s \times \text{MSE}_{\text{ref}} + (1 - r_s) \times \text{MSE}_{\epsilon}, \quad (7.37)$$

where r_s is the rate of skipped MBs within the frame under analysis, MSE_{ref} is the MSE of the reference frame(s) and MSE_{ϵ} is the mean square error estimate

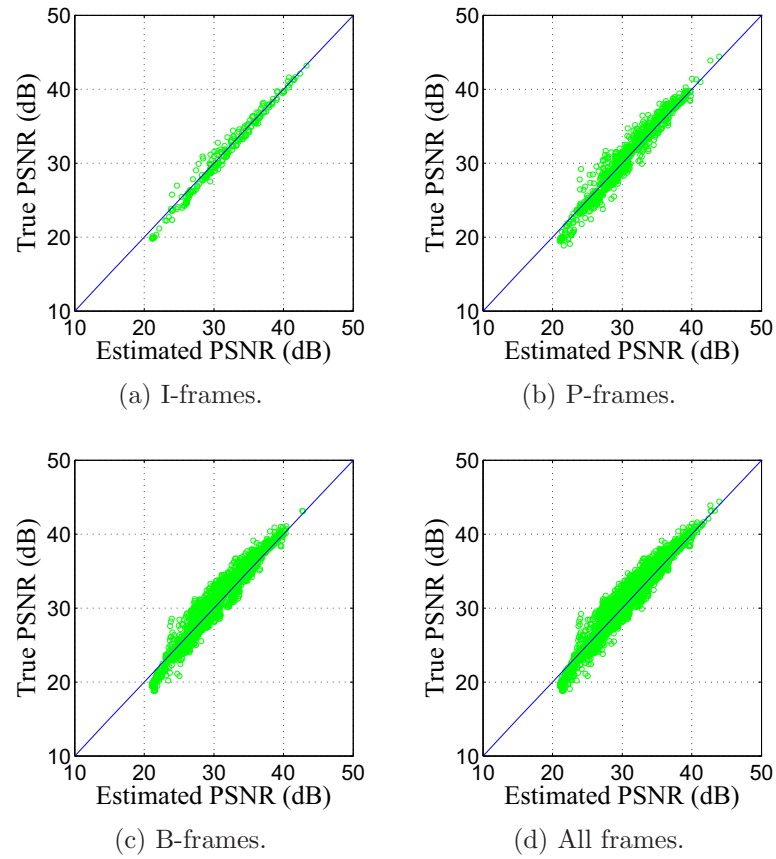


Figure 7.5: No-reference PSNR estimation *vs.* true PSNR – training set.

computed by the algorithm, considering the nonskipped MBs only.

Additionally, the plots depicted in Figure 7.7 show the temporal evolution of the PSNR estimates in four video sequences with similar bitrates (about 512 kbit/s). As can be observed, PSNR estimates closely follow their true values in spite of large PSNR variations within the same video sequence.

The algorithm proposed by Eden in [89] has been implemented for comparison purposes. This algorithm models coefficient distribution using a Laplace PDF, and uses a low complexity parameter estimation method for computing λ , which is given by:

$$\hat{\lambda}_{Eden} = -\frac{\log(1 - r_0)}{\alpha \bar{q}}, \quad (7.38)$$

where \bar{q} is the average quantization step used at a given DCT frequency within one frame, r_0 is the the rate of DCT coefficients quantized to zero and α is the parameter that controls the width of the quantizer's dead zone around 0. Additionally, the algorithm addresses the “all coefficients quantized to zero” problem by imposing

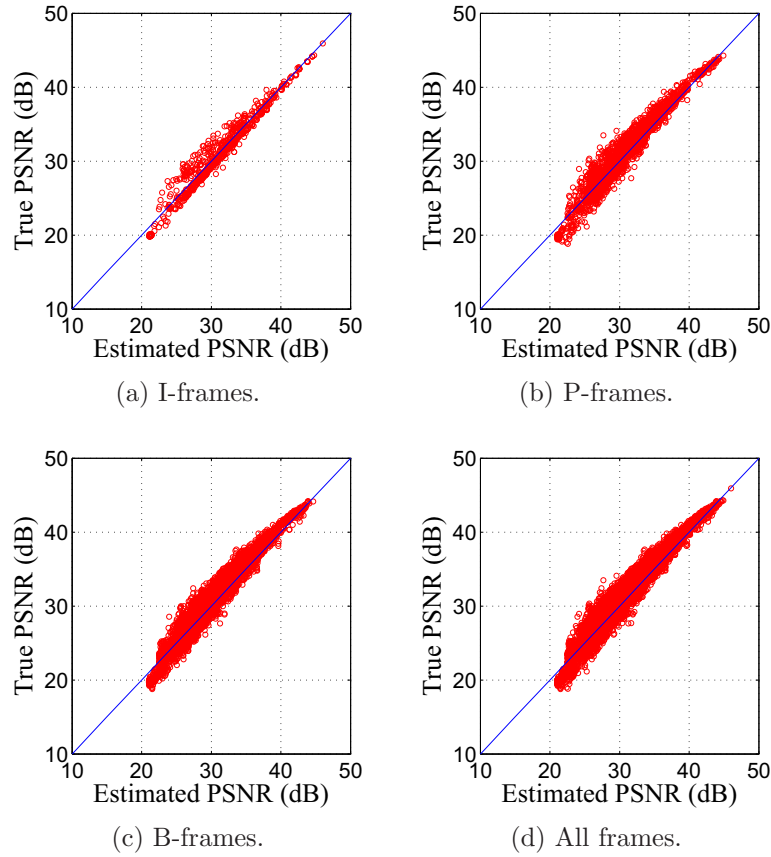


Figure 7.6: No-reference PSNR estimation *vs.* true PSNR – test set.

a bound on the value of λ in those situations. Based on information provided by the author, these bounds have been set to the maximum value of λ found in lower frequencies, since it is not likely to get smaller values of λ as frequency increases.

Table 7.3 compares the proposed method and the implementation of [89]. The symbols ϵ_{avg} , ϵ_{rms} and ρ represent, respectively, the average error, the root mean square error and the correlation, between true and estimated PSNR values. As can be observed from the table, the proposed method shows higher PSNR estimation accuracy regardless of the frame type.

7.5.3 Quality assessment

The results for quality assessment have been evaluated by comparing the quality scores retrieved by the algorithm with the ones that result from the subjective tests described in Section 3.5.

Figure 7.8-a) depicts the the value of the propose perceptual distortion metric, \hat{D}_g ,

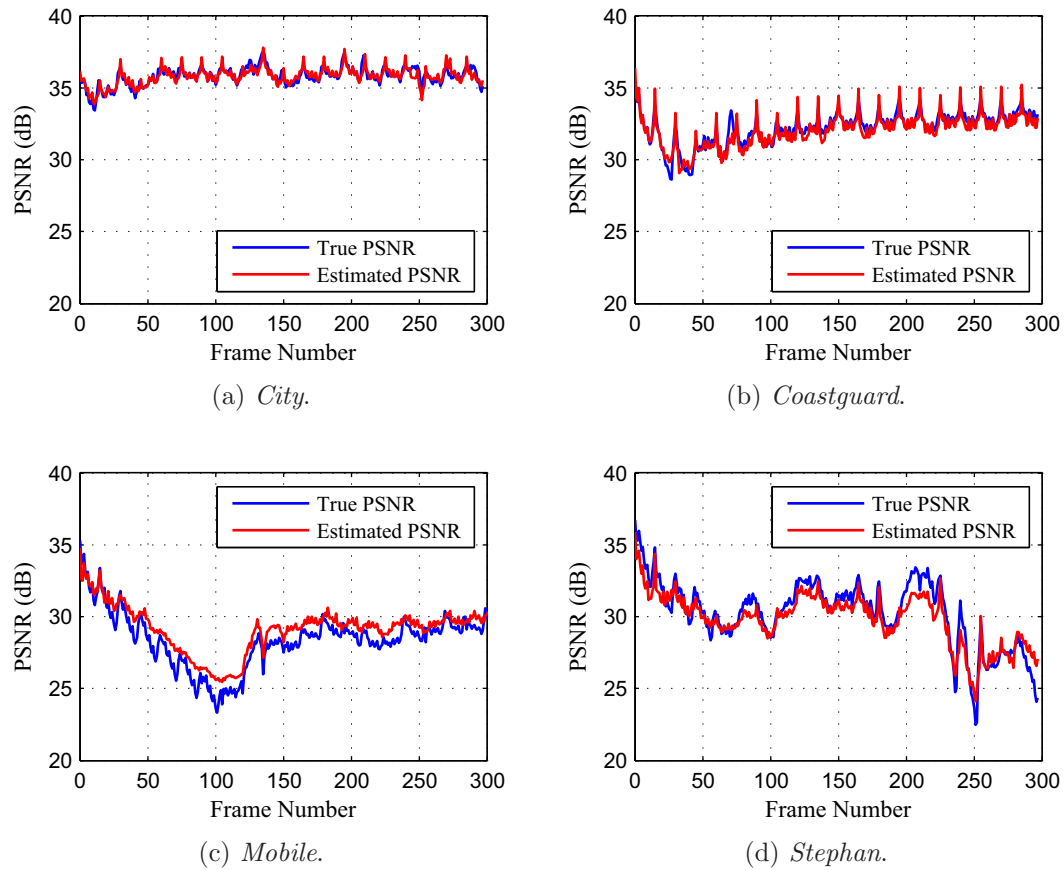


Figure 7.7: Temporal evolution of PSNR estimates (all sequences encoded at 512 kbit/s).

that results from (7.36), versus the corresponding true MOS values. Similarly to what has been performed in the results section of the previous chapters, a logistic function was used in order to map the \hat{D}_g values into the MOS range 1–5, used in the subjective experiments. The estimated MOS values are therefore the result of:

$$\text{Estimated MOS} = a_0 + \frac{a_1}{1 + e^{a_2 + a_3 \hat{D}_g}}, \quad (7.39)$$

where a_0 to a_3 are curve fitting parameters. In order to compute these parameters, the available video sequences, in a total of 50 sequences, have been split into training and validation sets, using one half of the sequences for each set. Parameter values are those that result from minimizing the square differences between true and estimated MOS scores in the training set, using the *Levenberg-Marquardt* method. A sketch of the resulting curve is also depicted in Figure 7.8-a).

Figure 7.8-b) shows the resulting normalized MOS estimates versus their true values.

Frame Type	Eden's [89]			Proposed		
	ϵ_{avg}	ϵ_{rms}	ρ	ϵ_{avg}	ϵ_{rms}	ρ
I	1.30	1.57	0.99	0.72	0.91	0.99
P	2.07	2.52	0.97	0.82	1.09	0.98
B	2.79	3.22	0.97	0.87	1.12	0.98
All	2.50	3.96	0.97	0.84	1.10	0.98

Table 7.3: PSNR estimation accuracy.

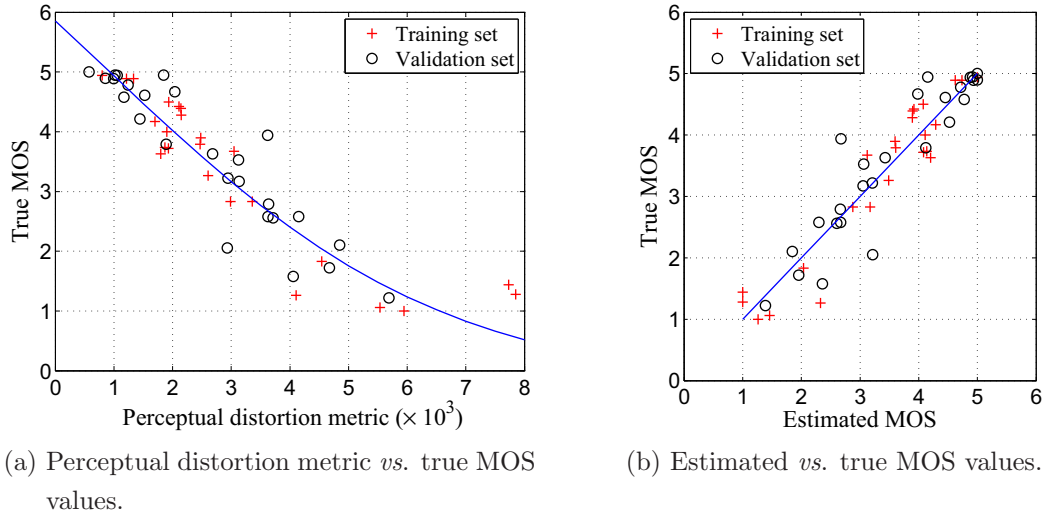


Figure 7.8: MOS estimation results.

As can be observed, the NR objective quality scores resulting from the proposed algorithm are well correlated with the subjective quality assessment data.

7.5.4 Comparison and discussion

The performance indicators suggested by VQEG (see Section 4.7) have been computed using the validation set. The results can be observed in Table 7.4. Pearson correlation and Spearman rank order coefficients are both above 0.9, which are good results for video. The RMS is smaller than 0.5, which means that most of the MOS estimates computed by the metric are within the grades given by the observers.

Compared with other results found on the literature, the proposed method seems to outperform the algorithms designed for similar purposes. In [28], Ries *et al.* propose

Root mean square error (RMS)	0.432
Pearson correlation coefficient (ρ_c)	0.939
Spearman rank order coefficient (ρ_s)	0.921
Outlier ratio (O_r)	0.020

Table 7.4: Evaluation of the proposed metric.

a no-reference video quality assessment metric where the quality scores result from combining a set of motion features extracted at the decoder. The method is improved in [29], where a different parametrization for estimating MOS is used according to a previous classification of the video content. These methods were evaluated using SIF (352×240 pixels) H.264 encoded video sequences, and the declared performance in [28] and [29] are $\rho_c = 0.80$ and $\rho_c = 0.86$, respectively, which are below the results of the method proposed in this paper.

In [27], Oelbaum and Diepold propose a reduced reference method for H.264 encoded sequences where several features extracted from the video are combined (most of them are artifact measurements and motion oriented features), and the results are adjusted based on two parameter values sent through a side channel. The declared performance of this method is $\rho_c = 0.84$, $\rho_s = 0.80$ and $O_r = 0.58$, which are also below the results achieved by the algorithm proposed in this paper.

A standard for reduced reference quality assessment of cable television signals is given in Recommendation ITU-T J.246 [3]. This metric – *Edge-PSNR* – is based on edge maps extracted from the original signals, which are sent to the receiver. The performance of this metric increases as the side channel bandwidth increases (*i.e.*, as the number of points in the sent edge map increases). The resulting values for ρ_c are in the range $0.81 - 0.83$. Again, our method shows better performance. However, it must be kept in mind that the method proposed in this paper is adapted to DCT-based video encoding while the standardized method [3] is not distortion specific.

7.6 Summary

A no-reference quality assessment algorithm for H.264 encoded video sequences has been proposed in this chapter. Similarly to the algorithm presented in the previous chapter for still images, it comprises a local error estimation module followed by

an error weighting module. The error weighting model is based on a perceptual spatio-temporal model adapted from the work of Kelly and Daly.

The error estimation module is able to compute PSNR estimates based on the quantization steps and DCT coefficient values taken from an H.264 bitstream. The results of this module outperform the state-of-the-art algorithm in [89]. The no-reference quality scores are then computed based on the error estimates and on the motion vectors extracted from the bit stream. These MOS estimates correlate well with the human perception of quality and show better results than other algorithms (derived for the same purpose) described in literature [27–29].

Chapter 8

Conclusion

In the past few years, image and video quality assessment has become an increasingly important subject, as systems are moving from the analog to the digital world. In the context of digital video delivery, the quality perceived at the receivers is mainly associated to the lossy encoding method that is used and to transmission errors. These two factors require the development of new quality metrics that are able to produce automatic quality scores.

The algorithms proposed along the Thesis perform quality assessment of the encoded images or videos by estimating the errors between the original and quantized DCT coefficient values, and then weighting those errors using a perceptual model. This approach is close to a typical full reference quality assessment algorithm based on perceptual error weighting. However, the proposed algorithms do not require the presence of a reference signal, belonging to the no-reference quality metrics class. The perceptual impact of the distortion due to quantization of the DCT coefficients is computed using elements extracted from the encoded image or video bitstream. Two main approaches for this problem have been investigated: a watermarking-based quality assessment algorithm and an algorithm that relies on the statistical properties of the DCT coefficients of natural images.

In Chapter 2, the Thesis started by providing a brief insight into the image quality assessment field and to the concepts that are directly related with its context. After a brief discussion on the factors that influence the perception of quality, it presented the most relevant characteristics of human visual system. The understanding of the HVS plays an important role in the development of image quality assessment algorithms. Since the work presented on the Thesis is focused on the quality resulting from lossy encoding of media contents, this chapter also provided the basics of

image and video coding, ending with a discussion about the artifacts caused by the standardized encoding methods and their impact on image and video quality.

The main concepts associated with subjective quality assessment were presented in Chapter 3: preparation of subjective quality assessment tests, a review of the existing standardized procedures and procedures for computing MOS values from the data collected in the subjective tests. The chapter ended with a short description of the subjective quality assessment tests that were performed in the scope of the Thesis.

An overview of the research work on objective quality assessment metrics was given in Chapter 4. Quality assessment metrics have been organized according to two criteria: by considering the use (or not) of the reference signal for computing the quality score; and by considering the nature of the data used for computing the metric's result. After presenting these possibilities of classification, a state-of-the-art on objective quality assessment metrics was presented. It emphasized the metrics belonging to the no-reference quality assessment class and it also presented the recent standardized procedures for objective assessment.

Chapter 5 is the first chapter focused on the work produced during the course of the Thesis. It presented a new no-reference image quality assessment algorithm based on watermarking techniques. The distortion associated to lossy encoding is estimated from the extracted watermark signal, using two different strategies: an empirical weighting strategy based on the watermark's extraction bit error rate and a strategy that is based on the statistical distribution of the DCT coefficients. The former was not able to provide accurate PSNR estimates while the latter showed good results for blindly computing local error estimations in the presence of JPEG lossy encoding. Using the most accurate error estimation strategy, image quality scores resulted from combining those error estimates with a perceptual model. The resulting no-reference quality scores showed a strong relation with subjective quality assessment data.

Following the work presented in Chapter 5, namely the strategy for distortion estimation based on the statistics of DCT coefficients, Chapter 6 presented a no-reference quality assessment algorithm that does not require the use of a watermark signal. In short, the statistical distribution of the original DCT coefficients is estimated from their quantized values, using a methodology that explores the correlation between distribution parameters at adjacent DCT frequencies. The proposed parameter estimation technique led to PSNR estimates that are more accurate than those provided by [88]. Those estimates were then used for computing a no-reference quality score

for images subject to JPEG encoding. Similarly to Chapter 5, error estimates were weighted using a perceptual model by Watson. The proposed algorithm was able to compute accurate PSNR estimates and image quality scores highly correlated with the human perception of quality.

Finally, Chapter 7 presented the main achievement of the Thesis, an algorithm that assesses the quality of H.264/AVC encoded video sequences without using a reference signal. Similarly to the algorithm presented in the previous chapter for still images, it also comprises a local error estimation module followed by an error weighting module. The error weighting model is based on a perceptual spatio-temporal model adapted from the work of Kelly and Daly. The error estimation module is able to compute PSNR estimates based on the quantization steps and DCT coefficient values taken from an H.264 bit stream. No reference quality scores are then computed based on the error estimates and on the motion vectors extracted from the bit stream. These MOS estimates correlate well with the human perception of quality and have shown better results than other state-of-the-art algorithms.

The major topic for future work, where significant research can still be performed, is to consider the case of video transmission errors. The results of the algorithm presented in Chapter 7 could be combined with specific packet-based transmission features (*e.g.*, packet loss rate), in order to produce a quality score for the distortion due to both the lossy encoding and the transmission processes. Another topic that may be worth to investigate is to apply the ideas related with DCT coefficients distribution estimation, presented in Chapters 6 and 7, for other applications, such as image and video denoising and error concealment.

Bibliography

- [1] H. R. Wu and K. R. Rao, *Digital video image uality and perceptual vision*. CRC Press, 2006.
- [2] G. Ghinea, G.-M. Muntean, P. Frossard, M. Etoh, F. Speranza, and H. Wu, “IEEE Transactions on Broadcasting – Special issue on “quality issues on mobile multimedia broadcasting”, vol. 7, no. 3, part II,” September 2008.
- [3] ITU-T, “Recommendation J.246 – Perceptual visual quality measurement techniques for multimedia services over digital cable television networks in the presence of a reduced bandwidth reference,” 2008.
- [4] ITU-T, “Recommendation J.247 – Objective perceptual multimedia video quality measurement in the presence of a full reference,” 2008.
- [5] ITU-R, “Recommendation BT.500-11 – Methodology for the subjective assessment of the quality of television pictures,” 1974–2002.
- [6] ITU-T, “Recommendation P.910 – Subjective video quality assessment methods for multimedia applications,” 1999.
- [7] ITU-T, “Recommendation G.1070 – Opinion model for video-telephony applications,” 2007.
- [8] A. B. Watson, “DCT quantization matrices optimized for individual images,” in *proc. of SPIE Human Vision, Visual Processing, and Digital Display IV*, S. Jose, USA, 1993.
- [9] T. Brandão and M. P. Queluz, “Blind PSNR estimation of video sequences, through non-uniform quantization watermarking,” in *proc. of International Conference on Image Analysis and Recognition (ICIAR)*, Póvoa de Varzim, Portugal, September 2006.

- [10] T. Brandão and M. P. Queluz, "Towards objective metrics for blind assessment of images quality," in *proc. of IEEE International Conference on Image Processing (ICIP)*, Atlanta, USA., October 2006.
- [11] T. Brandão and M. P. Queluz, "Blind perceptual quality assessment method for DCT-based encoded images," in *proc. of European Signal Processing Conference (EUSIPCO)*, 2007.
- [12] T. Brandão and M. P. Queluz, "Blind PSNR estimation of video sequences using quantized DCT coefficient data," in *proc. of Picture Coding Symposium (PCS)*, Lisbon, Portugal, November 2007.
- [13] T. Brandão and M. P. Queluz, "Estimation of DCT coefficient statistics from their quantized values: application to image quality evaluation," in *proc. of Conference on Telecommunications*, Peniche, Portugal, May 2007.
- [14] T. Brandão and M. P. Queluz, "No-reference PSNR estimation algorithm for H.264 encoded video sequences," in *proc. of European Signal Processing Conference (EUSIPCO)*, Lausanne, Switzerland, August 2008.
- [15] H. Wu and M. Yuen, "A generalized block-edge impairment metric for video coding," *IEEE Signal Processing Letters*, vol. 4, no. 11, pp. 317–320, November 1997.
- [16] S. Oğus, Y. Hu, and T. Nguyen, "Image coding ringing artifact reduction using morphological post-filtering," in *proc. of IEEE Workshop on Multimedia Signal Processing*, Redondo Beach, USA, December 1998, pp. 628–633.
- [17] X. Marichal, W.-Y. Ma, and H.-J. Zhang, "Blur determination in the compressed domain using DCT information," in *proc. of IEEE International Conference on Image Processing (ICIP)*, Kobe, Japan, October 1999, pp. 386–390.
- [18] Z. Wang, A. Bovik, and B. Evans, "Blind measurement of blocking artifacts in images," in *proc. of IEEE International Conference on Image Processing (ICIP)*, Vancouver, Canada, September 2000, pp. 981–984.
- [19] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "A no-reference perceptual blur metric," in *proc. of IEEE International Conference on Image Processing (ICIP)*, vol. 3, Rochester, USA, September "2002", pp. 57–60.

- [20] S. Liu and A. Bovik, "Efficient DCT-domain blind measurement and reduction of blocking artifacts," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 12, pp. 1139–1149, December 2002.
- [21] Z. Wang, H. Sheikh, and A. Bovik, "No-reference perceptual quality assessment of JPEG compressed images," in *proc. of IEEE International Conference on Image Processing (ICIP)*, vol. 1, Rochester, USA, September 2002, pp. 477–480.
- [22] S. Suthaharan, "A perceptually significant block-edge impairment metric for digital video coding," in *proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASP)*, vol. 3, Hong-Kong, China, April 2003, pp. 681–684.
- [23] F. Pan, X. Lin, S. Rahardja, W. Lin, E. Ong, S. Yao, Z. Lu, and X. Yang, "A locally adaptive algorithm for measuring blocking artifacts in images and videos," *Signal Processing: Image Communication*, vol. 19, no. 6, pp. 499–506, July 2004.
- [24] P. Marziliano, F. Dufaux, S. Winkler, and T. Ebrahimi, "Perceptual blur and ringing metrics: application to JPEG2000," *Image Communication - Special issue on Objective Video Quality Metrics*, vol. 19, no. 2, pp. 163–172, February 2004.
- [25] R. Barland and A. Saadane, "Reference free quality metric for JPEG2000 compressed images," in *proc. of International Symposium on Signal Processing and its Applications*, Sydney, Australia, August 2005, pp. 351–354.
- [26] H. Liu, N. Klomp, and I. Heynderickx, "A no-reference metric for perceived ringing artifacts in images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 4, pp. 529–539, April 2010.
- [27] T. Oelbaum and K. Diepold, "A reduced reference video quality metric for AVC/H.264," in *proc. of European Signal Processing Conference (EUSIPCO)*, Poznan, Poland, September 2007, pp. 1265–1269.
- [28] M. Ries, O. Nemethova, and M. Rupp, "Motion based reference-free quality estimation for H.264/AVC video streaming," in *proc. of International Symposium on Wireless Pervasive Computing*, S. Juan, Puerto Rico, February 2007.

- [29] M. Ries, O. Nemethova, and M. Rupp, “Performance evaluation of mobile video quality estimators,” in *proc. of European Signal Processing Conference (EUSIPCO)*, Poznan, Poland, September 2007.
- [30] T. Brandão and M. P. Queluz, “No-reference image quality assessment based on DCT domain statistics,” *Signal Processing*, vol. 88, no. 4, pp. 822–833, April 2008.
- [31] T. Brandão, L. Roque, and M. P. Queluz, “Quality assessment of H.264/AVC encoded video,” in *proc. of Conference on Telecommunications*, Santa Maria da Feira, Portugal, April 2009.
- [32] T. Brandão and M. P. Queluz, “No-reference quality assessment of H.264 encoded video,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 11, pp. 1437–1447, November 2010.
- [33] T. Brandão and M. P. Queluz, “No-reference perceptual quality metric for H.264/AVC encoded video,” in *proc. of International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM)*, Scottsdale, USA, January 2010.
- [34] A. Ahumada Jr. and C. Null, “Image quality: a multidimensional problem,” in *Digital images and human vision*, A. B. Watson, Ed. Cambridge, MA, USA: MIT Press, 1993, pp. 141–148.
- [35] S. A. Klein, “Image quality and image compression: a psychophysicist’s viewpoint,” in *Digital Images and Human Vision*, A. B. Watson, Ed. Cambridge, USA: MIT Press, 1993, pp. 73–88.
- [36] A. Savakis, S. Etz, and A. Loui, “Evaluation of image appeal in consumer photography,” in *proc. of SPIE Human Vision and Electronic Imaging V*, no. 3959, 2000, pp. 111–120.
- [37] S. Rihs, *MOSAIC handbook*, 1996, ch. The influence of audio on perceived picture quality and subjective audio-video delay tolerance, pp. 183–187.
- [38] B. E. Rogowitz, “The human visual system: a guide for the display technologist,” in *proc. of SID*, 1983.
- [39] S. Winkler, *Digital video quality*. Wiley, 2005.

- [40] J. Mannos and D. Sakrison, “The effects of a visual fidelity criterion on the encoding images,” *IEEE Transactions on Information Theory*, vol. 20, no. 4, pp. 525–536, July 1974.
- [41] S. Daly, “Engineering observations from spatiovelocity and spatiotemporal visual models,” in *Vision model and applications to image and video processing*. Kluwer, 2001.
- [42] W. J. Tam, L. Stelmach, L. Wang, D. Lauzon, and P. Gray, “Visual masking at scene cuts,” in *proc. of SPIE Human Vision, Visual Processing, and Digital Display VI*, vol. 2411, no. 111, January 1995.
- [43] A. Ahumada Jr., B. Beard, and R. Eriksson, “Spatio-temporal discrimination model predicts temporal masking functions,” in *proc. of SPIE Human Vision and Electronic Imaging III*, vol. 3299, S. Jose, USA, 1998, pp. 120–127.
- [44] B. Breitmeyer and H. Ogmen, “Recent model and findings in visual backward masking: a comparison, review and update,” *Perception & Psychophysics*, vol. 62, no. 8, pp. 1572–1595, November 2000.
- [45] H.-T. Quan and M. Ghanbari, “Asymmetrical temporal masking near video scene change,” in *proc. of IEEE International Conference on Image Processing (ICIP)*, S. Diego, USA, October 2008, pp. 2568–2571.
- [46] A. B. Watson, J. Hu, and J. F. McGowan, “DVQ: A digital video quality metric based on human vision,” *Journal of Electronic Imaging*, vol. 10, no. 1, pp. 20–29, January 2001.
- [47] C. Lambrecht, “Perceptual model and architectures for video coding applications,” Ph.D. dissertation, École Polytechnique Fédérale de Lausanne, 1996.
- [48] ITU-T, “Recommendation T.81 – Digital compression and coding of continuous-tone till images: requirements and guidelines.”
- [49] IJG, “Independent JPEG group - free library for JPEG image compression, release 6b,” March 1998, available online at <http://www.ijg.org/>.
- [50] ISO/IEC, “Information technology: JPEG 2000 image coding system: Core coding system,” 2000—2004.
- [51] M. Gormish, D. Lee., and M. Marcellin, “JPEG 2000: overview, architecture, and applications,” in *proc. of IEEE International Conference on Image Processing (ICIP)*, Vancouver, Canada, September 2000.

- [52] M. Rabbani and R. Jones, “An overview of the JPEG 2000 still image compression standard,” *Signal Processing: Image Communication*, vol. 17, no. 1, pp. 3–48, January 2002.
- [53] ISO/IEC, “Information technology: generic coding of moving pictures and associated audio information,” 1996–2007.
- [54] ITU-T, “Recommendation H.264 – Advanced video coding for generic audiovisual services,” 2005.
- [55] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.
- [56] J. Ostermann, J. Bormans, P. List, D. Marpe, M. Narroschke, F. Pereira, T. Stockhammer, and T. Wedi, “Video coding with H.264/AVC: tools, performance, and complexity,” *IEEE Circuits and Systems Magazine*, vol. 4, no. 1, pp. 7–28, Quarter 2004.
- [57] I. Richardson, *H.264 and MPEG-4 Video Compression*. John Wiley and Sons, 2003.
- [58] Heinrich-Hertz-Institut, “JM 12.4 – H.264 reference software,” December 2007, available online at <http://iphome.hhi.de/suehring/tml/>.
- [59] ITU-T, “Recommendation P.930 – Principles of a reference impairment system for video,” 1993–1996.
- [60] R. Lienhart, I. Kozintsev, Y.-K. Chen, M. Holliman, M. Yeung, A. Zaccarin, and R. Puri, *The handbook of video databases: design and applications*. CRC Press, 2003, ch. 38 - Challenges in distributed video management and delivery, pp. 961–990.
- [61] S. Winkler, “Video quality and beyond,” in *proc. of European Signal Processing Conference (EUSIPCO)*, Poznan, Poland, September 2007.
- [62] B. Girod, “What’s wrong with mean-square error?” in *Digital Images and Human Vision*, A. B. Watson, Ed. MIT Press, 1993.
- [63] Z. Wang and A. Bovik, “Mean squared error: love it or leave it? a new look at signal fidelity measures,” *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98–117, January 2009.

- [64] A. Eskicioglu and P. Fisher, "Image quality measures and their performance," *IEEE Transactions on Communications*, vol. 43, no. 12, pp. 2959–2965, 1995.
- [65] S. Winkler and P. Mohandas, "The evolution of video quality measurement: from PSNR to hybrid metrics," *IEEE Transactions on Broadcasting*, vol. 54, no. 3, pp. 660–668, September 2008.
- [66] Z. Wang and A. Bovik, "A universal image quality index," *IEEE Signal Processing Letters*, vol. 9, no. 3, pp. 81–84, March 2002.
- [67] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, April 2004.
- [68] Z. Wang, L. Lu, and A. Bovik, "Video quality assessment based on structural distortion measurement," *Signal Processing: Image Communication*, vol. 19, no. 2, pp. 121–132, February 2004.
- [69] D. H. Kelly, "Motion and vision II: stabilized spatio-temporal threshold surface," *Journal of the Optical Society of America*, vol. 69, no. 10, pp. 1340–1349, October 1979.
- [70] A. B. Watson, "DCTune: a technique for visual optimization of DCT quantization matrices for individual images," *Society for Information Display Digest of Technical Papers*, vol. 24, pp. 946–949, 1993.
- [71] S. Winkler, "Issues in vision model for perceptual video quality assessment," *Signal Processing*, vol. 78, no. 2, pp. 231–252, October 1999.
- [72] J. Lubin, "A visual discrimination mode for imaging system design and evaluation," in *Visual Models for Target Detection and Recognition*, E. Peli, Ed. World Scientific Publishers, 1995, pp. 245–283.
- [73] J. Lubin, "A human vision system model for objective picture quality measurements," in *proc. of International Broadcasting Convention*, Amsterdam, Netherlands, September 1997.
- [74] C. Lambrecht and O. Verscheure, "Perceptual quality measure using a spatio-temporal model of the human visual system," in *proc. of SPIE Digital Video Compression: Algorithms and Technologies*, vol. 2668, S. Jose, USA, January 1996, pp. 450–461.

- [75] S. Winkler, "A perceptual distortion metric for digital color images," in *proc. of IEEE International Conference on Image Processing (ICIP)*, vol. 3, Chicago, USA, 1998, pp. 399–403.
- [76] S. Winkler, "A perceptual distortion metric for digital color video," in *proc. of SPIE Human Vision and Electronic Imaging IV*, vol. 3644, S. Jose, USA, 1999, pp. 175–184.
- [77] B. Girod, "Information theoretical significance of spatial and temporal masking in video signals," in *proc. of SPIE Human Vision, Visual Processing, and Digital Display*, vol. 1077, S. Jose, USA, January 1989, pp. 178–187.
- [78] J. Caviedes and S. Gurbuz, "No-reference sharpness metric based on local edge kurtosis," in *proc. of IEEE International Conference on Image Processing (ICIP)*, vol. 3, Rochester, USA, September 2002, pp. 53–56.
- [79] J. Caviedes and F. Oberti, "A new sharpness metric based on local kurtosis, edge and energy information," *Image Communication - Special issue on Objective Video Quality Metrics*, vol. 19, no. 2, pp. 147–161, February 2004.
- [80] P. Gastaldo and R. Zunino, "No-reference quality assessment of JPEG images by using CBP neural networks," in *proc. of International Symposium on Circuits and Systems (ISCAS)*, vol. 5, Vancouver, Canada, May 2004, pp. 772–775.
- [81] P. Gastaldo, R. Zunino, I. Heynderickx, and E. Vicario, "Objective quality assessment of displayed images by using neural networks," *Signal Processing: Image Communication*, vol. 20, no. 7, pp. 643–661, August 2005.
- [82] R. Babu and A. Perkis, "An HVS-based no-reference perceptual quality assessment of JPEG coded images using neural networks," in *proc. of IEEE International Conference on Image Processing (ICIP)*, vol. 1, Genova, Italy, September 2005, pp. 433–436.
- [83] M. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312–322, September 2004.
- [84] C. Lee, S. Cho, J. Choe, T. Jeong, W. Ahn, and E. Lee, "Objective video quality assessment," *SPIE Optical Engineering*, vol. 45, no. 1, pp. 17004–17015, 2006.

- [85] S. Kanamuri, P. Cosman, A. Reibman, and V. Vaishampayan, "Modeling packet-loss visibility in MPEG-2 video," *IEEE Transactions on Multimedia*, vol. 8, no. 2, pp. 341–355, April 2006.
- [86] S. Kanamuri, P. Cosman, A. Reibman, and V. Vaishampayan, "Predicting H.264 packet-loss visibility using a generalized linear model," in *proc. of IEEE International Conference on Image Processing (ICIP)*, Atlanta, USA, October 2006, pp. 2245–2248.
- [87] D. S. Turaga, Y. Chen, and J. Caviedes, "No-reference PSNR estimation for compressed pictures," *Image Communication - Special issue on Objective Video Quality Metrics*, vol. 19, no. 2, pp. 173–184, February 2004.
- [88] A. Ichigaya, M. Kurozumi, N. Hara, Y. Nishida, and E. Nakasu, "A method of estimating coding PSNR using quantized DCT coefficients," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 2, pp. 251–259, February 2006.
- [89] A. Eden, "No-reference estimation of the coding PSNR for H.264-coded sequences," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 667–674, May 2007.
- [90] S.-Y. Shim, J.-H. Moon, and J.-K. Han, "PSNR estimation scheme using coefficient distribution of frequency domain in H.264 decoder," *IET Electronics Letters*, vol. 44, no. 2, pp. 108–109, January 2008.
- [91] ITU-T, "Recommendation J.144 – Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference," 2001.
- [92] ANSI, "Digital transport of one-way video signals - parameters for objective performance assessment," 2003.
- [93] A. Hekstra, J. Beerends, D. Ledermann, R. de Caluwe, F.E. Kohler, H. Koenen, S. Rihs, M. Ehram, and D. Schlauss, "PVQM: a perceptual video quality measure," *Signal Processing: Image Communication*, vol. 17, no. 10, pp. 781–798, November 2002.
- [94] I. Cox, M. Miller, and J. Bloom, *Digital watermarking*, F. E., Ed. Morgan Kaufmann, 2002.

- [95] B. Chen and G. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Transactions on Information Theory*, vol. 47, no. 4, pp. 1423–1443, May 2001.
- [96] S. Wang, J. Zhao, W. J. Tam, and F. Speranza, "Image quality measurement by using watermarking based on discrete wavelet transform," in *proc. of Biennial Symposium on Communications*, Kingston, Canada, May–June 2004.
- [97] D. Zheng, J. Zhao, W. J. Tam, and F. Speranza, "Image quality measurement by using digital watermarking," in *proc. of IEEE International Workshop on Haptic, Audio and Visual Environments and their Applications*, Ottawa, Canada, September 2003, pp. 65–70.
- [98] P. Campisi, G. Giunta, and A. Neri, "Object-based quality of service assessment using fragile watermarking in MPEG-4 video cellular services," in *proc. of IEEE International Conference on Image Processing (ICIP)*, vol. II, New York, USA, September 2002, pp. 881–884.
- [99] P. Campisi, M. Carli, G. Giunta, and A. Neri, "Tracing watermarking for multimedia communication quality assessment," in *proc. of IEEE International Conference on Communications*, 2002.
- [100] P. Campisi, M. Carli, G. Giunta, and A. Neri, "Blind quality assessment system for multimedia communications using tracing watermarking," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 996–1002, April 2003.
- [101] M. Farias and M. Carli, "Video objective metric using data hiding," in *proc. of IEEE Workshop on Multimedia Signal Processing*, St. Thomas, USA, December 2002.
- [102] M. Farias, M. Carli, A. Neri, and S. Mitra, "Video quality assessment based on data hiding driven by optical flow information," in *proc. of SPIE Image Quality and System Performance*, vol. 5294, S. Jose, USA, January 2004.
- [103] S. Saviotti, F. Mapelli, and R. Lancini, "Video quality analysis using a watermarking technique," in *proc. of International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, Lisbon, Portugal, April 2004.
- [104] O. Sugimoto, R. Kawada, M. Wada, and S. Matsumoto, "Objective measurement scheme for perceived picture quality degradation caused by MPEG

- encoding without any reference pictures,” in *proc. of SPIE Visual Communications and Image Processing*, vol. 4310, S. Jose, USA, January 2001, pp. 932–939.
- [105] S. Bossi, F. Mapelli, and R. Lancini, “Objective video quality evaluation by using semi-fragile watermarking,” in *proc. of Picture Coding Symposium (PCS)*, S. Francisco, USA, December 2004.
- [106] M. Holliman and M. Yeung, “Watermarking for automatic quality monitoring,” in *proc. of SPIE Security and Watermarking of Multimedia Contents*, vol. 4675, S. Jose, USA, January 2002.
- [107] E. Lam and J. Goodman, “A mathematical analysis of the DCT coefficient distributions for images,” *IEEE Transactions on Image Processing*, vol. 9, no. 10, pp. 1661–1666, October 2000.
- [108] J. Price and M. Rabbani, “Biased reconstruction for JPEG decoding,” *IEEE Signal Processing Letters*, vol. 6, no. 12, pp. 297–299, December 1999.
- [109] H. Sheikh, Z. Wang, L. Cormack, and A. Bovik, “LIVE Image quality assessment database, release 2,” 2006, available online at <http://live.ece.utexas.edu/research/quality>.
- [110] VQEG, “Final report from the video quality experts group on the validation of objective models of video quality assessment, phase II,” www.vqeg.org, Tech. Rep., August 2003.
- [111] A. Ichigaya, Y. Nishida, and E. Nakasu, “Nonreference method for estimating PSNR of MPEG-2 coded video using DCT coefficients and picture energy,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 6, pp. 817–826, June 2008.
- [112] F. Muller, “Distribution shape of two-dimensional DCT coefficients of natural images,” *Electronic Letters*, vol. 29, no. 22, pp. 1935–1936, October 1993.
- [113] T. Eude, R. Grisel, H. Cherifi, and R. Debrie, “On the distribution of the DCT coefficients,” in *proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASP)*, vol. 5, Adelaide, Australia, April 1994, pp. 365–368.

-
- [114] J.-H. Chang, J. W. Shin, N. S. Kim, and S. Mitra, “Image probability distribution based on generalized gamma function,” *IEEE Signal Processing Letters*, vol. 12, no. 4, pp. 325–328, April 2005.
 - [115] R. Duda, P. Hart, and D. Stork, *Pattern classification - 2nd edition*. Wiley-Interscience, 2000.
 - [116] J. Eggerton and M. Srinath, “Statistical distributions of image DCT coefficients,” *Computers & Electrical Engineering*, vol. 12, no. 3–4, pp. 137–145, January 1986.
 - [117] Y. Altunbasak and N. Kamaci, “An analysis of the DCT coefficient distribution with the H.264 video coder,” in *proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASP)*, vol. 3, Montreal, Canada, May 2004, pp. 177–180.
 - [118] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning*. Springer, 2001.